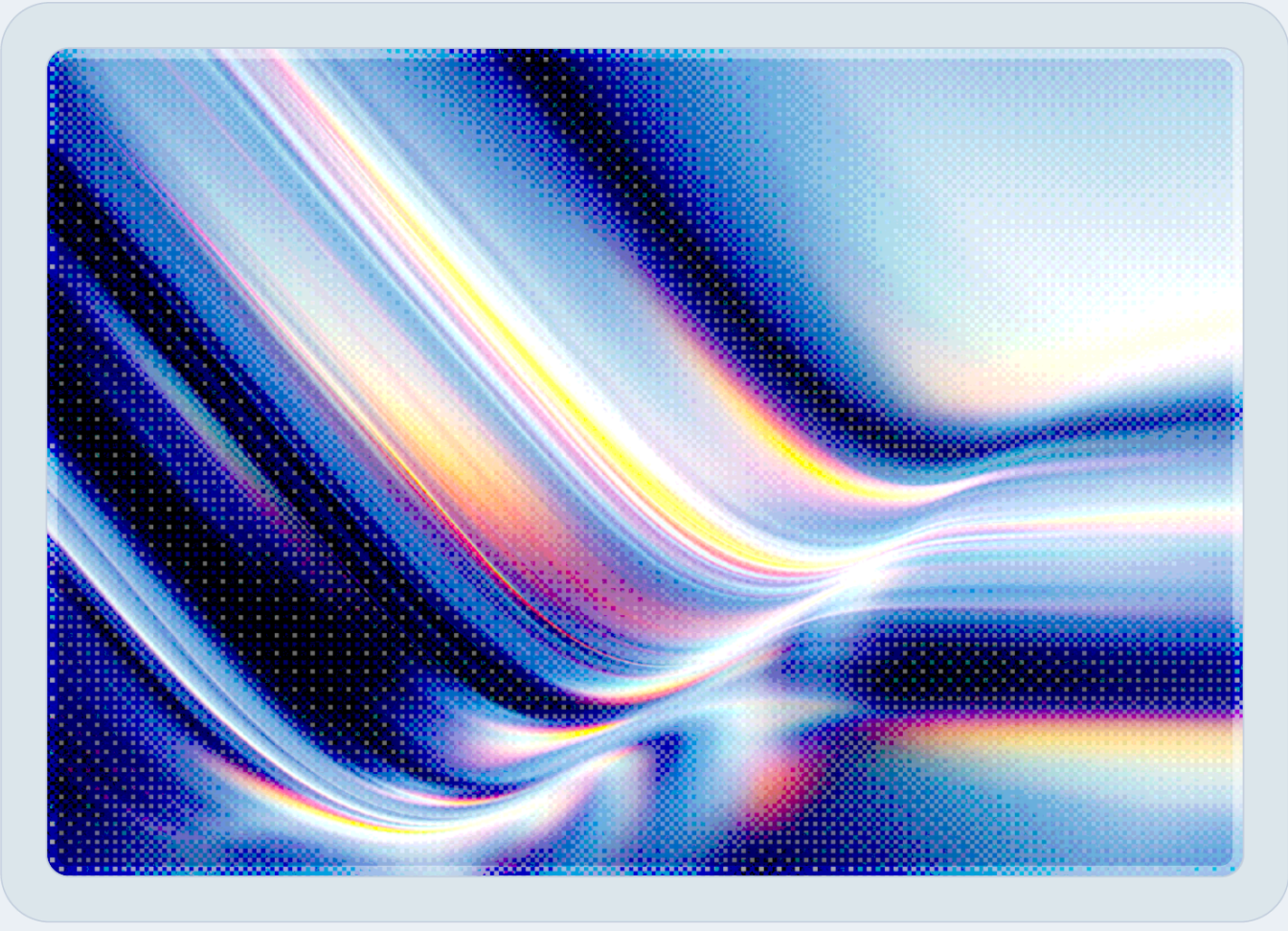
CSAI Foundation | Cloud Security Alliance

Blinding the Watchmen: Cloud Logging as an Attack Surface

How Adversaries Exploit Audit Log Gaps for Systematic Defense Evasion

2026-06-10

AI-assisted Rapid Research



CSAI

cloud security alliance®

Key Takeaways

Cloud audit logs are not passive records — they are the primary evidence source for incident detection, forensic investigation, and regulatory compliance. Adversaries have recognized this, and disabling or manipulating cloud logging services has become a documented, standardized phase of cloud intrusions rather than an ad hoc evasion measure.

- MITRE ATT&CK catalogues cloud log manipulation under T1562.008 (Impair Defenses: Disable or Modify Cloud Logs), updated in October 2025, with specific sub-techniques for AWS CloudTrail, GCP Audit Logs, and Azure Monitor diagnostic settings [1].
- The BRICKSTORM/UNC5221 espionage campaign maintained undetected access in compromised networks for 393 days [2], a dwell time that structurally exceeds the 90-day log retention window common in enterprise environments — creating a forensic blind spot even after discovery.
- Sysdig's investigation of the SCARLETEEL cloud operation documented CloudTrail disablement as a deliberate in-campaign action, used to degrade visibility during active exploitation [3].
- Datadog Security Labs identified an undocumented AWS API service called `iamadmin` that allowed read-only IAM enumeration actions — including `ListAccessKeys` and `GetRole` — to execute without generating CloudTrail entries, a blind spot that also suppressed corresponding GuardDuty alerts [4].
- Google Cloud's threat research documented that ransomware actors frequently deleted logs, core dumps, and backups before issuing ransom demands, reducing victim organizations' ability to recover independently or establish the full scope of exposure [5].
- CISA Binding Operational Directive 25-01, issued December 2024, requires federal agencies to implement SCuBA cloud configuration baselines, including continuous audit logging controls — reflecting government recognition of logging integrity as a foundational security control [6].

Background

Cloud audit logging occupies a unique position in enterprise security architecture: it is simultaneously the primary mechanism for detecting attacks in progress and the record required to reconstruct them after the fact. Services such as AWS CloudTrail, Google Cloud Audit Logs, and Azure Monitor Diagnostic Settings capture API activity, authentication events, configuration changes, and data access across cloud environments. Security information and event management platforms, cloud-native detection tools like Amazon GuardDuty and Microsoft Defender for Cloud, and third-party security operations platforms all depend on these log streams as their foundational data source. When those streams are altered or interrupted, detection capabilities degrade silently — the security tooling continues to function, but it operates on an incomplete or falsified picture of the environment.

This dependency creates an obvious adversarial opportunity. An attacker who disables cloud audit logging before conducting lateral movement, privilege escalation, or data exfiltration may effectively operate outside the cloud audit detection layer, unless independent telemetry sources — endpoint detection, network monitoring, identity platform logs — are also in place. Unlike disabling host-based endpoint detection, which typically generates an alert on the endpoint itself or on adjacent systems, disabling a centralized cloud logging service can be accomplished with a single API call, requires no malware, and leaves no trace in the very log store it eliminates. The action may surface in billing records or through out-of-band controls — but only if those controls are independently configured and monitored.

The technique predates the cloud era. Attackers have long targeted log files and audit mechanisms to conceal their activity. What distinguishes the cloud context is scale and architecture: a single CloudTrail trail or GCP logging sink may aggregate audit records across thousands of resources spanning multiple accounts or projects. Disabling it at the service configuration level affects visibility holistically, in a way that disabling individual host logs does not. The centralization that makes cloud audit logging operationally convenient also makes it a high-value target for disruption.

Named intrusion sets operating across the technology, legal, and critical infrastructure sectors have incorporated logging manipulation into documented kill chains. Ransomware operators have elevated it to a pre-extortion procedure. Structural gaps — from undocumented API surfaces to log retention policies shorter than observed attacker dwell times — mean that even organizations that have not been directly targeted may already have blind spots that would prevent them from knowing.

Security Analysis

Platform-Specific Techniques: AWS, GCP, and Azure

Each major cloud provider exposes logging configuration through API endpoints that, if abused, allow an attacker to suppress or corrupt audit records. MITRE ATT&CK T1562.008 documents the primary mechanisms as of its October 2025 revision [1], and incident response investigations have confirmed their use in real campaigns.

On AWS, three particularly consequential operations are `StopLogging` (which suspends a CloudTrail trail while leaving its configuration intact), `DeleteTrail` (which removes the trail entirely), and `UpdateTrail` combined with `PutEventSelectors` (which silently filters out management or data events while the trail continues to appear active and healthy). The last technique is particularly dangerous from a detection standpoint: a trail that selectively excludes high-sensitivity events — such as IAM role assumption, S3 `GetObject` calls, or KMS decryption operations — will pass routine monitoring checks that verify only whether a trail exists and is logging. The apparent continuity masks substantive gaps.

On Google Cloud Platform, logging sink configurations and Pub/Sub topics represent an exposed attack surface. The GCP configuration API operation `google.logging.v2.ConfigServiceV2.UpdateSink` can redirect or suppress log exports to external monitoring tools without halting audit log generation at the source [1][7], introducing gaps in SIEM ingestion while maintaining a false appearance of normal logging activity. The difficulty of detecting these changes compounds the risk: the modifications themselves should be captured by audit logs, but only if monitoring for audit configuration changes is independently configured and actively reviewed.

On Microsoft Azure, MITRE ATT&CK T1562.008 documents platform-specific techniques including deletion of Azure diagnostic settings via `MICROSOFT.INSIGHTS/DIAGNOSTICSETTINGS/DELETE` and disruption of log forwarding pipelines through Event Hub and Network Watcher modifications [1]. These operations are recorded in Azure Activity Logs — but only if those logs are themselves being monitored, which creates a circular dependency that adversaries can exploit by targeting the monitoring configuration before proceeding to operational goals.

How Log Manipulation Degrades Downstream Detection

Cloud audit logging and the security tooling that depends on it are tightly coupled in ways that have direct consequences when logging streams are disrupted. GuardDuty and Defender for Cloud rely substantially on CloudTrail and equivalent event streams for their detection coverage; coverage gaps in those streams propagate directly into detection gaps for the affected event categories. When the underlying log source is disabled, suppressed, or filtered, the detection service continues to run without generating alerts — because from its perspective, nothing is happening. The absence of malicious events in the log stream is indistinguishable from the absence of malicious activity.

Datadog Security Labs documented this dependency in a January 2023 disclosure covering an undocumented AWS API service, `iamadmin`, which allowed read-only IAM enumeration actions — including `ListGroupPolicies`, `ListAccessKeys`, and `GetRole` — to execute without generating CloudTrail entries [4]. Because GuardDuty's cloud enumeration detections rely on CloudTrail as their data source for these event categories, these actions also bypassed GuardDuty alerting. The vulnerability, patched by AWS in October 2022 and disclosed publicly in January 2023, illustrated a structural class of risk: undocumented or non-obvious API surfaces that fall outside the logging coverage model an organization believes to be complete. AWS has since addressed this specific gap, but the incident established that CloudTrail coverage cannot be assumed to be exhaustive without independent verification.

Documented Threat Actor Campaigns

The SCARLETEEL campaign, investigated and published by Sysdig in 2023, provides one of the clearest documented examples of cloud logging manipulation in an active intrusion [3]. The threat actor targeted a cloud-native application environment and disabled CloudTrail logs in the compromised AWS account as a deliberate step in the attack chain, specifically to degrade forensic visibility during the investigation period. The campaign also involved theft of proprietary software, credential harvesting via Terraform state files, and lateral movement across AWS accounts — a scope that would have been far harder to reconstruct had logging been suppressed earlier in the intrusion.

The BRICKSTORM activity cluster, tracked as UNC5221 and attributed to Chinese state-sponsored actors, presents a different dimension of the logging threat: the relationship between attacker dwell time and log retention policy [2]. Mandiant's reporting documented BRICKSTORM maintaining covert access in compromised networks for 393 days across targets in the legal, technology, and business process outsourcing sectors [2][9]. The campaign relied on a stealthy backdoor that evaded conventional detection during this period. Cloud provider default retention for audit logs is commonly 90 days or less — AWS CloudTrail's event history console retains records for 90 days by default — a window shorter than

documented espionage-campaign dwell times. When dwell time exceeds the retention window, organizations cannot reconstruct initial access methods, lateral movement paths, or the full scope of data exposure even after they discover the intrusion. The attacker need not disable logging at all — the passage of time accomplishes the same erasure, and structural retention policies effectively extend the attacker's forensic immunity.

Google Cloud's threat research published in 2025–2026 has documented a shift in ransomware operator behavior toward systematic pre-extortion destruction of logging infrastructure [5]. Ransomware actors frequently deleted logs, core dumps, and backup infrastructure before issuing ransom demands. The operational logic is clear: destroying recovery infrastructure and forensic records increases victim pressure to pay, because the cost of independent recovery — both financial and reputational — rises when the organization cannot determine what was accessed or exported. Bling Libra, the threat actor group behind the ShinyHunters ransomware operation, has been documented by Palo Alto Networks Unit 42 as having evolved from data sale to extortion [8], a pattern consistent with the broader ransomware shift toward infrastructure destruction documented by Google Cloud [5].

Structural Risk: Retention Gaps, AI Workloads, and New Blind Spots

Beyond deliberate manipulation, structural characteristics of cloud environments create logging gaps that adversaries can exploit passively. Log retention policies are often driven by storage cost considerations, defaulting to provider minimums rather than forensic timelines, and the gap between retention windows and observed threat actor dwell times is not incidental — it is increasingly predictable. If a threat actor can maintain access for longer than an organization retains records, the forensic record is guaranteed to be incomplete regardless of whether any active log manipulation occurred. Aligning retention policies to realistic threat timelines is therefore a defensive configuration choice, not merely a compliance consideration.

The rapid proliferation of AI workloads introduces additional complexity. Agentic AI systems, AI development pipelines, model inference infrastructure, and AI gateway services such as LLM proxy layers each generate API call patterns, identity assumptions, and data movement that differ substantially from conventional application workloads. Most enterprise SIEM implementations have not yet been updated to ingest AI-specific telemetry sources — including MCP server invocations, tool call sequences, prompt and completion logs, and agent-to-agent API traffic — into their cloud audit logging pipelines. This creates functional blind spots analogous to the early days of container and serverless adoption, when organizations discovered after incidents that key activity had fallen outside their established logging coverage. Adversaries targeting AI infrastructure will encounter these gaps as a natural consequence of incomplete log source integration.

Recommendations

Immediate Actions

Organizations should begin by verifying the integrity of their cloud audit logging configuration rather than assuming it reflects intended design. In AWS, this means confirming that CloudTrail is enabled in all regions, including the global services region, and that multi-region trails are configured with `IncludeGlobalServiceEvents` set to true. Event selector configurations should be reviewed to confirm that management events and, where appropriate, data events for high-sensitivity services (S3, KMS, Lambda, IAM) are captured [11]. In GCP, organizations should verify that all Data Access audit log types are enabled — Admin Activity logs are always on, but Data Read and Data Write logs require explicit configuration and are often left at default-off for cost reasons. In Azure, diagnostic settings should be confirmed for all critical services including Azure Active Directory, Key Vault, and Storage, with forwarding validated to an independent monitoring destination.

Equally important is monitoring for configuration changes to the logging infrastructure itself. CloudTrail management events include `StopLogging`, `DeleteTrail`, and `UpdateTrail`; GCP captures sink modifications in Admin Activity logs; Azure Activity Logs record diagnostic settings changes. Alert rules for these events should exist in the SIEM or security operations platform independent of GuardDuty or Defender for Cloud, because those services may be silenced by the same actions they would otherwise detect. A logging disruption alert that routes to the security operations center through the same pipeline it monitors provides no protection.

Short-Term Mitigations

Log retention policies should be reviewed against realistic threat actor dwell time distributions. The 393-day BRICKSTORM dwell time documented by Mandiant is at the extreme end, but M-Trends reporting confirms that dwell times exceeding 90 days are not uncommon in espionage-motivated intrusions [9]. Organizations operating in sectors targeted by persistent threat actors — technology, legal, financial services, critical infrastructure — should evaluate whether 90-day retention provides adequate forensic coverage for their threat model. Extended retention for management-plane logs (which are typically orders of magnitude lower in volume than data-plane events such as S3 object-level access) is substantially less expensive than extended retention for data-plane logs and represents a proportionate risk mitigation.

Organizations should also implement CloudTrail log file integrity validation and equivalent controls in GCP and Azure. CloudTrail's SHA-256 digest files allow post-hoc verification that log files have not been modified or deleted after delivery to S3. Storing CloudTrail logs in S3 buckets owned by a dedicated security account with restricted access prevents the same credentials used to disable logging from also being used to delete the delivered log files. The security account should have no permissions on production workload accounts, and production account IAM roles should have no permissions to modify the security account's S3 buckets.

For AI and agentic workloads, organizations should audit what telemetry their AI infrastructure generates and whether it is currently ingested into their SIEM. AI gateway services, MCP servers, agentic orchestration frameworks, and model inference endpoints all generate API call logs that may reveal compromise, prompt injection, or tool abuse. Integrating these sources into existing detection workflows before an incident occurs is substantially easier than reconstructing coverage after one.

Strategic Considerations

The fundamental tension in cloud logging security is that the controls an organization relies on for detection and investigation are themselves cloud-managed services — and therefore configurable by anyone with sufficient cloud permissions. This places a premium on access controls that prevent logging configuration changes from being authorized by the same credentials most likely to be compromised: high-privilege user accounts, CI/CD pipeline service accounts, and developer roles with broad permissions. Service control policies in AWS Organizations and equivalent constructs in GCP and Azure provide organization-wide guardrails that cannot be overridden by actions within individual accounts, and should be used to deny `cloudtrail:StopLogging` and analogous operations for all principals except dedicated security administration roles [12][13].

Organizations should also consider the log supply chain holistically. Audit log records pass through multiple systems — cloud provider API, log aggregation service, SIEM ingestion pipeline, storage — and each transition represents a potential point of tampering or loss. Immutable log storage, cryptographic integrity verification, and independent copy retention across accounts or cloud providers all raise the cost of a complete logging disruption attack. The goal is not to make logging disruption impossible — determined, privileged adversaries will find paths — but to ensure that any disruption leaves detectable evidence somewhere outside the attacker's reach.

CSA Resource Alignment

This research note connects to several active CSA frameworks and programs. The Cloud Adversarial Vectors, Exploits, and Threats (CAVEaT) matrix documents cloud-specific adversarial behaviors and provides a taxonomy for mapping logging manipulation techniques to broader cloud threat categories, supporting the kind of structured threat modeling that surfaces log suppression as a distinct risk requiring independent controls [10]. CAVEaT's coverage of cloud defense evasion complements MITRE ATT&CK T1562.008 for organizations seeking cloud-native framing for their threat model.

CSA's MAESTRO framework for agentic AI threat modeling is directly relevant to the AI workload blind spots discussed in this note. MAESTRO's threat analysis covers AI agent observability and the risk of agents operating outside their intended supervision context — a category that includes scenarios where AI agent activity is not captured by enterprise audit logging pipelines. Organizations deploying agentic AI systems should review MAESTRO guidance on monitoring and explainability controls to ensure that AI-specific telemetry is treated as a first-class logging requirement.

The CSA Cloud Controls Matrix (CCM) and its AI extension, the AI Controls Matrix (AICM), address logging and monitoring requirements across multiple control domains. CCM control domain SEF (Security Incident Management, E-Discovery, and Cloud Forensics) establishes baseline requirements for evidence preservation, log integrity, and forensic readiness. AICM extends these requirements to AI system audit trails, recognizing that AI workloads generate novel evidence categories that conventional cloud forensics frameworks do not fully address. CCM control LOG-08 specifically addresses log retention, and organizations should map their retention policies to this control's requirements as a baseline.

CSA's Zero Trust guidance is also relevant: the principle that no network position or cloud service should be trusted by default extends logically to logging infrastructure itself. Log pipelines, SIEM connectors, and log export configurations should be treated as sensitive, attack-surface-aware components requiring the same access control scrutiny applied to other critical infrastructure. Alerts on logging configuration changes are a foundational implementation of Zero Trust applied to the observability layer.

References

[1] MITRE. "Impair Defenses: Disable or Modify Cloud Logs (T1562.008)." MITRE ATT&CK, October 2025.

[2] Google Cloud Threat Intelligence. "Another BRICKSTORM: Stealthy Backdoor Enabling Espionage into Tech and Legal Sectors." Google Cloud Blog, 2025. See also: SecurityWeek. "Chinese Hackers Lurked Nearly 400 Days in Networks With Stealthy BrickStorm Malware." SecurityWeek, 2025.

[3] Sysdig. "SCARLETEEL: Operation Leveraging Terraform, Kubernetes, and AWS for Data Theft." Sysdig Blog, 2023.

[4] Datadog Security Labs. "Bypassing CloudTrail in AWS Service Catalog, and Other Logging Research." Datadog Security Labs, 2023. See also: Datadog Security Labs. "AWS CloudTrail Vulnerability: Undocumented API Allows CloudTrail Bypass." Datadog Security Labs, January 2023.

[5] Google Cloud. "Ransomware Tactics, Techniques, and Procedures in a Shifting Threat Landscape." Google Cloud Blog, 2026.

[6] CISA. "BOD 25-01: Implementing Secure Practices for Cloud Services." Cybersecurity and Infrastructure Security Agency, December 2024.

[7] Google Cloud. "Best Practices for Cloud Audit Logs." Google Cloud Documentation, 2025.

[8] Palo Alto Networks Unit 42. "Bling Libra's Tactical Evolution: The Threat Actor Group Behind ShinyHunters Ransomware." Unit 42 Threat Intelligence, 2024.

[9] Mandiant / Google Cloud. "M-Trends 2026: Data, Insights, and Strategies From the Frontlines." Google Cloud Security, 2026.

[10] Cloud Security Alliance. "Cloud Adversarial Vectors, Exploits, and Threats (CAVEaT)." CSA, 2024.

[11] AWS. "Security Best Practices in AWS CloudTrail." AWS CloudTrail User Guide, 2025.

[12] CISA. "Secure Cloud Business Applications (SCuBA) Project." CISA, 2024–2025.

[13] Microsoft. "Microsoft Cloud Security Benchmark v2 — Logging and Threat Detection." Microsoft Learn, 2025.