FORENSICS IN THE CLOUD

Reconstructing Attacks Across AWS, Azure, GCP, and Oracle Cloud

FIND EVIDENCE | PRESERVE INTEGRITY | ANALYZE LOGS & CONFIGS | RECONSTRUCT ATTACKS

MIKE CARTER

Forensics in the Cloud

Reconstructing Attacks Across AWS, Azure, GCP, and Oracle Cloud

Steve Publications

This book is available at https://leanpub.com/forensicsinthecloud

This version was published on 2026-09-06

This is a Leanpub book. Leanpub empowers authors and publishers with the Lean Publishing process. Lean Publishing is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

Contents

Reconstructing Attacks Across AWS, Azure, GCP, and Oracle Cloud

About this book: Cloud forensics demands different skills than traditional digital forensics. Public clouds generate rich, structured logs across dozens of services, yet attackers can compromise, tamper with, or destroy that evidence faster than responders can collect it. This book teaches you how to investigate security incidents across AWS, Microsoft Azure, Google Cloud Platform, and Oracle Cloud Infrastructure by showing you where evidence lives, how long it persists, how to collect it without destroying integrity, and how to reconstruct attacker activity from logs, snapshots, and configuration changes. You will find production-quality code examples, provider-specific investigation workflows, real-world incident walkthroughs using synthetic data, and practical guidance on maintaining chain-of-custody in environments you do not fully control. This is a hands-on reference for defenders who need to understand what happened in their cloud, who did it, and how to prove it.

Introduction: The New Crime Scene

The 3:00 AM Alert: A Realistic Cloud Incident

It begins with an alert. At 3:14 AM on a Tuesday, your SIEM fires on a GuardDuty finding: a critical severity alert labeled `CryptoCurrency:EC2/CoinMiner`. An EC2 instance in your us-east-1 VPC is making outbound HTTPS connections to a known cryptocurrency mining pool. You pull up CloudTrail and discover the instance was launched two hours earlier by an IAM user named `dev-jenkins`, but the session originated from an IP address in a country where none of your developers live. The same user had assumed a cross-account role thirty minutes before instance creation and had spent those thirty minutes enumerating IAM policies, listing S3 buckets, and disabling CloudTrail logging in one region.

You now have roughly ninety minutes before that instance finishes downloading its mining binary, before it uses your AWS API keys to scan for other vulnerable instances, and before it discovers the S3 bucket containing your customer database.

This is the new crime scene.

In a traditional data center, a security incident would start with a physical server in a locked room. You could isolate the server from the network, image its hard drive, dump its memory, and analyze the evidence at your leisure. The evidence was tangible, it stayed in one place, and you had full control over it.

Cloud environments remove almost all of those assumptions. Your compromised resource lives in someone else's data center. You cannot reach in and pull a drive. Your evidence is a mix of logs stored in object storage, snapshots of ephemeral block volumes, and metadata in services you do not directly manage. Resources spin up and down by the minute. Multi-tenancy means you share physical hardware with other customers and cannot inspect the hypervisor or network infrastructure. Attacker tools like Pacu or cloud-native backdoors can modify IAM policies, disable logging, and establish persistence across multiple accounts within minutes.

And yet, cloud environments also offer something traditional environments cannot: comprehensive, structured, machine-readable audit trails for virtually every operation. If your logging is properly configured and you know where to look, the cloud will tell you exactly who did what, when they did it, from where they did it, and using which credentials. The challenge is not a lack of evidence. The challenge is knowing how to find it, collect it before it disappears, and reconstruct an attack from dozens of services across multiple regions before the attacker realizes you are watching.

Why Cloud Forensics Is Not Just Server Forensics in a Data Center

If you are coming from traditional host forensics, you must unlearn some habits. The differences are not cosmetic; they are structural.

First, evidence is distributed. In a traditional forensics investigation, you image a disk and extract file system artifacts, registry entries, prefetch data, and memory artifacts from one system. In the cloud, evidence about a single compromise may live in AWS CloudTrail management events across three regions, S3 data events in a logging bucket, VPC Flow Logs from a subnet, GuardDuty findings in the security service, and the instance metadata of a compromised EC2 host. An Azure incident might spread across Activity Logs, Log Analytics workspaces, Entra ID sign-in logs, and Network Security Group flow logs. Correlation is not optional; it is the investigation.

Second, evidence is ephemeral. Cloud resources are designed to be short-lived. Auto-scaling groups terminate instances. Serverless functions execute and disappear. Kubernetes pods reschedule. If you do not collect memory from a compromised instance before it is terminated, that evidence is gone forever. If an attacker deletes a CloudTrail trail and destroys the S3 bucket receiving logs, hours of audit data may vanish. Retention policies matter. You must know your defaults: CloudTrail event history retains management events for nine days by default; GCP Data Access audit logs are retained for thirty days unless exported; OCI audit logs are retained for 365 days in the console but require explicit export for longer retention. These defaults define your investigative window.

Third, you do not own the infrastructure. In the cloud, the hypervisor, network fabric, physical storage controllers, and much of the control plane are

managed by the provider. You cannot inspect the host for rootkits or extract hypervisor memory. You must trust the provider's security services and audit logs as part of your evidence chain. This introduces a layer of complexity: your forensic conclusions rest partly on data collected and stored by an entity you do not control, and legal processes for obtaining provider-held evidence vary by jurisdiction.

Fourth, identity has replaced perimeter. Traditional forensics often started with a compromised host and worked outward to determine who accessed it. In the cloud, the most reliable forensic evidence often starts with identity. A single compromised IAM user, Azure AD account, or GCP service account can be used from any IP address to perform virtually any action the identity is permitted. Investigating a cloud incident almost always means investigating which identity was compromised, when the compromise occurred, what permissions it held, and how the attacker used those permissions across services. As the MITRE ATT&CK for IaaS matrix documents, techniques like credential access, privilege escalation via IAM policy abuse, and lateral movement through role assumption are among the most common cloud attack patterns.

These differences do not make cloud forensics harder in an absolute sense. In many ways, they make it easier: the logs are more structured, more complete, and more queryable than any on-premises environment you are likely to encounter. The difficulty comes from the scale, the velocity, and the need to understand provider-specific services and their forensic value. That is what this book is designed to teach you.

What You Will Learn and How to Use This Book

This book is structured to take you from foundational concepts to advanced multi-cloud investigations. Every chapter is designed to be useful on its own, but the material builds progressively.

The book is divided into three parts.

Part 1, Chapters 1 through 4, establishes the foundation. You will learn the principles of cloud forensics, how the shared responsibility model affects evidence collection, what types of artifacts exist in each major provider, and how audit logs work across AWS, Azure, GCP, and OCI. These chapters are essential for anyone new to cloud forensics, and they provide the vocabulary and framework for everything that follows.

Part 2, Chapters 5 through 13, covers specific forensic domains. Each chapter focuses on a layer of cloud activity: authentication events, compute instances, containers, serverless functions, network telemetry, storage and databases, configuration changes, persistence mechanisms, credential compromise, and privilege escalation. You will learn what evidence to look for, how to interpret it, what false positives to expect, and how to automate collection with Python scripts, CLI commands, and provider APIs. Every code example is production-quality, documented line by line, and designed to be adapted to your environment.

Part 3, Chapters 14 through 18, focuses on synthesis and advanced topics. Chapter 14 walks through five fully developed incident scenarios using synthetic data to show how evidence from multiple sources combines into a coherent investigation. Chapter 15 covers evidence preservation and chain-of-custody. Chapter 16 addresses multi-cloud investigations. Chapter 17 shows you how to build a cloud-native forensics automation platform. Chapter 18 examines emerging challenges, limitations, and future directions.

You should approach this book in one of two ways.

If you are new to cloud forensics, read it sequentially from Chapter 1 onward. The early chapters build concepts you will reference throughout the book.

If you already have experience investigating cloud incidents, treat this as a reference. Jump to the chapters relevant to your immediate needs, but be aware that earlier chapters contain context that will help you understand the provider-specific details later.

Throughout the book, pay attention to the distinctions between what can be reliably inferred from evidence and what remains a hypothesis. A suspicious API call is not proof of compromise; it is a lead that requires investigation. An unusual outbound connection in VPC Flow Logs does not prove data exfiltration; it proves data transfer that requires context. A well-conducted investigation weighs evidence from multiple sources and builds a conclusion that is consistent across all available data. That mindset is as important as any specific tool or technique.

A Note on Scope, Ethics, and the Defensive Posture

This book is written from a defensive, incident-response perspective. Every technique, every script, every investigation workflow is designed to help

defenders understand what has happened in their environment and how to respond. The incident scenarios use synthetic data and realistic attack patterns. They do not provide step-by-step instructions for compromising systems.

The code examples target AWS, Azure, GCP, and OCI APIs for purposes of log retrieval, evidence collection, and forensic analysis. All examples assume you have appropriate authorization to access the resources you are investigating. Running these scripts against environments you do not own or have permission to investigate is both unethical and potentially illegal.

The techniques described here align with standard incident response frameworks including NIST SP 800-61r2, NIST CSF 2.0, and the MITRE ATT&CK framework for IaaS and Cloud platforms. Where specific guidance exists in provider documentation or in authoritative frameworks like the NIST Cloud Computing Forensic Reference Architecture (NIST SP 800-201, published July 2024), this book references it.

If you are using this book to build an incident response capability for your organization, treat it as a starting point. Supplement it with your organization's legal requirements, regulatory obligations, and internal policies. Work with your legal team to ensure that evidence collection procedures satisfy chain-of-custody requirements for your jurisdiction. Test your procedures in a controlled environment before you face a real incident.

The alert you are preparing for will not give you a textbook scenario. It will arrive with incomplete information, ambiguous signals, and time pressure. This book cannot eliminate those challenges, but it can give you the knowledge, the tools, and the confidence to respond when the alert finally comes.

Chapter 1: Foundations of Cloud Forensics

This chapter establishes the conceptual framework for cloud forensics. We will examine the shared responsibility model as a forensic boundary, explore how ephemeral infrastructure changes evidence timelines, classify cloud forensic artifacts by priority and reliability, discuss legal and chain-of-custody considerations, and define the fundamental principles that govern all cloud investigations.

The Shared Responsibility Model as a Forensic Boundary

Every cloud provider operates under a shared responsibility model. In simple terms, the provider secures the infrastructure, and the customer secures everything they put into it. For forensics, this model creates a hard boundary: you can collect and analyze evidence for the layers you control, but the provider retains control over the layers it manages.

At AWS, the model divides responsibility as follows. AWS secures the physical data centers, the hardware, the hypervisor, and the network infrastructure. AWS also secures the control plane for managed services like RDS, DynamoDB, and Lambda. The customer is responsible for operating system configuration on EC2 instances, IAM policies, security group rules, data encryption, application security, and logging configuration. For forensics, this means you can collect logs from CloudTrail, inspect EC2 instance disks, and analyze application logs. You cannot inspect the hypervisor, access another tenant's data, or request raw hardware forensic images from AWS.

Microsoft Azure uses an analogous model. Azure secures the data center, host network, and host operating system. The customer manages the guest operating system, applications, identity, and data. Azure provides audit logs through Activity Logs and diagnostic data through Log Analytics, but the underlying host evidence remains with Microsoft.

Google Cloud Platform follows the same pattern. Google secures the global infrastructure and manages the control plane. The customer manages IAM policies, VPC configurations, application data, and instance configuration. Google provides audit logs and VPC flow logs, but the physical and hypervisor layers are opaque.

Oracle Cloud Infrastructure applies the shared responsibility model similarly. Oracle manages the physical infrastructure and control plane security. Customers manage identity policies, network security lists, data encryption, and instance configurations.

The forensic implications are significant. You must design your investigations around the assumption that you will never see the host hardware, never access another tenant's traffic, and never obtain provider internal logs without a formal legal process. Your evidence comes from three sources: the artifacts you generate and preserve in your environment, the audit and operational logs the provider exposes through APIs and consoles, and the security findings from provider-managed services like GuardDuty, Security Center, or Cloud Armor.

This boundary also affects legal proceedings. If evidence you need resides solely in provider-managed systems, you may need to work through legal channels to obtain it. Subpoenas, court orders, and preservation requests directed at cloud providers follow specific procedures that vary by jurisdiction and provider. Understanding the shared responsibility model tells you when you can collect evidence yourself and when you must engage legal processes.

Ephemeral Infrastructure and the Race Against Time

Traditional forensics had a significant advantage: the evidence stayed put. A compromised server sat in a rack until investigators arrived, disconnected it from the network, and imaged its disk. Cloud forensics has the opposite characteristic: evidence disappears constantly.

Consider a compromised container in a Kubernetes cluster. The container runs for a few hours, exfiltrates data, and then terminates when the pod is rescheduled or when resource limits are reached. If you did not capture the running container's memory or filesystem changes before it terminated, the evidence inside the container is gone. Consider a serverless function that an attacker modifies to exfiltrate data. The function executes, sends data to an external endpoint, and then sleeps. There is no persistent host to image.

The only evidence is in CloudTrail logs, function execution logs, and network flow logs, all of which may be overwritten or expired if retention policies are misconfigured.

Even virtual machines are ephemeral in the cloud. Auto-scaling groups replace unhealthy instances automatically. Instance store volumes lose data when the instance stops. Security groups can be modified to block your investigative access. IAM credentials can be revoked, disabling your ability to query APIs. An attacker who realizes they are being investigated can accelerate all of these destruction vectors.

This creates a race condition. The moment you detect an incident, you must begin collecting evidence before it is lost. The speed of collection depends on how well you prepared. Organizations that have automated evidence collection, centralized log storage, and documented playbooks can preserve critical evidence within minutes. Organizations that discover an incident and then scramble to enable logging may find that hours of evidence have already expired.

The lesson is straightforward: treat logging and evidence collection as a runtime requirement, not a post-incident consideration. If your incident response plan requires you to enable CloudTrail or activate VPC flow logs after a compromise is discovered, your plan is broken. Those services must be enabled before the incident occurs.

Cloud Forensic Artifacts: Classification and Priority

Cloud forensic artifacts fall into several categories based on where they are generated, how they are preserved, and how much investigative value they provide. Understanding these categories helps you prioritize evidence collection during an incident.

The highest-value artifacts are identity and authentication logs. These logs record who signed in, from where, with what credentials, and what actions they performed. AWS CloudTrail records every API call along with the IAM principal that made it. Azure Activity Logs and Entra ID sign-in logs serve the same function. GCP Audit Logs capture API calls and identity information. OCI Audit logs track all API activity. Because cloud operations are identity-based, these logs answer the most important forensic questions: which identity performed the action and when.

Second in priority are control-plane audit logs. These logs record configuration changes: new IAM users, modified security groups, changed network rules, created instances, altered policies. They establish a timeline of what changed in the environment and help investigators identify attacker actions like privilege escalation, persistence establishment, or log tampering.

Third are data-plane logs. These logs record actual data access: reads from an S3 bucket, queries to a database, invocations of a Lambda function, or API calls to a specific service. Data-plane logs are critical for determining the scope of data exposure but are often disabled by default because they generate high volume and cost. Organizations must decide which data-plane logs to enable based on risk and budget before an incident occurs.

Fourth are network telemetry logs. These include VPC Flow Logs on AWS, NSG Flow Logs on Azure, VPC Flow Logs on GCP, and VCN Flow Logs on OCI. They record IP traffic metadata: source and destination addresses, ports, protocols, packet counts, and byte counts. Network logs are essential for detecting lateral movement, command-and-control communication, and data exfiltration. Unlike audit logs, flow logs do not identify the user or service that generated the traffic; they must be correlated with other evidence.

Fifth are host-level artifacts. These include disk snapshots, memory dumps, process listings, and operating system logs from compromised instances. Host artifacts provide the most detailed evidence of what an attacker did on a specific machine, but they are often the most difficult to collect in the cloud because they require agent installation or live acquisition techniques.

Sixth are managed service artifacts. These include security findings from services like AWS GuardDuty, Azure Defender, GCP Security Command Center, and OCI Cloud Guard. These services use machine learning and threat intelligence to flag suspicious activity. They are not raw evidence, but they are often the trigger for an investigation and can point investigators toward relevant audit logs and network data.

Finally, there are configuration artifacts. These include IAM policy documents, network security group rules, resource configurations, and infrastructure-as-code templates. Configuration artifacts help investigators understand the environment's security posture at the time of the incident and identify misconfigurations that enabled the attack.

Not all artifacts are equally reliable. Identity and audit logs generated by the provider's control plane are highly reliable because they are immutable

and recorded independently of customer systems. Host-level artifacts are less reliable because an attacker may have tampered with them. Managed service findings are probabilistic and must be validated against raw logs. Every investigator must weigh the reliability of each evidence source when building a case.

Legal, Compliance, and Chain-of-Custody Considerations

Cloud forensics does not exist in a legal vacuum. Depending on your jurisdiction and the nature of the incident, evidence you collect may be used in criminal prosecutions, civil litigation, regulatory proceedings, or internal disciplinary actions. Each context has different requirements for evidence handling.

The foundational legal concepts are authenticity, integrity, and completeness. Authenticity means you can demonstrate that the evidence originated from the system it claims to represent. Integrity means the evidence has not been altered since collection. Completeness means you have collected all relevant evidence and not selectively excluded data that might contradict your findings.

In cloud environments, establishing these properties requires deliberate procedures. When you collect a CloudTrail log file from an S3 bucket, you must preserve the original file and compute its cryptographic hash. When you create an EBS snapshot for forensic analysis, you must document the snapshot creation time, the instance it was taken from, and the identity that created it. When you query a Log Analytics workspace, you must record the exact query, the time range, and the results.

Chain-of-custody documentation must track every action performed on evidence from collection through analysis to storage. In traditional forensics, this meant physical seals, evidence bags, and custody logs. In cloud forensics, chain of custody lives in audit trails, access logs, and cryptographic hashes. Every person who accesses evidence must be identified. Every copy of evidence must be tracked. Every analysis performed on evidence must be documented and reproducible.

Regulatory requirements add complexity. GDPR imposes restrictions on personal data processing and cross-border data transfers. HIPAA governs protected health information in healthcare organizations. PCI-DSS regulates payment card data. SOX requirements apply to financial reporting systems.

Each regulation may impose specific requirements for log retention, evidence preservation, and data handling that affect your forensic procedures.

Cross-border considerations are particularly important in cloud forensics. Cloud providers distribute data across regions globally. Evidence relevant to your investigation may reside in a jurisdiction with different privacy laws, data sovereignty requirements, or legal assistance processes. Organizations with multi-region or multi-cloud deployments must understand where their data resides and what legal frameworks govern it.

Working with law enforcement introduces additional procedures. Law enforcement agencies can issue preservation requests, subpoenas, search warrants, and court orders to cloud providers. Providers have dedicated legal response teams and specific procedures for processing these requests. Organizations should coordinate with legal counsel before responding to law enforcement inquiries to ensure that evidence preservation obligations are met and that ongoing incident response efforts are not compromised.

NIST IR 8006 and NIST SP 800-201 provide detailed guidance on the legal and procedural challenges of cloud forensics. These documents identify key stakeholder roles, jurisdictional issues, and technical challenges that investigators must navigate. Familiarity with these frameworks strengthens the defensibility of your forensic procedures.

The Three Laws of Cloud Evidence Collection

After years of cloud incident investigations, three principles have emerged that are as fundamental to cloud forensics as Locard's Exchange Principle is to traditional crime scene investigation. These principles are not legal statutes, but they capture lessons learned from real incidents where evidence was lost, contaminated, or rendered inadmissible because basic rules were not followed.

First law: preserve before you modify. In cloud environments, it is tempting to immediately terminate a compromised instance, delete a malicious Lambda function, or revoke an attacker's credentials. These actions are appropriate for containment, but they should follow evidence preservation, not precede it. Taking a disk snapshot before terminating an instance preserves the disk evidence. Enabling CloudWatch Logs or exporting existing logs before modifying logging configurations preserves historical data. Reversing a security group change before restoring proper access rules preserves evidence of the

attacker's network access. If you modify evidence before you preserve it, you may lose the ability to reconstruct what happened.

The key insight is that preservation must be automated and immediate. In practice, this means having scripts or runbooks that capture evidence in the first minutes of incident detection. A well-designed incident response automation framework takes snapshots, exports logs, and revokes credentials in a controlled sequence that preserves evidence while limiting further damage.

Second law: collect from the outside in. When investigating a cloud incident, begin with the evidence sources that are most independent of the compromised system. Provider audit logs, identity logs, and network flow logs are generated outside the compromised instance and are less likely to have been tampered with. Host-level logs, file system artifacts, and process listings are more likely to be modified by an attacker. Build your investigation timeline from the independent sources first, then use host-level artifacts to fill in details and confirm hypotheses. If host-level evidence contradicts independent evidence, trust the independent evidence and investigate why the host evidence is unreliable.

Third law: assume logs have been tampered with. Sophisticated attackers know where cloud audit logs live and how to disable or delete them. An attacker with IAM permissions can delete CloudTrail trails, disable VPC flow log configurations, or modify Log Analytics data retention policies. An attacker in Azure can delete Activity Log diagnostic settings or purge Log Analytics workspaces. An attacker in GCP can disable Data Access audit logs or delete log sinks.

Every cloud forensics investigation must begin by checking whether logging was disabled or evidence was deleted during the incident window. CloudTrail logs itself: if someone deletes a CloudTrail trail, that deletion appears in the event history. Azure Activity Logs records changes to logging configurations. GCP Audit Logs records log sink modifications. Look for these events early in the investigation. Their presence is often the clearest indicator of a sophisticated attacker.

When log tampering is detected, the investigation shifts to recovering deleted or missing evidence. Provider services may have backup copies. Cross-account logging may have captured events that were deleted locally. SIEM systems may have ingested logs before they were deleted. Third-party monitoring tools may have their own event records. The more redundant

logging channels you maintain, the harder it is for an attacker to erase all evidence.

These three laws form the core methodology for cloud evidence collection. Every chapter in this book builds on them. The scripts, the queries, the workflows, and the investigation procedures all assume that you preserve before you modify, collect from the outside in, and verify logging integrity early and often.

Understanding these foundations is essential before we move to the provider-specific details. The next chapter examines how cloud architecture affects forensic visibility and what evidence is available in each layer of the cloud stack.

Chapter 2: Cloud Architecture and Forensic Visibility

Cloud architecture determines what evidence you can see and where it lives. This chapter examines how tenancy models affect visibility, maps the visibility spectrum across compute, network, storage, and database layers, explores the forensic challenges of managed services, shows how to design logging and monitoring architectures for forensic readiness, and addresses the complications of regional boundaries and cross-account access.

Public Cloud Tenancy Models and What You Can See

When you deploy workloads in a public cloud, your resources run on shared physical infrastructure. Multiple customers' virtual machines may execute on the same physical server. Your network traffic may traverse the same switches and routers as another customer's traffic. Your data may reside on the same storage arrays as data belonging to other organizations.

This multi-tenancy has direct implications for forensics. You can see your resources and your traffic. You cannot see resources belonging to other customers, and you cannot see the shared infrastructure that connects them.

At the hypervisor layer, the situation is clear. AWS Nitro, Azure Hyper-V based hosts, Google Compute Engine hypervisors, and OCI compute hypervisors are all managed exclusively by the provider. They enforce isolation between tenants. You can request forensic evidence from the hypervisor only through a formal legal process directed at the provider, and even then, providers may not be able to provide the evidence without risking tenant confidentiality.

At the network layer, you have visibility into traffic to and from your resources but not traffic between other tenants. VPC flow logs on AWS, NSG flow logs on Azure, VPC flow logs on GCP, and VCN flow logs on OCI capture traffic involving your virtual network interfaces. They do not capture traffic between other customers' resources. If an attacker uses another customer's

compromised instance to scan your environment, you will see the inbound traffic in your flow logs, but you will not see the attacker's reconnaissance activity from the other customer's perspective.

At the storage layer, you can inspect your own disks and buckets but not shared storage infrastructure. An EBS volume is dedicated to your AWS account. An Azure managed disk belongs to your subscription. A GCP persistent disk is associated with your project. OCI block volumes are scoped to your tenancy. Within your account or subscription, you can snapshot disks, mount them to forensic instances, and analyze them with standard tools. Beyond your boundaries, you cannot.

The shared responsibility model extends to control-plane services as well. In managed services like AWS Lambda, Azure Functions, or GCP Cloud Run, the provider manages the underlying runtime and infrastructure. You deploy code and configuration, but you do not have access to the operating system or host processes. Your forensic evidence is limited to logs, execution records, and configuration snapshots. You cannot extract runtime memory or inspect process listings from a Lambda function the way you would from an EC2 instance.

Understanding these boundaries is essential for setting realistic expectations. A cloud forensics investigation cannot answer every question. It can only answer the questions that your logging, your permissions, and your provider's capabilities allow you to investigate.

Compute, Network, Storage, and Database Layers: Visibility Spectrum

Cloud infrastructure can be organized into four layers for forensic purposes: compute, network, storage, and database. Each layer generates different types of evidence and offers different levels of visibility.

The compute layer includes virtual machines and containers. On AWS, this is primarily EC2 and Elastic Kubernetes Service. On Azure, it is Virtual Machines and AKS. On GCP, it is Compute Engine and GKE. On OCI, it is Compute instances and OKE.

Compute-layer visibility is the richest in the cloud. You can access the guest operating system, collect process listings, examine running services, review

operating system logs, and inspect file systems. For forensic purposes, the standard approach is to create a snapshot of the instance's disk volume, attach the snapshot to a forensic analysis instance, and examine the disk without modifying the original. Memory can be captured from a running instance using tools deployed through Systems Manager or similar remote execution services. These techniques mirror traditional host forensics but must be executed through cloud APIs.

The network layer includes virtual networks, load balancers, firewalls, and traffic flow logs. Network-layer visibility is critical for understanding how an attacker moved through your environment, communicated with external infrastructure, and transferred data.

Flow logs provide the most important network-layer evidence. AWS VPC Flow Logs record IP traffic at the subnet or network interface level. They capture source and destination IP addresses, source and destination ports, protocols, packet counts, byte counts, and whether traffic was accepted or rejected by security groups or network ACLs. Azure NSG Flow Logs provide similar visibility. GCP VPC Flow Logs and OCI VCN Flow Logs offer equivalent capabilities.

Flow logs have limitations. They do not capture application-layer data such as HTTP headers or payload content. They do not identify the user or process that generated the traffic. They must be correlated with audit logs to establish identity. They must be enabled before traffic occurs; you cannot retroactively enable them for past traffic. And they generate significant volume; organizations must configure sampling or filtering policies to manage costs while maintaining forensic utility.

Load balancer access logs provide another network evidence source. AWS Application Load Balancers, Azure Load Balancers, GCP Cloud Load Balancing, and OCI Load Balancers all generate access logs that record client IP addresses, request paths, response codes, and response times. These logs are valuable for web application incident investigations.

The storage layer includes object storage, block storage, and file storage. AWS provides S3, EBS, and EFS. Azure provides Blob Storage, Managed Disks, and Files. GCP provides Cloud Storage, Persistent Disks, and Filestore. OCI provides Object Storage, Block Volumes, and File Systems.

Storage-layer visibility is excellent for detecting data access and exfiltration. S3 server access logs and CloudTrail data events record every object-level

operation on a bucket. Azure Blob Storage diagnostic logs track read, write, and delete operations. GCP Cloud Storage audit logs record data access events. OCI Object Storage logging tracks operations through the Audit service.

Storage logs are essential for answering the question that matters most to organizations: did the attacker access or copy sensitive data. By analyzing storage logs, investigators can identify which objects were read, by which identity, at what time, and from which IP address. They can determine whether bulk download patterns consistent with data exfiltration occurred.

The database layer includes managed databases and data warehouses. AWS offers RDS, DynamoDB, Aurora, and Redshift. Azure provides SQL Database, Cosmos DB, and Synapse. GCP provides Cloud SQL, Firestore, BigQuery, and AlloyDB. OCI offers Autonomous Database, MySQL, PostgreSQL, and Object Storage based solutions.

Database-layer visibility is the weakest link in many cloud environments. Managed databases abstract away the underlying infrastructure, which means you cannot access the host file system, examine database log files directly, or inspect memory. Your evidence comes from the provider's audit logs and query logs.

AWS RDS provides audit logs through the service-specific audit logging features. Azure SQL Database provides query store and audit logs through Azure Monitor. GCP Cloud SQL provides audit logs through Cloud Audit Logs. OCI Autonomous Database provides audit trails.

These database logs are essential for detecting unauthorized data access, bulk queries consistent with exfiltration, and database schema modifications that may indicate persistence mechanisms. However, their coverage depends on configuration. Organizations must enable database-specific audit logging before incidents occur.

Managed Services and the Forensic Black Box Problem

The more managed a service is, the less forensic visibility you have. This is an inherent trade-off in cloud architecture. Managed services reduce operational overhead by handling infrastructure management, patching, scaling, and security on your behalf. They also remove your ability to access the underlying system for forensic investigation.

Consider AWS Lambda. A Lambda function executes in a managed runtime environment. You deploy code and configuration. AWS manages the execution environment, scales it automatically, and isolates it from other functions. When a security incident involves a Lambda function, you cannot SSH into the runtime, examine running processes, or inspect memory. Your evidence is limited to CloudWatch Logs generated by the function, CloudTrail invocation logs, and the function's configuration.

The same limitation applies to Azure Functions, GCP Cloud Functions, and OCI Functions. All of these serverless platforms operate within managed environments that do not expose host-level access.

Consider AWS RDS. An RDS instance runs a database engine on managed infrastructure. You cannot access the operating system. You cannot run system commands on the host. Your evidence comes from database-specific audit logs, CloudTrail management events, and VPC Flow Logs. If an attacker exploits a database vulnerability and installs a backdoor at the operating system level, you may not be able to detect it directly.

Consider GCP Cloud Run or Azure Container Instances. These managed container platforms run containers without exposing node-level access. Forensic evidence is limited to container logs, execution records, and configuration data.

The forensic black box problem forces a change in investigative approach. For managed services, you must rely on API audit logs and network logs to reconstruct attacker activity. These logs are often sufficient to determine what an attacker did even without host access, but they lack the detail and context that host-level evidence provides.

Organizations must accept this trade-off. The operational benefits of managed services outweigh the forensic limitations in most cases. But the trade-off requires that logging be configured comprehensively. If a managed service does not expose host access, its audit logs become your only source of evidence, and incomplete logging becomes an unacceptable risk.

Architecting for Forensics: Logging, Monitoring, and Evidence Preservation by Design

The most important factor in cloud forensic success is not tools, techniques, or even experience. It is logging. An organization with comprehensive logging

can investigate virtually any incident. An organization without logging is blind, regardless of its forensic capabilities.

Architecting for forensics means designing your logging, monitoring, and evidence preservation infrastructure before an incident occurs. This design has several components.

First, centralize logs in a dedicated logging account or subscription. AWS best practices recommend creating a centralized security and logging account in your AWS Organization and configuring all member accounts to send CloudTrail logs, VPC Flow Logs, and GuardDuty findings to this account. Azure recommends centralizing Log Analytics workspaces. GCP recommends using a shared logging project. OCI recommends a centralized logging tenancy structure.

Centralization serves three purposes. It protects logs from attackers who compromise a single account. If an attacker disables logging in a compromised AWS account, logs that were sent to the centralized account remain intact. It simplifies analysis by providing a single location to search for evidence. It enables cross-account correlation.

Second, implement immutability for log storage. Logs that can be deleted or modified are not reliable evidence. AWS S3 Object Lock with compliance mode prevents deletion for a specified retention period. Azure Blob Storage immutable storage with WORM policies provides equivalent protection. GCP Cloud Storage object retention policies achieve the same goal. OCI Object Storage retention policies enforce retention periods.

Configure log storage buckets with immutability and ensure that no user, including administrators, can delete or modify logs before the retention period expires. This protects logs from both attackers and negligent administrators.

Third, enable all relevant logging by default. Management audit logs should be enabled for every account, subscription, project, or tenancy. Network flow logs should be enabled for all production networks. Data-plane logs should be enabled for sensitive services. Security service logging should be activated wherever available.

Fourth, preserve evidence automatically. Configure automated workflows that create snapshots of compromised instances, export logs to external storage, and revoke compromised credentials. AWS Step Functions, Azure Logic Apps, GCP Workflows, and OCI Functions can orchestrate these workflows.

Fifth, separate forensic analysis from production environments. Forensic investigations should occur in isolated environments that do not share IAM credentials, network access, or data stores with production. AWS recommends a dedicated forensics account. Azure recommends a separate security subscription. GCP recommends a dedicated analysis project. OCI recommends compartment isolation.

This separation prevents contamination of evidence, protects production systems from accidental disruption, and ensures that forensic investigators operate with the minimum necessary access.

Regional Boundaries, Cross-Account Access, and Evidence Scoping

Cloud environments are global. Resources span regions, accounts, subscriptions, and tenancies. Evidence about a single incident may be distributed across this global topology, and investigators must know how to scope their search.

Regional boundaries matter for logging. AWS CloudTrail is region-aware; each region logs API calls for that region separately. However, CloudTrail trails can be configured as organization trails that aggregate logs across all accounts and regions. Azure Activity Logs are subscription-level but can be centralized through Log Analytics. GCP Audit Logs are project-level but can be aggregated through log sinks to a central project. OCI Audit logs are tenancy-wide.

Investigators must ensure that their search includes all regions where the affected identity had access. An attacker who compromises an IAM user in us-east-1 may pivot to resources in eu-west-1 or ap-southeast-1. An attacker who compromises an Azure subscription may access resources in other Azure regions through role assignments. An attacker who compromises a GCP service account may operate across any project to which it is granted access.

Cross-account access complicates evidence scoping. AWS supports cross-account role assumption through IAM role trust policies. An attacker who assumes a cross-account role appears in the destination account's CloudTrail logs as a different principal than the original compromised identity. Azure supports cross-subscription access through Entra ID and role assignments. GCP supports cross-project access through service account impersonation

and IAM bindings. OCI supports cross-compartment access through IAM policies.

To trace an attacker across account boundaries, investigators must correlate identity information across logs. AWS CloudTrail records session issuer information for assumed roles. Azure Activity Logs record the principal ID and authentication context. GCP Audit Logs record the principal email and delegated identity information. OCI Audit logs record the identity domain and user information.

Evidence scoping must begin with the compromised identity and follow that identity's access path through the environment. Query audit logs in all accounts, subscriptions, projects, or compartments where the identity had permissions. Check for role assumption events, cross-account API calls, and resource creation in unexpected environments.

The broader the environment, the more thorough the scoping must be. A comprehensive incident investigation treats cross-boundary access as a normal part of cloud operations, not as an edge case.

Chapter 3: Identity and Access Management as the Forensic Center

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

IAM Fundamentals: Users, Roles, Groups, and Policies

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Mapping Identity Across AWS IAM, Azure AD/Entra ID, GCP IAM, and OCI IAM

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Authentication Events: Sign-Ins, MFA, and Anomalous Patterns

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Authorization Decisions: Why Did This User or Service Have Access?

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Cross-Account, Cross-Tenant, and Federated Identity Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Parsing and Analyzing IAM Activity Logs

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Chapter 4: Cloud Audit and Activity Logs

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

AWS CloudTrail: Logs, Insights, and Data Events

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Azure Activity Logs and Diagnostic Settings

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Google Cloud Audit Logs and Organization Policy

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

OCI Audit Service and Events

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Cross-Provider Comparison: What Each Logs and What It Misses

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Building a Unified Audit Log Search Tool

Chapter 5: Authentication and Sign-In Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Sign-In Events: What to Look for in Authentication Logs

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Impossible Travel, Geolocation Anomalies, and Velocity Analysis

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Token Theft, Session Hijacking, and Replay Attacks

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Key-Based Authentication and Access Key Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

MFA Bypass, Prompt Fatigue, and Consent Attacks

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Detecting Suspicious Sign-In Patterns Across Providers

Chapter 6: Compute Instance Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Instance Metadata and the Initial Reconnaissance Trail

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Snapshot Forensics: Capturing Disks Before and During Incidents

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Memory Acquisition in the Cloud: Tools, Timing, and Limitations

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Process and Command-Line Artifacts

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Instance Creation and Configuration as Evidence

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Automated Instance Evidence Collection Script

Chapter 7: Container and Kubernetes Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Container Lifecycle and Forensic Implications

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Image Layer Forensics and Registry Investigation

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Runtime Forensics: Processes, Networks, and Volumes in Containers

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Kubernetes API Server Logs and Audit Trail

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

EKS, AKS, GKE, and OKE: Managed Kubernetes Forensic Differences

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Container and Kubernetes Forensic Analysis Tool

Chapter 8: Serverless Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Serverless Architecture and Where Evidence Lives

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

AWS Lambda Forensics: Execution Logs and Invocation Patterns

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Azure Functions and GCP Cloud Functions Investigation

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Injection and Function Misconfiguration as Attack Vectors

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Timing Attacks, Trigger Manipulation, and Cold-Start Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Serverless Function Log Analysis Pipeline

Chapter 9: Network Forensics and Telemetry

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Cloud Network Models: VPC, VNet, VPC, and Virtual Cloud Network

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Flow Logs: AWS VPC Flow Logs, Azure NSG Flow Logs, GCP Flow Logs, OCI Flow Logs

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

DNS Query Logs and Resolution Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Load Balancer, CDN, and Gateway Logs

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Network-Level Indicators: Port Scanning, Beaconing, and Data Egress

Code Example: Network Flow Log Analysis and Visualization

Chapter 10: Storage and Data Access Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Object Storage Forensics: S3, Blob Storage, Cloud Storage, and Object Storage

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Access Patterns, Policy Changes, and Public Exposure Events

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Database Forensics: RDS, Azure SQL, Cloud SQL, and Autonomous Database

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Key Management Service Forensics: KMS, Azure Key Vault, Cloud KMS, OCI Vault

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Detecting Data Exfiltration: Volume, Timing, and Destination Analysis

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Storage Access Log Analysis for Data Loss Indicators

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Chapter 11: Configuration Change and Infrastructure Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Infrastructure as Code and Change Management as Evidence

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Security Group, Network Security List, and Firewall Rule Modifications

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

New IAM Roles, Users, and Permission Escalation Events

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Unusual Resource Creation: Instances, Buckets, and Functions

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Detecting and Tracking Configuration Drift

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Infrastructure Change Detection and Alerting

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Chapter 12: Persistence and Defense Evasion in the Cloud

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Cloud-Specific Persistence Mechanisms

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Log Tampering, Deletion, and Suppression

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Cloud API Proxying and C2 Infrastructure

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Compromised CI/CD Pipelines and Supply Chain Attacks

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Resource Misuse: Cryptominers, Botnets, and Spam Infrastructure

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Persistence and Defense Evasion Detection Script

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Chapter 13: Compromised Credentials and Privilege Escalation

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Credential Acquisition: Phishing, Key Exposure, and Metadata Service Abuse

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Analyzing Compromised Access Key Usage Patterns

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Role Assumption and Privilege Escalation Paths

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Policy Misconfigurations That Enable Escalation

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Containment Strategies and Impact Assessment

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Credential Abuse and Privilege Escalation Analysis

Chapter 14: Incident Scenarios: Walkthrough Investigations

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Scenario 1: Compromised Developer Credentials on AWS

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Scenario 2: Lateral Movement via Azure Role Elevation

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Scenario 3: Data Exfiltration from GCP Cloud Storage

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Scenario 4: Multi-Cloud Persistence and Command-and-Control

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Scenario 5: Supply Chain Attack via CI/CD Pipeline Compromise

Chapter 15: Evidence Preservation and Chain of Custody

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Preservation Principles: Integrity, Authenticity, and Reproducibility

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Snapshotting, Exporting, and Archiving Evidence

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Hashing and Verification in Cloud Environments

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Chain of Custody: Documentation, Access Controls, and Legal Requirements

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Working with Providers: Preservation Requests and Legal Holds

Code Example: Evidence Preservation and Verification Toolkit

Chapter 16: Multi-Cloud and Cross-Provider Forensics

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Multi-Cloud Architecture and Attack Surface Considerations

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Unified Log Aggregation and Cross-Provider Correlation

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Federated Identity and Cross-Cloud Access Patterns

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Data Replication, Synchronization, and Evidence Fragmentation

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Building a Multi-Cloud Investigation Playbook

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Multi-Cloud Log Correlation Engine

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Chapter 17: Automation, Tooling, and Building a Cloud Forensics Platform

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Forensics Automation Philosophy: When to Automate and What to Trust

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Building a Centralized Log Lake for Forensic Investigations

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

SIEM Integration: Splunk, Sentinel, Chronicle, and Open-Source Alternatives

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Scripting Evidence Collection: Python, Bash, and Provider CLIs

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Building Custom Detection Rules and Alerting Pipelines

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Code Example: Complete Cloud Forensics Automation Framework

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Chapter 18: Advanced Topics and Future Directions

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Confidential Computing and Forensic Limitations

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

AI and Machine Learning in Cloud Incident Investigation

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Adversarial Adaptation: Attackers Evolving Around Cloud Defenses

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Regulatory Changes, Cross-Border Evidence, and Compliance

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Where Cloud Forensics Is Headed: Trends and Recommendations

Conclusion: Preparing for the Next Incident

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

The Forensic Readiness Checklist

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Building Organizational Muscle Memory

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

Key Takeaways and Recommended Next Steps

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.

References

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/forensicsinthecloud.