CONFIDENTIAL — SAMPLE / DEMONSTRATION

# CLOUD PENETRATION TESTING REPORT

## AstraVault Technologies, Inc. — Fictional Sample Assessment

*2026 Illustrative Sample Deliverable — Amazon Web Services (AWS) Environment*

| | |
|---|---|
| **Organization** | AstraVault Technologies, Inc. (Fictional) |
| **Industry** | FinTech / Enterprise SaaS — Cloud-Based Financial Workflow & Analytics Platform |
| **Cloud Provider** | Amazon Web Services (AWS) — Fictional Multi-Account Environment |
| **Prepared By** | TMG Security |
| **Assessment Type** | Cloud Penetration Test — AWS Configuration, IAM, and Workload Security Assessment |
| **Assessment Period** | 07 September 2026 – 25 September 2026 |
| **Report Date** | 30 September 2026 |
| **Classification** | CONFIDENTIAL — SAMPLE / DEMONSTRATION |

**FICTIONAL SAMPLE — NOT AN ACTUAL CLIENT CLOUD PENETRATION TEST**

*AstraVault Technologies, Inc. is fictional. All AWS accounts, resources, findings, and evidence are illustrative.*

# 02 DOCUMENT CONTROL

| | |
|---|---|
| **Document Title** | Cloud Penetration Testing Report — AstraVault Technologies, Inc. (Fictional Sample) |
| **Document ID** | TMG-CLOUD-AVT-2026-001 |
| **Version** | 1.0 |
| **Issue Date** | 30 September 2026 |
| **Assessment Period** | 07 September 2026 – 25 September 2026 |
| **Assessment Type** | Cloud Penetration Test (AWS) — External, Configuration Review, Limited-Privilege IAM, Assumed-Breach |
| **Client** | AstraVault Technologies, Inc. (Fictional) |
| **Cloud Provider** | Amazon Web Services (AWS) |
| **Accounts In Scope** | AstraVault Production (123456789012), Security (123456789013), Development (123456789014) — all fictional |
| **Primary Domains** | console.astravault-example.com, api.astravault-example.com, app.astravault-example.com (fictional, non-resolving) |
| **Environment** | Fictional AWS environment provisioned for this illustrative engagement |
| **Standards Referenced** | CVSS v3.1-inspired scoring, CWE, AWS Well-Architected Framework (Security Pillar) — illustrative use only |
| **Prepared By** | TMG Security Offensive Security Team — Cloud Security Practice |
| **Technical Lead** | TMG Security — Lead Cloud Penetration Tester (Fictional Engagement Role) |
| **Reviewed By** | TMG Security Quality Assurance Lead |
| **Classification** | CONFIDENTIAL — SAMPLE / DEMONSTRATION |

## Version History

| Version | Date | Author | Summary |
|---|---|---|---|
| 1.0 | 30 September 2026 | TMG Security Offensive Security Team | Initial issue — fictional sample deliverable. |

## Distribution List

The following fictional distribution list reflects the recipients TMG Security would designate for an engagement of this type; no real individuals are named.

| Recipient (Fictional Role) | Organization | Purpose |
|---|---|---|
| Chief Information Security Officer | AstraVault Technologies, Inc. | Executive sponsor and primary report recipient. |
| VP of Cloud Infrastructure | AstraVault Technologies, Inc. | Remediation ownership and cloud-engineering prioritization. |
| Director of Platform Security | AstraVault Technologies, Inc. | Coordinates retest scheduling and finding tracking. |
| Engagement Manager | TMG Security | Client relationship management and delivery oversight. |
| Quality Assurance Lead | TMG Security | Independent technical review of this report prior to issue. |

## Authorship

| Name (Fictional Role) | Role | Contribution |
|---|---|---|
| Samir Okonkwo-Reyes | Lead Cloud Penetration Tester | IAM/privilege-escalation testing, attack-path validation, report lead. |
| Lena Vasquez-Ito | Cloud Security Consultant | S3/network/workload security testing, evidence collection. |
| Priya Raman | Quality Assurance Lead | Independent technical review and report quality assurance. |

| Name (Fictional Role) | Role | Contribution |
|---|---|---|
| David Okafor | Engagement Manager | Scoping, rules of engagement, and client coordination. |

# 03 CONFIDENTIALITY

This document and its contents are classified CONFIDENTIAL — SAMPLE / DEMONSTRATION by TMG Security. In a real client engagement, a report of this type and classification would be subject to strict handling controls; the notes below describe those controls as they would normally apply, for illustrative purposes.

## Intended Handling (Illustrative)

- Distribution limited to the named recipients in Section 02's Distribution List and any individual they explicitly authorize.
- Not to be forwarded, published, or disclosed outside the receiving organization without the prior written consent of TMG Security.
- Electronic copies would normally be encrypted at rest and in transit, with printed copies logged and tracked.
- Findings, evidence, and remediation detail would normally be shared only with personnel who have a legitimate operational need to know.

## This Sample Report

Because this is a fictional, illustrative sample report prepared for demonstration and portfolio purposes, none of the above handling requirements apply to any real data, and no real confidential information of any kind is contained in this document. The CONFIDENTIAL marking and classification language are retained throughout this sample to accurately demonstrate the format, tone, and markings a genuine TMG Security cloud penetration testing report would carry.

This is a fictional sample report. No real confidential client information is contained anywhere in this document.

# 04 IMPORTANT DISCLAIMER

**FICTIONAL SAMPLE REPORT — NOT AN ACTUAL CLIENT CLOUD PENETRATION TEST**

This report has been produced solely for demonstration and portfolio purposes on behalf of TMG Security. It is intended to illustrate the structure, technical depth, and professional reporting standard of a cloud penetration testing deliverable for an AWS environment. To ensure there is no ambiguity regarding the nature of this document, TMG Security sets out the following clarifications:

- AstraVault Technologies, Inc. is a fictional entity. It does not correspond to any real company, and any resemblance to an actual organization is purely coincidental.
- All AWS accounts referenced in this report — including account IDs 123456789012, 123456789013, and 123456789014 — are fictional and do not correspond to any real AWS account.
- All domains referenced throughout this report (console.astravault-example.com, api.astravault-example.com, app.astravault-example.com, and astravault-example.com generally) are fictional, non-resolving placeholder domains used only for illustration.
- No real client was engaged and no real AWS environment, account, resource, or workload was tested.
- No production environment, system, network, cloud resource, container, database, or code repository of any kind was accessed, scanned, or tested in the creation of this report.
- No real user data, personal information, payment data, or customer records were accessed, viewed, or processed.
- No real credentials — including IAM access keys, database passwords, API keys, or third-party integration secrets — were used or are disclosed anywhere in this report. All credential-shaped values shown (e.g., AKIAEXAMPLE-prefixed access key IDs) are non-functional illustrative placeholders.
- All AWS CLI output, IAM policy documents, JSON responses, S3 configuration, security-group rules, CloudTrail events, and IAM trust policies shown in this report are synthetic and constructed for illustration only.
- All vulnerabilities, findings, risk ratings, CVSS-style scores, dates, and remediation statuses described in this report are fictional and were constructed for demonstration purposes.
- All attack paths and Proofs of Concept (PoCs) described in this report are illustrative, synthetic, and non-destructive; none were executed against a real or live AWS account, and no real cloud resource was exploited, modified, or deleted.
- All screenshots and tool-style evidence panels (AWS CLI output, IAM console-style output, security-dashboard mockups) are synthetic mock-ups created specifically for this sample report, not real tool captures.
- No real exploitation occurred at any point during the preparation of this document, and no actual customer or business impact occurred.
- TMG Security is not affiliated with, endorsed by, reviewed by, or certified by AWS, PTES, NIST, OWASP, or any other standards organization unless explicitly stated otherwise. Nothing in this report implies AWS certification, partnership, or endorsement of TMG Security or of this assessment.
- This report does not represent an actual TMG Security client engagement and must not be relied upon as evidence of the security posture of any real organization, cloud account, or application.

## Intended Use

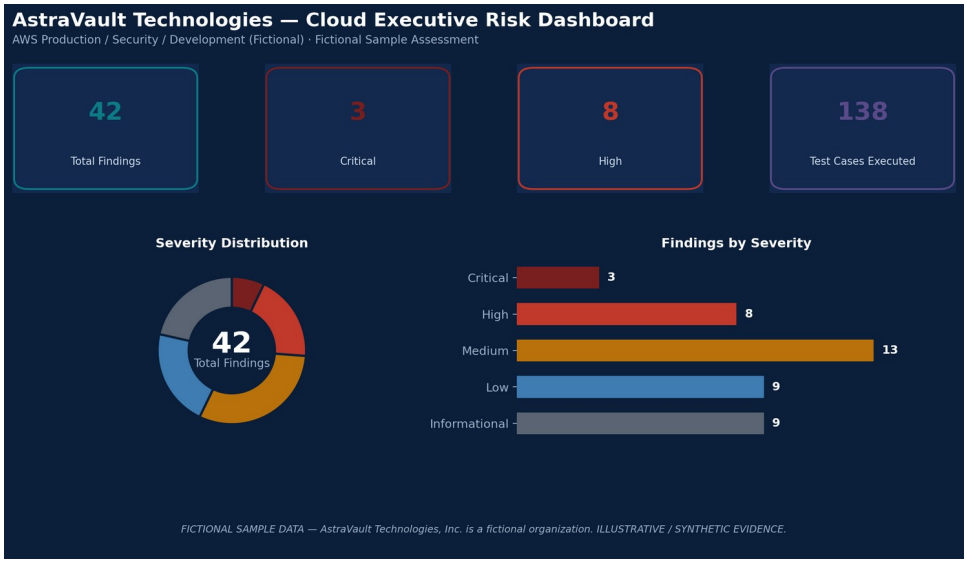
This sample report may be shared with prospective clients, partners, and other interested parties to illustrate TMG Security's cloud security testing methodology, technical reporting standards, and evidence-presentation quality. It should not be relied upon for any compliance, legal, contractual, or procurement decision, and does not constitute security advice regarding any real AWS environment or organization.

# TABLE OF CONTENTS

# 05 EXECUTIVE SUMMARY

TMG Security performed a fictional cloud penetration test of AstraVault Technologies, Inc.'s AWS environment between 07 September 2026 and 25 September 2026. AstraVault Technologies, Inc. is a fictional FinTech / enterprise SaaS company providing a cloud-based financial workflow and analytics platform. This illustrative assessment combined external black-box reconnaissance, authenticated cloud configuration review, limited-privilege IAM testing, and assumed-breach/compromised-workload testing across AstraVault's three fictional AWS accounts (Production, Security, and Development) to identify exploitable weaknesses in identity and access management, data storage, network architecture, workload configuration, logging and detection, and cloud deployment pipelines.

This assessment identified 42 fictional findings: 3 Critical, 8 High, 13 Medium, 9 Low, and 9 Informational. The three Critical findings, taken together, describe a realistic and complete cloud attack path: an over-permissive cross-account IAM role capable of full administrative privilege escalation, a publicly accessible S3 bucket exposing sensitive fictional financial data, and an EC2 workload metadata exposure that converts a server-side request forgery (SSRF) condition directly into stolen cloud credentials. Section 40 (Attack Path Analysis) demonstrates how these and other findings chain together into four composite, non-destructive fictional attack scenarios.




*Illustrative executive risk dashboard — ILLUSTRATIVE / FICTIONAL SAMPLE DATA, AstraVault Technologies, Inc.*

## Key Metrics

| | | | |
|---|---|---|---|
| **42** Total Fictional Findings | **3** Critical | **8** High | **138** Test Cases Executed |

| | | | |
|---|---|---|---|
| **3** Fictional AWS Accounts In Scope | **4** Composite Attack Chains Validated | **13** Medium | **18** Low + Informational |

## Assessment Scope Summary

Testing covered AstraVault's fictional AWS-primary environment across IAM, S3, EC2, VPC, Security Groups, Lambda, API Gateway, ECS/ECR containers, RDS, KMS, Secrets Manager, CloudTrail, CloudWatch, GuardDuty, Security Hub, WAF, CloudFront, Route 53, SNS/SQS/EventBridge, Systems Manager, and AWS Backup — exercised only where technically relevant to a specific finding, rather than as a forced checklist of every available AWS service. Full scope detail appears in Section 08.

## Principal Risk Themes

- Identity and access management is the assessment's dominant risk theme: 2 of 3 Critical findings and 3 of 8 High findings trace directly to IAM misconfiguration — over-permissive cross-account trust, documented privilege-escalation policy chains, and execution roles far broader than the workloads that use them.
- Public data exposure remains a live risk: a production S3 bucket holding fictional customer financial statements was reachable by any unauthenticated internet user, with no server-side encryption enforcement as a compensating control.
- Workload-to-cloud credential theft (SSRF-to-metadata) was confirmed technically feasible end-to-end against a fictional internet-facing feature, illustrating how an application-layer weakness converts directly into cloud-account compromise.
- Detective controls lag behind the environment's actual risk: CloudTrail data-event coverage, CloudWatch alerting, and GuardDuty finding routing all show gaps that would slow real-world incident detection and response for several of this report's own finding categories.
- AstraVault's Development account consistently carries a weaker security baseline than Production, and an overly broad cross-account trust relationship means that gap is not fully contained — see CLOUD-005 and CLOUD-041.

### Findings by Severity

| Severity | Count | Description |
|---|---|---|
| Critical | 3 | Immediate, high-confidence risk of full account or sensitive-data compromise; remediation recommended within 0-14 days. |
| High | 8 | Significant risk requiring prompt remediation; recommended within 0-30 days. |
| Medium | 13 | Moderate risk warranting remediation in the near term; recommended within 31-60 days. |
| Low | 9 | Lower risk, often defense-in-depth or hygiene gaps; recommended within 61-90 days. |
| Informational | 9 | Observations and governance recommendations with no immediate exploit path; tracked for continuous improvement. |

### Positive Observations

This fictional assessment also confirmed a number of security controls operating effectively: the primary transactional RDS database is correctly placed in a private subnet with scoped security-group access; CloudTrail log-file validation and multi-region coverage are correctly enabled; the majority of production S3 buckets correctly enforce Block Public Access and default encryption; and GuardDuty and Security Hub are enabled organization-wide, even where finding-routing maturity has room to improve. Section 52 details these and other positive controls in full.

### Strategic Recommendation

TMG Security recommends AstraVault prioritize remediation in the following order: (1) the three Critical findings and the IAM/cross-account trust weaknesses underlying two of them, closing the composite attack paths in Section 40 first; (2) the eight High findings, with particular urgency on the publicly reachable RDS instance and the unpatched production container image; (3) the Medium findings, most of which are configuration and governance gaps rather than directly exploitable conditions; and (4) the Low and Informational findings as part of an ongoing cloud security hygiene program. A full remediation roadmap with suggested timelines appears in Section 49.

## 06 ASSESSMENT OVERVIEW

This section describes the fictional objective, type, and approach of the illustrative AstraVault Technologies, Inc. cloud penetration test, prepared to demonstrate TMG Security's engagement-scoping and reporting standards for AWS cloud security assessments, as the cloud counterpart to TMG Security's companion Web Application, API, Android, iOS, and Network sample reports.

### Engagement Type

This is a fictional Cloud Penetration Test conducted against AstraVault's AWS-primary environment, combining external black-box testing of internet-facing cloud assets, authenticated configuration review across all three fictional AWS accounts, limited-privilege IAM testing from a standard application-tier identity, and assumed-breach testing simulating a compromised workload. TMG Security did not have, and does not claim to have had, unrestricted AWS Management Console access at any point in this fictional engagement.

### Testing Window

Testing was conducted over the fictional assessment period of 07 September 2026 – 25 September 2026 (13 business days), with this report issued 30 September 2026. All Critical findings were communicated to the fictional AstraVault security team out-of-band within 24 hours of confirmation, consistent with the Rules of Engagement in Section 10.

This engagement overview describes a fictional, illustrative assessment. No real client engagement occurred and no real AWS environment was accessed.

## 07 ASSESSMENT OBJECTIVES

The fictional objective of this engagement was to identify and validate exploitable security weaknesses across AstraVault's AWS cloud environment, and to provide AstraVault's fictional engineering and security leadership with prioritized, actionable remediation guidance. Specifically, this assessment sought to:

- Determine what an unauthenticated external attacker could discover and reach across AstraVault's internet-facing cloud attack surface.
- Assess whether AstraVault's IAM configuration — users, roles, groups, policies, and cross-account trust relationships — enforces least privilege and resists documented privilege-escalation techniques.
- Evaluate the security of AstraVault's data storage services (S3, RDS, Secrets Manager, KMS) against unauthorized access, exposure, and weak encryption/backup practices.
- Assess network segmentation and security-group configuration across AstraVault's VPCs for unnecessary internet exposure of sensitive resources.
- Evaluate workload security (EC2, ECS, Lambda, ECR/container images) for over-permissioned execution roles, metadata-service exposure, and vulnerable dependencies.
- Assess AstraVault's logging, monitoring, and detection posture (CloudTrail, CloudWatch, GuardDuty, Security Hub) for coverage gaps that would delay incident detection and response.
- Review AstraVault's fictional CI/CD and cloud deployment pipeline for secure handling of deployment credentials, artifact integrity, and image-scanning enforcement.
- Construct and validate realistic, non-destructive composite attack chains demonstrating how individual findings combine into material business risk.
- Translate technical findings into clear, prioritized business impact and an actionable remediation roadmap.

## 08 SCOPE

The following fictional scope table defines the boundaries of this illustrative assessment. Full account, resource, and identity inventories appear in Appendices B-I.

### In Scope

| Area | Detail |
|---|---|
| AWS Accounts | AstraVault Production (123456789012), AstraVault Security (123456789013), AstraVault Development (123456789014) — all fictional, non-existent account IDs |
| Cloud Provider | Amazon Web Services (AWS), us-east-1 primary region (fictional) |
| External Attack Surface | Public DNS (astravault-example.com), public S3 buckets, CloudFront distributions, public API Gateway endpoints, internet-reachable EC2/RDS network paths |

| Area | Detail |
|---|---|
| Identity & Access Management | IAM users, roles, groups, policies, permission boundaries, STS/AssumeRole, cross-account trust relationships |
| Storage | S3 (bucket policies, encryption, versioning, logging, access points), RDS, Secrets Manager, KMS |
| Compute & Workloads | EC2 (instances, AMIs, instance profiles), ECS (task definitions, task roles), Lambda (execution roles, configuration), ECR (image scanning, lifecycle policies) |
| Network | VPC, subnets, route tables, Internet/NAT Gateways, Security Groups, Network ACLs, VPC endpoints, peering, Transit Gateway |
| Edge & Delivery | CloudFront, Route 53, AWS WAF, API Gateway |
| Logging & Detection | CloudTrail, CloudWatch, GuardDuty, Security Hub, AWS Config |
| Messaging & Events | SNS, SQS, EventBridge (where relevant to a specific finding) |
| Operations | Systems Manager (Patch Manager, Session Manager), AWS Backup |
| CI/CD & Deployment | Fictional cloud deployment pipeline: source → build → container build → ECR → deployment role → production, including OIDC/workload-identity federation |
| Domains In Scope | console.astravault-example.com, api.astravault-example.com, app.astravault-example.com (all fictional, non-resolving) |
| Assessment Perspectives | External Black-Box; Authenticated Cloud Configuration Review; Limited-Privilege IAM Assessment; Assumed-Breach / Compromised Workload Perspective |

### Out of Scope

| Area | Detail |
|---|---|
| Non-AWS Infrastructure | Not included; this engagement is AWS-only (see the companion Network, Web, API, Android, and iOS sample reports for other testing disciplines) |
| Denial-of-Service Testing | Not performed under any circumstances |
| Destructive Testing | Not performed under any circumstances — no resource was created, modified, or deleted in a way that could affect availability |
| Social Engineering | Not included in this engagement |
| Physical Security | Not included in this engagement |
| Unrestricted Console Access | TMG Security operated from the four defined assessment perspectives only, never with unrestricted AWS Management Console access |
| Third-Party SaaS Integrations | Only client-side/AWS-side integration configuration reviewed; third-party vendor infrastructure itself (e.g., the fictional payment aggregator) not tested |
| Application-Layer Source Code Review | Not performed; this is an infrastructure/cloud-configuration assessment, not a source-code audit |
| End-User Device Security | Not included in this engagement |

## 09 OUT OF SCOPE

Beyond the area-level scope boundaries in Section 08, the following specific fictional components and considerations were explicitly excluded from direct testing in this engagement, and are flagged here for transparency.

| Component / Area | Reason Excluded |
|---|---|
| Payment-aggregator vendor infrastructure | Third-party managed service; only AstraVault-side integration and secret handling reviewed (see CLOUD-007) |
| AWS's own infrastructure and control plane | Testing of AWS's own infrastructure is prohibited under the AWS Customer Support Policy for Penetration Testing; this fictional engagement respects that boundary |
| Non-production customer data | No real customer data of any kind exists in this fictional exercise; all data referenced is synthetic |
| Mobile and web application source code | Covered by TMG Security's companion Android, iOS, Web Application, and API sample reports, not this cloud-infrastructure assessment |

| Component / Area | Reason Excluded |
|---|---|
| Physical data center security | AWS-managed; outside TMG Security's assessment boundary and AWS's own shared-responsibility model for customer testing |
| End-user endpoint security (laptops, mobile devices) | Outside this engagement's cloud-infrastructure scope |
| Organizational security awareness / phishing simulation | Not included in this engagement's rules of engagement |

Exclusion from direct testing does not imply these components are free of risk; it reflects this fictional engagement's defined scope, duration, and the AWS shared-responsibility model.

# 10 RULES OF ENGAGEMENT

The following fictional rules of engagement governed this illustrative assessment and are documented here for completeness, consistent with TMG Security's standard cloud penetration testing practice.

## Testing Constraints

- No destructive testing: no production resource was created, modified, or deleted in a way that could affect availability, integrity, or a real customer.
- No denial-of-service testing of any kind, including resource-exhaustion or high-volume load testing beyond limited, controlled rate-limit validation.
- No real credential material was generated, used, or retained; all evidence in this report uses fictional placeholder values.
- For this illustrative sample, S3 testing is assumed to have been explicitly authorized by the relevant AWS customer/account owner and, where required under applicable AWS policy, coordinated with AWS Support or the applicable AWS account representative; this fictional engagement does not test AWS-owned infrastructure or services, consistent with the boundary described in Section 09.
- Critical findings were to be reported to the fictional AstraVault security team within 24 hours of confirmation, out-of-band from the final report.
- Testing was restricted to the fictional AWS account IDs and domains explicitly listed in Section 08; no other AWS account or organization was in scope.
- All testing activity was to be conducted from TMG Security's documented, allow-listed fictional source IP ranges, coordinated in advance with the fictional AstraVault cloud team to distinguish assessment activity from genuine threat activity in GuardDuty/CloudTrail.

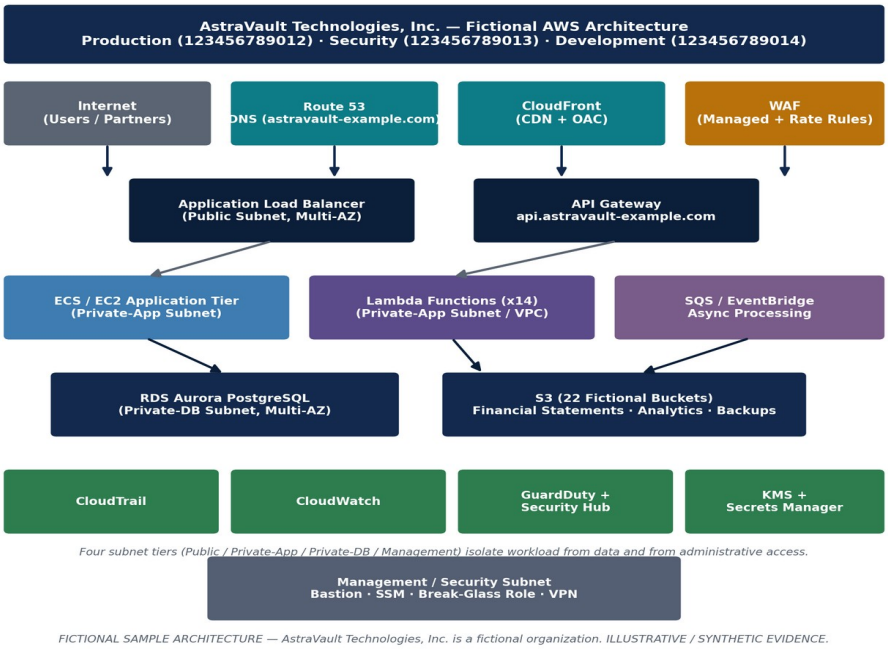
## Assessment Perspectives Used

| Perspective | Description |
|---|---|
| External Black-Box | No credentials or internal knowledge; internet-facing reconnaissance and testing only. |
| Authenticated Cloud Configuration Review | Read-only, cross-account audit-scoped role used to review configuration across all three fictional accounts. |
| Limited-Privilege IAM Assessment | A standard, low-privilege application/engineering-tier IAM identity used to test privilege-escalation and cross-account trust paths. |
| Assumed-Breach / Compromised Workload Perspective | Simulated a workload already compromised (e.g., via SSRF or a vulnerable container) to assess blast radius from that vantage point, without performing the initial compromise technique against a real system. |

*Full test-account and assumed-role detail for each perspective appears in Appendix O.*

# 11 CLOUD ARCHITECTURE

The following fictional architecture diagram illustrates how AstraVault's AWS environment is organized: a public-facing content-delivery and web path (Route 53 → CloudFront → AWS WAF → Application Load Balancer → ECS/EC2 application workloads → RDS), a parallel API/event-driven path (API Gateway → Lambda → SQS/EventBridge → processing workloads), and a security-services layer (CloudTrail, CloudWatch, GuardDuty, Security Hub, KMS, Secrets Manager) spanning both. It is provided to give this sample report realistic technical grounding for the findings in later sections.




*Illustrative fictional AWS architecture — all account IDs, resource names, and infrastructure detail are fictional.*

## Architecture Notes

- The public web path terminates application traffic at an Application Load Balancer fronting ECS/EC2 application workloads, which in turn reach the primary transactional RDS Aurora cluster in a private database subnet — correctly segmented, as confirmed in Section 19-21.
- A parallel serverless/API path handles asynchronous processing: API Gateway routes to Lambda functions, which publish to SQS and EventBridge for downstream fan-out to processing workloads such as statement generation and fraud monitoring — the subject of Sections 23-24 and several Medium/High findings.
- Network tiering follows a four-subnet model: Public Subnets (ALB, NAT Gateway, bastion), Private Application Subnets (ECS/EC2/Lambda ENIs), Private Database Subnets (RDS), and a Management/Security Subnet (bastion, VPN, SSM-managed hosts) — with one notable exception, the analytics-reporting RDS instance, incorrectly placed in a public subnet (CLOUD-010).
- Security services (CloudTrail, CloudWatch, GuardDuty, Security Hub, KMS, Secrets Manager) span the environment and feed the centralized logging and detection pipeline in the fictional AstraVault Security account, though with coverage gaps documented in Sections 30-33.
- The Development account maintains a structurally similar but independently-provisioned environment with a materially weaker security baseline, discussed further in Section 12 and CLOUD-041.
- Container workloads (ECS on Fargate and EC2 capacity providers) pull images from ECR, itself fed by the fictional CI/CD pipeline reviewed in Section 38.

# 12 AWS ACCOUNT STRUCTURE

AstraVault's fictional AWS environment is organized under a single AWS Organization spanning three accounts, summarized below. The complete account register, including purpose and ownership detail, appears in Appendix B.

| Account | Account ID (Fictional) | Purpose | Environment |
|---|---|---|---|
| AstraVault Production | 123456789012 (fictional) | Primary production workloads, customer-facing API and application, transactional data | Production |
| AstraVault Security | 123456789013 (fictional) | Centralized logging, CloudTrail/GuardDuty/Security Hub aggregation, break-glass administrative access | Security / Shared Services |
| AstraVault Development | 123456789014 (fictional) | Engineering development and staging workloads, CI/CD build environment | Development / Staging |

### Account Segmentation Model

AstraVault's intended segmentation model places production workloads and sensitive data in the Production account, centralizes logging and break-glass administrative access in the Security account, and isolates engineering/staging work in the Development account. This is a sound baseline pattern; however, this assessment found the segmentation is not fully enforced in practice — both the Production-to-Security cross-account trust (CLOUD-001) and the Production-to-Development cross-account trust (CLOUD-005) are broader than the segmentation model intends, and the Development account's own baseline controls lag behind Production's (CLOUD-041).

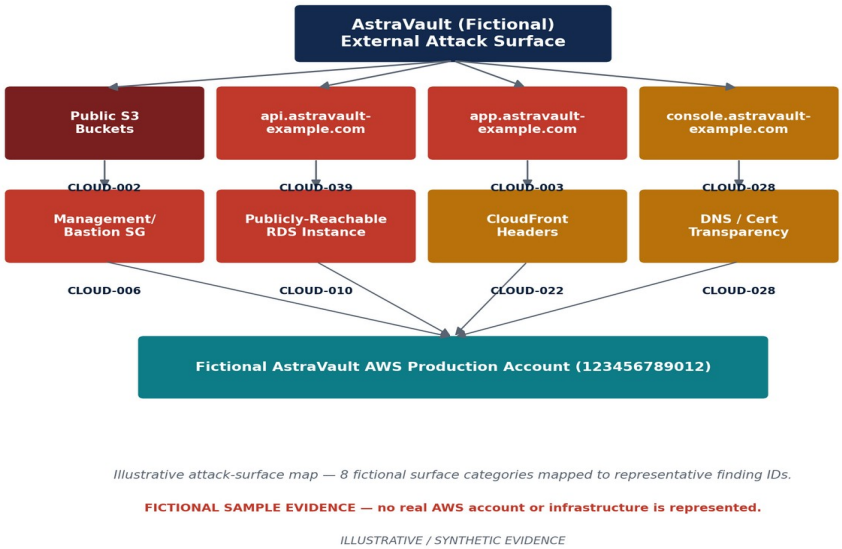
### Organization-Level Controls

- AWS Organizations Service Control Policies (SCPs) are in use for a small set of baseline guardrails but do not yet fully enforce the IAM and tagging constraints recommended throughout this report's findings.
- Security Hub and GuardDuty are enabled organization-wide via delegated administration from the Security account, though standard-subscription and finding-routing maturity vary by account (CLOUD-029, CLOUD-034).
- AWS Config is enabled with an aggregator in the Security account, providing the configuration-history basis for several observations in this report.

All account IDs, organization structure, and account names in this section are fictional and do not correspond to any real AWS Organization.

## 13 EXTERNAL ATTACK SURFACE

TMG Security began this fictional assessment from a purely external, unauthenticated black-box perspective, mapping every internet-reachable AWS resource associated with AstraVault's fictional domains before any authenticated testing began.




*Illustrative fictional external cloud attack surface map — 8 discovered surface categories mapped to representative finding IDs.*

## Discovered External Surface

| Surface Category | Fictional Observation | Related Finding |
|---|---|---|
| Public DNS / Certificate Transparency | Enumeration of astravault-example.com subdomains revealed the production S3 bucket naming pattern | CLOUD-002 |
| Public S3 Buckets | 1 of 22 production buckets reachable with unauthenticated GetObject/ListBucket | CLOUD-002 |
| CloudFront Distribution | app.astravault-example.com served via CloudFront with incomplete edge security-header enforcement | CLOUD-022 |
| Public API Gateway | api.astravault-example.com exposing 27 routes, including the SSRF-vulnerable report-fetch feature | CLOUD-003, CLOUD-030 |
| Public EC2 / Bastion | Management/bastion host reachable on SSH/RDP from the internet | CLOUD-006 |
| Public RDS Endpoint | Analytics-reporting database reachable at the network layer from the internet | CLOUD-010 |
| Route 53 Hosted Zone | Public hosted zone with no query logging enabled | CLOUD-028 |
| WAF Coverage | Managed rule groups present but no rate-based rule on sensitive endpoints | CLOUD-030 |

No unauthenticated access to internal application data, administrative interfaces, or AWS Management Console functionality was achieved during the external black-box phase itself; the S3 bucket exposure (CLOUD-002) was the only Critical-severity finding directly reachable without any authentication. The remaining Critical and High findings required at least the limited-privilege or assumed-breach perspectives described in Section 10 to confirm.

# 14 METHODOLOGY

TMG Security's fictional cloud penetration testing methodology combines industry-standard cloud security assessment practice (informed by, but not certified against, frameworks such as the AWS Well-Architected Framework's Security Pillar, the MITRE ATT&CK Cloud matrix, and the CIS AWS Foundations Benchmark) with hands-on exploitation techniques specific to AWS. Testing proceeded through the following phases.

## Phase 1 — Reconnaissance & External Attack Surface Mapping

Unauthenticated enumeration of AstraVault's fictional public DNS, certificate-transparency logs, S3 bucket naming patterns, CloudFront distributions, and public API Gateway routes, establishing the external attack surface described in Section 13 before any credentialed access was used.

## Phase 2 — Authenticated Cloud Configuration Review

Using a fictional read-only, cross-account audit-scoped role, TMG Security systematically reviewed configuration across IAM, S3, EC2, VPC, Security Groups, Lambda, API Gateway, ECS/ECR, RDS, KMS, Secrets Manager, CloudTrail, CloudWatch, GuardDuty, Security Hub, WAF, CloudFront, Route 53, SNS/SQS/EventBridge, Systems Manager, and AWS Backup across all three fictional AWS accounts, comparing observed configuration against AWS security best practices and AstraVault's own fictional internal policies where available.

## Phase 3 — Limited-Privilege IAM & Privilege-Escalation Testing

From a fictional standard, low-privilege application/engineering-tier IAM identity, TMG Security tested documented IAM privilege-escalation patterns (policy chaining, PassRole combinations, cross-account AssumeRole reachability) using IAM policy simulation and, where safe and non-destructive, live AssumeRole testing against fictional test identities — producing the IAM privilege-escalation matrix in Section 17.

## Phase 4 — Assumed-Breach / Compromised-Workload Testing

TMG Security simulated the perspective of an attacker who has already obtained a foothold in a specific workload (via SSRF-to-metadata or a vulnerable container image), assessing what that foothold's actual IAM permissions and network reachability would allow — without performing any destructive action against a real system — to validate the composite attack chains presented in Section 40.

## Phase 5 — Attack Chain Construction & Business Impact Analysis

Individual findings were cross-referenced to construct 4 realistic, non-destructive composite attack chains (Section 40), and each Critical/High finding's technical impact was translated into fictional business impact and likelihood (Section 48) to support prioritized remediation.

## Phase 6 — Reporting & Quality Assurance

Findings were documented with CVSS-inspired scoring, CWE mapping, synthetic evidence, and specific AWS control recommendations, then passed through TMG Security's internal quality-assurance process — including the automated finding-count, ID-sequence, and cross-reference validation described in this report's closing Conclusion section — before issue.

TMG Security is not affiliated with, endorsed by, reviewed by, or certified by AWS, PTES, NIST, OWASP, or any other standards organization. References to industry frameworks describe methodology influences only and do not imply certification against them.

# 15 RISK RATING METHODOLOGY

Each finding in this report is assigned an illustrative CVSS v3.1-inspired base score and a qualitative severity rating. These scores are TMG Security's own sample estimations for this fictional exercise; they are not submitted to any CVSS numbering authority and should not be treated as officially registered CVSS scores.

## Severity Bands

| Severity | Illustrative CVSS Range | Description |
|---|---|---|
| Critical | 9.0 – 10.0 | Immediate, high-confidence risk of full account compromise, mass sensitive-data exposure, or complete loss of a critical security boundary. |
| High | 7.0 – 8.9 | Significant risk of material data exposure, privilege escalation, or compromise of a specific high-value resource. |
| Medium | 4.0 – 6.9 | Moderate risk, often requiring an additional precondition or affecting a less-critical resource; meaningful defense-in-depth or governance gap. |
| Low | 0.1 – 3.9 | Lower risk, typically a hygiene, logging, or hardening gap with limited standalone exploitability. |
| Informational | N/A | Observations and recommendations with no identified exploit path; tracked for continuous improvement rather than urgent remediation. |

## Scoring Factors

Consistent with the CVSS v3.1 framework's structure (used here as an illustrative scoring methodology only), each score reflects: Attack Vector (network, adjacent, local, physical), Attack Complexity, Privileges Required, User Interaction, Scope, and the Confidentiality/Integrity/Availability impact triad. For cloud-specific findings, TMG Security additionally weighs the AWS-specific factors below when calibrating likelihood and business impact.

## Cloud-Specific Risk Factors

- Credential/Role Reachability — how directly a finding converts into usable AWS credentials (e.g., SSRF-to-metadata, over-permissive execution roles).
- Blast Radius — how many downstream resources, accounts, or workloads are affected by a compromise of the specific identity or resource in question.
- Data Sensitivity — the classification of data reachable through the finding (fictional customer financial data is treated as the highest sensitivity tier).
- Detection Coverage — whether existing logging/alerting would likely detect exploitation of the finding before material damage occurs.
- Precondition Complexity — how many additional steps or separate weaknesses must chain together before the finding becomes exploitable in practice.

## Likelihood Ratings

| Likelihood | Description |
|---|---|
| High | Exploitable with commodity technique and minimal precondition; no specialized tooling or extended access required. |
| Medium | Requires a specific precondition (e.g., a compromised credential of a particular type) but no additional novel vulnerability. |
| Low | Requires multiple chained preconditions or a less-common attacker capability to reach exploitability. |
| N/A | Used for detection-gap, hygiene, and governance findings that are not independently exploitable. |

All CVSS-style scores and vectors in this report are illustrative sample estimations for this fictional exercise and are not official, registered CVSS scores.

# 16 IAM ASSESSMENT

TMG Security performed a comprehensive review of AstraVault's fictional Identity and Access Management (IAM) configuration across all three fictional AWS accounts, covering IAM users, roles, groups, inline and managed policies, resource and trust policies, STS/AssumeRole usage, permission boundaries, service roles, instance profiles, task roles, Lambda execution roles, and access-key lifecycle. The full principal inventory appears in Appendix C.

## Illustrative Tests Performed

- Enumerate all IAM users, roles, and groups and map each to a fictional business owner.
- Review the IAM credential report for stale (180+ day inactive) console/API access and unrotated access keys.
- Verify root-account access-key absence and hardware-MFA enforcement.
- Review IAM password policy, permission-boundary usage, and MFA enforcement for human principals.
- Review service-linked roles, instance profiles, and federated (IAM Identity Center) access configuration.

## Illustrative Observations

AstraVault's fictional IAM environment spans 41 IAM users, a small set of groups, and a broad inventory of service/instance-profile roles across the three fictional accounts, detailed in Appendix C. The IAM credential report identified 7 stale users with no activity in 180+ days — including two former-contractor accounts never deactivated at offboarding (CLOUD-017) — and 11 users carrying access keys older than 365 days against AstraVault's own documented 90-day rotation policy (CLOUD-018). Root-account hygiene, password policy, and IAM Identity Center federated-access configuration were all found consistent with good practice. The most significant IAM weaknesses identified relate not to individual user hygiene but to over-permissioned roles and trust relationships, detailed in Section 17 and cross-referenced throughout Sections 18-40.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-017 | MEDIUM | Stale IAM Users With Long-Unused Console and API Access |
| CLOUD-018 | MEDIUM | Unused and Unrotated IAM Access Keys Exceeding Recommended Age |

## IAM Principal Summary

The table below summarizes the fictional IAM principal population by type and account. The complete principal-level register, including permissions and MFA status, appears in Appendix C.

| Category | Production | Security | Development | Total |
|---|---|---|---|---|
| IAM Users | 24 | 6 | 11 | 41 |
| IAM Roles (human-assumable) | 9 | 4 | 3 | 16 |
| IAM Roles (service/instance-profile) | 22 | 5 | 8 | 35 |
| IAM Groups | 5 | 2 | 2 | 9 |

# 17 PRIVILEGE ESCALATION

Building on the IAM principal inventory in Section 16, TMG Security tested AstraVault's fictional environment against a fictional IAM privilege-escalation matrix — a structured set of 10 documented AWS IAM privilege-escalation patterns — using IAM policy simulation and, where safe, live AssumeRole testing from the limited-privilege test identity described in Appendix O.

## Illustrative Tests Performed

- Test each of the 10 documented escalation patterns in the matrix below against every human-assumable IAM group and role.
- Test cross-account AssumeRole reachability from each fictional account's lower-privileged identities toward higher-privileged roles in other accounts.
- Validate that permission boundaries, where applied, actually constrain the escalation patterns they are intended to block.

- Confirm whether any discovered escalation path was reachable from the specific limited-privilege test identity used in this assessment, not merely theoretically possible for an unconstrained principal.

## Illustrative Observations

Of the 10 documented escalation patterns tested, 3 were found reachable (or contributing) in AstraVault's fictional environment: the iam:PassRole + iam:CreatePolicyVersion + lambda:CreateFunction chain available to the platform-engineers group (CLOUD-004); direct cross-account AssumeRole reachability via overly broad trust-policy principals (CLOUD-001, CLOUD-005); and a wildcard-resource iam:PassRole grant that, while not independently sufficient for escalation, materially widens the blast radius of CLOUD-001. The remaining 7 patterns were either not found or correctly blocked by permission boundaries, indicating AstraVault's IAM team has applied boundary-based controls thoughtfully in some areas even though gaps remain in others.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-004 | HIGH | Privilege Escalation via IAM Policy Chaining (iam:PassRole + iam:CreatePolicyVersion) |
| CLOUD-001 | CRITICAL | Overly Permissive IAM Role Enables Cross-Account Administrative Privilege Escalation |
| CLOUD-005 | HIGH | Excessive Trust Relationship Between Production and Development AWS Accounts |

## Fictional IAM Privilege-Escalation Matrix

The matrix below documents all 10 escalation patterns tested against AstraVault's fictional environment, consistent with publicly documented AWS IAM privilege-escalation research applied here for illustrative purposes only.

| Pattern | Permission Combination | Escalation Technique | Result | Finding |
|---|---|---|---|---|
| EP-01 | `iam:CreateAccessKey on another user` | Create access keys for a higher-privileged user and authenticate as them | Not Found | — |
| EP-02 | `iam:CreateLoginProfile / UpdateLoginProfile on another user` | Set/reset a console password for a higher-privileged user | Not Found | — |
| EP-03 | `iam:AttachUserPolicy / AttachGroupPolicy (self)` | Attach AdministratorAccess (or similar) directly to self or own group | Not Found (boundary enforced) | — |
| EP-04 | `iam:PassRole + iam:CreatePolicyVersion + lambda:CreateFunction` | Create/modify a policy version on a higher-privileged role, pass it to a new Lambda function, execute under the elevated identity | Found | CLOUD-004 |
| EP-05 | `iam:PassRole + ec2:RunInstances (with instance profile)` | Launch an EC2 instance with a higher-privileged instance profile attached | Not Found (scoped) | — |
| EP-06 | `iam:UpdateAssumeRolePolicy` | Modify a role's trust policy to add a self-controlled principal | Not Found (boundary enforced) | — |
| EP-07 | `sts:AssumeRole on an overly-permissive cross-account trust` | Assume a role with a trust policy scoped to an entire account principal rather than a named identity | Found | CLOUD-001, CLOUD-005 |
| EP-08 | `iam:SetDefaultPolicyVersion (rollback to a permissive prior version)` | Roll back a policy to a previously-more-permissive version still retained in policy version history | Not Found | — |
| EP-09 | `iam:CreatePolicy + iam:AttachRolePolicy (via a wildcard PassRole boundary)` | Attach a newly created broad policy to a role reachable via a wildcard-resource iam:PassRole grant | Found (contributing factor) | CLOUD-001 |
| EP-10 | `cloudformation:CreateStack with an elevated iam:PassRole capability` | Deploy a CloudFormation stack that provisions resources under a higher-privileged service role | Not Found (scoped) | — |

# 18 S3 ASSESSMENT

TMG Security reviewed all 22 fictional production S3 buckets across bucket policies, Block Public Access configuration, ACLs and Object Ownership, default encryption, versioning, server access logging, access points, presigned URL usage, cross-account access, and backup/version retention. The complete bucket register appears in Appendix D.

## Illustrative Tests Performed

- Test unauthenticated anonymous GetObject/ListBucket against every discovered fictional bucket name.
- Review Block Public Access configuration at both the bucket and account level.
- Review bucket policies and ACLs for wildcard Principal grants or unnecessary cross-account access.
- Review default (SSE-KMS) encryption enforcement and Object Ownership configuration.
- Review S3 Versioning, Server Access Logging, and lifecycle/backup-retention configuration.
- Review presigned URL generation and expiry behavior in application upload/download flows.

## Illustrative Observations

One of AstraVault's 22 fictional production buckets — astravault-prod-financial-statements, storing customer account statement PDFs — was found fully reachable by unauthenticated internet requests, with Block Public Access disabled and a wildcard-Principal bucket policy (CLOUD-002), and the same bucket's versioning is suspended, removing a native recovery path against accidental or malicious overwrite (CLOUD-014). 6 of the 22 buckets lack default SSE-KMS encryption (CLOUD-013), 5 lack Server Access Logging (CLOUD-026), and the fictional vendor-integration bucket retains an overly broad cross-account resource policy left over from a since-changed integration pattern (CLOUD-012). The remaining buckets, including all buckets holding CloudTrail logs, Terraform state, and application artifacts, correctly enforce Block Public Access and encryption.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-002 | CRITICAL | Publicly Accessible S3 Bucket Exposes Sensitive Financial Data |
| CLOUD-012 | MEDIUM | Overly Broad S3 Bucket Resource Policy Grants Unnecessary Cross-Account Access |
| CLOUD-013 | MEDIUM | Missing Server-Side Encryption Enforcement on S3 Buckets Storing Sensitive Data |
| CLOUD-014 | MEDIUM | S3 Bucket Versioning Disabled on Buckets Storing Financial Records |
| CLOUD-026 | LOW | S3 Access Logging Not Enabled on Selected Buckets |

## S3 Bucket Register (Summary)

The table below summarizes the fictional bucket inventory reviewed during this assessment. The complete register with encryption and versioning detail for all 22 buckets appears in Appendix D.

| Bucket | Public Access | Finding |
|---|---|---|
| astravault-prod-financial-statements | Block Public Access: Disabled | CLOUD-002, CLOUD-014 |
| astravault-prod-analytics-exports | Block Public Access: Disabled | CLOUD-002 |
| astravault-prod-user-uploads | Block Public Access: Enabled | CLOUD-013 |
| astravault-prod-vendor-integration-drop | Block Public Access: Enabled | CLOUD-012 |
| astravault-prod-backups | Block Public Access: Enabled | — |
| astravault-prod-app-static-assets | Block Public Access: Enabled (OAC only) | — |
| astravault-prod-cloudtrail-logs | Block Public Access: Enabled | — |
| astravault-prod-access-logs | Block Public Access: Enabled | CLOUD-026 |
| astravault-prod-terraform-state | Block Public Access: Enabled | — |
| astravault-prod-lambda-artifacts | Block Public Access: Enabled | — |

*(Showing 10 of 22 fictional buckets; the complete register appears in Appendix D.)*

# 19 EC2 ASSESSMENT

TMG Security reviewed AstraVault's fictional EC2 fleet across launch-template/Auto Scaling group configuration, Instance Metadata Service (IMDS) settings, instance-profile role scope, golden AMI pipeline integrity, and Systems Manager patch-compliance coverage. The complete workload register appears in Appendix E.

## Illustrative Tests Performed

- Enumerate EC2 instances, launch templates, and Auto Scaling groups and review IMDS (IMDSv1/IMDSv2) configuration.
- Review instance-profile role scope attached to each EC2 workload tier.
- Review the golden AMI build pipeline for image signing and SBOM generation.
- Review Systems Manager Patch Manager compliance status across the production fleet.
- Test for unencrypted EBS volumes attached to production instances.

## Illustrative Observations

The fictional astravault-prod-api-worker Auto Scaling group, backing the public report-fetch feature, permits legacy IMDSv1 (HttpTokens: optional) rather than requiring IMDSv2, and was confirmed exploitable end-to-end via an SSRF-to-metadata chain (CLOUD-003). The instance profile attached to this fleet (astravault-ec2-app-role) also carries broader S3/DynamoDB access than the workload's function requires, directly amplifying CLOUD-003's impact and cross-referencing CLOUD-001's over-permission theme. The golden AMI pipeline lacks image signing and SBOM generation (CLOUD-021), and 6 of 44 fleet instances report no patch-compliance data to Systems Manager (CLOUD-038). EBS encryption was confirmed enabled fleet-wide.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-003 | CRITICAL | EC2 Workload Metadata Exposure Enables Credential Theft and Privilege Escalation |
| CLOUD-021 | MEDIUM | Missing Workload Integrity Controls on EC2 AMI Build Pipeline |
| CLOUD-038 | INFORMATIONAL | Systems Manager Patch Compliance Reporting Gaps on EC2 Fleet |

## EC2 / Workload Register (Summary)

| Resource | Type | IMDS Config | Finding |
|---|---|---|---|
| astravault-prod-api-worker (ASG, 6 instances) | EC2 Auto Scaling Group | IMDSv1 permitted (HttpTokens: optional) | CLOUD-003 |
| astravault-prod-mgmt-bastion | EC2 Instance | IMDSv2 required | CLOUD-006 |
| astravault-prod-batch-processor (ASG, 3 instances) | EC2 Auto Scaling Group | IMDSv2 required | — |
| astravault-prod-ecs-cluster | ECS Cluster (Fargate + EC2 capacity providers) | N/A (task-level metadata endpoint) | — |
| astravault-golden-ami (v2026.08) | AMI | Not enforced at image level | CLOUD-021 |
| astravault-dev-api-worker (ASG, 2 instances) | EC2 Auto Scaling Group | IMDSv1 permitted | — |
| astravault-prod-vpn-endpoint | EC2 Instance (Client VPN adjunct) | IMDSv2 required | — |
| astravault-prod-jump-analytics | EC2 Instance | IMDSv2 required | — |

# 20 VPC ASSESSMENT

TMG Security mapped AstraVault's fictional VPC topology across all three accounts — subnets, route tables, Internet/NAT Gateways, VPC endpoints, peering, and Transit Gateway attachments — to validate the segmentation model described in Section 11. The complete VPC/subnet/security-group register appears in Appendix F.

### Illustrative Tests Performed

- Map VPC, subnet, and route-table topology across all three fictional accounts.
- Review Internet Gateway and NAT Gateway attachment for unintended public routing.
- Review VPC Endpoint configuration for private connectivity to S3/Secrets Manager/KMS.
- Review VPC peering and Transit Gateway attachments for unintended cross-environment routes.
- Review VPC Flow Logs enablement across all VPCs in all three accounts.

### Illustrative Observations

AstraVault's Production VPC correctly implements the intended four-tier subnet model (public, private-application, private-database, management/security), and Transit Gateway route tables correctly segment Production from Development at the routing layer. However, VPC Flow Logs are not enabled on the fictional Development-account VPC (CLOUD-027), and VPC Endpoints for S3/Secrets Manager/KMS exist but are not yet used to eliminate the broad security-group egress rules flagged in Section 22 (CLOUD-016).

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-027 | LOW | VPC Flow Logs Not Enabled on Selected VPCs |

### VPC Register (Summary)

| Resource | Type | Notes | Finding |
|---|---|---|---|
| vpc-0fictprod01 | VPC | Public, Private-App, Private-DB, Management/Security subnet tiers | — |
| vpc-0fictsec01 | VPC | Log aggregation, break-glass tooling | — |
| vpc-0fictdev01 | VPC | Single-tier development/staging network | CLOUD-027 |
| subnet-pub-a/b (Production) | Public Subnet (x2, multi-AZ) | ALB, NAT Gateway, bastion | — |
| subnet-app-a/b (Production) | Private Application Subnet (x2, multi-AZ) | ECS/EC2 application workloads, Lambda ENIs | — |
| subnet-db-a/b (Production) | Private Database Subnet (x2, multi-AZ) | RDS primary/replica — astravault-analytics-reporting-db incorrectly placed in public subnet | CLOUD-010 |
| subnet-mgmt-a (Production) | Management/Security Subnet | Bastion, VPN endpoint, SSM-managed hosts | — |
| igw-0fictprod | Internet Gateway | Attached to public subnets only | — |

*(Showing VPC-level resources; security-group entries appear in Section 22 and the full register in Appendix F.)*

## 21 NETWORK SEGMENTATION

Beyond individual VPC and security-group configuration, TMG Security tested actual network-layer segmentation between AstraVault's fictional public, private-application, private-database, and management subnets, and between the Production and Development accounts.

### Illustrative Tests Performed

- Verify public/private/management subnet tiering matches the documented architecture.
- Test network-layer reachability from public subnet resources to the private database subnet.
- Review Network ACL rules as a defense-in-depth layer behind security groups.
- Test management/security subnet isolation from application workload subnets.
- Review Transit Gateway route-table segmentation between Production and Development.

### Illustrative Observations

Segmentation between the public, application, and management subnet tiers was confirmed effective, and Transit Gateway routing correctly isolates Production from Development at the network layer. The one segmentation exception identified is the analytics-reporting RDS instance's public-subnet placement (CLOUD-010, detailed further in Section 27), which places it outside the private-database-subnet boundary the rest of the environment correctly enforces. Network ACLs were found configured with the AWS default allow-all posture throughout, providing no meaningful defense-in-depth beyond security groups — an observation rather than a standalone finding, since security-group configuration is the primary control layer in AstraVault's design.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-010 | HIGH | RDS Database Instance Publicly Reachable From the Internet |

# 22 SECURITY GROUPS

TMG Security reviewed all 24 fictional production security groups' inbound and outbound (egress) rules, cross-referencing internet-facing rules against external port-scan results. The complete register appears in Appendix F.

## Illustrative Tests Performed

- Enumerate all security groups and flag inbound rules from 0.0.0.0/0 on administrative ports.
- Test external reachability of flagged administrative ports from the internet.
- Review egress rule scope across all production security groups.
- Review security-group-to-security-group referencing for tiered application architectures.
- Review RDS/database security-group source scoping for direct internet exposure.

## Illustrative Observations

The fictional astravault-prod-mgmt-sg security group, attached to both the management bastion and (via a shared association) the analytics-reporting RDS instance, permits inbound SSH (22), RDP (3389), and PostgreSQL (5432) from 0.0.0.0/0 (CLOUD-006), and 18 of the 24 reviewed security groups retain the AWS default unrestricted egress rule rather than a scoped policy (CLOUD-016). The primary application-tier and primary-database security groups were, by contrast, found correctly scoped to security-group-to-security-group references rather than broad CIDR ranges, demonstrating the pattern AstraVault should extend to the remaining groups.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-006 | HIGH | Security Group Permits Unrestricted Administrative Access to Management Ports |
| CLOUD-016 | MEDIUM | Permissive Security Group Egress Rules Allow Unrestricted Outbound Access |

## Security Group Register (Summary)

| Security Group | Scope | Notes | Finding |
|---|---|---|---|
| sg-0fictmgmt01 (astravault-prod-mgmt-sg) | Bastion + RDS-adjacent hosts | Inbound 22/3389/5432 from 0.0.0.0/0 | CLOUD-006 |
| sg-0fictapp01 (astravault-prod-app-sg) | Application/API worker tier | Inbound 443 from ALB SG only; default all-traffic egress | CLOUD-016 |
| sg-0fictdb01 (astravault-prod-db-sg) | RDS primary transactional database | Inbound 5432 from app-sg only (correctly scoped) | — |
| sg-0fictanalyticsdb (astravault-analytics-db-sg) | Analytics reporting RDS instance | Inbound 5432 from 0.0.0.0/0 (public-subnet placement) | CLOUD-010 |
| nacl-0fictprodapp | Private Application Subnet | Default allow-all (no additional restriction beyond security groups) | — |

# 23 LAMBDA ASSESSMENT

TMG Security reviewed all 14 fictional production Lambda functions across execution-role scope, environment-variable configuration, resource-based policies, and configured timeout/memory versus observed usage. The complete Lambda/API register appears in Appendix G.

## Illustrative Tests Performed

- Enumerate all Lambda functions and map each to its execution role.
- Review execution-role permission scope against each function's actual resource-access needs.
- Review function environment variables for plaintext secret material.
- Review function timeout/memory configuration against observed CloudWatch Insights metrics.
- Test function resource-based policies for unintended public or cross-account invoke access.
- Review Lambda function URL / API Gateway integration authorization configuration.

## Illustrative Observations

All 14 fictional Lambda functions share a single execution role (astravault-lambda-exec-role) granting s3:*, dynamodb:*, and secretsmanager:GetSecretValue on all resources — far exceeding what any individual function needs, and concentrating the blast radius of any one function's compromise across all 14 (CLOUD-009). The astravault-prod-webhook-processor function, the most internet-adjacent of the 14, is additionally provisioned with the maximum timeout and memory allocation against observed usage two orders of magnitude smaller (CLOUD-020). No function was found with plaintext secrets in its environment configuration, and resource-based policies were correctly scoped throughout.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-009 | HIGH | Lambda Execution Role Grants Excessive IAM Permissions Beyond Function Requirements |
| CLOUD-020 | MEDIUM | Lambda Function Configured With Excessive Timeout and Memory Beyond Operational Need |

# 24 API GATEWAY

TMG Security tested AstraVault's public and internal fictional API Gateway deployments — 27 routes on api.astravault-example.com and a separate internal operations API — for authorization correctness, request validation, rate limiting, and SSRF resistance in the report-fetch feature specifically.

## Illustrative Tests Performed

- Enumerate all API Gateway REST/HTTP APIs, stages, and authorization configuration.
- Test object/broken-object-level authorization on report-fetch and account-data endpoints.
- Review request-validation model configuration across all public routes.
- Test rate-limiting and throttling behavior on authentication and funds-transfer endpoints.
- Review API Gateway access-logging configuration for audit-trail completeness.
- Test the report-fetch feature's outbound URL handling for SSRF (metadata and internal-range redirection).

## Illustrative Observations

The report-fetch feature on api.astravault-example.com does not validate the destination of its outbound fetch, and was confirmed exploitable for SSRF-to-metadata credential theft (CLOUD-003, detailed fully in Section 39). No rate-based WAF rule protects the login or funds-transfer endpoints (CLOUD-030, cross-referenced from Section 34), and 3 of 27 public routes lack a request-validation model, though no exploitable condition resulted during testing (CLOUD-039, informational). Broken object-level authorization testing against account-data and transfer-confirmation endpoints found correct authorization enforcement throughout — a positive result given this class of vulnerability's prevalence in real-world API assessments.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-003 | CRITICAL | EC2 Workload Metadata Exposure Enables Credential Theft and Privilege Escalation |

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-030 | LOW | WAF Rule Set Missing Rate-Based Rule on Public API Gateway |
| CLOUD-039 | INFORMATIONAL | API Gateway Missing Request Validation on Selected Public Endpoints |

## Lambda / API Register (Summary)

| Function / API | Purpose | Finding |
|---|---|---|
| astravault-prod-statement-generator | Generates monthly customer account statements | — |
| astravault-prod-webhook-processor | Processes inbound partner/payment-aggregator webhooks | CLOUD-009, CLOUD-020 |
| astravault-prod-fraud-check | Fraud-monitoring pipeline consumer (SNS subscriber) | — |
| astravault-prod-transaction-router | Routes transaction events between services (EventBridge target) | CLOUD-037 |
| astravault-prod-report-export-worker | Async generation of analytics export files | — |
| astravault-prod-user-notification | Sends transactional email/SMS notifications | — |
| astravault-prod-kyc-doc-processor | Processes uploaded KYC documents | — |
| astravault-prod-session-cleanup | Scheduled cleanup of expired session records | — |
| astravault-prod-audit-log-shipper | Ships application audit logs to centralized pipeline | — |
| astravault-prod-payment-reconciliation | Daily payment-aggregator reconciliation job | — |
| astravault-prod-pdf-render | Renders statement PDFs from templated data | — |
| astravault-prod-webhook-retry | Retries failed outbound partner webhook deliveries | — |
| astravault-prod-analytics-aggregator | Aggregates daily analytics rollups | — |
| astravault-prod-support-ticket-sync | Syncs support tickets with fictional CRM | — |
| api.astravault-example.com (REST API) | Primary customer-facing / mobile-and-web backend API (27 routes) | CLOUD-030, CLOUD-039 |
| console.astravault-example.com (Internal API) | Internal operations/admin console API | — |

# 25 CONTAINER SECURITY

TMG Security reviewed ECS task-definition security configuration across AstraVault's fictional container workloads, including privilege mode, Linux capabilities, task-role scoping, network mode, and runtime-isolation configuration.

## Illustrative Tests Performed

- Review ECS task definitions for privileged mode and added Linux capabilities.
- Review ECS task role and execution role scoping per service.
- Test container runtime isolation and host-mount configuration for workload separation.
- Review ECS service network mode and security-group assignment.
- Review GuardDuty ECS Runtime Monitoring / Malware Protection coverage for container workloads.

## Illustrative Observations

The fictional astravault-api-gateway-service ECS task definition is not fully privileged but explicitly adds the SYS_ADMIN and NET_ADMIN Linux capabilities beyond the service's documented functional need, meaningfully increasing container-escape feasibility in the event of code-level compromise (CLOUD-024). Task roles were otherwise found individually scoped per service (in contrast to the shared Lambda execution-role pattern in Section 23), host-mount configuration showed no unnecessary bind-mounts, and GuardDuty ECS Runtime Monitoring is enabled but not yet routed to active alerting (cross-referenced from CLOUD-029 in Section 32).

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-024 | MEDIUM | ECS Task Definition Runs Containers With Elevated Linux Capabilities |

# 26 ECR

TMG Security reviewed all 11 fictional production ECR repositories for image-scanning configuration, current scan findings on deployed image tags, repository cross-account access policy, and image lifecycle-policy configuration. The complete register appears in Appendix H.

## Illustrative Tests Performed

- Enumerate all ECR repositories and review image-scanning (scan-on-push) configuration.
- Review ECR scan findings for currently-deployed production image tags.
- Verify whether the CI/CD pipeline gates deployment on Critical/High ECR scan findings.
- Review ECR repository policy for unintended cross-account pull/push access.
- Review ECR image lifecycle-policy configuration for unbounded image retention.

## Illustrative Observations

The fictional astravault/analytics-worker repository's currently-deployed production image tag carries a Critical-severity scan finding for an outdated, remotely-exploitable base-image package, and the CI/CD pipeline does not gate deployment on scan severity — the vulnerable image was deployed despite the finding being available at build time (CLOUD-008). 9 of the 11 reviewed repositories have no image lifecycle policy, resulting in unbounded retention of historical (and potentially more-vulnerable) image tags (CLOUD-031). Repository access policies were correctly scoped with no unintended cross-account grants.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-008 | HIGH | Vulnerable Container Image With Exploitable Package in Production ECR Repository |
| CLOUD-031 | LOW | ECR Repository Missing Image Lifecycle Policy (Unbounded Image Retention) |

## Container / ECR Register (Summary)

| Repository / Task | Scan Status | Finding |
|---|---|---|
| astravault/api-gateway-service | Pass (0 Critical/High) | CLOUD-024 |
| astravault/analytics-worker | Fail (1 Critical, 4 High) | CLOUD-008 |
| astravault/webhook-processor-container | Pass (0 Critical/High) | CLOUD-031 |
| astravault/statement-generator | Pass (0 Critical/High) | CLOUD-031 |
| astravault/fraud-detection-model-server (fictional) | Pass (2 Medium) | CLOUD-031 |
| astravault/internal-admin-console | Pass (0 Critical/High) | CLOUD-031 |
| astravault/reconciliation-batch | Pass (1 Medium) | CLOUD-031 |
| astravault/notification-service | Pass (0 Critical/High) | CLOUD-031 |
| astravault/kyc-doc-processor | Pass (0 Critical/High) | CLOUD-031 |
| astravault/support-sync-service | Pass (0 Critical/High) | CLOUD-031 |
| astravault/legacy-batch-runner (fictional, deprecated) | Fail (3 High — unscanned since deprecation) | CLOUD-031 |
| astravault-api-gateway-service (ECS task def :41) | privileged: false; capabilities.add: SYS_ADMIN, NET_ADMIN | CLOUD-024 |

# 27 RDS

TMG Security reviewed all 5 fictional RDS/managed-database resources for public accessibility, storage/snapshot encryption, backup retention, and IAM database authentication availability. The complete database register appears in Appendix I.

## Illustrative Tests Performed

- Enumerate all RDS instances and review Publicly Accessible configuration.
- Test external network-layer reachability of RDS endpoints (connection only, no authentication bypass attempted).
- Review RDS storage and snapshot encryption configuration (KMS).
- Review RDS automated backup retention and cross-region snapshot copy configuration.
- Review RDS IAM database authentication availability as an alternative to static passwords.

## Illustrative Observations

The fictional astravault-analytics-reporting-db instance is configured with Publicly Accessible: true and resides in a public subnet, confirmed reachable at the network layer from the internet (CLOUD-010) — inconsistent with the private-subnet pattern correctly applied to the primary transactional astravault-prod-financial-db instance. Storage encryption was confirmed enabled on all 5 databases. Backup retention on the primary financial database is set to only 7 days with no cross-region copy or Vault Lock immutability (CLOUD-019, detailed in Section 37). IAM database authentication is available but not yet adopted for the primary financial database, tracked as part of the Secrets Management rotation gap in Section 28.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-010 | HIGH | RDS Database Instance Publicly Reachable From the Internet |
| CLOUD-019 | MEDIUM | Weak AWS Backup Configuration for Production Database Resources |

## Database Register

| Database | Engine | Publicly Accessible | Finding |
|---|---|---|---|
| astravault-prod-financial-db | Aurora PostgreSQL 15 (fictional) | No | CLOUD-019 |
| astravault-analytics-reporting-db | RDS PostgreSQL 15 (fictional) | Yes | CLOUD-010 |
| astravault-prod-sessions-cache (fictional) | ElastiCache Redis (fictional) | No | — |
| astravault-dev-financial-db-replica | Aurora PostgreSQL 15 (fictional, Development account) | No | CLOUD-041 |
| astravault-prod-audit-log-db (fictional) | DynamoDB (fictional) | No (private VPC endpoint only) | — |

# 28 SECRETS MANAGEMENT

TMG Security reviewed AstraVault's fictional secret-handling practices across ECS/Lambda workload configuration, AWS Secrets Manager, and Systems Manager Parameter Store, searching for plaintext secret exposure and reviewing rotation configuration for stored credentials.

## Illustrative Tests Performed

- Search ECS/Lambda/EC2 workload configuration for plaintext secret material.
- Enumerate Secrets Manager secrets and review rotation configuration.
- Review Secrets Manager resource policies for unintended cross-account access.
- Review Systems Manager Parameter Store SecureString usage for sensitive configuration values.
- Test for secrets committed to the fictional application's build artifacts or container layers.

### Illustrative Observations

The fictional astravault-api-gateway-service ECS task definition sets the production database password and a fictional payment-aggregator API key directly as plaintext environment variables rather than referencing Secrets Manager (CLOUD-007) — the most significant secrets-management finding in this assessment. Separately, the properly-stored astravault/prod/rds/financial-db-credentials Secrets Manager entry does not have automatic rotation configured despite native RDS rotation-Lambda support being readily available (CLOUD-032). No secrets were found committed to build artifacts or container image layers, and Parameter Store SecureString usage was consistent throughout the remainder of the environment.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-007 | HIGH | Secrets Exposed Through Workload Environment Configuration |
| CLOUD-032 | LOW | Secrets Manager Rotation Not Configured for Long-Lived Database Credentials |

## 29 KMS

TMG Security reviewed AstraVault's fictional customer-managed KMS key inventory, focusing on key-policy scope, rotation configuration, and condition-based restriction (kms:ViaService, kms:CallerAccount) on keys protecting the most sensitive data stores.

### Illustrative Tests Performed

- Enumerate all customer-managed KMS keys and review key-policy scope.
- Review KMS key rotation configuration for all customer-managed keys.
- Test kms:ViaService and kms:CallerAccount condition enforcement on sensitive key policies.
- Review use of AWS-managed vs. customer-managed keys across sensitive data stores.
- Review CloudTrail coverage of kms:Decrypt calls for anomaly-detection readiness.

### Illustrative Observations

The fictional alias/astravault-prod-data-key customer-managed key, used to encrypt several sensitive S3 buckets and Secrets Manager entries, retains the default "Enable IAM User Permissions" key-policy statement alongside application-specific grants, effectively allowing any sufficiently-permissive Production IAM principal to decrypt protected data independent of a dedicated key-specific grant (CLOUD-023). Key rotation was confirmed enabled on all customer-managed keys, and AWS-managed keys were appropriately used for lower-sensitivity data stores.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-023 | MEDIUM | KMS Key Policy Grants Overly Broad Decrypt Permissions |

# 30 CLOUDTRAIL

TMG Security reviewed AstraVault's fictional CloudTrail configuration — the astravault-org-trail organization trail — for multi-region coverage, management- and data-event logging scope, log-file integrity validation, and log-delivery bucket protection.

## Illustrative Tests Performed

- Verify the organization trail is multi-region and covers all three fictional accounts.
- Review CloudTrail data-event selector configuration for sensitive S3 buckets.
- Review CloudTrail data-event selector configuration for Lambda function invocations.
- Test CloudTrail log-file validation (digest) enablement and log-file integrity.
- Review the CloudTrail log-delivery S3 bucket policy for tamper-resistance.

## Illustrative Observations

The organization trail correctly covers all three fictional accounts with multi-region logging and management events enabled, and log-file validation with a tamper-resistant delivery-bucket policy is correctly configured — both positive controls. However, no data-event selectors are configured for either S3 (on the sensitive financial-statements and analytics-exports buckets) or Lambda invocations, leaving no object-level or invocation-level audit trail for the environment's most sensitive resources (CLOUD-011).

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-011 | HIGH | CloudTrail Logging Coverage Gap on Security-Sensitive API Activity |

# 31 CLOUDWATCH

TMG Security reviewed CloudWatch metric-filter alarm coverage against a 12-alarm baseline aligned to high-risk security events, alongside Log Group retention configuration and dashboard/on-call visibility.

## Illustrative Tests Performed

- Review CloudWatch metric-filter alarm coverage against the CIS AWS Foundations-aligned baseline.
- Test alarm routing and escalation for root-account-usage and unauthorized-API-call alarms.
- Review CloudWatch Log Group retention configuration across all log groups.
- Review CloudWatch Logs Insights query access scoping for sensitive log data.
- Review CloudWatch dashboard and composite alarm coverage for executive/on-call visibility.

## Illustrative Observations

Of the 12 recommended baseline high-risk-event alarms, only 4 are currently configured — root-account usage and unauthorized-API-call patterns are covered, but IAM policy changes, security-group changes, Network ACL changes, and CloudTrail/GuardDuty configuration changes are not (CLOUD-015). Separately, 9 of 31 CloudWatch Log Groups are configured with indefinite retention rather than a policy-aligned period, a minor cost and data-minimization observation (CLOUD-040).

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-015 | MEDIUM | Insufficient CloudWatch Alerting on High-Risk IAM and Security Events |
| CLOUD-040 | INFORMATIONAL | CloudWatch Log Group Retention Set to Indefinite on Non-Critical Logs |

# 32 GUARDDUTY

TMG Security reviewed GuardDuty enablement, protection-plan coverage, finding-suppression configuration, and delegated-administrator setup across the fictional AWS Organization.

### Illustrative Tests Performed

- Verify GuardDuty is enabled with appropriate protection plans (S3, EKS, Malware, RDS) across all accounts.
- Review EventBridge routing of GuardDuty findings to an alerting/chat-ops channel.
- Review GuardDuty finding-suppression rules for overly broad exclusions.
- Test GuardDuty delegated-administrator configuration for centralized findings visibility.

### Illustrative Observations

GuardDuty is enabled organization-wide with appropriate protection plans and centralized delegated-administrator visibility from the fictional Security account — a solid baseline. However, no EventBridge rule routes findings to an active alerting or chat-ops channel, meaning GuardDuty's detective value currently depends on a team member proactively checking the console rather than being proactively notified (CLOUD-029). No overly broad suppression rules were identified.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-029 | LOW | GuardDuty Findings Not Routed to a Centralized Alerting Channel |

## 33 SECURITY HUB

TMG Security reviewed Security Hub standard-subscription configuration, finding aggregation across GuardDuty/Config/Inspector, and custom-insight configuration across the fictional AWS Organization.

### Illustrative Tests Performed

- Verify Security Hub is enabled and subscribed to the AWS Foundational Security Best Practices standard in all accounts.
- Review Security Hub finding aggregation across GuardDuty, Config, and Inspector.
- Review Security Hub custom-insight configuration for high-risk finding tracking.
- Test cross-account Security Hub delegated-administrator visibility from the Security account.

### Illustrative Observations

Security Hub is enabled with the AWS Foundational Security Best Practices standard subscribed in the Production and Security accounts, correctly aggregating GuardDuty, Config, and Inspector findings with centralized delegated-administrator visibility. The Development account, however, is not subscribed to any standard (CLOUD-034), consistent with the broader Development-account baseline gap discussed in Section 12 and CLOUD-041.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-034 | INFORMATIONAL | Security Hub Standard Subscriptions Incomplete Across Accounts |

# 34 WAF

TMG Security reviewed the fictional astravault-public-api-waf Web ACL protecting the public API Gateway stage, covering managed rule-group coverage, rate-based rules, bot protection, and logging.

## Illustrative Tests Performed

- Enumerate WAF Web ACLs and review AWS Managed Rule Group coverage.
- Review rate-based rule coverage on authentication and funds-transfer endpoints.
- Test WAF Bot Control / CAPTCHA coverage on the login and account-creation flows.
- Review WAF logging configuration for blocked/allowed request visibility.
- Test WAF managed-rule coverage against common injection and known-bad-input payloads (non-destructive).

## Illustrative Observations

The Web ACL correctly includes AWS Managed Rule Groups for common threats (SQL injection, known bad inputs) and logging is enabled with full visibility into blocked/allowed requests. However, no rate-based rule is configured, leaving the API without WAF-layer protection against high-volume credential-stuffing or scraping traffic on sensitive endpoints such as login and funds-transfer beyond API Gateway's own coarser account-level throttling (CLOUD-030). Injection and known-bad-input payload testing was correctly blocked by the existing managed rules.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-030 | LOW | WAF Rule Set Missing Rate-Based Rule on Public API Gateway |

# 35 CLOUDFRONT

TMG Security reviewed the fictional CloudFront distribution (E1FICTIONALCF01) fronting app.astravault-example.com, covering origin access control, response-header policy enforcement, and geo-restriction/signed-URL configuration.

## Illustrative Tests Performed

- Enumerate CloudFront distributions and review origin access control configuration.
- Review Response Headers Policy attachment for security-header enforcement at the edge.
- Capture and review actual response headers across a representative sample of application routes.
- Test CloudFront-to-origin traffic for TLS enforcement and origin custom-header validation.
- Review CloudFront geo-restriction and signed-URL/cookie configuration for restricted content paths.

## Illustrative Observations

Origin Access Control is correctly configured, preventing direct origin access that bypasses the CDN, and origin traffic correctly enforces TLS with a custom-header validation check. However, no CloudFront Response Headers Policy is attached, and response-header capture across sampled routes confirmed inconsistent (and on several routes, entirely absent) Content-Security-Policy, Strict-Transport-Security, X-Frame-Options, and X-Content-Type-Options headers, since the origin application itself sets them inconsistently and there is no edge-level backstop (CLOUD-022).

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-022 | MEDIUM | Insufficient Security-Header Controls at CloudFront Edge |

# 36 CROSS-ACCOUNT SECURITY

Building on the IAM trust-relationship review in Sections 16-17, TMG Security specifically enumerated and live-tested every cross-account IAM trust relationship across AstraVault's three fictional AWS accounts, since account segmentation is only as strong as the trust relationships that bridge it.

## Illustrative Tests Performed

- Enumerate all cross-account IAM trust relationships across the three fictional accounts.
- Test AssumeRole reachability of the Security-account cross-account role from a Production low-privilege identity.
- Test AssumeRole reachability of the Production support role from a Development-account identity.
- Review ExternalId and MFA condition enforcement on all cross-account trust policies.
- Review AWS Organizations SCP coverage restricting cross-account privilege escalation.

## Illustrative Observations

Two of AstraVault's cross-account trust relationships were confirmed overly broad: the Security account's log-aggregation role trusts the entire Production account principal with iam:*-level permissions once assumed (CLOUD-001), and the Production support role trusts the entire, lower-assurance Development account principal for read access to Production secrets (CLOUD-005). Neither trust policy requires an ExternalId or MFA condition. A small number of other cross-account roles — including the fictional break-glass administrative role — were found correctly scoped to named principals with hardware-MFA and approval logging.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-001 | CRITICAL | Overly Permissive IAM Role Enables Cross-Account Administrative Privilege Escalation |
| CLOUD-005 | HIGH | Excessive Trust Relationship Between Production and Development AWS Accounts |

# 37 BACKUP & RECOVERY

TMG Security reviewed AWS Backup plan coverage, retention configuration, cross-region/cross-account copy, and Vault Lock immutability protection for AstraVault's fictional production data resources.

## Illustrative Tests Performed

- Review AWS Backup plan coverage and retention configuration for production database resources.
- Review AWS Backup Vault Lock configuration for immutability against deletion.
- Test cross-region/cross-account backup copy configuration for disaster-recovery resilience.
- Review backup-vault access policy for overly broad delete permissions.
- Review restore-testing cadence and documentation for production backup sets.

## Illustrative Observations

The AWS Backup plan covering the primary fictional financial database retains backups for only 7 days with no cross-region copy and no Vault Lock immutability protection, meaning any principal with backup:DeleteRecoveryPoint permission could remove recent recovery points entirely (CLOUD-019). AstraVault's documented internal target is 35 days' retention for financial database backups, a gap the fictional cloud team acknowledged during the authenticated review interview. Restore-testing documentation exists but was last exercised over a year prior to this assessment, a secondary observation folded into the same finding.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-019 | MEDIUM | Weak AWS Backup Configuration for Production Database Resources |

# 38 CI/CD SECURITY

TMG Security reviewed AstraVault's fictional cloud deployment pipeline — source repository (name withheld/generic per this report's scope), build system, container build, ECR push, and the OIDC-federated deployment role used to reach production — for secure credential handling, artifact integrity, and deployment gating.

## Illustrative Tests Performed

- Review CI/CD pipeline OIDC/workload-identity federation configuration for the deployment role.
- Review production deployment-role permission scope against least privilege.
- Test whether ECR scan findings block the CI/CD deployment pipeline.
- Review pipeline secret handling for hard-coded credentials in build configuration.
- Review artifact integrity verification (image signing) between build and deployment.
- Review Infrastructure-as-Code drift between templated and deployed resource configuration.

## Illustrative Observations

The deployment pipeline correctly uses OIDC/workload-identity federation rather than long-lived static credentials, and no hard-coded secrets were found in build configuration — both positive, modern practices. However, the pipeline does not gate deployment on ECR scan severity, allowing the Critical-severity vulnerable image identified in Section 26 to reach production (CLOUD-008), the AMI build pipeline lacks image signing/SBOM generation (CLOUD-021, cross-referenced from Section 19), and a fictional terraform plan dry-run identified 12 resources with configuration drift from their templated definition, mostly manually-applied security-group changes made during past incident response (CLOUD-036).

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-008 | HIGH | Vulnerable Container Image With Exploitable Package in Production ECR Repository |
| CLOUD-021 | MEDIUM | Missing Workload Integrity Controls on EC2 AMI Build Pipeline |
| CLOUD-036 | INFORMATIONAL | Infrastructure-as-Code Drift Observed Between Deployed and Templated Resources |

# 39 METADATA SERVICE SECURITY

TMG Security specifically tested AstraVault's fictional Instance Metadata Service (IMDS) configuration and the application-layer SSRF resistance of the report-fetch feature identified as internet-facing in Section 13, since this combination is one of the highest-impact cloud attack patterns and underlies this report's third Critical finding.

## Illustrative Tests Performed

- Verify IMDSv2 (HttpTokens: required) enforcement across all EC2 launch templates and Auto Scaling groups.
- Test HttpPutResponseHopLimit configuration as a defense-in-depth SSRF mitigation.
- Test application-layer outbound-fetch features for link-local (169.254.0.0/16) destination validation.
- Review ECS task metadata endpoint (v4) access scoping within the container network namespace.
- Test DNS-rebinding resistance of the report-fetch feature's outbound URL validation.

## Illustrative Observations

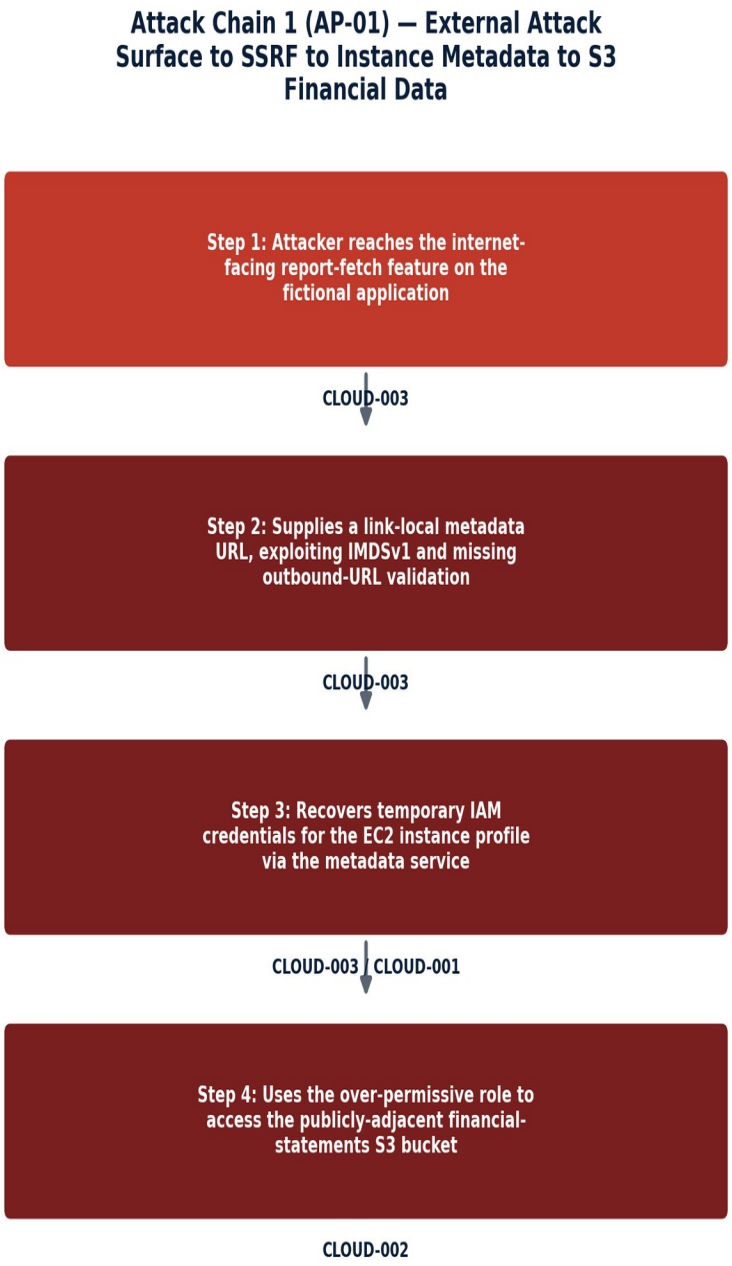
The fictional astravault-prod-api-worker Auto Scaling group permits IMDSv1 with no session-token requirement, and the report-fetch feature performs no destination validation on user-supplied export URLs — TMG Security confirmed the full SSRF-to-metadata-to-credential-theft chain end-to-end in a controlled, non-destructive manner (CLOUD-003, and see Attack Chain 1 in Section 40). ECS task metadata endpoint access was found correctly scoped within the container network namespace for other workloads, and DNS-rebinding testing against the report-fetch feature is tracked as a supplementary observation pending the primary IMDSv2/allow-list remediation.

| Finding ID | Severity | Title |
|---|---|---|
| CLOUD-003 | CRITICAL | EC2 Workload Metadata Exposure Enables Credential Theft and Privilege Escalation |

# 40 ATTACK PATH ANALYSIS

Individual findings, viewed in isolation, can understate real-world cloud risk. This section chains this report's findings into 4 realistic, fictional, non-destructive composite attack scenarios, each grounded entirely in findings documented elsewhere in this report and cross-referenced by finding ID. TMG Security validated the technical feasibility of each step in a controlled manner — no chain was executed end-to-end against a live system, and no destructive or state-changing action was taken at any point.

## Attack Chain 1 — External Attack Surface → SSRF → Instance Metadata → S3 Financial Data



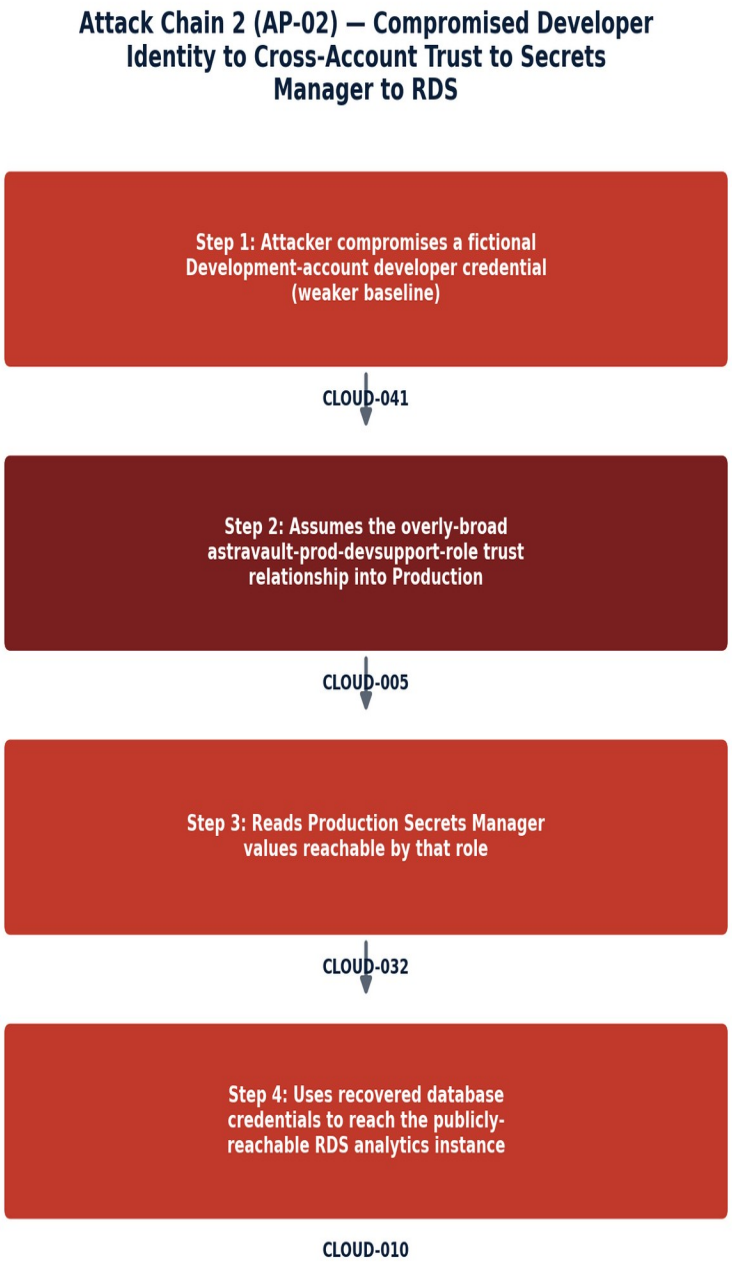

*Fictional composite attack chain 1 — illustrative sample only. ILLUSTRATIVE / SYNTHETIC EVIDENCE.*

- Step 1: An unauthenticated attacker discovers the internet-facing report-fetch feature on api.astravault-example.com during external reconnaissance (Section 13).
- Step 2: The attacker supplies a link-local metadata URL as the export destination; the feature performs no destination validation, and the backing EC2 fleet permits legacy IMDSv1 with no session-token requirement (CLOUD-003).
- Step 3: The attacker recovers live, temporary IAM credentials for the astravault-ec2-app-role instance profile directly from the metadata response.

- Step 4: That role's permissions — broader than the workload's actual function requires, echoing the over-permission pattern in CLOUD-001 — allow read access to the publicly-adjacent astravault-prod-financial-statements S3 bucket (CLOUD-002), which was already independently confirmed reachable by any unauthenticated user.

*Business impact: this chain demonstrates that AstraVault's most severe fictional exposure requires no insider access and no social engineering — only a single unauthenticated application-layer weakness, converted into cloud credential theft and sensitive financial-data access through two further, independently-documented misconfigurations.*

## Attack Chain 2 — Compromised Developer Identity → Cross-Account Trust → Secrets Manager → RDS Access



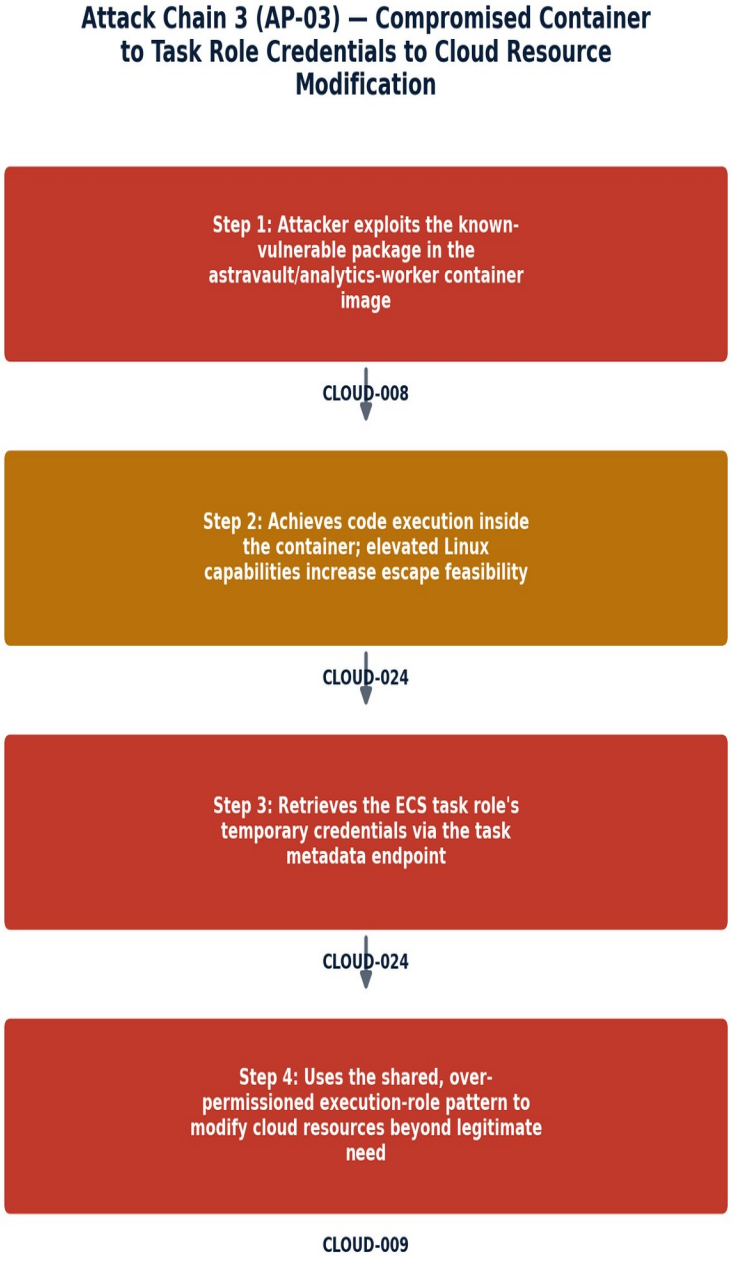

*Fictional composite attack chain 2 — illustrative sample only. ILLUSTRATIVE / SYNTHETIC EVIDENCE.*

- Step 1: An attacker compromises a fictional Development-account developer credential — the more attainable target given the Development account's weaker baseline controls (CLOUD-041).
- Step 2: The attacker assumes the astravault-prod-devsupport-role using the overly broad Production-to-Development cross-account trust relationship, which requires no MFA or ExternalId (CLOUD-005).
- Step 3: Using the assumed role's read access, the attacker retrieves the financial-database credential from Secrets Manager — correctly stored, but not rotated on a defined schedule (CLOUD-032).

- Step 4: The attacker uses the recovered credential to connect to the astravault-analytics-reporting-db instance, which is independently reachable at the network layer from the internet (CLOUD-010).

*Business impact: this chain shows how a gap in account segmentation (Section 12, Section 36) allows a foothold in AstraVault's lowest-assurance environment to reach production database credentials and a network-exposed database instance, entirely bypassing the isolation the account structure is intended to provide.*

## Attack Chain 3 — Compromised Container → Task Role Credentials → Excessive IAM Permissions → Cloud Resource Modification



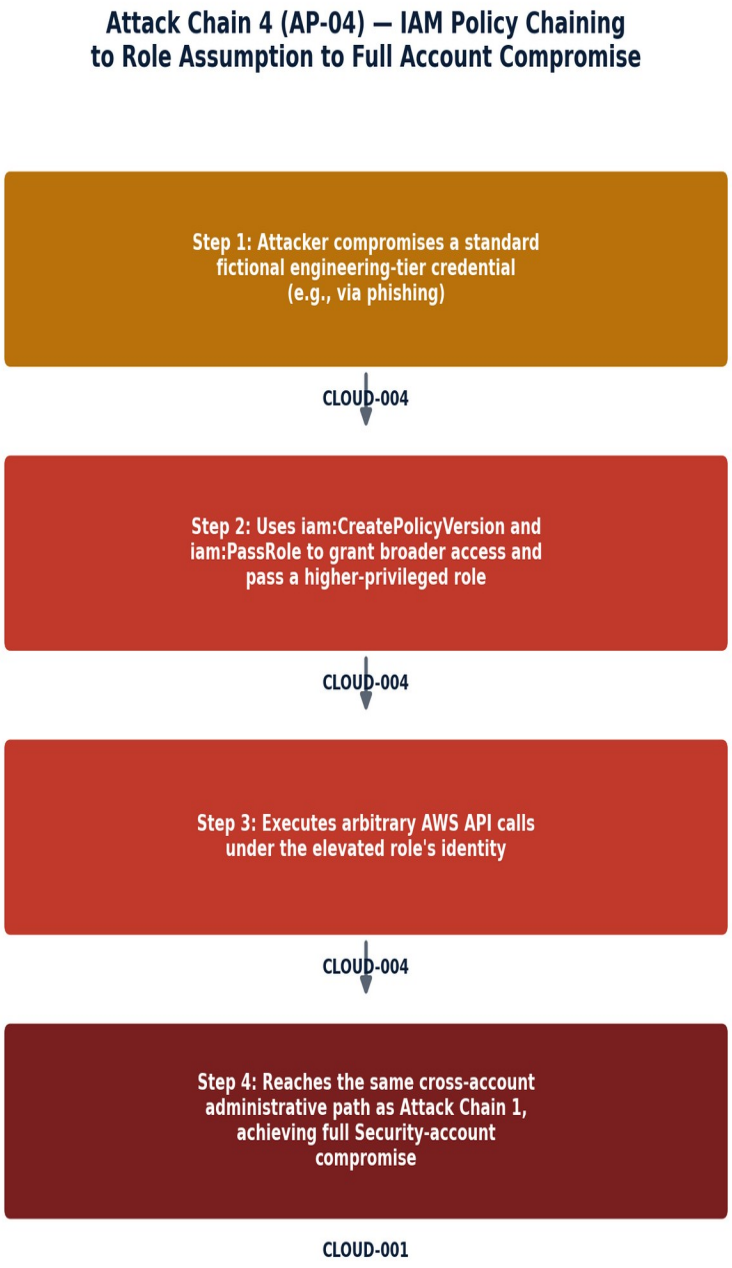

Fictional composite attack chain — illustrative sample only. ILLUSTRATIVE / SYNTHETIC EVIDENCE.

*Fictional composite attack chain 3 — illustrative sample only. ILLUSTRATIVE / SYNTHETIC EVIDENCE.*

- Step 1: An attacker exploits the known-vulnerable base-image package in the currently-deployed astravault/analytics-worker container image, which the CI/CD pipeline did not block despite an available Critical-severity scan finding (CLOUD-008).
- Step 2: The container's elevated Linux capabilities (SYS_ADMIN, NET_ADMIN), broader than its documented functional need, increase the feasibility of moving from application-level code execution toward the container boundary (CLOUD-024).
- Step 3: The attacker retrieves the ECS task role's temporary credentials via the in-container task metadata endpoint.
- Step 4: Because Lambda/task execution roles in AstraVault's environment follow a broadly-shared, over-permissioned pattern (CLOUD-009), the attacker uses the recovered credentials to modify cloud resources (e.g., S3 objects, DynamoDB records) beyond the compromised service's legitimate function.

*Business impact: this chain illustrates how a single unpatched container dependency, combined with excess container privilege and over-permissioned execution roles, escalates from application-level compromise to unauthorized cloud-resource modification — all three contributing weaknesses independently remediable.*

## Attack Chain 4 — IAM Policy Chaining → Role Assumption → Full Account Compromise




*Fictional composite attack chain 4 — illustrative sample only. ILLUSTRATIVE / SYNTHETIC EVIDENCE.*

- Step 1: An attacker compromises a standard fictional engineering-tier credential (e.g., via phishing) — a much lower bar than compromising an already-privileged identity.
- Step 2: Using the astravault-platform-engineers group's iam:CreatePolicyVersion and iam:PassRole permissions, the attacker creates a new policy version granting broader access and passes an elevated role to a newly created Lambda function — a documented IAM privilege-escalation pattern (CLOUD-004).
- Step 3: The attacker executes arbitrary AWS API calls under the elevated role's identity, including further AssumeRole calls.

- Step 4: This reaches the same overly-broad cross-account trust path documented in Attack Chain 1 and CLOUD-001, achieving full fictional Security-account compromise from a starting point that required no privileged access at all.

*Business impact: this chain demonstrates that AstraVault's most critical exposure (CLOUD-001) is reachable not only by directly compromising a highly-privileged identity, but via a documented, automatable technique from a standard engineering credential — meaningfully widening the realistic population of attack starting points.*

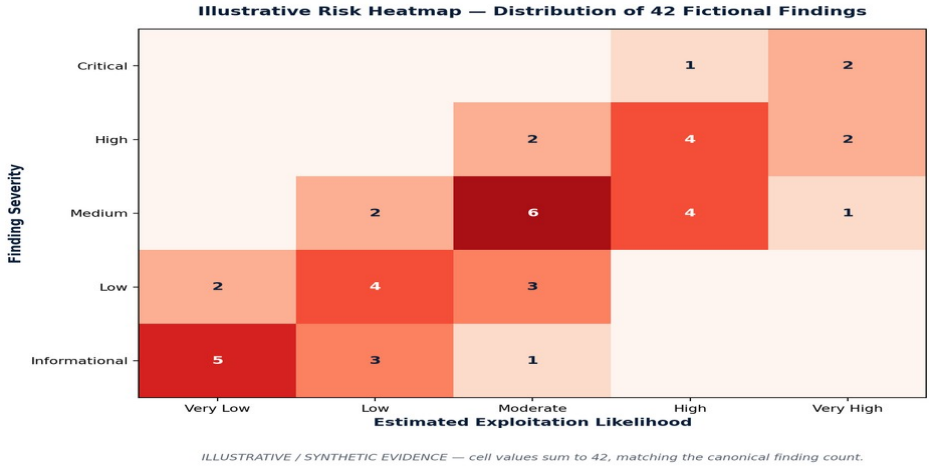
## Attack Path Summary Table

| Path | Findings Involved | Entry Point |
|---|---|---|
| AP-01 | CLOUD-003, CLOUD-001, CLOUD-002 | 1) Attacker reaches the internet-facing report-fetch feature. |
| AP-02 | CLOUD-005, CLOUD-032, CLOUD-010 | 1) Attacker compromises a fictional Development-account developer credential (the weaker of the two account baselines). |
| AP-03 | CLOUD-008, CLOUD-024, CLOUD-009 | 1) Attacker exploits the known-vulnerable package in the astravault/analytics-worker container image. |
| AP-04 | CLOUD-004, CLOUD-001 | 1) Attacker compromises a standard fictional engineering-tier credential (e. |

All four attack chains above are fictional, illustrative, and non-destructive. No real AWS account, credential, or resource was accessed, modified, or exploited in their construction or validation.

# 41 FINDINGS SUMMARY

This fictional assessment identified 42 illustrative findings across AstraVault Technologies, Inc.'s AWS cloud environment, summarized below: 3 Critical, 8 High, 13 Medium, 9 Low, and 9 Informational. Full detail for every finding, including synthetic evidence and specific AWS control recommendations, is provided in Sections 42-46, organized by severity.




*Illustrative risk heatmap — likelihood versus impact — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE.*

## Findings at a Glance

| Finding ID | Severity | Title | Status |
|---|---|---|---|
| CLOUD-001 | CRITICAL | Overly Permissive IAM Role Enables Cross-Account Administrative Privilege Escalation | Remediation In Progress |
| CLOUD-002 | CRITICAL | Publicly Accessible S3 Bucket Exposes Sensitive Financial Data | Remediation In Progress |
| CLOUD-003 | CRITICAL | EC2 Workload Metadata Exposure Enables Credential Theft and Privilege Escalation | Remediation In Progress |
| CLOUD-004 | HIGH | Privilege Escalation via IAM Policy Chaining (iam:PassRole + iam:CreatePolicyVersion) | Remediation Planned |
| CLOUD-005 | HIGH | Excessive Trust Relationship Between Production and Development AWS Accounts | Remediation Planned |
| CLOUD-006 | HIGH | Security Group Permits Unrestricted Administrative Access to Management Ports | Remediation Planned |
| CLOUD-007 | HIGH | Secrets Exposed Through Workload Environment Configuration | Remediation In Progress |
| CLOUD-008 | HIGH | Vulnerable Container Image With Exploitable Package in Production ECR Repository | Remediation In Progress |
| CLOUD-009 | HIGH | Lambda Execution Role Grants Excessive IAM Permissions Beyond Function Requirements | Remediation Planned |
| CLOUD-010 | HIGH | RDS Database Instance Publicly Reachable From the Internet | Remediation Planned |
| CLOUD-011 | HIGH | CloudTrail Logging Coverage Gap on Security-Sensitive API Activity | Remediation Planned |
| CLOUD-012 | MEDIUM | Overly Broad S3 Bucket Resource Policy Grants Unnecessary Cross-Account Access | Remediation Planned |
| CLOUD-013 | MEDIUM | Missing Server-Side Encryption Enforcement on S3 Buckets Storing Sensitive Data | Remediation Planned |
| CLOUD-014 | MEDIUM | S3 Bucket Versioning Disabled on Buckets Storing Financial Records | Remediation Planned |
| CLOUD-015 | MEDIUM | Insufficient CloudWatch Alerting on High-Risk IAM and Security Events | Remediation Planned |

| Finding ID | Severity | Title | Status |
|---|---|---|---|
| CLOUD-016 | MEDIUM | Permissive Security Group Egress Rules Allow Unrestricted Outbound Access | Remediation Planned |
| CLOUD-017 | MEDIUM | Stale IAM Users With Long-Unused Console and API Access | Remediation In Progress |
| CLOUD-018 | MEDIUM | Unused and Unrotated IAM Access Keys Exceeding Recommended Age | Remediation Planned |
| CLOUD-019 | MEDIUM | Weak AWS Backup Configuration for Production Database Resources | Remediation Planned |
| CLOUD-020 | MEDIUM | Lambda Function Configured With Excessive Timeout and Memory Beyond Operational Need | Remediation Planned |
| CLOUD-021 | MEDIUM | Missing Workload Integrity Controls on EC2 AMI Build Pipeline | Remediation Planned |
| CLOUD-022 | MEDIUM | Insufficient Security-Header Controls at CloudFront Edge | Remediation Planned |
| CLOUD-023 | MEDIUM | KMS Key Policy Grants Overly Broad Decrypt Permissions | Remediation Planned |
| CLOUD-024 | MEDIUM | ECS Task Definition Runs Containers With Elevated Linux Capabilities | Remediation Planned |
| CLOUD-025 | LOW | Incomplete Resource Tagging Undermines Cost and Security Governance | Remediation Planned |
| CLOUD-026 | LOW | S3 Access Logging Not Enabled on Selected Buckets | Remediation Planned |
| CLOUD-027 | LOW | VPC Flow Logs Not Enabled on Selected VPCs | Remediation Planned |
| CLOUD-028 | LOW | Route 53 DNS Zone Lacks Query Logging | Remediation Planned |
| CLOUD-029 | LOW | GuardDuty Findings Not Routed to a Centralized Alerting Channel | Remediation Planned |
| CLOUD-030 | LOW | WAF Rule Set Missing Rate-Based Rule on Public API Gateway | Remediation Planned |
| CLOUD-031 | LOW | ECR Repository Missing Image Lifecycle Policy (Unbounded Image Retention) | Remediation Planned |
| CLOUD-032 | LOW | Secrets Manager Rotation Not Configured for Long-Lived Database Credentials | Remediation Planned |
| CLOUD-033 | LOW | SNS Topic Policy Permits Overly Broad Publish Access | Remediation Planned |
| CLOUD-034 | INFORMATIONAL | Security Hub Standard Subscriptions Incomplete Across Accounts | Remediation Planned |
| CLOUD-035 | INFORMATIONAL | Inconsistent Resource Naming Convention Complicates Governance | Remediation Planned |
| CLOUD-036 | INFORMATIONAL | Infrastructure-as-Code Drift Observed Between Deployed and Templated Resources | Remediation Planned |
| CLOUD-037 | INFORMATIONAL | EventBridge Rule Lacks Dead-Letter Queue for Failed Invocations | Remediation Planned |
| CLOUD-038 | INFORMATIONAL | Systems Manager Patch Compliance Reporting Gaps on EC2 Fleet | Remediation Planned |
| CLOUD-039 | INFORMATIONAL | API Gateway Missing Request Validation on Selected Public Endpoints | Remediation Planned |
| CLOUD-040 | INFORMATIONAL | CloudWatch Log Group Retention Set to Indefinite on Non-Critical Logs | Remediation Planned |
| CLOUD-041 | INFORMATIONAL | Development Account Lacks Equivalent Security Baseline to Production | Remediation Planned |
| CLOUD-042 | INFORMATIONAL | Documentation Gap in Incident Response Runbook for Cloud-Specific Scenarios | Remediation Planned |

## Finding Format

Each finding card in Sections 42-46 presents: Finding ID, Title, Severity, CVSS-style Score and Vector (where applicable), CWE reference, Affected Cloud Service, Affected Fictional Resource, Description, Technical Details, Attack Scenario, Impact, Business Impact, Likelihood, Evidence Reference, Recommendation, Recommended AWS Controls, Management Response,

Owner/Priority/Target Date/Status, and Validation/Retest Guidance. Critical and High findings add a Technical Evidence Summary and a full synthetic evidence panel.

## Synthetic Evidence Standard

Every piece of evidence in this report is illustrative and fictional. All evidence references the fictional AstraVault Technologies, Inc. AWS accounts (123456789012, 123456789013, 123456789014), the fictional non-resolving domains console.astravault-example.com / api.astravault-example.com / app.astravault-example.com, and placeholder credential values such as AKIAEXAMPLE-prefixed access key IDs and REDACTED secret values. No evidence in this report was captured from a real AWS account, and no PoC is destructive, weaponized, or intended to enable real-world exploitation.

## CVSS Scoring Note

All CVSS-style scores and vectors shown in this report — including the illustrative vector strings on each Critical, High, and select Medium finding — are sample estimations constructed for this fictional exercise. They are not derived from any real vulnerability, are not submitted to any CVSS numbering authority, and should not be treated as authoritative, registered scoring for a real system.

**ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE — every AWS CLI output, IAM policy document, JSON response, security-group rule, CloudTrail event, screenshot, CVSS score/vector, and evidence reference in Sections 42-46 is synthetic.**

# 42 CRITICAL FINDINGS

The 3 Critical-severity findings below represent AstraVault's most urgent fictional cloud risk and, together, form the backbone of Attack Chains 1 and 4 in Section 40. TMG Security recommends remediation within 0-14 days for all three.

| CLOUD-001 — Overly Permissive IAM Role Enables Cross-Account Administrative Privilege Escalation | CRITICAL |
|---|---|
| **CVSS-Style Score: 9.6 (Critical) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-269 — Improper Privilege Management** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:H/A:H — Illustrative vector only` | |
| **Description** | The fictional astravault-security-crossaccount-role, deployed in the AstraVault Security account (123456789013) to support fictional centralized log aggregation, carries an inline policy granting iam:*, sts:AssumeRole on Resource "*", and organizations:* rather than the narrow logs:PutLogEvents / logs:CreateLogStream scope its stated purpose requires. Its trust policy allows sts:AssumeRole from the entire fictional AstraVault Production account (123456789012) principal (arn:aws:iam::123456789012:root) with no external ID, no MFA condition, and no source-identity restriction, rather than from a single scoped role. |
| **Technical Details** | During the limited-privilege IAM assessment, TMG Security enumerated assumable roles reachable from a fictional low-privilege AstraVault Production IAM user (astravault-app-readonly) using a fictional iam:GetRole / sts:AssumeRole probe sequence. The astravault-security-crossaccount-role's trust policy permitted assumption from any principal in the Production account, and the assumed role's own permissions (iam:*, sts:AssumeRole on "*") allowed further escalation — including creating a new IAM user with AdministratorAccess attached and generating long-lived fictional access keys for it — entirely within the nominally separate AstraVault Security account. |
| **Attack Scenario** | A fictional attacker who compromises any credential inside the AstraVault Production account (e.g., a leaked CI/CD deployment key, a phished developer session, or a compromised EC2 instance profile) can assume astravault-security-crossaccount-role without further authorization, then use its iam:* permissions to create a new administrative IAM user in the Security account, fully defeating the account-boundary segmentation AstraVault's fictional cloud team believed was in place. |
| **Impact** | Complete compromise of the fictional AstraVault Security account from a foothold in the fictional Production account, including the ability to create new administrative identities, modify CloudTrail/GuardDuty configuration, and delete security tooling used to detect the intrusion. |
| **Business Impact** | Loss of the security-account boundary undermines every other control in AstraVault's fictional environment: an attacker who reaches this role can disable detection (GuardDuty, Security Hub, CloudTrail) before proceeding further, converting a single-account compromise into an organization-wide breach with no reliable audit trail. |
| **Likelihood** | High — requires only a single compromised Production-account credential and no additional exploitation technique beyond a documented AssumeRole call. |
| **Evidence Reference** | *EV-CLOUD-001, EV-CLOUD-002 (fictional IAM role policy export, fictional sts:AssumeRole trace) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Scope the cross-account role's permissions to only logs:PutLogEvents and logs:CreateLogStream on the specific fictional log group; restrict the trust policy to a single named Production role ARN (not the account root principal); require sts:ExternalId and an MFA condition (aws:MultiFactorAuthPresent) on the trust policy; apply a permission boundary to the role capping it below iam:* regardless of any future policy change. |
| **Recommended AWS Controls** | IAM permission boundaries on all cross-account roles; SCP (Service Control Policy) at the AWS Organizations level denying iam:CreateUser / iam:AttachUserPolicy for non-administrative OUs; IAM Access Analyzer cross-account findings review; mandatory ExternalId and MFA condition on every cross-account trust policy. |
| **Management Response** | *AstraVault cloud engineering (fictional) has accepted this finding as the top remediation priority from this fictional assessment and will re-scope the role and trust policy ahead of the next change window.* |
| **Owner / Priority / Target / Status** | **Cloud Security Engineering Lead (Fictional Role)** \| Immediate — 0-14 Days \| 14 October 2026 \| **Remediation In Progress** |
| **Validation / Retest Guidance** | *Retest by re-attempting the fictional AssumeRole probe from the Production low-privilege identity against the remediated trust policy and confirming the call is denied without a valid ExternalId and MFA-present session, and that the resulting session (where permitted) cannot perform iam:CreateUser or iam:AttachUserPolicy.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 02 · IAM & Identity Security / 19 · Cross-Account Trust | IAM, STS, Cross-Account Trust Policies | arn:aws:iam::123456789013:role/astravault-security-crossaccount-role (fictional, AstraVault Security account 123456789013) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

**Fictional IAM Trust Policy Export (Fictional / Synthetic)**

```
$ aws iam get-role --role-name astravault-security-crossaccount-role --profile astravault-prod-readonly
{
  "Role": {
    "RoleName": "astravault-security-crossaccount-role",
    "Arn": "arn:aws:iam::123456789013:role/astravault-security-crossaccount-role",
    "AssumeRolePolicyDocument": {
      "Version": "2012-10-17",
      "Statement": [{
        "Effect": "Allow",
        "Principal": { "AWS": "arn:aws:iam::123456789012:root" },
        "Action": "sts:AssumeRole"
      }]
    }
  }
}
```

**Fictional AssumeRole and Privilege Check (Fictional / Synthetic)**

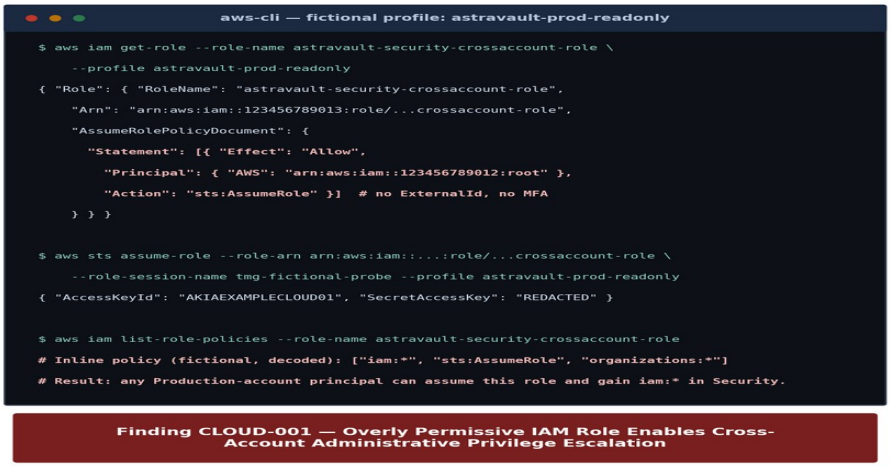
```
$ aws sts assume-role --role-arn arn:aws:iam::123456789013:role/astravault-security-crossaccount-role \
    --role-session-name tmg-fictional-probe --profile astravault-prod-readonly
{
  "Credentials": {
    "AccessKeyId": "AKIAEXAMPLECLOUD01",
    "SecretAccessKey": "REDACTED",
    "SessionToken": "REDACTED-FICTIONAL-SESSION-TOKEN"
  }
}
$ aws iam list-attached-role-policies --role-name astravault-security-crossaccount-role
{
  "AttachedPolicies": [],
  "InlinePolicies": ["astravault-security-crossaccount-inline"]
}
# Inline policy (fictional, decoded): Action: ["iam:*", "sts:AssumeRole", "organizations:*"], Resource: "*"
```

Expected: The role should grant only logs:PutLogEvents/logs:CreateLogStream on a specific log group, and its trust policy should require a named source role ARN plus ExternalId and MFA — not the entire Production account root principal with no conditions.

Observed: The role granted iam:*/sts:AssumeRole/organizations:* on all resources and could be assumed by any principal in the Production account with no ExternalId or MFA condition.

**Impact: Full compromise of the fictional AstraVault Security account is achievable from any Production-account foothold, including creation of new administrative identities.**

ILLUSTRATIVE / SYNTHETIC EVIDENCE
*no real AWS account or data*




Finding CLOUD-001 — Overly Permissive IAM Role Enables Cross-Account Administrative Privilege Escalation

*Synthetic mock-panel evidence — ILLUSTRATIVE / SYNTHETIC EVIDENCE. No real AWS account, console, or data is depicted.*

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | A fictional attacker who compromises any credential inside the AstraVault Production account (e.g., a leaked CI/CD deployment key, a phished developer session, or a compromised EC2 instance profile) can assume astravault-security-crossaccount-role without further authorization, then use its iam:* permissions to create a new administrative IAM user in the Security account, fully defeating the account-boundary segmentation AstraVault's fictional cloud team believed was in place. |
| **Likelihood** | High — requires only a single compromised Production-account credential and no additional exploitation technique beyond a documented AssumeRole call. |
| **Security Impact** | Complete compromise of the fictional AstraVault Security account from a foothold in the fictional Production account, including the ability to create new administrative identities, modify CloudTrail/GuardDuty configuration, and delete security tooling used to detect the intrusion. |
| **Business Impact** | Loss of the security-account boundary undermines every other control in AstraVault's fictional environment: an attacker who reaches this role can disable detection (GuardDuty, Security Hub, CloudTrail) before proceeding further, converting a single-account compromise into an organization-wide breach with no reliable audit trail. |
| **Evidence Collected** | EV-CLOUD-001, EV-CLOUD-002 (fictional IAM role policy export, fictional sts:AssumeRole trace) |
| **Validation / Retest Guidance** | Retest by re-attempting the fictional AssumeRole probe from the Production low-privilege identity against the remediated trust policy and confirming the call is denied without a valid ExternalId and MFA-present session, and that the resulting session (where permitted) cannot perform iam:CreateUser or iam:AttachUserPolicy. |

| **CLOUD-002 — Publicly Accessible S3 Bucket Exposes Sensitive Financial Data** | **CRITICAL** |
|---|---|
| **CVSS-Style Score: 9.1 (Critical) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-284 — Improper Access Control** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N — Illustrative vector only` | |
| **Description** | The fictional astravault-prod-financial-statements bucket, used to store generated customer account statements and analytics exports, has S3 Block Public Access disabled at the bucket level and carries a bucket policy with a Principal of "*" and s3:GetObject / s3:ListBucket actions with no IP, VPC-endpoint, or condition restriction. The bucket also lacks default (SSE-KMS) encryption enforcement, so objects can be uploaded without encryption. |
| **Technical Details** | From an unauthenticated external black-box perspective, TMG Security enumerated the bucket name via a fictional certificate-transparency and subdomain-permutation exercise against *.astravault-example.com, then confirmed public listability and object retrieval with a plain, unauthenticated HTTPS request — no AWS credentials, signed URL, or account membership was required. |
| **Attack Scenario** | Any unauthenticated internet user who discovers or guesses the fictional bucket name can list its full contents and download every object, including fictional customer account statements and analytics exports containing PII-adjacent and financial data, with no logging correlation back to a specific requester identity beyond a source IP. |
| **Impact** | Unauthenticated disclosure of the full contents of a fictional bucket holding customer-facing financial documents, with no authentication barrier and no server-side encryption guarantee. |
| **Business Impact** | A real-world equivalent of this finding would constitute a reportable data-exposure event for a FinTech platform, triggering regulatory notification obligations, customer-trust damage, and potential liability under U.S. state data-breach notification laws — entirely from a passive, unauthenticated discovery technique with no exploitation skill required. |
| **Likelihood** | High — public S3 buckets are routinely discovered by automated internet-wide scanning; no authentication or social engineering is required. |
| **Evidence Reference** | *EV-CLOUD-003, EV-CLOUD-004 (fictional public-bucket listing capture, fictional anonymous GetObject response) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Enable S3 Block Public Access at both the bucket and account level; remove the wildcard-principal bucket policy statement; restrict access to a specific VPC endpoint or a narrow set of application roles; enable default SSE-KMS encryption with bucket-key enabled; enable S3 Access Logging and enforce it via SCP across the organization. |
| **Recommended AWS Controls** | S3 Block Public Access (account-level SCP enforcement), bucket policies scoped to VPC endpoints or specific role ARNs, default SSE-KMS encryption, AWS Config rule s3-bucket-public-read-prohibited, GuardDuty S3 protection, Macie for sensitive-data discovery on remaining buckets. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and applied an emergency fictional remediation restoring Block Public Access within 24 hours of the fictional out-of-band notification that accompanies every Critical finding in this engagement's rules of engagement.* |
| **Owner / Priority / Target / Status** | **Cloud Platform Engineering Lead (Fictional Role)** \| Immediate — 0-14 Days \| 08 September 2026 (emergency fix) / 14 October 2026 (full remediation incl. encryption enforcement) \| **Remediation In Progress** |
| **Validation / Retest Guidance** | *Retest by re-attempting the fictional anonymous GetObject/ListBucket request against the remediated bucket and confirming Access Denied, and by confirming Block Public Access is enabled at both bucket and account level and cannot be toggled off by any role below the security-account boundary.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 04 · S3 & Object Storage Security | S3 | s3://astravault-prod-financial-statements (fictional bucket, AstraVault Production account 123456789012) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

**Fictional Anonymous Bucket Listing (Fictional / Synthetic)**

```
$ curl -s https://astravault-prod-financial-statements.s3.amazonaws.com/?list-type=2
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult>
  <Name>astravault-prod-financial-statements</Name>
  <Contents>
    <Key>statements/2026-08/acct-<FICTIONAL_ACCT_ID>-statement.pdf</Key>
    <Size>184320</Size>
  </Contents>
  <Contents>
    <Key>analytics-exports/2026-08/monthly-transaction-export.csv</Key>
    <Size>5242880</Size>
  </Contents>
  <!-- fictional listing truncated -->
</ListBucketResult>
```

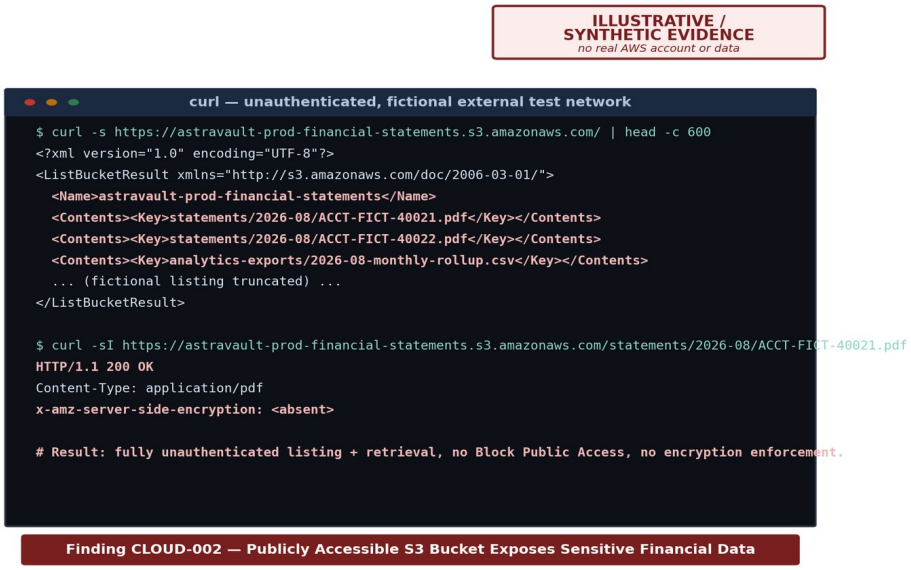
**Fictional Bucket Policy (Retrieved via Authenticated Review Perspective) (Fictional / Synthetic)**

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "FictionalPublicReadLegacyIntegration",
    "Effect": "Allow",
    "Principal": "*",
    "Action": ["s3:GetObject", "s3:ListBucket"],
    "Resource": [
      "arn:aws:s3:::astravault-prod-financial-statements",
      "arn:aws:s3:::astravault-prod-financial-statements/*"
    ]
  }]
}
```

Expected: The bucket should have Block Public Access enabled, no wildcard-Principal policy statement, and should require an authenticated, authorized request scoped to a VPC endpoint or specific application role.

Observed: The bucket allowed fully unauthenticated listing and object retrieval with Block Public Access disabled and a wildcard-Principal bucket policy.

**Impact: Complete unauthenticated disclosure of fictional customer financial statements and analytics exports stored in the bucket.**




*Synthetic mock-panel evidence — ILLUSTRATIVE / SYNTHETIC EVIDENCE. No real AWS account, console, or data is depicted.*

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | Any unauthenticated internet user who discovers or guesses the fictional bucket name can list its full contents and download every object, including fictional customer account statements and analytics exports containing PII-adjacent and financial data, with no logging correlation back to a specific requester identity beyond a source IP. |
| **Likelihood** | High — public S3 buckets are routinely discovered by automated internet-wide scanning; no authentication or social engineering is required. |

| | |
|---|---|
| **Security Impact** | Unauthenticated disclosure of the full contents of a fictional bucket holding customer-facing financial documents, with no authentication barrier and no server-side encryption guarantee. |
| **Business Impact** | A real-world equivalent of this finding would constitute a reportable data-exposure event for a FinTech platform, triggering regulatory notification obligations, customer-trust damage, and potential liability under U.S. state data-breach notification laws — entirely from a passive, unauthenticated discovery technique with no exploitation skill required. |
| **Evidence Collected** | EV-CLOUD-003, EV-CLOUD-004 (fictional public-bucket listing capture, fictional anonymous GetObject response) |
| **Validation / Retest Guidance** | Retest by re-attempting the fictional anonymous GetObject/ListBucket request against the remediated bucket and confirming Access Denied, and by confirming Block Public Access is enabled at both bucket and account level and cannot be toggled off by any role below the security-account boundary. |

| **CLOUD-003 — EC2 Workload Metadata Exposure Enables Credential Theft and Privilege Escalation** | **CRITICAL** |
|---|---|
| **CVSS-Style Score: 9.3 (Critical) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-918 — Server-Side Request Forgery (SSRF)** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:N — Illustrative vector only` | |
| **Description** | The fictional astravault-prod-api-worker EC2 fleet, which backs a report-fetch feature on https://api.astravault-example.com that retrieves a user-supplied statement-export URL, does not validate or restrict the destination of that outbound fetch and is provisioned with the legacy Instance Metadata Service configured to allow IMDSv1 (HttpTokens: optional) rather than requiring IMDSv2 session tokens. This combination allows a classic SSRF-to-metadata chain. |
| **Technical Details** | From the assumed-breach/compromised-workload testing perspective, TMG Security supplied a fictional statement-export URL pointing at the link-local metadata address (http://169.254.169.254/latest/meta-data/iam/security-credentials/astravault-ec2-app-role) to the report-fetch feature. Because IMDSv1 was permitted, no session token was required, and the application's outbound-fetch logic returned the metadata response content to the requester, yielding temporary IAM credentials for the astravault-ec2-app-role instance profile. |
| **Attack Scenario** | An attacker who can influence the destination of the report-fetch feature's outbound request (directly, or via a chained vulnerability in an upstream component) retrieves the EC2 instance's temporary IAM credentials from IMDS without any additional exploitation step, then uses those credentials — which the following findings show are themselves over-permissioned (see CLOUD-001-style role scoping issues) — to reach the fictional S3 buckets and other resources the instance profile can access. |
| **Impact** | Theft of live, temporary IAM credentials for the EC2 instance profile without any authentication to AWS itself, enabling any action the instance profile's attached policies allow. |
| **Business Impact** | SSRF-to-metadata chains are among the most consequential cloud attack patterns because they convert an application-layer bug into full cloud-credential theft; combined with the instance profile's actual permission scope, this finding is a direct precondition for the Attack Chain 1 composite scenario described in Section 40 of this report. |
| **Likelihood** | High — SSRF-to-metadata is a well-known, low-complexity technique, and IMDSv1 was reachable with no additional barrier. |
| **Evidence Reference** | *EV-CLOUD-005, EV-CLOUD-006 (fictional SSRF request/response capture, fictional recovered IMDS credential payload) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Require IMDSv2 (HttpTokens: required) on all EC2 launch templates and Auto Scaling groups organization-wide; set a low HttpPutResponseHopLimit to break most SSRF-to-metadata chains even where IMDSv2 alone is insufficient; add strict allow-list validation and DNS-rebinding protection to the report-fetch feature's outbound URL handling; scope the instance profile to only the specific resources the workload needs. |
| **Recommended AWS Controls** | IMDSv2 enforcement via SCP (ec2:MetadataHttpTokens condition) and AWS Config rule ec2-imdsv2-check; VPC egress controls limiting outbound fetch destinations; instance-profile least-privilege scoping; GuardDuty EC2 finding types for IMDS credential exfiltration. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will enforce IMDSv2 fleet-wide and add outbound URL allow-listing to the report-fetch feature ahead of the next release.* |
| **Owner / Priority / Target / Status** | **Platform Engineering Lead (Fictional Role)** \| Immediate — 0-14 Days \| 14 October 2026 \| **Remediation In Progress** |
| **Validation / Retest Guidance** | *Retest by repeating the fictional SSRF-to-metadata request against the remediated fleet and confirming IMDSv1 requests are rejected, IMDSv2 is required, and the report-fetch feature refuses link-local and non-allow-listed destinations.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 05 · EC2 & Workload Security / 22 · SSRF & Metadata Service Exposure | EC2, Instance Metadata Service (IMDS), IAM Instance Profiles | i-0fict0a1b2c3d4e5f6 (fictional EC2 instance, astravault-prod-api-worker Auto Scaling group, instance profile astravault-ec2-app-role) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

**Fictional SSRF Request via Report-Fetch Feature (Fictional / Synthetic)**

```
POST /v2/reports/export-fetch HTTP/1.1
Host: api.astravault-example.com
Authorization: Bearer <FICTIONAL_LOW_PRIV_SESSION_TOKEN>
Content-Type: application/json

{"exportUrl": "http://169.254.169.254/latest/meta-data/iam/security-credentials/astravault-ec2-app-role"}
```

**Fictional Recovered IMDS Credential Response (Fictional / Synthetic)**

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "Code": "Success",
  "AccessKeyId": "AKIAEXAMPLECLOUD02",
  "SecretAccessKey": "REDACTED",
  "Token": "REDACTED-FICTIONAL-IMDS-SESSION-TOKEN",
  "Expiration": "2026-09-14T18:00:00Z"
}
```

Expected: IMDSv2 should be required (no unauthenticated IMDSv1 fallback), and the report-fetch feature should reject link-local (169.254.0.0/16) and other non-allow-listed destinations before making the outbound request.

Observed: IMDSv1 was permitted with no session token, and the report-fetch feature relayed the metadata response content back to the requester with no destination validation.

**Impact: Full theft of live, temporary IAM credentials for the EC2 instance profile, usable for any action the profile's attached policies allow until natural expiration.**


*Synthetic mock-panel evidence — ILLUSTRATIVE / SYNTHETIC EVIDENCE. No real AWS account, console, or data is depicted.*

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | An attacker who can influence the destination of the report-fetch feature's outbound request (directly, or via a chained vulnerability in an upstream component) retrieves the EC2 instance's temporary IAM credentials from IMDS without any additional exploitation step, then uses those credentials — which the following findings show are themselves over-permissioned (see CLOUD-001-style role scoping issues) — to reach the fictional S3 buckets and other resources the instance profile can access. |
| **Likelihood** | High — SSRF-to-metadata is a well-known, low-complexity technique, and IMDSv1 was reachable with no additional barrier. |
| **Security Impact** | Theft of live, temporary IAM credentials for the EC2 instance profile without any authentication to AWS itself, enabling any action the instance profile's attached policies allow. |
| **Business Impact** | SSRF-to-metadata chains are among the most consequential cloud attack patterns because they convert an application-layer bug into full cloud-credential theft; combined with the instance profile's actual permission scope, this finding is a direct precondition for the Attack Chain 1 composite scenario described in Section 40 of this report. |
| **Evidence Collected** | EV-CLOUD-005, EV-CLOUD-006 (fictional SSRF request/response capture, fictional recovered IMDS credential payload) |
| **Validation / Retest Guidance** | Retest by repeating the fictional SSRF-to-metadata request against the remediated fleet and confirming IMDSv1 requests are rejected, IMDSv2 is required, and the report-fetch feature refuses link-local and non-allow-listed destinations. |

# 43 HIGH FINDINGS

The 8 High-severity findings below represent significant fictional risk warranting prompt remediation, recommended within 0-30 days.

| CLOUD-004 — Privilege Escalation via IAM Policy Chaining (iam:PassRole + iam:CreatePolicyVersion) | HIGH |
|---|---|
| **CVSS-Style Score: 8.4 (High) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-269 — Improper Privilege Management** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:H/A:N — Illustrative vector only` | |
| **Description** | Members of the fictional astravault-platform-engineers IAM group hold iam:CreatePolicyVersion and iam:PassRole on a broad resource scope alongside their intended lambda:CreateFunction / lambda:UpdateFunctionCode permissions. This combination is a documented IAM privilege-escalation pattern: a member can create a new version of any customer-managed policy attached to a higher-privileged role (including their own group's boundary-adjacent policies) and pass that role to a new Lambda function to execute arbitrary AWS API calls under its identity. |
| **Technical Details** | TMG Security's limited-privilege IAM assessment confirmed via the fictional IAM privilege-escalation matrix (Section 17 / Appendix C) that the combination of iam:PassRole (Resource "*") and iam:CreatePolicyVersion (Resource "*") granted to this group maps directly to a known escalation path, without needing to actually execute a live escalation against the fictional environment. |
| **Attack Scenario** | A fictional engineer-level credential (or an attacker who phishes one) creates a new default version of a customer-managed policy attached to a more privileged role, granting themselves additional actions, then passes that role to a newly created Lambda function to execute under the elevated identity — achieving privilege escalation without ever touching a flagged "escalation-only" permission directly. |
| **Impact** | A standard engineering-tier credential can escalate to a materially higher permission set through a documented, automatable IAM policy-chaining technique. |
| **Business Impact** | Because this affects an entire engineering group rather than a single account, the practical blast radius of a single phished engineer credential is far larger than the group's intended permission set implies. |
| **Likelihood** | Medium-High — requires a compromised engineering-tier credential, but no additional vulnerability beyond documented IAM API usage. |
| **Evidence Reference** | *EV-CLOUD-007 (fictional IAM policy simulation output, IAM privilege-escalation matrix cross-reference) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Split iam:PassRole and iam:CreatePolicyVersion out of the engineering group's baseline policy into a separate, tightly scoped deployment-role policy used only by the fictional CI/CD pipeline's OIDC role; apply permission boundaries to the engineering group capping escalation-relevant actions; run IAM Access Analyzer policy generation to right-size the group's actual usage pattern. |
| **Recommended AWS Controls** | IAM Access Analyzer, permission boundaries on all human IAM principals, separation of deployment-time IAM permissions from day-to-day engineering permissions, scheduled privilege-escalation-path scanning (e.g., PMapper/Cloudsplaining-style analysis) as a recurring control. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will split the deployment-time permissions into a dedicated CI/CD role ahead of the next quarterly access review.* |
| **Owner / Priority / Target / Status** | **IAM & Identity Lead (Fictional Role)** \| 0-30 Days \| 10 October 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by re-running the fictional IAM privilege-escalation matrix analysis against the remediated group policy and confirming no remaining combination of granted actions maps to a documented escalation pattern.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 03 · Privilege Escalation | IAM | IAM group astravault-platform-engineers (fictional, AstraVault Production account 123456789012) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

```
Fictional iam simulate-principal-policy Output (Fictional / Synthetic)
$ aws iam simulate-principal-policy \
    --policy-source-arn arn:aws:iam::123456789012:group/astravault-platform-engineers \
    --action-names iam:PassRole iam:CreatePolicyVersion lambda:CreateFunction
{
  "EvaluationResults": [
    {"EvalActionName": "iam:PassRole", "EvalDecision": "allowed", "EvalResourceName": "*"},
```

```
    {"EvalActionName": "iam:CreatePolicyVersion", "EvalDecision": "allowed", "EvalResourceName": "*"},
    {"EvalActionName": "lambda:CreateFunction", "EvalDecision": "allowed", "EvalResourceName": "*"}
  ]
}
# Matches fictional IAM privilege-escalation matrix pattern EP-04 (PassRole + CreatePolicyVersion +
CreateFunction).
```

Expected: Escalation-relevant action combinations should not co-occur in a single engineering-tier group policy; deployment-time actions should be isolated to a dedicated, narrowly-scoped CI/CD role.

Observed: The astravault-platform-engineers group policy combined iam:PassRole, iam:CreatePolicyVersion, and lambda:CreateFunction on broad resource scopes, matching a documented escalation pattern.

**Impact: Any compromised engineering-tier credential can escalate to a materially higher permission set via documented IAM policy chaining.**

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | A fictional engineer-level credential (or an attacker who phishes one) creates a new default version of a customer-managed policy attached to a more privileged role, granting themselves additional actions, then passes that role to a newly created Lambda function to execute under the elevated identity — achieving privilege escalation without ever touching a flagged "escalation-only" permission directly. |
| **Likelihood** | Medium-High — requires a compromised engineering-tier credential, but no additional vulnerability beyond documented IAM API usage. |
| **Security Impact** | A standard engineering-tier credential can escalate to a materially higher permission set through a documented, automatable IAM policy-chaining technique. |
| **Business Impact** | Because this affects an entire engineering group rather than a single account, the practical blast radius of a single phished engineer credential is far larger than the group's intended permission set implies. |
| **Evidence Collected** | EV-CLOUD-007 (fictional IAM policy simulation output, IAM privilege-escalation matrix cross-reference) |
| **Validation / Retest Guidance** | Retest by re-running the fictional IAM privilege-escalation matrix analysis against the remediated group policy and confirming no remaining combination of granted actions maps to a documented escalation pattern. |

| **CLOUD-005 — Excessive Trust Relationship Between Production and Development AWS Accounts** | **HIGH** |
|---|---|
| **CVSS-Style Score: 7.9 (High) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-284 — Improper Access Control** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:L/A:N — Illustrative vector only` | |
| **Description** | The fictional astravault-prod-devsupport-role in the Production account trusts the entire AstraVault Development account (123456789014) principal for sts:AssumeRole, and grants read access to production Secrets Manager secrets and RDS snapshot metadata — access intended only for a small on-call subset of engineers, not the full, lower-assurance Development account. |
| **Technical Details** | The Development account is provisioned with looser baseline controls (see CLOUD-041) than Production, including broader default IAM permissions and no mandatory hardware-MFA policy. Because any Development-account principal can assume the Production support role, the Development account's weaker baseline becomes the effective security boundary for a subset of Production secrets. |
| **Attack Scenario** | A fictional attacker who compromises any Development-account credential — a more likely target given its weaker baseline controls — can assume astravault-prod-devsupport-role and read Production Secrets Manager values without needing to compromise anything in Production directly. |
| **Impact** | Production secret material is reachable via the weaker of the two accounts' security postures rather than the stronger one, defeating the intended purpose of account-level segmentation. |
| **Business Impact** | Account separation is a foundational control in AstraVault's fictional cloud architecture; a trust relationship this broad materially reduces the assurance that separation is meant to provide, particularly for the RDS credential material this role can reach. |
| **Likelihood** | Medium — requires a compromised Development-account credential, which is more attainable than a Production one given the baseline gap. |
| **Evidence Reference** | *EV-CLOUD-008 (fictional cross-account trust policy export, fictional Development-account baseline comparison) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Restrict the trust policy to a small, named set of on-call engineer role ARNs rather than the full Development account principal; require MFA and a short session duration on assumption; move the small legitimate on-call use case to a dedicated break-glass role with additional approval logging. |
| **Recommended AWS Controls** | Named-principal trust policies, mandatory MFA condition on cross-account AssumeRole, session-duration capping, CloudTrail alerting on this specific role's assumption events routed to the Security account. |

| | |
|---|---|
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will narrow the trust policy to the on-call engineer role set in the next remediation cycle.* |
| **Owner / Priority / Target / Status** | **Cloud Security Engineering Lead (Fictional Role)** \| 0-30 Days \| 17 October 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming the remediated trust policy rejects AssumeRole from Development-account principals outside the named on-call role set, and that MFA is enforced on assumption.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 19 · Cross-Account Trust | IAM, Organizations | arn:aws:iam::123456789012:role/astravault-prod-devsupport-role (fictional, trusted by AstraVault Development account 123456789014) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

```
Fictional Trust Policy Export (Fictional / Synthetic)
$ aws iam get-role --role-name astravault-prod-devsupport-role
"AssumeRolePolicyDocument": {
  "Statement": [{
    "Effect": "Allow",
    "Principal": { "AWS": "arn:aws:iam::123456789014:root" },
    "Action": "sts:AssumeRole"
  }]
}
```

Expected: Trust should be scoped to a small, named set of on-call role ARNs with MFA required, not the entire Development-account root principal.

Observed: Any principal in the fictional Development account could assume the Production support role with no MFA condition.

**Impact: Production secret and RDS-snapshot-metadata access is only as strong as the Development account's weaker baseline controls.**

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | A fictional attacker who compromises any Development-account credential — a more likely target given its weaker baseline controls — can assume astravault-prod-devsupport-role and read Production Secrets Manager values without needing to compromise anything in Production directly. |
| **Likelihood** | Medium — requires a compromised Development-account credential, which is more attainable than a Production one given the baseline gap. |
| **Security Impact** | Production secret material is reachable via the weaker of the two accounts' security postures rather than the stronger one, defeating the intended purpose of account-level segmentation. |
| **Business Impact** | Account separation is a foundational control in AstraVault's fictional cloud architecture; a trust relationship this broad materially reduces the assurance that separation is meant to provide, particularly for the RDS credential material this role can reach. |
| **Evidence Collected** | EV-CLOUD-008 (fictional cross-account trust policy export, fictional Development-account baseline comparison) |
| **Validation / Retest Guidance** | Retest by confirming the remediated trust policy rejects AssumeRole from Development-account principals outside the named on-call role set, and that MFA is enforced on assumption. |

| **CLOUD-006 — Security Group Permits Unrestricted Administrative Access to Management Ports** | **HIGH** |
|---|---|
| **CVSS-Style Score: 8.1 (High) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-284 — Improper Access Control** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:N — Illustrative vector only` | |
| **Description** | The fictional astravault-prod-mgmt-sg security group, attached to the bastion/management EC2 instances and the production RDS Aurora cluster, allows inbound TCP 22 (SSH), 3389 (RDP), and 5432 (PostgreSQL) from 0.0.0.0/0 rather than from a scoped corporate IP range or a Systems-Manager-Session-Manager-only design. |
| **Technical Details** | External black-box port scanning against the fictional bastion host's public IP confirmed all three ports reachable from the internet. Authenticated configuration review confirmed the underlying security group rule source as 0.0.0.0/0 with no compensating Network ACL restriction. |
| **Attack Scenario** | Combined with weak or reused credentials (a separate risk class outside this finding's scope), an internet-reachable management port on a database-adjacent host materially increases the population of attackers who can attempt brute-force or credential-stuffing access against production infrastructure. |
| **Impact** | Administrative and database management ports are reachable from any internet host, removing network-layer defense-in-depth for these services. |
| **Business Impact** | Even with strong credentials, internet-exposed management ports on production-adjacent infrastructure represent an |

| | |
|---|---|
| | unnecessary and easily-remediated increase in attack surface for a FinTech platform's most sensitive systems. |
| **Likelihood** | Medium — exploitation requires an additional credential-level weakness, but exposure itself is trivially discoverable via routine internet scanning. |
| **Evidence Reference** | *EV-CLOUD-009 (fictional external port-scan result), EV-CLOUD-010 (fictional security group rule export) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Restrict inbound rules to AstraVault's fictional corporate VPN CIDR range only; migrate SSH/RDP access to AWS Systems Manager Session Manager (removing the need for inbound 22/3389 entirely); move RDS to a private subnet with no direct internet-reachable security group path. |
| **Recommended AWS Controls** | Systems Manager Session Manager for host access, security-group rule restricted to VPN/bastion CIDR only, AWS Config rule restricted-ssh / restricted-common-ports, Network ACL as defense-in-depth. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will migrate to Session Manager and restrict the security group in the next remediation cycle.* |
| **Owner / Priority / Target / Status** | **Network Engineering Lead (Fictional Role)** \| 0-30 Days \| 17 October 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by re-running the fictional external port scan against the bastion and RDS-adjacent hosts and confirming ports 22/3389/5432 are no longer reachable from 0.0.0.0/0.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 07 · Security Groups & Network ACLs | EC2 Security Groups | sg-0fictmgmt01 (astravault-prod-mgmt-sg, fictional) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

**Fictional Security Group Description (Fictional / Synthetic)**

```
$ aws ec2 describe-security-groups --group-ids sg-0fictmgmt01
{
  "IpPermissions": [
    {"FromPort": 22, "ToPort": 22, "IpProtocol": "tcp", "IpRanges": [{"CidrIp": "0.0.0.0/0"}]},
    {"FromPort": 3389, "ToPort": 3389, "IpProtocol": "tcp", "IpRanges": [{"CidrIp": "0.0.0.0/0"}]},
    {"FromPort": 5432, "ToPort": 5432, "IpProtocol": "tcp", "IpRanges": [{"CidrIp": "0.0.0.0/0"}]}
  ]
}
```

Expected: Inbound management and database ports should be restricted to a scoped VPN/bastion CIDR, with host access migrated to Session Manager wherever feasible.

Observed: TCP 22, 3389, and 5432 were all open to 0.0.0.0/0 on the management/database security group.

**Impact: Internet-wide reachability of administrative and database ports on production-adjacent infrastructure.**

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | Combined with weak or reused credentials (a separate risk class outside this finding's scope), an internet-reachable management port on a database-adjacent host materially increases the population of attackers who can attempt brute-force or credential-stuffing access against production infrastructure. |
| **Likelihood** | Medium — exploitation requires an additional credential-level weakness, but exposure itself is trivially discoverable via routine internet scanning. |
| **Security Impact** | Administrative and database management ports are reachable from any internet host, removing network-layer defense-in-depth for these services. |
| **Business Impact** | Even with strong credentials, internet-exposed management ports on production-adjacent infrastructure represent an unnecessary and easily-remediated increase in attack surface for a FinTech platform's most sensitive systems. |
| **Evidence Collected** | EV-CLOUD-009 (fictional external port-scan result), EV-CLOUD-010 (fictional security group rule export) |
| **Validation / Retest Guidance** | Retest by re-running the fictional external port scan against the bastion and RDS-adjacent hosts and confirming ports 22/3389/5432 are no longer reachable from 0.0.0.0/0. |

| **CLOUD-007 — Secrets Exposed Through Workload Environment Configuration** | **HIGH** |
|---|---|
| **CVSS-Style Score: 7.6 (High) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-798 — Use of Hard-coded Credentials** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:C/C:H/I:N/A:N — Illustrative vector only` | |
| **Description** | The fictional astravault-api-gateway-service ECS task definition sets the production RDS database password and a fictional third-party payment-aggregator API key directly as plaintext environment variables in the task definition rather than referencing them from Secrets Manager via the valueFrom mechanism. |
| **Technical Details** | Because ECS task definitions are readable by any principal with ecs:DescribeTaskDefinition permission — a far broader |

| | |
|---|---|
| | population than those with direct Secrets Manager access to the underlying secret — the plaintext values are effectively exposed to a wider audience than intended, and also appear in fictional CloudTrail data-event logs and ECS console history. |
| **Attack Scenario** | Any fictional engineer or automated tool with read-only ECS describe permissions (a common, broadly-granted permission for observability tooling) can retrieve the production database password and payment-aggregator API key without needing any Secrets Manager permission at all. |
| **Impact** | Production database credentials and a third-party payment-aggregator API key are readable by a broader population than intended, bypassing the access boundary Secrets Manager is meant to enforce. |
| **Business Impact** | Credential sprawl of this kind is a leading root cause of real-world cloud breaches; for a FinTech platform, exposure of payment-aggregator API keys specifically raises third-party financial-integration risk beyond AstraVault's own environment. |
| **Likelihood** | Medium-High — requires only a broadly-granted read permission, not a specialized exploitation technique. |
| **Evidence Reference** | *EV-CLOUD-011 (fictional ecs describe-task-definition output showing plaintext environment values) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Migrate all task-definition secrets to Secrets Manager references via the secrets (valueFrom) field rather than environment; rotate the exposed fictional database password and payment-aggregator API key; add a CI/CD pipeline check that rejects task definitions containing plaintext values matching known secret patterns. |
| **Recommended AWS Controls** | ECS secrets integration with Secrets Manager, IAM least-privilege scoping of ecs:DescribeTaskDefinition, Secrets Manager automatic rotation, pre-deployment secret-scanning in the CI/CD pipeline. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding, rotated the exposed fictional credentials, and will migrate to Secrets Manager references in the next deployment.* |
| **Owner / Priority / Target / Status** | **Platform Engineering Lead (Fictional Role)** \| 0-30 Days \| 17 October 2026 \| **Remediation In Progress** |
| **Validation / Retest Guidance** | *Retest by confirming the remediated task definition contains no plaintext secret values and that ecs:DescribeTaskDefinition no longer returns credential material.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 13 · Secrets Management | ECS, Lambda | astravault-prod-ecs-cluster task definition astravault-api-gateway-service:41 (fictional) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

**Fictional ecs describe-task-definition Output (Fictional / Synthetic)**

```
$ aws ecs describe-task-definition --task-definition astravault-api-gateway-service:41
"containerDefinitions": [{
  "name": "api-gateway",
  "environment": [
    {"name": "DB_PASSWORD", "value": "REDACTED-FICTIONAL-PLAINTEXT-VALUE"},
    {"name": "PAYMENT_AGGREGATOR_API_KEY", "value": "REDACTED-FICTIONAL-PLAINTEXT-VALUE"}
  ]
}]
```

Expected: Sensitive values should be referenced via the ECS secrets (valueFrom) mechanism pointing at Secrets Manager, never set directly in the environment array.

Observed: Both the database password and a third-party payment-aggregator API key were present as plaintext environment values in the task definition.

**Impact: Credential exposure to any principal with ecs:DescribeTaskDefinition, a broader population than intended Secrets Manager access.**

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | Any fictional engineer or automated tool with read-only ECS describe permissions (a common, broadly-granted permission for observability tooling) can retrieve the production database password and payment-aggregator API key without needing any Secrets Manager permission at all. |
| **Likelihood** | Medium-High — requires only a broadly-granted read permission, not a specialized exploitation technique. |
| **Security Impact** | Production database credentials and a third-party payment-aggregator API key are readable by a broader population than intended, bypassing the access boundary Secrets Manager is meant to enforce. |
| **Business Impact** | Credential sprawl of this kind is a leading root cause of real-world cloud breaches; for a FinTech platform, exposure of payment-aggregator API keys specifically raises third-party financial-integration risk beyond AstraVault's own environment. |
| **Evidence Collected** | EV-CLOUD-011 (fictional ecs describe-task-definition output showing plaintext environment values) |
| **Validation / Retest Guidance** | Retest by confirming the remediated task definition contains no plaintext secret values and that ecs:DescribeTaskDefinition no longer returns credential material. |

| CLOUD-008 — Vulnerable Container Image With Exploitable Package in Production ECR Repository | HIGH |
|---|---|
| **CVSS-Style Score: 8.2 (High) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-1104 — Use of Unmaintained Third-Party Components** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:L — Illustrative vector only` | |
| **Description** | The fictional astravault/analytics-worker container image currently deployed to production bundles a fictional outdated base-image package with a publicly disclosed remote-code-execution class vulnerability (illustrative CVE-style reference: FICTIONAL-CVE-2025-99999), and the ECR repository has image scanning configured but its findings are not gated in the CI/CD deployment pipeline — the vulnerable image was deployed despite a Critical-severity scan result being available. |
| **Technical Details** | TMG Security reviewed the fictional ECR scan findings for the currently-deployed image tag and cross-referenced the deployment pipeline configuration, confirming that ECR scan-on-push results are generated but not consumed as a deployment gate; the pipeline proceeds regardless of scan severity. |
| **Attack Scenario** | An attacker who can reach the analytics-worker service's exposed interface (directly, or via a chained application-layer vulnerability) could attempt to exploit the fictional unpatched package to achieve remote code execution within the container, from which the container's task role credentials become reachable (see the composite scenario in Attack Chain 3, Section 40). |
| **Impact** | A production container image with a known, high-severity vulnerable component is actively deployed with no compensating gate preventing its continued use. |
| **Business Impact** | Unpatched, internet-adjacent container workloads are a common initial-access vector in real-world cloud incidents; the absence of a deployment gate means this class of risk will recur with every future vulnerable dependency unless the pipeline itself is fixed, not just this one image. |
| **Likelihood** | Medium — requires a working exploit for the specific fictional vulnerable package and network reachability to the affected service. |
| **Evidence Reference** | *EV-CLOUD-012 (fictional ECR image scan finding export) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Rebuild the image from a patched base and redeploy immediately; configure the CI/CD pipeline to block deployment on any Critical or High ECR scan finding; adopt a minimal, regularly-rebuilt base image and enable ECR enhanced scanning with continuous rescanning. |
| **Recommended AWS Controls** | ECR scan-on-push with deployment-pipeline gating, scheduled base-image rebuilds, Amazon Inspector for continuous container scanning, image signing/attestation (e.g., cosign) verified at deploy time. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding, redeployed a patched image, and will add scan-result gating to the pipeline as part of the CI/CD security remediation tracked under CLOUD-024's related workstream.* |
| **Owner / Priority / Target / Status** | **DevOps / Platform Engineering Lead (Fictional Role)** \| 0-30 Days \| 17 October 2026 \| **Remediation In Progress** |
| **Validation / Retest Guidance** | *Retest by confirming the redeployed image resolves the fictional vulnerable package and that a deliberately reintroduced Critical-severity test finding is correctly blocked by the pipeline gate.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 11 · ECR / Image Security | ECR, ECS | astravault/analytics-worker:2026-08-12 (fictional ECR repository image, digest sha256:fictionaldigest7a91) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

**Fictional ECR Scan Result (Fictional / Synthetic)**

```
$ aws ecr describe-image-scan-findings --repository-name astravault/analytics-worker --image-id
imageTag=2026-08-12
{
  "imageScanFindings": {
    "findingSeverityCounts": {"CRITICAL": 1, "HIGH": 4, "MEDIUM": 11},
    "findings": [{
      "name": "FICTIONAL-CVE-2025-99999",
      "severity": "CRITICAL",
      "description": "Illustrative remote code execution in a fictional outdated base-image package."
    }]
  }
}
```

Expected: A Critical-severity ECR scan finding should block deployment via a CI/CD pipeline gate until remediated.

Observed: The Critical finding was present in the scan results, but the deployment pipeline proceeded with no gate consuming the scan severity.

**Impact: A production, internet-adjacent container workload runs with a known, high-severity remotely exploitable component.**

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | An attacker who can reach the analytics-worker service's exposed interface (directly, or via a chained application-layer vulnerability) could attempt to exploit the fictional unpatched package to achieve remote code execution within the container, from which the container's task role credentials become reachable (see the composite scenario in Attack Chain 3, Section 40). |
| **Likelihood** | Medium — requires a working exploit for the specific fictional vulnerable package and network reachability to the affected service. |
| **Security Impact** | A production container image with a known, high-severity vulnerable component is actively deployed with no compensating gate preventing its continued use. |
| **Business Impact** | Unpatched, internet-adjacent container workloads are a common initial-access vector in real-world cloud incidents; the absence of a deployment gate means this class of risk will recur with every future vulnerable dependency unless the pipeline itself is fixed, not just this one image. |
| **Evidence Collected** | EV-CLOUD-012 (fictional ECR image scan finding export) |
| **Validation / Retest Guidance** | Retest by confirming the redeployed image resolves the fictional vulnerable package and that a deliberately reintroduced Critical-severity test finding is correctly blocked by the pipeline gate. |

| **CLOUD-009 — Lambda Execution Role Grants Excessive IAM Permissions Beyond Function Requirements** | **HIGH** |
|---|---|
| **CVSS-Style Score: 7.7 (High) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-269 — Improper Privilege Management** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:M/A:N — Illustrative vector only` | |
| **Description** | A single fictional execution role, astravault-lambda-exec-role, is shared across all 14 of AstraVault's fictional Lambda functions rather than each function having a dedicated, minimally-scoped role. The shared role's policy grants s3:*, dynamodb:*, and secretsmanager:GetSecretValue on Resource "*", far exceeding what any individual function (including the fictional astravault-prod-statement-generator function, which only needs to read one bucket and one secret) requires. |
| **Technical Details** | TMG Security reviewed the fictional Lambda console/CLI configuration for all 14 functions and confirmed each references the same shared execution role, with the aggregate permission set representing the union of every function's requirements rather than each function's individual need. |
| **Attack Scenario** | Exploitation of a code-level vulnerability (e.g., insecure deserialization or a dependency vulnerability) in the lowest-sensitivity of the 14 functions — a fictional public-facing webhook-processor with a comparatively larger attack surface — would grant an attacker the full shared role's permissions, including access to S3 buckets and Secrets Manager entries the webhook processor has no legitimate need to reach. |
| **Impact** | Compromise of any single one of 14 Lambda functions grants the attacker the union of all 14 functions' effective permissions rather than that function's own minimal footprint. |
| **Business Impact** | Shared over-permissioned execution roles are a common serverless anti-pattern that concentrates blast radius; for a platform that processes financial workflows across many discrete functions, this significantly amplifies the impact of any single function-level vulnerability. |
| **Likelihood** | Medium — requires a code-level vulnerability in at least one of the 14 functions, but no additional cloud-specific exploitation technique. |
| **Evidence Reference** | *EV-CLOUD-013 (fictional shared execution role policy export, fictional function-to-role mapping) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Create a dedicated, minimally-scoped execution role per Lambda function reflecting only that function's actual resource needs; use IAM Access Analyzer policy generation from CloudTrail activity to right-size each new role; add a CI/CD check rejecting shared-role reuse across functions. |
| **Recommended AWS Controls** | Per-function IAM roles, IAM Access Analyzer, AWS Config rule for Lambda function public access and role scope review, scheduled least-privilege audits. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will migrate to per-function roles over the next two remediation cycles, prioritizing the internet-facing webhook-processor function first.* |
| **Owner / Priority / Target / Status** | **Serverless Engineering Lead (Fictional Role)** \| 0-30 Days \| 24 October 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming each of the 14 Lambda functions has a distinct execution role scoped to only its own required resources, and that no function retains s3:* / dynamodb:* on Resource "*".* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 08 · Serverless / Lambda Security | Lambda, IAM | arn:aws:iam::123456789012:role/astravault-lambda-exec-role (fictional, shared across 14 fictional Lambda functions) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

**Fictional Shared Role Policy (Fictional / Synthetic)**

```
$ aws iam get-role-policy --role-name astravault-lambda-exec-role --policy-name astravault-lambda-shared-inline
{
  "PolicyDocument": {
    "Statement": [{
      "Effect": "Allow",
      "Action": ["s3:*", "dynamodb:*", "secretsmanager:GetSecretValue"],
      "Resource": "*"
    }]
  }
}
# Referenced identically by all 14 fictional Lambda functions, including astravault-prod-webhook-processor.
```

Expected: Each Lambda function should have a distinct execution role scoped to only the specific resources it needs.

Observed: All 14 fictional Lambda functions shared a single execution role granting s3:*, dynamodb:*, and secretsmanager:GetSecretValue on all resources.

**Impact: Compromise of the lowest-sensitivity function grants the attacker the union of every function's effective permissions.**

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | Exploitation of a code-level vulnerability (e.g., insecure deserialization or a dependency vulnerability) in the lowest-sensitivity of the 14 functions — a fictional public-facing webhook-processor with a comparatively larger attack surface — would grant an attacker the full shared role's permissions, including access to S3 buckets and Secrets Manager entries the webhook processor has no legitimate need to reach. |
| **Likelihood** | Medium — requires a code-level vulnerability in at least one of the 14 functions, but no additional cloud-specific exploitation technique. |
| **Security Impact** | Compromise of any single one of 14 Lambda functions grants the attacker the union of all 14 functions' effective permissions rather than that function's own minimal footprint. |
| **Business Impact** | Shared over-permissioned execution roles are a common serverless anti-pattern that concentrates blast radius; for a platform that processes financial workflows across many discrete functions, this significantly amplifies the impact of any single function-level vulnerability. |
| **Evidence Collected** | EV-CLOUD-013 (fictional shared execution role policy export, fictional function-to-role mapping) |
| **Validation / Retest Guidance** | Retest by confirming each of the 14 Lambda functions has a distinct execution role scoped to only its own required resources, and that no function retains s3:* / dynamodb:* on Resource "*". |

| **CLOUD-010 — RDS Database Instance Publicly Reachable From the Internet** | **HIGH** |
|---|---|
| **CVSS-Style Score: 8.0 (High) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-284 — Improper Access Control** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N — Illustrative vector only` | |
| **Description** | The fictional astravault-analytics-reporting-db RDS instance, used by the analytics team for fictional ad-hoc reporting, is configured with Publicly Accessible set to true and resides in a public subnet, rather than the private-database-subnet pattern used by the primary transactional database. Network-layer exposure is currently narrowed only by the security-group rule reviewed in CLOUD-006-adjacent scope, not by subnet placement. |
| **Technical Details** | External black-box testing confirmed the instance's public DNS endpoint resolves and accepts TCP connections on port 5432 from the internet at the network layer (authentication was not bypassed — see Rules of Engagement); authenticated configuration review confirmed the Publicly Accessible flag and public-subnet placement directly in the RDS console/CLI configuration. |
| **Attack Scenario** | Even without a credential-level compromise, a publicly reachable database endpoint significantly increases exposure to automated credential-stuffing, brute-force, and future authentication-bypass vulnerabilities in the database engine itself, none of which would be possible if the instance were reachable only from within the VPC. |
| **Impact** | A production-adjacent analytics database is reachable at the network layer from any internet host, removing an entire layer of defense-in-depth. |
| **Business Impact** | Analytics databases frequently contain replicated or derived copies of sensitive production data; internet exposure of this instance is inconsistent with the private-subnet pattern correctly applied to AstraVault's primary transactional database and represents an unnecessary, avoidable risk. |
| **Likelihood** | Medium — requires a credential or database-engine vulnerability to progress beyond network reachability, but the exposure itself is immediately discoverable. |
| **Evidence Reference** | *EV-CLOUD-014 (fictional external network reachability confirmation), EV-CLOUD-015 (fictional RDS configuration export) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Set Publicly Accessible to false and move the instance to the private database subnet; if external reporting access is required, front it with a VPN, bastion, or a purpose-built reporting API rather than direct database exposure; rotate database credentials as a precaution. |

| | |
|---|---|
| **Recommended AWS Controls** | RDS private-subnet placement, AWS Config rule rds-instance-public-access-check, VPC-only security-group scoping, IAM database authentication where supported. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will migrate the instance to a private subnet in the next maintenance window.* |
| **Owner / Priority / Target / Status** | **Data Engineering Lead (Fictional Role)** \| 0-30 Days \| 24 October 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming the remediated instance's public DNS endpoint no longer resolves to a routable public IP and that connection attempts from outside the VPC fail at the network layer.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 12 · RDS & Database Security | RDS | astravault-analytics-reporting-db (fictional RDS PostgreSQL instance, Publicly Accessible: true) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

```
Fictional Network Reachability Test (Fictional / Synthetic)
$ nmap -Pn -p 5432 astravault-analytics-reporting-db.fictional-rds.us-east-1.amazonaws.com
PORT     STATE SERVICE
5432/tcp open  postgresql

$ aws rds describe-db-instances --db-instance-identifier astravault-analytics-reporting-db \
    --query 'DBInstances[0].PubliclyAccessible'
true
```

Expected: The RDS instance should have Publicly Accessible set to false and reside in a private database subnet with no direct internet route.

Observed: The instance accepted TCP connections on port 5432 from the internet and was confirmed configured with Publicly Accessible: true.

**Impact: Network-layer exposure of a production-adjacent analytics database to any internet host.**

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | Even without a credential-level compromise, a publicly reachable database endpoint significantly increases exposure to automated credential-stuffing, brute-force, and future authentication-bypass vulnerabilities in the database engine itself, none of which would be possible if the instance were reachable only from within the VPC. |
| **Likelihood** | Medium — requires a credential or database-engine vulnerability to progress beyond network reachability, but the exposure itself is immediately discoverable. |
| **Security Impact** | A production-adjacent analytics database is reachable at the network layer from any internet host, removing an entire layer of defense-in-depth. |
| **Business Impact** | Analytics databases frequently contain replicated or derived copies of sensitive production data; internet exposure of this instance is inconsistent with the private-subnet pattern correctly applied to AstraVault's primary transactional database and represents an unnecessary, avoidable risk. |
| **Evidence Collected** | EV-CLOUD-014 (fictional external network reachability confirmation), EV-CLOUD-015 (fictional RDS configuration export) |
| **Validation / Retest Guidance** | Retest by confirming the remediated instance's public DNS endpoint no longer resolves to a routable public IP and that connection attempts from outside the VPC fail at the network layer. |

| **CLOUD-011 — CloudTrail Logging Coverage Gap on Security-Sensitive API Activity** | **HIGH** |
|---|---|
| **CVSS-Style Score: 7.3 (High) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-778 — Insufficient Logging** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H — Illustrative vector only (Availability reflects detection/response impact)` | |
| **Description** | The fictional astravault-org-trail organization trail captures management events but has S3 data-event logging disabled for the astravault-prod-financial-statements and astravault-prod-analytics-exports buckets, and does not log Lambda data events at all. This leaves object-level access to the most sensitive S3 buckets, and Lambda invocation activity, without an audit trail. |
| **Technical Details** | TMG Security's authenticated configuration review of the CloudTrail trail configuration confirmed IsMultiRegionTrail: true and management-event logging enabled, but the fictional Data Events / Advanced Event Selectors configuration showed no S3 or Lambda data-event selectors configured, consistent with the absence of object-level audit trail evidence when TMG Security's own fictional test access to the CLOUD-002 bucket was reviewed. |
| **Attack Scenario** | An attacker who accesses sensitive S3 object data directly (as in CLOUD-002) or invokes a Lambda function leaves no CloudTrail data-event record of the specific objects accessed or function invocations made, materially slowing incident response and forensic reconstruction even after the underlying access-control gap is fixed. |

| | |
|---|---|
| **Impact** | No audit trail exists for object-level S3 access or Lambda invocation activity on the environment's most sensitive resources. |
| **Business Impact** | Detection and incident-response capability depends on logging coverage matching the sensitivity of the underlying resources; this gap means a real-world equivalent breach of the financial-statements bucket could not be fully scoped from CloudTrail evidence alone. |
| **Likelihood** | N/A — this finding describes a detection gap rather than an exploitable technical vulnerability; likelihood is assessed in terms of undetected exploitation of other findings, which is High given the gap directly covers the CLOUD-002 and CLOUD-003 resource set. |
| **Evidence Reference** | *EV-CLOUD-016 (fictional CloudTrail event-selector configuration export) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Enable S3 data-event logging for all buckets classified as sensitive (starting with astravault-prod-financial-statements and astravault-prod-analytics-exports); enable Lambda data-event logging organization-wide; route resulting events to the centralized fictional Security account log-aggregation pipeline with appropriate retention. |
| **Recommended AWS Controls** | CloudTrail advanced event selectors for S3/Lambda data events, centralized log aggregation to the Security account, CloudWatch Logs Insights or a SIEM-equivalent for correlation, AWS Config rule cloudtrail-s3-dataevents-enabled. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will enable the missing data-event selectors in the next change window.* |
| **Owner / Priority / Target / Status** | **Security Operations Lead (Fictional Role)** \| 0-30 Days \| 24 October 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming S3 and Lambda data events for the in-scope resources now appear in CloudTrail within the expected latency window after a fictional test access event.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 16 · CloudTrail | CloudTrail | astravault-org-trail (fictional organization CloudTrail, AWS Organizations management account) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

```
Fictional Event Selector Export (Fictional / Synthetic)
$ aws cloudtrail get-event-selectors --trail-name astravault-org-trail
{
  "EventSelectors": [{
    "ReadWriteType": "All",
    "IncludeManagementEvents": true,
    "DataResources": []
  }]
}
# No S3 or Lambda data-event resources configured.
```

Expected: S3 data events should be enabled for sensitive buckets and Lambda data events enabled organization-wide, in addition to management-event logging.

Observed: Only management events were logged; the DataResources array was empty.

**Impact: No object-level S3 or Lambda-invocation audit trail exists for the environment's most sensitive resources.**

**TECHNICAL EVIDENCE SUMMARY**

| | |
|---|---|
| **Attack Scenario (Summary)** | An attacker who accesses sensitive S3 object data directly (as in CLOUD-002) or invokes a Lambda function leaves no CloudTrail data-event record of the specific objects accessed or function invocations made, materially slowing incident response and forensic reconstruction even after the underlying access-control gap is fixed. |
| **Likelihood** | N/A — this finding describes a detection gap rather than an exploitable technical vulnerability; likelihood is assessed in terms of undetected exploitation of other findings, which is High given the gap directly covers the CLOUD-002 and CLOUD-003 resource set. |
| **Security Impact** | No audit trail exists for object-level S3 access or Lambda invocation activity on the environment's most sensitive resources. |
| **Business Impact** | Detection and incident-response capability depends on logging coverage matching the sensitivity of the underlying resources; this gap means a real-world equivalent breach of the financial-statements bucket could not be fully scoped from CloudTrail evidence alone. |
| **Evidence Collected** | EV-CLOUD-016 (fictional CloudTrail event-selector configuration export) |
| **Validation / Retest Guidance** | Retest by confirming S3 and Lambda data events for the in-scope resources now appear in CloudTrail within the expected latency window after a fictional test access event. |

# 44 MEDIUM FINDINGS

The 13 Medium-severity findings below represent moderate fictional risk, typically requiring an additional precondition or affecting a less-critical resource. TMG Security recommends remediation within 31-60 days.

| CLOUD-012 — Overly Broad S3 Bucket Resource Policy Grants Unnecessary Cross-Account Access | MEDIUM |
|---|---|
| **CVSS-Style Score: 6.5 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-284 — Improper Access Control** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N — Illustrative vector only` | |
| **Description** | The fictional astravault-prod-vendor-integration-drop bucket policy grants s3:GetObject/PutObject to the entire fictional AstraVault Development account (123456789014) principal, rather than a single scoped vendor-integration role, to support a legacy fictional third-party data exchange that was later re-architected to use a dedicated IAM role. |
| **Technical Details** | Configuration review confirmed the bucket's resource policy retains the broad Development-account grant even though the fictional application owner confirmed only one specific role (astravault-vendor-integration-role) currently uses it. |
| **Attack Scenario** | Any Development-account principal, not just the intended vendor-integration role, can read or write vendor exchange files, creating an unnecessary path for accidental or malicious data exposure or tampering from a broader population than intended. |
| **Impact** | Unnecessary read/write access to vendor exchange data from a broader Development-account population than intended. |
| **Business Impact** | Stale, overly broad resource policies accumulate risk over time as architecture evolves without corresponding policy cleanup; this is a common finding class that individually is moderate but collectively erodes least-privilege posture. |
| **Likelihood** | Low-Medium — requires a compromised or malicious Development-account principal outside the intended role. |
| **Evidence Reference** | *EV-CLOUD-017 (fictional bucket policy export, application-owner interview notes) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Narrow the bucket policy Principal to only astravault-vendor-integration-role's ARN; add a scheduled quarterly review of all cross-account bucket policies against current application ownership records. |
| **Recommended AWS Controls** | IAM Access Analyzer external-access findings review, quarterly resource-policy audit, S3 Access Points for fine-grained per-consumer access scoping. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will narrow the policy in the next scheduled access review.* |
| **Owner / Priority / Target / Status** | **Cloud Platform Engineering Lead (Fictional Role)** \| 31-60 Days \| 14 November 2026 \| Remediation Planned |
| **Validation / Retest Guidance** | *Retest by confirming the bucket policy Principal is scoped to the single vendor-integration role ARN.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 04 · S3 & Object Storage Security | S3 | s3://astravault-prod-vendor-integration-drop (fictional bucket) |

**FULL EVIDENCE PANEL — ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE**

```
Fictional Bucket Policy Excerpt (Fictional / Synthetic)
"Principal": { "AWS": "arn:aws:iam::123456789014:root" },
"Action": ["s3:GetObject", "s3:PutObject"]
```

Expected: Principal should be scoped to the specific vendor-integration role ARN only.

Observed: Principal granted access to the entire Development-account root.

**Impact: Unnecessary read/write access from a broader population than the intended single role.**

| CLOUD-013 — Missing Server-Side Encryption Enforcement on S3 Buckets Storing Sensitive Data | MEDIUM |
|---|---|
| **CVSS-Style Score: 5.9 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-311 — Missing Encryption of Sensitive Data** |

| | |
|---|---|
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:H/PR:H/UI:N/S:U/C:L/I:N/A:N — Illustrative vector only` | |
| **Description** | 6 of 22 fictional production S3 buckets, including astravault-prod-user-uploads, lack a default bucket-level encryption configuration (BucketKeyEnabled / SSE-KMS), meaning objects can be uploaded unencrypted if the uploading application does not explicitly set an encryption header. |
| **Technical Details** | Configuration review of all 22 fictional production buckets' default-encryption settings found 16 correctly enforcing SSE-KMS with a customer-managed key, and 6 with no default encryption configuration set at the bucket level. |
| **Attack Scenario** | This is a defense-in-depth gap rather than a directly exploitable path: any application-layer bug or misconfiguration that uploads without an explicit encryption header would result in unencrypted objects at rest with no bucket-level backstop. |
| **Impact** | Objects in the affected buckets may be stored unencrypted at rest if the uploading application does not explicitly request encryption. |
| **Business Impact** | Encryption-at-rest is frequently a compliance and contractual baseline expectation for a FinTech platform; inconsistent enforcement across buckets creates unnecessary variance in AstraVault's actual security posture versus its likely stated posture. |
| **Likelihood** | Low — requires an application-layer omission of the encryption header; most current upload paths do set it correctly. |
| **Evidence Reference** | *EV-CLOUD-018 (fictional bucket encryption configuration audit across all 22 production buckets) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Apply default SSE-KMS bucket-level encryption with BucketKeyEnabled to all 6 affected buckets; add an SCP or AWS Config rule that denies s3:PutObject without an encryption header organization-wide as a backstop. |
| **Recommended AWS Controls** | S3 default encryption, AWS Config rule s3-bucket-server-side-encryption-enabled, SCP denying unencrypted PutObject, KMS customer-managed keys with rotation enabled. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will apply default encryption to the remaining 6 buckets in the next change window.* |
| **Owner / Priority / Target / Status** | **Cloud Platform Engineering Lead (Fictional Role)** \| 31-60 Days \| 14 November 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming all 22 production buckets now show a default SSE-KMS encryption configuration.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 04 · S3 & Object Storage Security / 14 · KMS & Encryption | S3, KMS | 6 of 22 fictional production S3 buckets, including astravault-prod-user-uploads |

| **CLOUD-014 — S3 Bucket Versioning Disabled on Buckets Storing Financial Records** | **MEDIUM** |
|---|---|
| **CVSS-Style Score: 5.3 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-693 — Protection Mechanism Failure** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:U/C:N/I:L/A:L — Illustrative vector only` | |
| **Description** | The fictional astravault-prod-financial-statements bucket has S3 Versioning set to Suspended rather than Enabled, meaning an accidental or malicious overwrite/delete of a financial-statement object cannot be recovered through S3's native version history. |
| **Technical Details** | Configuration review confirmed the versioning state directly via the bucket's Versioning configuration API response, cross-referenced against the fictional change-management log showing versioning was suspended during a 2025 migration and never re-enabled. |
| **Attack Scenario** | Any principal with s3:PutObject/DeleteObject on this bucket (intentionally or via a compromised credential) can irrecoverably overwrite or delete a fictional customer financial statement with no built-in recovery path beyond the separate backup finding tracked in CLOUD-019. |
| **Impact** | No native version-history recovery path exists for accidental or malicious modification of financial-statement objects. |
| **Business Impact** | Financial record integrity and recoverability is a reasonable baseline expectation for a FinTech platform; combined with the backup gap in CLOUD-019, this increases the risk of unrecoverable data loss for regulated financial documents. |
| **Likelihood** | Low — requires either an operational mistake or a compromised credential with write/delete access to this specific bucket. |
| **Evidence Reference** | *EV-CLOUD-019 (fictional bucket versioning configuration export) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Re-enable S3 Versioning on this and all buckets storing financial or regulated records; combine with S3 Object Lock (compliance mode) for the subset of records with a defined regulatory retention requirement. |
| **Recommended AWS Controls** | S3 Versioning, S3 Object Lock for compliance-retention data, AWS Config rule s3-bucket-versioning-enabled, MFA Delete for the highest-sensitivity buckets. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will re-enable versioning in the next change window.* |
| **Owner / Priority / Target / Status** | **Cloud Platform Engineering Lead (Fictional Role)** \| 31-60 Days \| 14 November 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming the bucket's Versioning configuration returns Enabled and that a test object overwrite produces a* |

| | |
|---|---|
| | *recoverable prior version.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 04 · S3 & Object Storage Security / 20 · Backup & Recovery Security | S3 | s3://astravault-prod-financial-statements (fictional; versioning: Suspended) |

| **CLOUD-015 — Insufficient CloudWatch Alerting on High-Risk IAM and Security Events** | **MEDIUM** |
|---|---|
| **CVSS-Style Score: 5.7 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-778 — Insufficient Logging** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:N/A:H — Illustrative vector only (Availability reflects detection/response impact)` | |
| **Description** | Of a recommended baseline set of 12 high-risk-event CloudWatch metric-filter alarms (e.g., root-account usage, IAM policy changes, security-group changes, failed console logins, disabling of CloudTrail/GuardDuty), only 4 are currently configured and routed to the fictional astravault-security-alerts SNS topic. |
| **Technical Details** | Review of the fictional CloudWatch Alarms and metric-filter configuration confirmed alarms exist for root-account usage and unauthorized-API-call patterns, but not for IAM policy changes, security-group changes, network-ACL changes, or CloudTrail/GuardDuty configuration changes. |
| **Attack Scenario** | Several of this report's own findings (e.g., CLOUD-001's cross-account role modification, CLOUD-006's security-group exposure) would not, on their own, trigger any alert to the fictional security team even after remediation, if a similar misconfiguration were reintroduced. |
| **Impact** | Several classes of high-risk configuration change would occur without any automated alert to the security team. |
| **Business Impact** | Detective controls are a compensating layer for preventive-control gaps; the current alerting coverage means several of this report's own finding classes could recur silently. |
| **Likelihood** | N/A — detection gap rather than a directly exploitable vulnerability. |
| **Evidence Reference** | *EV-CLOUD-020 (fictional CloudWatch alarm inventory vs. recommended baseline) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Implement the remaining 8 baseline CloudWatch metric-filter alarms (IAM/security-group/NACL/CloudTrail/GuardDuty configuration changes) and route all 12 to the security-alerts SNS topic with a defined on-call escalation path. |
| **Recommended AWS Controls** | CloudWatch metric filters and alarms per the CIS AWS Foundations-aligned baseline, EventBridge rules for GuardDuty/Security Hub finding routing, a defined on-call runbook. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will implement the remaining alarms in the next sprint.* |
| **Owner / Priority / Target / Status** | **Security Operations Lead (Fictional Role)** \| 31-60 Days \| 14 November 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming all 12 baseline alarms exist and fire correctly against a fictional test event for each covered change type.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 15 · Logging & Monitoring | CloudWatch, EventBridge, SNS | astravault-security-alerts SNS topic (fictional; 4 of 12 recommended CloudWatch metric-filter alarms configured) |

| **CLOUD-016 — Permissive Security Group Egress Rules Allow Unrestricted Outbound Access** | **MEDIUM** |
|---|---|
| **CVSS-Style Score: 5.4 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-284 — Improper Access Control** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:A/AC:H/PR:L/UI:N/S:U/C:L/I:N/A:N — Illustrative vector only` | |
| **Description** | 18 of 24 fictional production security groups retain the AWS default egress rule permitting all outbound traffic to 0.0.0.0/0 on all ports, rather than a scoped egress policy limiting workloads to only their required destinations (e.g., the RDS cluster, specific AWS service endpoints, and approved third-party APIs). |
| **Technical Details** | Configuration review of all 24 fictional production security groups found 6 with scoped egress rules and 18 retaining the unmodified AWS default. |
| **Attack Scenario** | Following any workload compromise (such as the CLOUD-008 container vulnerability), unrestricted egress makes data exfiltration and command-and-control communication to attacker-controlled infrastructure straightforward, where scoped egress would have blocked or at least flagged the anomalous outbound connection. |
| **Impact** | Compromised workloads face no network-layer barrier to outbound data exfiltration or command-and-control communication. |

| | |
|---|---|
| **Business Impact** | Egress filtering is a well-established defense-in-depth control against data exfiltration; its absence compounds the impact of any successful initial-access finding elsewhere in this report. |
| **Likelihood** | Low — requires a prior workload compromise to become relevant; not independently exploitable. |
| **Evidence Reference** | *EV-CLOUD-021 (fictional security group egress rule audit across 24 production groups) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Define and apply scoped egress rules per workload tier (application, database, management) limiting outbound traffic to required destinations only; use VPC endpoints for AWS service traffic to avoid needing broad internet egress at all. |
| **Recommended AWS Controls** | Scoped security-group egress rules, VPC endpoints (S3, DynamoDB, Secrets Manager, KMS), AWS Network Firewall or a NAT Gateway with flow-log-based anomaly detection, GuardDuty for anomalous outbound traffic detection. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will roll out scoped egress rules tier-by-tier over the next two remediation cycles.* |
| **Owner / Priority / Target / Status** | **Network Engineering Lead (Fictional Role)** \| 31-60 Days \| 21 November 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming each of the 24 production security groups now has a scoped egress rule set consistent with its workload tier's documented requirements.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 07 · Security Groups & Network ACLs | EC2 Security Groups | 18 of 24 fictional production security groups (default egress: all traffic, 0.0.0.0/0) |

| **CLOUD-017 — Stale IAM Users With Long-Unused Console and API Access** | **MEDIUM** |
|---|---|
| **CVSS-Style Score: 5.0 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-262 — Not Using Password Aging** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:L/A:N — Illustrative vector only` | |
| **Description** | The fictional IAM credential report shows 7 of 41 IAM users with no console login and no API activity (via access keys) for 180 or more days, including two fictional former-contractor accounts that were never deactivated at offboarding. |
| **Technical Details** | TMG Security reviewed the fictional aws iam generate-credential-report output, cross-referencing PasswordLastUsed and AccessKeyLastUsed fields against a 180-day threshold consistent with AstraVault's own fictional internal access-review policy. |
| **Attack Scenario** | Stale, un-offboarded accounts represent unnecessary standing attack surface — a former contractor's still-active credential is a target that requires no new vulnerability to exploit, only credential compromise (phishing, credential-stuffing against a reused password) of an account nobody is actively monitoring. |
| **Impact** | 7 unnecessary standing identities, including 2 confirmed former-contractor accounts, remain active with no legitimate current business need. |
| **Business Impact** | Unmanaged identity lifecycle is a common root cause of real-world unauthorized-access incidents; this finding is straightforward to remediate and represents low-hanging governance risk. |
| **Likelihood** | Low — requires compromise of a specific stale credential, but no exploitation technique beyond standard credential-theft methods. |
| **Evidence Reference** | *EV-CLOUD-022 (fictional IAM credential report export, filtered to 180+ day inactivity) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Deactivate or delete all 7 identified stale accounts, prioritizing the 2 former-contractor accounts immediately; implement an automated 90-day inactivity deactivation policy and integrate IAM offboarding into AstraVault's fictional HR offboarding workflow. |
| **Recommended AWS Controls** | IAM credential reports reviewed on a recurring schedule, automated inactive-credential deactivation (e.g., via a scheduled Lambda function), HR-system-to-IAM offboarding integration, AWS IAM Identity Center for centralized lifecycle management. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and deactivated the 2 former-contractor accounts immediately upon notification; the remaining 5 are under review.* |
| **Owner / Priority / Target / Status** | **IAM & Identity Lead (Fictional Role)** \| 31-60 Days \| 21 November 2026 \| **Remediation In Progress** |
| **Validation / Retest Guidance** | *Retest by regenerating the fictional IAM credential report and confirming no user exceeds the 90-day inactivity threshold without a documented exception.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 02 · IAM & Identity Security | IAM | 7 of 41 fictional IAM users with no console or API activity for 180+ days (per fictional IAM credential report) |

| CLOUD-018 — Unused and Unrotated IAM Access Keys Exceeding Recommended Age | MEDIUM |
|---|---|
| CVSS-Style Score: 4.9 (Medium) — Illustrative CVSS v3.1-inspired score | CWE: CWE-798 — Use of Hard-coded Credentials |
| CVSS Vector (Illustrative): `CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:L/A:N — Illustrative vector only` | |
| Description | 11 of 41 fictional IAM users carry at least one access key older than 365 days with no rotation, exceeding AstraVault's fictional own documented 90-day rotation policy by a wide margin. |
| Technical Details | The same fictional IAM credential report used in CLOUD-017 was reviewed for AccessKey1LastRotated/AccessKey2LastRotated fields, confirming 11 users with a key age exceeding 365 days, including 3 users whose keys have never been rotated since initial creation. |
| Attack Scenario | Long-lived, unrotated access keys increase the window of usefulness for any historically leaked or logged key value (e.g., accidentally committed to a fictional code repository at some point in the past) and are inconsistent with AstraVault's own stated policy. |
| Impact | Extended exposure window for any historically compromised or leaked access key value tied to these 11 identities. |
| Business Impact | Access-key rotation is a low-cost, high-value control; the gap between AstraVault's documented policy and actual practice suggests a governance/enforcement issue rather than a policy-design issue. |
| Likelihood | Low — requires a separate leak or compromise event involving one of the 11 long-lived keys. |
| Evidence Reference | *EV-CLOUD-023 (fictional IAM credential report, access-key-age analysis) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| Recommendation | Rotate all 11 identified keys immediately; migrate long-lived human-user access keys to short-lived federated access via IAM Identity Center or AWS SSO wherever feasible, eliminating the need for long-lived keys entirely for most users. |
| Recommended AWS Controls | IAM Access Analyzer unused-access findings, automated key-age alerting, IAM Identity Center federated access, scheduled key-rotation enforcement via a Config rule (access-keys-rotated). |
| Management Response | *AstraVault engineering (fictional) has accepted this finding and will rotate the identified keys and evaluate IAM Identity Center migration in the next quarter.* |
| Owner / Priority / Target / Status | **IAM & Identity Lead (Fictional Role)** \| 31-60 Days \| 21 November 2026 \| Remediation Planned |
| Validation / Retest Guidance | *Retest by confirming all 11 identified keys have been rotated within the prior 90 days and that the AWS Config access-keys-rotated rule reports compliant.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 02 · IAM & Identity Security | IAM | 11 of 41 fictional IAM users with an access key older than 365 days |

| CLOUD-019 — Weak AWS Backup Configuration for Production Database Resources | MEDIUM |
|---|---|
| CVSS-Style Score: 5.2 (Medium) — Illustrative CVSS v3.1-inspired score | CWE: CWE-693 — Protection Mechanism Failure |
| CVSS Vector (Illustrative): `CVSS:3.1/AV:N/AC:H/PR:H/UI:N/S:U/C:N/I:N/A:H — Illustrative vector only` | |
| Description | The fictional AWS Backup plan covering astravault-prod-financial-db retains backups for only 7 days with no cross-region copy configured, and the backup vault does not have Vault Lock enabled, leaving backups modifiable or deletable by any principal with backup:DeleteRecoveryPoint permission. |
| Technical Details | Review of the fictional AWS Backup plan and backup-vault access policy confirmed the 7-day retention window and absence of both cross-region copy and Vault Lock, against AstraVault's fictional internal data-retention target of 35 days for financial database backups. |
| Attack Scenario | A ransomware-style scenario (destructive testing itself was out of scope per the Rules of Engagement, but the configuration was reviewed) combining database compromise with deletion of recent recovery points would leave AstraVault with recovery options limited to whatever falls within the 7-day window, and no protection against an attacker with backup-delete permissions removing those recovery points entirely. |
| Impact | Limited recovery window and no immutability protection for production database backups. |
| Business Impact | For a FinTech platform, backup resilience is directly tied to business continuity and regulatory recovery-point expectations; a 7-day window with no immutability is below a reasonable baseline for financial data. |
| Likelihood | Low — requires either an extended undetected compromise or an attacker with specific backup-delete permissions. |
| Evidence Reference | *EV-CLOUD-024 (fictional AWS Backup plan and vault-policy configuration export) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| Recommendation | Extend retention to at least 35 days with a cross-region copy to the fictional Security account; enable AWS Backup Vault Lock in compliance mode for the production financial database backup vault to prevent deletion even by account |

| | |
|---|---|
| | administrators within the locked retention period. |
| **Recommended AWS Controls** | AWS Backup Vault Lock, cross-region/cross-account backup copies, scheduled restore testing, AWS Config rule backup-plan-min-frequency-and-min-retention-check. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will extend retention and enable Vault Lock in the next change window.* |
| **Owner / Priority / Target / Status** | **Data Engineering Lead (Fictional Role)** \| 31-60 Days \| 21 November 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming the backup plan's retention window, cross-region copy configuration, and Vault Lock status meet the revised policy.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 20 · Backup & Recovery Security | AWS Backup, RDS | astravault-prod-financial-db (fictional; AWS Backup plan retention: 7 days, no cross-region copy) |

| **CLOUD-020 — Lambda Function Configured With Excessive Timeout and Memory Beyond Operational Need** | **MEDIUM** |
|---|---|
| **CVSS-Style Score: 4.6 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-400 — Uncontrolled Resource Consumption** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:U/C:N/I:N/A:L — Illustrative vector only` | |
| **Description** | The fictional astravault-prod-webhook-processor function is configured with the maximum allowed 900-second timeout and 3008 MB memory allocation, while fictional observed CloudWatch metrics show a p99 runtime of 4 seconds and peak memory usage of roughly 256 MB, indicating configuration far beyond operational need. |
| **Technical Details** | TMG Security reviewed fictional CloudWatch Lambda Insights metrics for the function against its configured limits, confirming a large gap between actual and provisioned resources. |
| **Attack Scenario** | In combination with a code-level vulnerability enabling resource-exhaustion or recursive-invocation abuse, the oversized timeout and memory allocation increase both the potential cost-based denial-of-wallet impact and the window available for an attacker to abuse a compromised invocation before the function is forcibly terminated. |
| **Impact** | Unnecessarily large resource ceiling increases both cost-abuse potential and the operational window available to a compromised invocation. |
| **Business Impact** | This is a hardening/cost-governance finding rather than a directly exploitable vulnerability on its own, but right-sizing reduces both cloud spend risk and the blast radius of any future function-level compromise. |
| **Likelihood** | Low — requires a separate code-level vulnerability to become a meaningful abuse vector. |
| **Evidence Reference** | *EV-CLOUD-025 (fictional CloudWatch Lambda Insights metric export vs. configured limits) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Right-size the function's timeout and memory allocation based on observed p99 metrics with reasonable headroom (e.g., 30s timeout, 512 MB memory); apply AWS Budgets alerting on Lambda cost anomalies as a compensating control. |
| **Recommended AWS Controls** | Lambda Power Tuning for right-sizing, AWS Budgets cost-anomaly alerts, reserved concurrency limits to cap invocation-based cost abuse. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will right-size the function configuration in an upcoming maintenance cycle.* |
| **Owner / Priority / Target / Status** | **Serverless Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming the function's configured timeout and memory align with observed p99 usage plus reasonable headroom.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 08 · Serverless / Lambda Security | Lambda | astravault-prod-webhook-processor (fictional; timeout: 900s, memory: 3008 MB against an observed p99 runtime of 4s / 256 MB) |

| **CLOUD-021 — Missing Workload Integrity Controls on EC2 AMI Build Pipeline** | **MEDIUM** |
|---|---|
| **CVSS-Style Score: 5.6 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-345 — Insufficient Verification of Data Authenticity** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:H/PR:H/UI:N/S:U/C:N/I:H/A:N — Illustrative vector only` | |
| **Description** | The fictional AMI build pipeline that produces astravault-golden-ami, used to launch the production EC2 fleet, does not |

| | |
|---|---|
| | sign the resulting image or generate a software bill of materials (SBOM), and the AMI-launch process does not verify image provenance before use in Auto Scaling group launch templates. |
| **Technical Details** | TMG Security reviewed the fictional AMI pipeline's build and publish steps and confirmed no signing (e.g., via EC2 Image Builder's image signing support) or SBOM generation occurs, and the launch template references the AMI by ID with no additional integrity check. |
| **Attack Scenario** | A compromise of the AMI build pipeline's credentials (a separate CI/CD-security concern) could allow an attacker to publish a tampered golden AMI that would be trusted and launched fleet-wide with no provenance check to catch the substitution. |
| **Impact** | No integrity verification exists between AMI build and fleet-wide launch, relying entirely on pipeline-credential security as the sole control. |
| **Business Impact** | Golden-image integrity is a foundational assumption for fleet-wide EC2 security; without provenance verification, this assumption has no independent enforcement. |
| **Likelihood** | Low — requires compromise of the AMI build pipeline's own credentials, a separate and currently reasonably-controlled surface. |
| **Evidence Reference** | *EV-CLOUD-026 (fictional AMI pipeline configuration and launch-template review) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Adopt EC2 Image Builder with image signing enabled; generate and retain an SBOM for each golden AMI build; add a launch-time or Config-rule-based check verifying the AMI's signature/provenance before it is permitted in production launch templates. |
| **Recommended AWS Controls** | EC2 Image Builder with signing, SBOM generation (e.g., via Syft or EC2 Image Builder components), AWS Config custom rule for AMI provenance verification. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will evaluate EC2 Image Builder adoption in the next planning cycle.* |
| **Owner / Priority / Target / Status** | **Platform Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming the remediated pipeline produces signed AMIs with an associated SBOM and that launch templates verify provenance before use.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 05 · EC2 & Workload Security / 24 · CI/CD & Cloud Deployment Security | EC2, Systems Manager, CI/CD Pipeline | astravault-golden-ami build pipeline (fictional; no image signing or SBOM generation) |

| **CLOUD-022 — Insufficient Security-Header Controls at CloudFront Edge** | **MEDIUM** |
|---|---|
| **CVSS-Style Score: 5.1 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-1021 — Improper Restriction of Rendered UI Layers** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:L/I:L/A:N — Illustrative vector only` | |
| **Description** | The fictional CloudFront distribution fronting app.astravault-example.com does not attach a Response Headers Policy, so responses lack Content-Security-Policy, X-Frame-Options, Strict-Transport-Security, and X-Content-Type-Options headers at the edge, even though the origin application partially sets some of these inconsistently across routes. |
| **Technical Details** | TMG Security's external testing captured response headers across a representative sample of fictional application routes served through the distribution and confirmed inconsistent and, on several routes, entirely absent security headers, then confirmed via authenticated review that no CloudFront Response Headers Policy was attached to enforce them centrally. |
| **Attack Scenario** | Inconsistent security-header enforcement increases the residual risk of clickjacking and certain client-side injection classes on the subset of routes where the origin itself does not set the headers, since there is no edge-level backstop. |
| **Impact** | Missing or inconsistent browser-security headers on a subset of application routes. |
| **Business Impact** | This is a defense-in-depth gap rather than a standalone critical exposure, but centralizing header enforcement at the CDN edge is a low-cost control that closes the gap for all current and future routes at once. |
| **Likelihood** | Low — requires an additional client-side vulnerability class to be meaningfully exploitable. |
| **Evidence Reference** | *EV-CLOUD-027 (fictional response-header capture across sampled application routes) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Attach a CloudFront Response Headers Policy enforcing Content-Security-Policy, Strict-Transport-Security (with includeSubDomains and preload), X-Frame-Options: DENY, and X-Content-Type-Options: nosniff across the distribution, independent of origin-level header behavior. |
| **Recommended AWS Controls** | CloudFront Response Headers Policy, AWS WAF managed rule groups for common web exploits, periodic header-configuration testing as part of routine change validation. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will attach a Response Headers Policy in the next front-end release.* |
| **Owner / Priority / Target / Status** | **Web Platform Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |

| | |
|---|---|
| **Validation / Retest Guidance** | *Retest by re-capturing response headers across the same sampled routes and confirming consistent enforcement of all four header classes.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 18 · CloudFront / CDN Security | CloudFront | E1FICTIONALCF01 (fictional CloudFront distribution serving app.astravault-example.com) |

| **CLOUD-023 — KMS Key Policy Grants Overly Broad Decrypt Permissions** | **MEDIUM** |
|---|---|
| **CVSS-Style Score: 6.1 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-284 — Improper Access Control** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N — Illustrative vector only` | |
| **Description** | The fictional alias/astravault-prod-data-key customer-managed KMS key, used to encrypt several sensitive S3 buckets and Secrets Manager entries, has a key policy granting kms:Decrypt to the entire fictional AstraVault Production account principal via the default key-policy statement, rather than a narrower set of roles that actually need decrypt access. |
| **Technical Details** | Review of the fictional key policy confirmed the default "Enable IAM User Permissions" statement was left in place alongside application-specific grants, effectively allowing any Production-account principal with a permissive IAM policy to decrypt data protected by this key, independent of whether that principal has a legitimate need. |
| **Attack Scenario** | Encryption-at-rest provides limited additional protection beyond the underlying IAM access controls when the KMS key policy itself does not narrow who can decrypt; an attacker who obtains any sufficiently-permissive Production IAM credential can decrypt protected data without needing a dedicated key-specific grant. |
| **Impact** | KMS encryption provides less independent protection than intended because decrypt access is not scoped beyond the account's general IAM posture. |
| **Business Impact** | For sensitive financial data, encryption is often expected to provide a meaningful additional control layer beyond IAM alone; the current configuration reduces KMS to largely redundant with IAM rather than an independent layer of defense-in-depth. |
| **Likelihood** | Low-Medium — requires a sufficiently permissive IAM credential, which several other findings in this report show is more attainable than intended. |
| **Evidence Reference** | *EV-CLOUD-028 (fictional KMS key policy export) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Replace the default IAM-permissions key-policy statement with explicit, narrowly-scoped kms:Decrypt grants limited to the specific application and service roles that require it; add a kms:ViaService condition restricting decrypt to requests originating from the intended AWS service (e.g., S3, Secrets Manager). |
| **Recommended AWS Controls** | Scoped KMS key policies, kms:ViaService and kms:CallerAccount conditions, separate KMS keys per data-sensitivity tier, CloudTrail monitoring of kms:Decrypt calls for anomaly detection. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will re-scope the key policy in the next change window.* |
| **Owner / Priority / Target / Status** | **Cloud Security Engineering Lead (Fictional Role)** \| 31-60 Days \| 21 November 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming the remediated key policy denies kms:Decrypt from a fictional test principal outside the explicitly scoped role set.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 14 · KMS & Encryption | KMS | alias/astravault-prod-data-key (fictional customer-managed KMS key) |

| **CLOUD-024 — ECS Task Definition Runs Containers With Elevated Linux Capabilities** | **MEDIUM** |
|---|---|
| **CVSS-Style Score: 5.8 (Medium) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-250 — Execution with Unnecessary Privileges** |
| **CVSS Vector (Illustrative):** `CVSS:3.1/AV:L/AC:H/PR:L/UI:N/S:C/C:L/I:L/A:L — Illustrative vector only` | |
| **Description** | The fictional astravault-api-gateway-service ECS task definition is not fully privileged, but explicitly adds the SYS_ADMIN and NET_ADMIN Linux capabilities — both broader than the container's documented functional need — rather than running with the container runtime's default, more restrictive capability set. |
| **Technical Details** | Review of the fictional task definition's linuxParameters.capabilities.add array confirmed both capabilities were added, and the fictional application owner confirmed via interview that neither is required for the service's current functionality, appearing to be leftover from an earlier debugging configuration. |

| | |
|---|---|
| **Attack Scenario** | In the event of container-level code execution (e.g., via the CLOUD-008 vulnerable-image finding), the added SYS_ADMIN capability in particular meaningfully increases the feasibility of container-escape or host-level privilege-escalation techniques compared to a default, minimally-privileged capability set. |
| **Impact** | Elevated container privileges increase the feasibility of container-escape techniques following any container-level compromise. |
| **Business Impact** | Combined with CLOUD-008's vulnerable image, this finding increases the potential severity of a successful exploit against the same service from container-contained to host-level. |
| **Likelihood** | Low — requires a prior container-level compromise to become relevant; not independently exploitable. |
| **Evidence Reference** | *EV-CLOUD-029 (fictional ECS task definition linuxParameters export, application-owner interview notes) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Remove the SYS_ADMIN and NET_ADMIN capability grants from the task definition; run with the container runtime's default capability set (or narrower) unless a specific, documented functional need is identified and independently reviewed. |
| **Recommended AWS Controls** | ECS task definition capability minimization, AWS Config custom rule flagging added Linux capabilities, runtime-security tooling (e.g., GuardDuty ECS Runtime Monitoring) for anomalous syscall detection. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will remove the unnecessary capability grants in the next deployment.* |
| **Owner / Priority / Target / Status** | **Platform Engineering Lead (Fictional Role)** \| 31-60 Days \| 21 November 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming the remediated task definition no longer adds SYS_ADMIN or NET_ADMIN and that the service continues to function correctly without them.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 10 · Container Security / 23 · Container Escape & Workload Isolation | ECS | astravault-api-gateway-service task definition (fictional; privileged: false, but linuxParameters.capabilities.add includes SYS_ADMIN and NET_ADMIN) |

# 45 LOW FINDINGS

The 9 Low-severity findings below are primarily defense-in-depth, logging, and hygiene gaps with limited standalone exploitability. TMG Security recommends remediation within 61-90 days.

| CLOUD-025 — Incomplete Resource Tagging Undermines Cost and Security Governance | LOW |
|---|---|
| **CVSS-Style Score: 3.1 (Low) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-1059 — Insufficient Technical Documentation** |
| **Description** | A fictional organization-wide tag audit found approximately 38% of production resources missing one or more of AstraVault's three fictional mandatory tags (Owner, DataClassification, Environment), most commonly on EC2 instances and S3 buckets created outside the standard fictional Terraform-based provisioning workflow. |
| **Technical Details** | TMG Security used the fictional AWS Resource Groups Tag Editor to enumerate tag coverage across the Production account's resource inventory against the three mandatory tag keys. |
| **Attack Scenario** | This is a governance gap rather than a directly exploitable vulnerability; its relevance to this assessment is that missing DataClassification tags made several findings (including CLOUD-002's bucket) harder to prioritize quickly during the assessment itself, since sensitivity had to be confirmed through interview rather than tag metadata. |
| **Impact** | Reduced ability to programmatically identify resource ownership, sensitivity, and environment for both security triage and cost-allocation purposes. |
| **Business Impact** | Tagging gaps compound the time-to-detect and time-to-remediate for future findings by removing a fast, automatable way to identify what a given resource is and who owns it. |
| **Likelihood** | N/A — governance/hygiene finding, not independently exploitable. |
| **Evidence Reference** | *EV-CLOUD-030 (fictional tag-coverage audit export) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Enforce mandatory tagging via an SCP or AWS Config rule (required-tags) that flags or restricts creation of untagged resources; backfill tags on the existing 38% gap using the fictional Terraform state as the source of truth where available. |
| **Recommended AWS Controls** | AWS Config required-tags rule, SCP-level tag enforcement on resource creation, Terraform-enforced tagging module reused across all fictional provisioning. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will backfill tags and implement enforcement in the next quarter.* |
| **Owner / Priority / Target / Status** | **Cloud Platform Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by re-running the fictional tag-coverage audit and confirming compliance above 95% with enforcement blocking new untagged resources.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 21 · Cloud Misconfiguration | Resource Groups & Tag Editor (organization-wide) | Approximately 38% of fictional production resources missing one or more mandatory tags (Owner, DataClassification, Environment) |

| CLOUD-026 — S3 Access Logging Not Enabled on Selected Buckets | LOW |
|---|---|
| **CVSS-Style Score: 3.4 (Low) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-778 — Insufficient Logging** |
| **Description** | 5 of 22 fictional production S3 buckets, all lower-traffic internal-tooling buckets, do not have S3 Server Access Logging enabled, providing no request-level log independent of the CloudTrail data-event gap already tracked in CLOUD-011. |
| **Technical Details** | Configuration review of all 22 fictional production buckets' logging configuration found 17 with access logging enabled and 5 without. |
| **Attack Scenario** | Not independently exploitable; represents a supplementary logging gap that would slightly slow investigation of any incident involving these 5 lower-traffic buckets specifically. |
| **Impact** | Reduced request-level visibility for the 5 affected buckets. |
| **Business Impact** | Minor incremental risk given these are lower-traffic internal buckets and CloudTrail data-event logging (once CLOUD-011 is remediated) will provide overlapping coverage. |
| **Likelihood** | N/A — logging-hygiene finding. |

| | |
|---|---|
| **Evidence Reference** | *EV-CLOUD-031 (fictional bucket logging configuration audit) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Enable S3 Server Access Logging on the 5 identified buckets, directing logs to the existing fictional centralized logging bucket used by the other 17. |
| **Recommended AWS Controls** | S3 Server Access Logging, AWS Config rule s3-bucket-logging-enabled. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will enable logging on the remaining buckets in the next change window.* |
| **Owner / Priority / Target / Status** | **Cloud Platform Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming all 22 production buckets now have Server Access Logging enabled.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 04 · S3 & Object Storage Security / 15 · Logging & Monitoring | S3 | 5 of 22 fictional production buckets without S3 Server Access Logging enabled |

| **CLOUD-027 — VPC Flow Logs Not Enabled on Selected VPCs** | **LOW** |
|---|---|
| **CVSS-Style Score: 3.6 (Low) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-778 — Insufficient Logging** |
| **Description** | The fictional Development-account VPC does not have VPC Flow Logs enabled, unlike the Production and Security account VPCs, which both log to the centralized fictional log-aggregation pipeline. |
| **Technical Details** | Configuration review confirmed Flow Logs status per VPC across all three fictional accounts, finding only the Development VPC without them. |
| **Attack Scenario** | Given the cross-account trust gap identified in CLOUD-005, a Development-account compromise used as a stepping stone toward Production would leave a weaker network-flow audit trail in the account where initial access most plausibly occurs. |
| **Impact** | No network-flow audit trail exists for the Development account. |
| **Business Impact** | This gap is more significant in combination with CLOUD-005 than in isolation, since the Development account is the more likely initial-access point into the broader environment. |
| **Likelihood** | N/A — logging-hygiene finding. |
| **Evidence Reference** | *EV-CLOUD-032 (fictional VPC Flow Logs configuration audit across all three accounts) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Enable VPC Flow Logs on the Development-account VPC, routed to the same centralized log-aggregation pipeline used by Production and Security. |
| **Recommended AWS Controls** | VPC Flow Logs (all traffic), centralized log aggregation, AWS Config rule vpc-flow-logs-enabled applied organization-wide via Config aggregator. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will enable Flow Logs in the next change window.* |
| **Owner / Priority / Target / Status** | **Network Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming Flow Logs are enabled and flowing to the centralized pipeline for the Development VPC.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 06 · VPC & Network Security / 15 · Logging & Monitoring | VPC | vpc-0fictdev01 (fictional AstraVault Development-account VPC) |

| **CLOUD-028 — Route 53 DNS Zone Lacks Query Logging** | **LOW** |
|---|---|
| **CVSS-Style Score: 2.8 (Low) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-778 — Insufficient Logging** |
| **Description** | The fictional public Route 53 hosted zone for astravault-example.com does not have query logging enabled, limiting visibility into DNS-level reconnaissance or abuse patterns targeting AstraVault's fictional public domain. |
| **Technical Details** | Configuration review confirmed no query-logging configuration associated with the hosted zone. |
| **Attack Scenario** | Not independently exploitable; DNS query logging is a reconnaissance-detection and incident-forensics aid rather than a preventive control. |
| **Impact** | Reduced visibility into DNS-level reconnaissance activity against the public domain. |
| **Business Impact** | Minor; primarily affects investigative capability during and after an incident rather than preventing one. |
| **Likelihood** | N/A — logging-hygiene finding. |

| | |
|---|---|
| **Evidence Reference** | *EV-CLOUD-033 (fictional Route 53 hosted-zone configuration review) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Enable Route 53 Resolver query logging (or hosted-zone query logging) with logs routed to the centralized fictional log-aggregation pipeline. |
| **Recommended AWS Controls** | Route 53 query logging, CloudWatch Logs Insights for DNS query analysis. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will enable query logging in the next change window.* |
| **Owner / Priority / Target / Status** | **Network Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming query logging is enabled and log entries are flowing to the centralized pipeline.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 01 · External Cloud Attack Surface / 15 · Logging & Monitoring | Route 53 | astravault-example.com (fictional public hosted zone) |

| **CLOUD-029 — GuardDuty Findings Not Routed to a Centralized Alerting Channel** | **LOW** |
|---|---|
| **CVSS-Style Score: 3.9 (Low) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-778 — Insufficient Logging** |
| **Description** | GuardDuty is enabled organization-wide across all three fictional accounts, but findings are only reviewable by manually opening the console; no EventBridge rule routes Medium-or-higher findings to the security-alerts SNS topic or a fictional chat-ops channel. |
| **Technical Details** | TMG Security reviewed the fictional GuardDuty configuration and confirmed detector status enabled with default finding-publishing frequency, but found no associated EventBridge rule pattern matching GuardDuty finding events. |
| **Attack Scenario** | GuardDuty's detective value is significantly reduced without active routing, since findings could go unnoticed for an extended period absent a team member proactively checking the console. |
| **Impact** | GuardDuty findings, including any that would correlate with this report's own findings, are not proactively surfaced to the security team. |
| **Business Impact** | This compounds the CLOUD-015 alerting gap; GuardDuty is a relatively low-cost detective control whose value depends entirely on downstream routing and response. |
| **Likelihood** | N/A — detection-gap finding. |
| **Evidence Reference** | *EV-CLOUD-034 (fictional GuardDuty and EventBridge configuration review) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Create an EventBridge rule matching GuardDuty finding events at Medium severity and above, routed to the security-alerts SNS topic and a fictional chat-ops integration, with a defined triage SLA. |
| **Recommended AWS Controls** | EventBridge GuardDuty finding routing, SNS/chat-ops integration, Security Hub as a finding-aggregation layer across GuardDuty/Config/Inspector. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will implement the routing rule in the next sprint.* |
| **Owner / Priority / Target / Status** | **Security Operations Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by generating a fictional GuardDuty sample finding and confirming it is routed to the alerting channel within the expected latency window.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 25 · Cloud Persistence & Detection | GuardDuty, EventBridge, SNS | GuardDuty detector (fictional; findings visible only in-console, no EventBridge rule to SNS/chat routing) |

| **CLOUD-030 — WAF Rule Set Missing Rate-Based Rule on Public API Gateway** | **LOW** |
|---|---|
| **CVSS-Style Score: 3.8 (Low) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-770 — Allocation of Resources Without Limits or Throttling** |
| **Description** | The fictional WAF Web ACL protecting the public API Gateway stage includes AWS Managed Rule groups for common threats (SQLi, known bad inputs) but does not include a rate-based rule, leaving the API without WAF-layer protection against high-volume credential-stuffing or scraping traffic beyond API Gateway's own default account-level throttling. |
| **Technical Details** | Review of the fictional Web ACL's rule list confirmed the AWSManagedRulesCommonRuleSet and AWSManagedRulesSQLiRuleSet were present, but no RateBasedStatement rule was configured. |
| **Attack Scenario** | Without a rate-based rule, a single source (or small set of sources) can send a high volume of requests to sensitive endpoints such as login or password-reset before API Gateway's coarser, account-wide throttling engages, increasing |

| | |
|---|---|
| | feasibility of credential-stuffing or enumeration attacks. |
| **Impact** | Reduced protection against high-volume, single-source abusive traffic patterns targeting sensitive API endpoints. |
| **Business Impact** | Rate-based WAF rules are a standard, low-effort control for FinTech APIs handling authentication and financial transactions; their absence is a straightforward gap to close. |
| **Likelihood** | Low-Medium — does not require a code vulnerability, only sustained request volume from an attacker. |
| **Evidence Reference** | *EV-CLOUD-035 (fictional WAF Web ACL rule configuration export) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Add a rate-based WAF rule scoped to sensitive endpoints (login, password-reset, funds-transfer) with an appropriately tuned threshold and CAPTCHA/block action. |
| **Recommended AWS Controls** | AWS WAF rate-based rules, AWS WAF Bot Control for the login/authentication surface, API Gateway usage plans and per-client throttling. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will add the rate-based rule in the next WAF configuration update.* |
| **Owner / Priority / Target / Status** | **Web Platform Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming a rate-based rule is active and correctly throttles a fictional high-volume test request pattern against the login endpoint.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 17 · WAF & Edge Security | WAF, API Gateway | astravault-public-api-waf (fictional Web ACL attached to the public API Gateway stage) |

| **CLOUD-031 — ECR Repository Missing Image Lifecycle Policy (Unbounded Image Retention)** | **LOW** |
|---|---|
| **CVSS-Style Score: 2.6 (Low) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-1104 — Use of Unmaintained Third-Party Components** |
| **Description** | 9 of 11 fictional production ECR repositories have no image lifecycle policy configured, resulting in unbounded retention of every historical image tag, including old images built from since-patched vulnerable base images that remain pullable indefinitely. |
| **Technical Details** | TMG Security reviewed the lifecycle-policy configuration for all 11 fictional production ECR repositories and found only 2 (including astravault/analytics-worker, following an earlier unrelated cost-optimization effort) with a policy configured. |
| **Attack Scenario** | Not independently exploitable, but increases the population of pullable images — including old, more-vulnerable builds — that could be mistakenly redeployed by an automation error or a compromised deployment credential choosing an older tag. |
| **Impact** | Unbounded accumulation of historical images, including outdated and potentially more-vulnerable builds, remaining available for deployment indefinitely. |
| **Business Impact** | Primarily a hygiene and cost-governance finding; the security relevance is limited to reducing the chance of an old, vulnerable image being mistakenly redeployed. |
| **Likelihood** | N/A — hygiene finding, not independently exploitable. |
| **Evidence Reference** | *EV-CLOUD-036 (fictional ECR lifecycle-policy audit across 11 production repositories) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Apply a lifecycle policy to all production ECR repositories retaining a bounded number of recent tagged images (e.g., last 10) and expiring untagged images after 7 days. |
| **Recommended AWS Controls** | ECR lifecycle policies, ECR repository scanning on a recurring schedule for retained images, tagging immutability for production-promoted images. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will apply lifecycle policies across all repositories in the next change window.* |
| **Owner / Priority / Target / Status** | **DevOps / Platform Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming all 11 production ECR repositories now have an active lifecycle policy consistent with the revised retention standard.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 11 · ECR / Image Security | ECR | 9 of 11 fictional production ECR repositories with no lifecycle policy configured |

| CLOUD-032 — Secrets Manager Rotation Not Configured for Long-Lived Database Credentials | LOW |
|---|---|
| **CVSS-Style Score: 3.7 (Low) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-262 — Not Using Password Aging** |
| **Description** | The fictional astravault/prod/rds/financial-db-credentials secret, while correctly stored in Secrets Manager rather than in plaintext (contrast CLOUD-007), does not have automatic rotation configured, despite Secrets Manager's native RDS rotation-Lambda integration being readily available for this exact use case. |
| **Technical Details** | Configuration review confirmed the secret's RotationEnabled flag is false and no rotation Lambda is associated, with the fictional credential having been last changed at initial provisioning over a year prior to this assessment. |
| **Attack Scenario** | Not independently exploitable, but extends the useful window for any historical, currently-undetected leak of this credential, and is inconsistent with rotation practices AstraVault has correctly applied to other secrets in the same account. |
| **Impact** | Extended exposure window for the production financial database credential in the event of any undetected historical leak. |
| **Business Impact** | Low incremental risk given the credential is at least properly stored in Secrets Manager rather than exposed in plaintext, but automatic rotation is a readily available, low-effort improvement for this specific secret type. |
| **Likelihood** | N/A — hygiene finding. |
| **Evidence Reference** | *EV-CLOUD-037 (fictional Secrets Manager rotation configuration review) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Enable Secrets Manager's native automatic rotation for this secret using the standard RDS single-user rotation Lambda template, on a 30-day rotation schedule. |
| **Recommended AWS Controls** | Secrets Manager automatic rotation, RDS IAM database authentication as a longer-term alternative to password-based credentials entirely. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will enable rotation in the next maintenance window.* |
| **Owner / Priority / Target / Status** | **Data Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming automatic rotation is enabled and a scheduled rotation completes successfully without application disruption.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 13 · Secrets Management | Secrets Manager, RDS | astravault/prod/rds/financial-db-credentials (fictional Secrets Manager secret; rotation: disabled) |

| CLOUD-033 — SNS Topic Policy Permits Overly Broad Publish Access | LOW |
|---|---|
| **CVSS-Style Score: 3.2 (Low) — Illustrative CVSS v3.1-inspired score** | **CWE: CWE-284 — Improper Access Control** |
| **Description** | The fictional astravault-prod-transaction-notifications SNS topic, which fans out to a downstream fraud-monitoring Lambda subscriber, has a topic policy permitting sns:Publish from the entire fictional Production account principal rather than the specific application role that should be the sole publisher. |
| **Technical Details** | Review of the fictional topic policy confirmed the broad Principal grant; interview with the fictional application owner confirmed only one service should ever publish to this topic. |
| **Attack Scenario** | Any Production-account principal (not limited to the intended publisher) can publish messages to this topic, which could be used to inject spurious or misleading messages into the downstream fraud-monitoring pipeline, potentially masking real events or generating false positives. |
| **Impact** | Unnecessary publish access to a topic feeding a fraud-monitoring pipeline from a broader population than intended. |
| **Business Impact** | Low likelihood given it requires an already-compromised Production credential, but the downstream fraud-monitoring context gives this more relevance than a typical overly-broad topic policy. |
| **Likelihood** | Low — requires an already-compromised Production-account credential. |
| **Evidence Reference** | *EV-CLOUD-038 (fictional SNS topic policy export, application-owner interview notes) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Narrow the topic policy's Principal to the specific application role ARN that legitimately publishes to this topic. |
| **Recommended AWS Controls** | Scoped SNS topic policies, IAM Access Analyzer external/cross-principal access review, CloudTrail alerting on sns:Publish calls from unexpected principals. |
| **Management Response** | *AstraVault engineering (fictional) has accepted this finding and will narrow the topic policy in the next change window.* |

| | |
|---|---|
| **Owner / Priority / Target / Status** | **Platform Engineering Lead (Fictional Role)** \| 61-90 Days \| 12 December 2026 \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Retest by confirming the topic policy Principal is scoped to the single intended publisher role ARN.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 21 · Cloud Misconfiguration | SNS | astravault-prod-transaction-notifications (fictional SNS topic) |

# 46 INFORMATIONAL FINDINGS

The 9 Informational findings below are observations and governance recommendations with no identified exploit path, tracked for AstraVault's continuous cloud security improvement program rather than urgent remediation.

| **CLOUD-034 — Security Hub Standard Subscriptions Incomplete Across Accounts** | **INFORMATIONAL** |
|---|---|
| **CVSS-Style Score: N/A — Informational observation, not independently scored** | **CWE: N/A** |
| **Description** | Security Hub is enabled with the AWS Foundational Security Best Practices standard subscribed in the Production and Security accounts, but only partially configured in the Development account, which is not subscribed to any standard. |
| **Technical Details** | Configuration review across all three fictional accounts confirmed the subscription gap in Development. |
| **Attack Scenario** | N/A — observational finding. |
| **Impact** | Reduced automated configuration-posture visibility in the Development account specifically. |
| **Business Impact** | Low; Development holds less sensitive data directly, though CLOUD-005 and CLOUD-041 note its role as a weaker link relative to Production. |
| **Likelihood** | N/A |
| **Evidence Reference** | *EV-CLOUD-039 (fictional Security Hub standard-subscription review across three accounts) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Subscribe the Development account to at least the AWS Foundational Security Best Practices standard for consistency with Production and Security. |
| **Recommended AWS Controls** | Security Hub central configuration across the AWS Organization, consistent standard subscription policy. |
| **Management Response** | *AstraVault engineering (fictional) will align Development's Security Hub configuration with Production/Security in a future maintenance cycle.* |
| **Owner / Priority / Target / Status** | **Security Operations Lead (Fictional Role)** \| N/A — Informational \| Q1 2027 (fictional planning horizon) \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Confirm at next review that Development is subscribed to the same Security Hub standards as Production and Security.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 15 · Logging & Monitoring | Security Hub | AstraVault Development account (123456789014, fictional) |

| **CLOUD-035 — Inconsistent Resource Naming Convention Complicates Governance** | **INFORMATIONAL** |
|---|---|
| **CVSS-Style Score: N/A — Informational observation, not independently scored** | **CWE: N/A** |
| **Description** | Resource naming across the fictional environment mixes several conventions (astravault-prod-*, prod-astravault-*, and a small number of legacy nsp-* prefixed resources predating a company rebrand), making automated inventory and ownership mapping harder than a single consistent convention would allow. |
| **Technical Details** | Observed during the resource-inventory phase of this assessment while building the registers in Appendices B-I. |
| **Attack Scenario** | N/A — observational finding. |
| **Impact** | Minor operational friction for inventory, tagging, and ownership-mapping automation. |
| **Business Impact** | Low; a hygiene observation rather than a security control gap. |
| **Likelihood** | N/A |
| **Evidence Reference** | *EV-CLOUD-040 (fictional resource-inventory naming-pattern sample) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Adopt and enforce a single documented naming convention going forward, and rename legacy resources opportunistically during their next planned change. |
| **Recommended AWS Controls** | Documented naming-convention standard, Terraform module enforcement, periodic naming-compliance audit. |
| **Management Response** | *AstraVault engineering (fictional) acknowledges this observation and will incorporate it into ongoing platform hygiene efforts.* |

| | |
|---|---|
| **Owner / Priority / Target / Status** | **Cloud Platform Engineering Lead (Fictional Role)** \| N/A — Informational \| Ongoing (fictional) \| **Remediation Planned** |
| **Validation / Retest Guidance** | *N/A — tracked as an ongoing hygiene item rather than a discrete retest.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 21 · Cloud Misconfiguration | Organization-wide | Mixed naming patterns across EC2, S3, and Lambda resources (fictional) |

| **CLOUD-036 — Infrastructure-as-Code Drift Observed Between Deployed and Templated Resources** | **INFORMATIONAL** |
|---|---|
| **CVSS-Style Score: N/A — Informational observation, not independently scored** | **CWE: N/A** |
| **Description** | A fictional terraform plan dry-run against the current state showed 12 resources with configuration drift from their templated definition, most commonly manually-applied security-group rule changes made outside the IaC pipeline during past incident response. |
| **Technical Details** | Reviewed via the fictional Terraform state comparison provided by the AstraVault cloud team during the authenticated configuration review. |
| **Attack Scenario** | N/A — observational finding, though drift is how several of this report's other findings (e.g., stale security-group rules) likely originated and persisted undetected. |
| **Impact** | Reduces confidence that the IaC templates reflect actual deployed configuration, complicating both security review and change management. |
| **Business Impact** | Moderate long-term governance risk if unaddressed, as drift compounds over time and each instance represents an unreviewed manual change. |
| **Likelihood** | N/A |
| **Evidence Reference** | *EV-CLOUD-041 (fictional terraform plan drift summary) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Reconcile the 12 identified drifted resources back into Terraform state (either by updating the template to reflect an intentional change, or reverting the manual change); add a scheduled drift-detection job with alerting for future occurrences. |
| **Recommended AWS Controls** | Scheduled Terraform drift detection, change-management policy requiring all infrastructure changes to flow through IaC, break-glass process with mandatory reconciliation follow-up. |
| **Management Response** | *AstraVault engineering (fictional) acknowledges this observation and will reconcile the identified drift and implement scheduled detection.* |
| **Owner / Priority / Target / Status** | **Platform Engineering Lead (Fictional Role)** \| N/A — Informational \| Q1 2027 (fictional planning horizon) \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Confirm at next review that the 12 drifted resources are reconciled and scheduled drift detection is active.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 24 · CI/CD & Cloud Deployment Security | CloudFormation / Terraform (fictional IaC pipeline) | 12 of approximately 210 fictional Terraform-managed resources showing configuration drift |

| **CLOUD-037 — EventBridge Rule Lacks Dead-Letter Queue for Failed Invocations** | **INFORMATIONAL** |
|---|---|
| **CVSS-Style Score: N/A — Informational observation, not independently scored** | **CWE: N/A** |
| **Description** | The fictional astravault-prod-transaction-event-router EventBridge rule, which fans transaction events out to several downstream Lambda targets, has no configured dead-letter queue (DLQ), so failed target invocations are retried per the default policy and then silently dropped. |
| **Technical Details** | Reviewed via the fictional EventBridge rule target configuration during the serverless/API assessment. |
| **Attack Scenario** | N/A — observational finding; primarily a reliability rather than security concern, though silently dropped security-relevant events (e.g., a downstream fraud-check invocation) would be an availability-adjacent detection gap. |
| **Impact** | Failed downstream invocations, including potentially security-relevant ones, are not retained for investigation or replay. |
| **Business Impact** | Low-to-moderate; primarily an operational reliability gap with a secondary security-observability angle given one target is fraud-related. |
| **Likelihood** | N/A |

| | |
|---|---|
| **Evidence Reference** | *EV-CLOUD-042 (fictional EventBridge rule target configuration review) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Configure an SQS dead-letter queue on the EventBridge rule's targets with alerting on DLQ message arrival. |
| **Recommended AWS Controls** | EventBridge DLQ configuration, CloudWatch alarm on DLQ depth, scheduled DLQ review process. |
| **Management Response** | *AstraVault engineering (fictional) acknowledges this observation and will add DLQ configuration in a future sprint.* |
| **Owner / Priority / Target / Status** | **Serverless Engineering Lead (Fictional Role)** \| N/A — Informational \| Q1 2027 (fictional planning horizon) \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Confirm at next review that a DLQ is configured and alerting is active on message arrival.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 08 · Serverless / Lambda Security | EventBridge, SQS | astravault-prod-transaction-event-router (fictional EventBridge rule) |

| **CLOUD-038 — Systems Manager Patch Compliance Reporting Gaps on EC2 Fleet** | **INFORMATIONAL** |
|---|---|
| **CVSS-Style Score: N/A — Informational observation, not independently scored** | **CWE: N/A** |
| **Description** | 6 of 44 fictional production EC2 instances report a NoData patch-compliance status in Systems Manager, most commonly instances launched from a golden AMI variant that does not have the SSM Agent enabled by default. |
| **Technical Details** | Reviewed via the fictional Systems Manager Patch Compliance dashboard during the workload security assessment. |
| **Attack Scenario** | N/A — observational finding; these instances are not confirmed unpatched, only unreported, which limits AstraVault's ability to confirm their patch status either way. |
| **Impact** | Reduced visibility into the actual patch status of 6 fleet instances. |
| **Business Impact** | Low; primarily a visibility gap rather than a confirmed vulnerability, but it does prevent AstraVault from confirming these 6 instances meet the same patch baseline as the rest of the fleet. |
| **Likelihood** | N/A |
| **Evidence Reference** | *EV-CLOUD-043 (fictional SSM Patch Compliance dashboard export) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Ensure the SSM Agent is enabled and reporting on all production EC2 instances, including updating the golden AMI variant responsible for the gap. |
| **Recommended AWS Controls** | SSM Agent baked into all golden AMIs, Patch Manager maintenance windows, scheduled compliance-dashboard review. |
| **Management Response** | *AstraVault engineering (fictional) acknowledges this observation and will update the golden AMI variant to include the SSM Agent.* |
| **Owner / Priority / Target / Status** | **Platform Engineering Lead (Fictional Role)** \| N/A — Informational \| Q1 2027 (fictional planning horizon) \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Confirm at next review that all production instances report current patch-compliance status via Systems Manager.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 05 · EC2 & Workload Security | Systems Manager (Patch Manager) | 6 of 44 fictional production EC2 instances reporting as "NoData" in SSM Patch Compliance |

| **CLOUD-039 — API Gateway Missing Request Validation on Selected Public Endpoints** | **INFORMATIONAL** |
|---|---|
| **CVSS-Style Score: N/A — Informational observation, not independently scored** | **CWE: N/A** |
| **Description** | 3 of 27 fictional public API Gateway routes do not have a request-validation model attached, relying entirely on application-layer validation rather than rejecting malformed requests at the gateway. TMG Security did not identify an exploitable vulnerability resulting from this gap during testing, but notes it as a defense-in-depth observation. |
| **Technical Details** | Reviewed via the fictional API Gateway route configuration export during the API assessment. |
| **Attack Scenario** | N/A — observational finding; no exploitable condition was confirmed on the affected routes during testing. |
| **Impact** | Minor reduction in defense-in-depth for malformed-request handling on 3 routes. |
| **Business Impact** | Low; application-layer validation was confirmed present and functioning on all 3 routes during testing. |
| **Likelihood** | N/A |
| **Evidence Reference** | *EV-CLOUD-044 (fictional API Gateway route/model configuration review) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |

| | |
|---|---|
| **Recommendation** | Attach a request-validation model to the remaining 3 routes for consistency with the other 24 and as an additional defense-in-depth layer. |
| **Recommended AWS Controls** | API Gateway request-validation models, AWS WAF as a complementary edge-layer control. |
| **Management Response** | *AstraVault engineering (fictional) acknowledges this observation and will add validation models for consistency.* |
| **Owner / Priority / Target / Status** | **Web Platform Engineering Lead (Fictional Role)** \| N/A — Informational \| Q1 2027 (fictional planning horizon) \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Confirm at next review that all 27 public routes have a request-validation model attached.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 09 · API Gateway & Cloud APIs | API Gateway | 3 of 27 fictional public API Gateway routes without a request-validation model attached |

| **CLOUD-040 — CloudWatch Log Group Retention Set to Indefinite on Non-Critical Logs** | **INFORMATIONAL** |
|---|---|
| **CVSS-Style Score: N/A — Informational observation, not independently scored** | **CWE: N/A** |
| **Description** | 9 of 31 fictional CloudWatch Log Groups, mostly verbose application-debug logs, are configured with indefinite retention rather than a defined retention period aligned to AstraVault's fictional data-retention policy, increasing storage cost and slightly extending the window that any sensitive data accidentally logged would remain retrievable. |
| **Technical Details** | Reviewed via the fictional CloudWatch Logs retention-configuration audit across all 31 log groups. |
| **Attack Scenario** | N/A — observational finding; primarily a cost and data-minimization consideration rather than a security control gap. |
| **Impact** | Minor increase in cost and in the retention window for any sensitive data inadvertently written to these logs. |
| **Business Impact** | Low; a hygiene and data-minimization observation. |
| **Likelihood** | N/A |
| **Evidence Reference** | *EV-CLOUD-045 (fictional CloudWatch Logs retention configuration audit) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Set an explicit, policy-aligned retention period (e.g., 90 days for debug logs, longer for audit-relevant logs) on all 9 identified log groups. |
| **Recommended AWS Controls** | CloudWatch Logs retention policy, AWS Config rule cw-loggroup-retention-period-check, log-scrubbing for sensitive data in debug output. |
| **Management Response** | *AstraVault engineering (fictional) acknowledges this observation and will set explicit retention periods in the next change window.* |
| **Owner / Priority / Target / Status** | **Security Operations Lead (Fictional Role)** \| N/A — Informational \| Q1 2027 (fictional planning horizon) \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Confirm at next review that all 31 log groups have an explicit, policy-aligned retention period configured.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 15 · Logging & Monitoring | CloudWatch Logs | 9 of 31 fictional CloudWatch Log Groups with retention set to "Never Expire" |

| **CLOUD-041 — Development Account Lacks Equivalent Security Baseline to Production** | **INFORMATIONAL** |
|---|---|
| **CVSS-Style Score: N/A — Informational observation, not independently scored** | **CWE: N/A** |
| **Description** | The fictional Development account was found across several other findings in this report (CLOUD-005, CLOUD-027, CLOUD-034) to consistently carry a weaker security baseline than Production — broader default IAM permissions, no mandatory hardware-MFA policy, disabled VPC Flow Logs, and incomplete Security Hub subscription. This entry consolidates that pattern as a standalone observation. |
| **Technical Details** | Synthesized from the individual configuration gaps identified elsewhere in this assessment, all traced back to the same account. |
| **Attack Scenario** | N/A — consolidating observation; the individual exploitable elements are tracked under their own finding IDs (notably CLOUD-005). |
| **Impact** | The Development account functions as a consistently weaker link across multiple control domains simultaneously, rather than any single isolated gap. |

| | |
|---|---|
| **Business Impact** | Given CLOUD-005's finding that Production trusts Development for certain access, a pattern of weaker baseline controls in Development has outsized relevance to Production's own security posture. |
| **Likelihood** | N/A |
| **Evidence Reference** | *EV-CLOUD-046 (cross-reference summary of Development-account baseline gaps identified in this assessment) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Apply AstraVault's Production security baseline (MFA policy, IAM permission boundaries, VPC Flow Logs, Security Hub subscription, tagging enforcement) to the Development account as a standing policy, not a one-off catch-up exercise, using AWS Organizations SCPs and Control Tower-style guardrails to keep the baselines in sync going forward. |
| **Recommended AWS Controls** | AWS Control Tower or equivalent account-baseline automation, Organization-wide SCPs, consistent Security Hub/Config/GuardDuty enablement across all accounts. |
| **Management Response** | *AstraVault engineering (fictional) acknowledges this pattern and will pursue account-baseline standardization as a program-level initiative.* |
| **Owner / Priority / Target / Status** | **Cloud Security Engineering Lead (Fictional Role)** \| N/A — Informational \| Q1 2027 (fictional planning horizon) \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Confirm at next review that the Development account's baseline configuration matches Production across the specific controls cited above.* |

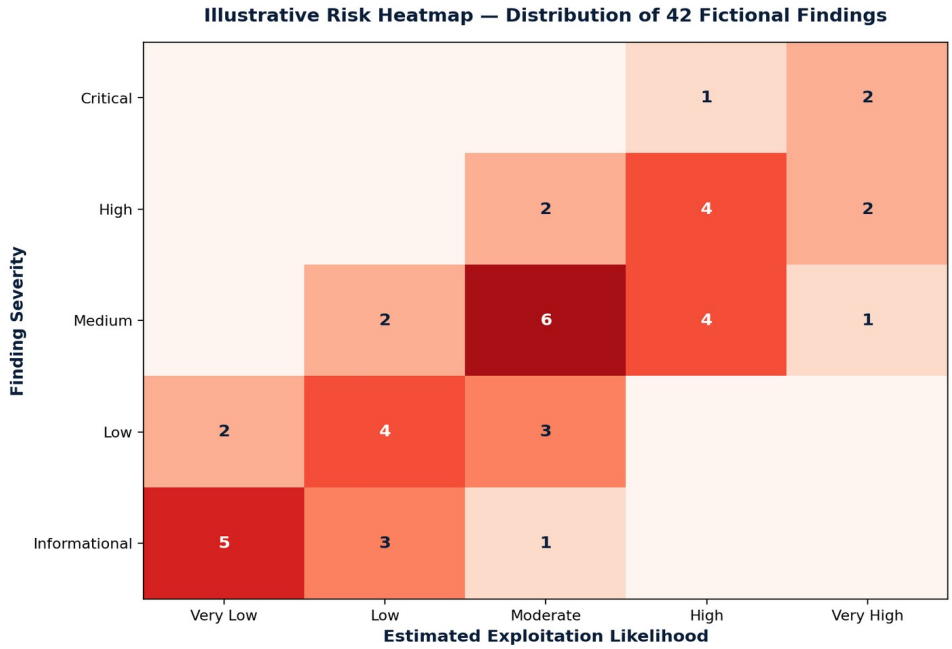
| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 21 · Cloud Misconfiguration | Organization-wide (Development account, 123456789014, fictional) | AstraVault Development account baseline (fictional) |

| **CLOUD-042 — Documentation Gap in Incident Response Runbook for Cloud-Specific Scenarios** | **INFORMATIONAL** |
|---|---|
| **CVSS-Style Score: N/A — Informational observation, not independently scored** | **CWE: N/A** |
| **Description** | AstraVault's fictional Incident Response runbook, reviewed as part of this engagement's documentation-review perspective, covers general application-security incident scenarios well but does not include cloud-specific playbooks for scenarios such as IAM credential compromise, cross-account role abuse, or S3 data-exposure response — the exact categories most of this report's Critical and High findings fall into. |
| **Technical Details** | Reviewed via a fictional document walkthrough with the AstraVault security team during the assessment kickoff and closing debrief. |
| **Attack Scenario** | N/A — process/documentation observation, not a technical finding. |
| **Impact** | In a real incident matching this report's finding categories, responders would have less specific guidance than for the application-security scenarios the runbook does cover well. |
| **Business Impact** | Moderate; well-tested, cloud-specific incident playbooks materially reduce response time and error rate during an actual incident, particularly for less-common scenarios like cross-account role abuse. |
| **Likelihood** | N/A |
| **Evidence Reference** | *EV-CLOUD-047 (fictional runbook review notes) [ILLUSTRATIVE / FICTIONAL SAMPLE EVIDENCE]* |
| **Recommendation** | Extend the Incident Response runbook with dedicated playbooks for IAM credential compromise, cross-account role abuse, S3 data exposure, and compromised-workload containment, using this report's own findings and attack chains (Section 40) as a starting reference set; conduct a fictional tabletop exercise against at least one cloud-specific scenario within the next two quarters. |
| **Recommended AWS Controls** | Cloud-specific incident-response playbooks, scheduled tabletop exercises, AWS Incident Detection and Response engagement or equivalent for critical accounts. |
| **Management Response** | *AstraVault engineering (fictional) acknowledges this observation and will extend the runbook and schedule a tabletop exercise in the coming quarter.* |
| **Owner / Priority / Target / Status** | **Security Operations Lead (Fictional Role)** \| N/A — Informational \| Q1 2027 (fictional planning horizon) \| **Remediation Planned** |
| **Validation / Retest Guidance** | *Confirm at next review that the runbook has been extended with the identified cloud-specific playbooks and that a tabletop exercise has been conducted.* |

| CLOUD SECURITY DOMAIN | AFFECTED CLOUD SERVICE | AFFECTED FICTIONAL RESOURCE |
|---|---|---|
| 25 · Cloud Persistence & Detection | Organization-wide (process/documentation) | AstraVault fictional Incident Response runbook (v3, 2025) |

# 47 RISK HEATMAP

The heatmap below plots all 42 fictional findings by severity against estimated exploitation likelihood, providing AstraVault's leadership with a single-glance view of where technical risk concentrates. Findings cluster toward the high-severity / high-likelihood quadrant in the IAM, cross-account trust, and metadata-exposure domains — consistent with the Attack Path Analysis in Section 40.

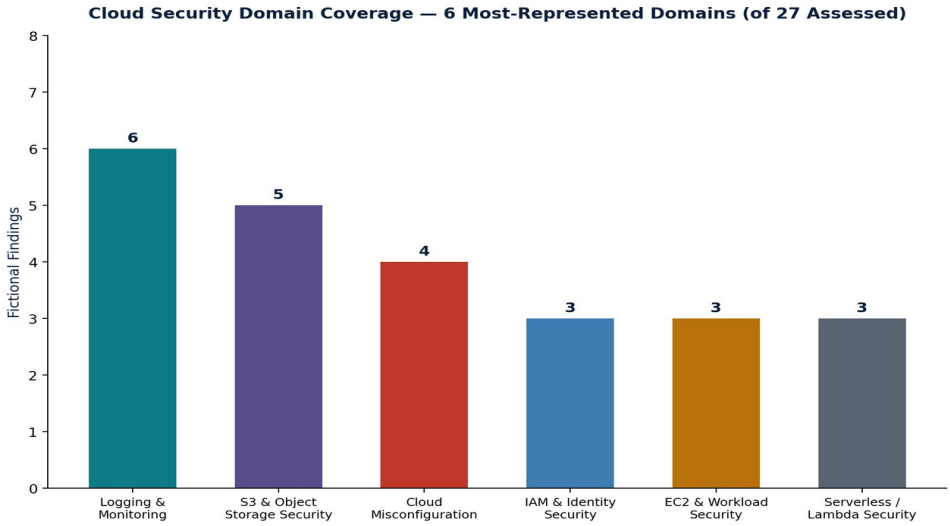



*ILLUSTRATIVE / SYNTHETIC EVIDENCE — cell values sum to 42, matching the canonical finding count.*

*Illustrative risk heatmap — severity versus likelihood, all 42 fictional findings — ILLUSTRATIVE / SYNTHETIC EVIDENCE.*

## Domain Concentration

Findings are not evenly distributed across AstraVault's 27 assessed cloud security domains. Logging & Monitoring, S3 & Object Storage Security, and Cloud Misconfiguration each contributed the largest number of findings, with IAM & Identity Security, EC2 & Workload Security, and Serverless/Lambda Security close behind — while domains such as CloudTrail, WAF, and CloudFront/CDN Security each contributed a single, more contained finding. This concentration pattern is itself useful for prioritization: the Critical and most severe High findings still concentrate in IAM and cross-account trust (Section 40's attack chains all pass through CLOUD-001 or CLOUD-004), so strengthening detection coverage and storage governance reduces the largest volume of findings, while strengthening IAM and cross-account trust controls breaks the highest-impact attack paths.




*Illustrative fictional finding distribution across the 6 most-represented cloud security domains — ILLUSTRATIVE / SYNTHETIC EVIDENCE.*

# 48 BUSINESS IMPACT

This section translates the technical findings in Sections 42-46 into fictional business terms for AstraVault's executive and board-level audiences, framed around the specific risks a FinTech / enterprise SaaS platform faces from cloud misconfiguration.

## Regulatory & Compliance Exposure

A real-world equivalent of CLOUD-002 (public exposure of customer financial statements) would likely trigger mandatory breach-notification obligations under applicable U.S. state data-breach laws and could implicate contractual obligations to AstraVault's fictional banking and payment partners. Several Medium and Low findings (missing encryption enforcement, weak backup immutability, incomplete logging) compound this exposure by reducing AstraVault's ability to scope and respond to a real incident quickly and completely.

## Financial Impact

- Direct incident-response and forensic investigation costs in a real-world equivalent breach scenario, particularly given the CloudTrail data-event and CloudWatch alerting gaps that would slow root-cause determination.
- Potential regulatory fines and mandatory customer notification costs associated with exposure of financial statement data.
- Reputational damage and customer-trust erosion for a FinTech platform whose core value proposition depends on data security.
- Business disruption costs if a real-world equivalent of Attack Chain 3 (container compromise → cloud resource modification) resulted in service disruption to core financial-workflow features.

### Third-Party & Partner Risk

CLOUD-007's exposure of a fictional payment-aggregator API key illustrates how a cloud misconfiguration inside AstraVault's own environment can create risk for AstraVault's external partners and integrations, potentially triggering contractual notification obligations to the third party independent of any direct AstraVault customer impact.

## Operational Impact

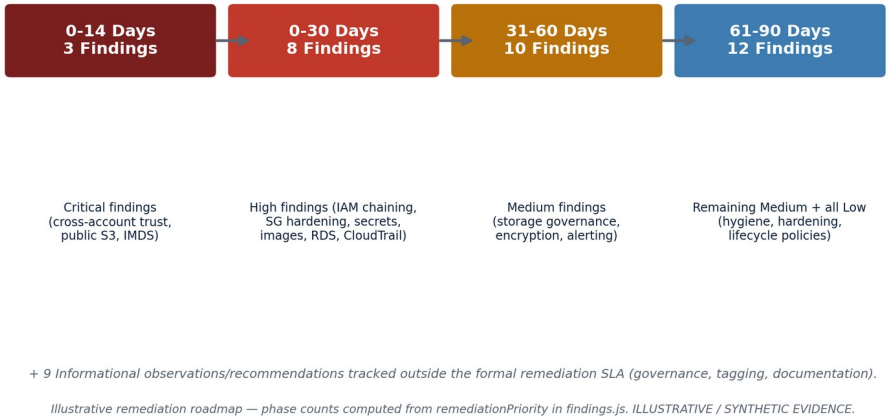
Beyond a specific incident scenario, several findings (CLOUD-004, CLOUD-009, the shared execution-role pattern) represent structural risk that would recur with each new feature or workload unless the underlying IAM patterns are corrected — meaning the business impact of these findings compounds over time rather than being a one-time exposure.

## Impact by Critical/High Finding

| Finding | Primary Business Impact |
|---|---|
| CLOUD-001 | Loss of the Security-account boundary; ability to disable detection before further compromise. |
| CLOUD-002 | Regulatory notification exposure; reputational damage from unauthenticated financial-data disclosure. |
| CLOUD-003 | Direct conversion of an application bug into cloud-account credential theft. |
| CLOUD-004 | Widens the realistic population of attackers who can reach CLOUD-001's impact. |
| CLOUD-005 | Production secret exposure gated only by the weaker Development-account baseline. |
| CLOUD-006 | Increased brute-force/credential-stuffing exposure on production-adjacent infrastructure. |
| CLOUD-007 | Third-party payment-aggregator API key exposure; broader-than-intended credential visibility. |
| CLOUD-008 | Known, high-severity remotely exploitable component in production. |
| CLOUD-009 | Single function compromise grants the union of 14 functions' permissions. |
| CLOUD-010 | Unnecessary internet exposure of a production-adjacent analytics database. |
| CLOUD-011 | Materially slower incident scoping for any breach of the environment's most sensitive data. |

# 49 REMEDIATION ROADMAP

The fictional remediation roadmap below phases all 42 findings into four windows aligned to severity and technical dependency, so AstraVault's engineering teams can sequence work efficiently — addressing the findings that unlock the Section 40 attack chains first.




*Illustrative fictional remediation roadmap by phase — ILLUSTRATIVE / SYNTHETIC EVIDENCE.*

## Phase Detail

*Phase counts below are generated directly from each finding's remediationPriority value in this report's canonical finding data, so this table and the Section 50 Management Action Plan can never drift apart.*

| Phase | Findings | Focus |
|---|---|---|
| Immediate — 0-14 Days (Critical) | 3 | Cross-account IAM trust scoping, public S3 bucket remediation, IMDSv2 enforcement — breaks Attack Chains 1 and 4 at their strongest links. |
| 0-30 Days (High) | 8 | IAM policy chaining, cross-account trust narrowing, security-group hardening, secret migration, container image patching, execution-role scoping, RDS network isolation, CloudTrail data-event coverage. |
| 31-60 Days (Medium, Part 1) | 10 | S3 governance, encryption/versioning enforcement, alerting expansion, egress scoping, KMS policy scoping. |
| 61-90 Days (Medium tail + Low) | 12 | Remaining Medium items plus all Low findings: IAM hygiene, backup hardening, container capability minimization, logging completeness, WAF rate-limiting, ECR lifecycle policies, secret rotation, SNS policy scoping. |
| Continuous (Informational) | 9 | Governance, tagging, IaC drift reconciliation, documentation, and process improvements tracked outside the formal remediation SLA. |

# 50 MANAGEMENT ACTION PLAN

The table below consolidates AstraVault's fictional management response, ownership, and target date for every finding in this report, providing a single reference for tracking remediation progress.

| Finding | Severity | Title | Owner | Target Date | Status |
|---|---|---|---|---|---|
| CLOUD-001 | CRITICAL | Overly Permissive IAM Role Enables Cross-Account Administrative Privilege Escalation | Cloud Security Engineering Lead (Fictional Role) | 14 October 2026 | Remediation In Progress |
| CLOUD-002 | CRITICAL | Publicly Accessible S3 Bucket Exposes Sensitive Financial Data | Cloud Platform Engineering Lead (Fictional Role) | 08 September 2026 (emergency fix) / 14 October 2026 (full remediation incl. encryption enforcement) | Remediation In Progress |
| CLOUD-003 | CRITICAL | EC2 Workload Metadata Exposure Enables Credential Theft and Privilege Escalation | Platform Engineering Lead (Fictional Role) | 14 October 2026 | Remediation In Progress |
| CLOUD-004 | HIGH | Privilege Escalation via IAM Policy Chaining (iam:PassRole + iam:CreatePolicyVersion) | IAM & Identity Lead (Fictional Role) | 10 October 2026 | Remediation Planned |

| Finding | Severity | Title | Owner | Target Date | Status |
|---|---|---|---|---|---|
| CLOUD-005 | HIGH | Excessive Trust Relationship Between Production and Development AWS Accounts | Cloud Security Engineering Lead (Fictional Role) | 17 October 2026 | Remediation Planned |
| CLOUD-006 | HIGH | Security Group Permits Unrestricted Administrative Access to Management Ports | Network Engineering Lead (Fictional Role) | 17 October 2026 | Remediation Planned |
| CLOUD-007 | HIGH | Secrets Exposed Through Workload Environment Configuration | Platform Engineering Lead (Fictional Role) | 17 October 2026 | Remediation In Progress |
| CLOUD-008 | HIGH | Vulnerable Container Image With Exploitable Package in Production ECR Repository | DevOps / Platform Engineering Lead (Fictional Role) | 17 October 2026 | Remediation In Progress |
| CLOUD-009 | HIGH | Lambda Execution Role Grants Excessive IAM Permissions Beyond Function Requirements | Serverless Engineering Lead (Fictional Role) | 24 October 2026 | Remediation Planned |
| CLOUD-010 | HIGH | RDS Database Instance Publicly Reachable From the Internet | Data Engineering Lead (Fictional Role) | 24 October 2026 | Remediation Planned |
| CLOUD-011 | HIGH | CloudTrail Logging Coverage Gap on Security-Sensitive API Activity | Security Operations Lead (Fictional Role) | 24 October 2026 | Remediation Planned |
| CLOUD-012 | MEDIUM | Overly Broad S3 Bucket Resource Policy Grants Unnecessary Cross-Account Access | Cloud Platform Engineering Lead (Fictional Role) | 14 November 2026 | Remediation Planned |
| CLOUD-013 | MEDIUM | Missing Server-Side Encryption Enforcement on S3 Buckets Storing Sensitive Data | Cloud Platform Engineering Lead (Fictional Role) | 14 November 2026 | Remediation Planned |
| CLOUD-014 | MEDIUM | S3 Bucket Versioning Disabled on Buckets Storing Financial Records | Cloud Platform Engineering Lead (Fictional Role) | 14 November 2026 | Remediation Planned |
| CLOUD-015 | MEDIUM | Insufficient CloudWatch Alerting on High-Risk IAM and Security Events | Security Operations Lead (Fictional Role) | 14 November 2026 | Remediation Planned |
| CLOUD-016 | MEDIUM | Permissive Security Group Egress Rules Allow Unrestricted Outbound Access | Network Engineering Lead (Fictional Role) | 21 November 2026 | Remediation Planned |
| CLOUD-017 | MEDIUM | Stale IAM Users With Long-Unused Console and API Access | IAM & Identity Lead (Fictional Role) | 21 November 2026 | Remediation In Progress |
| CLOUD-018 | MEDIUM | Unused and Unrotated IAM Access Keys Exceeding Recommended Age | IAM & Identity Lead (Fictional Role) | 21 November 2026 | Remediation Planned |
| CLOUD-019 | MEDIUM | Weak AWS Backup Configuration for Production Database Resources | Data Engineering Lead (Fictional Role) | 21 November 2026 | Remediation Planned |
| CLOUD-020 | MEDIUM | Lambda Function Configured With Excessive Timeout and Memory Beyond Operational Need | Serverless Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-021 | MEDIUM | Missing Workload Integrity Controls on EC2 AMI Build Pipeline | Platform Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-022 | MEDIUM | Insufficient Security-Header Controls at CloudFront Edge | Web Platform Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-023 | MEDIUM | KMS Key Policy Grants Overly Broad Decrypt Permissions | Cloud Security Engineering Lead (Fictional Role) | 21 November 2026 | Remediation Planned |
| CLOUD-024 | MEDIUM | ECS Task Definition Runs Containers With Elevated Linux Capabilities | Platform Engineering Lead (Fictional Role) | 21 November 2026 | Remediation Planned |
| CLOUD-025 | LOW | Incomplete Resource Tagging Undermines Cost and Security Governance | Cloud Platform Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-026 | LOW | S3 Access Logging Not Enabled on Selected Buckets | Cloud Platform Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-027 | LOW | VPC Flow Logs Not Enabled on Selected VPCs | Network Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-028 | LOW | Route 53 DNS Zone Lacks Query Logging | Network Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-029 | LOW | GuardDuty Findings Not Routed to a Centralized Alerting Channel | Security Operations Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-030 | LOW | WAF Rule Set Missing Rate-Based Rule on Public API Gateway | Web Platform Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-031 | LOW | ECR Repository Missing Image Lifecycle Policy (Unbounded Image Retention) | DevOps / Platform Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-032 | LOW | Secrets Manager Rotation Not Configured for Long-Lived Database Credentials | Data Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-033 | LOW | SNS Topic Policy Permits Overly Broad Publish Access | Platform Engineering Lead (Fictional Role) | 12 December 2026 | Remediation Planned |
| CLOUD-034 | INFORMATIONAL | Security Hub Standard Subscriptions Incomplete Across Accounts | Security Operations Lead (Fictional Role) | Q1 2027 (fictional planning horizon) | Remediation Planned |
| CLOUD-035 | INFORMATIONAL | Inconsistent Resource Naming Convention Complicates Governance | Cloud Platform Engineering Lead (Fictional Role) | Ongoing (fictional) | Remediation Planned |
| CLOUD-036 | INFORMATIONAL | Infrastructure-as-Code Drift Observed Between Deployed and Templated Resources | Platform Engineering Lead (Fictional Role) | Q1 2027 (fictional planning horizon) | Remediation Planned |
| CLOUD-037 | INFORMATIONAL | EventBridge Rule Lacks Dead-Letter Queue for Failed Invocations | Serverless Engineering Lead (Fictional Role) | Q1 2027 (fictional planning horizon) | Remediation Planned |
| CLOUD-038 | INFORMATIONAL | Systems Manager Patch Compliance Reporting Gaps on EC2 Fleet | Platform Engineering Lead (Fictional Role) | Q1 2027 (fictional planning horizon) | Remediation Planned |
| CLOUD-039 | INFORMATIONAL | API Gateway Missing Request Validation on Selected Public Endpoints | Web Platform Engineering Lead (Fictional Role) | Q1 2027 (fictional planning horizon) | Remediation Planned |
| CLOUD-040 | INFORMATIONAL | CloudWatch Log Group Retention Set to Indefinite on Non-Critical Logs | Security Operations Lead (Fictional Role) | Q1 2027 (fictional planning horizon) | Remediation Planned |
| CLOUD-041 | INFORMATIONAL | Development Account Lacks Equivalent Security Baseline to Production | Cloud Security Engineering Lead (Fictional Role) | Q1 2027 (fictional planning horizon) | Remediation Planned |
| CLOUD-042 | INFORMATIONAL | Documentation Gap in Incident Response Runbook for Cloud-Specific Scenarios | Security Operations Lead (Fictional Role) | Q1 2027 (fictional planning horizon) | Remediation Planned |

# 51 RETEST METHODOLOGY

TMG Security recommends a formal retest of this fictional environment following remediation, structured as follows.

## Retest Scope

- Every Critical and High finding retested individually against the specific validation/retest guidance documented on its finding card in Sections 42-43.
- A sample of Medium findings retested, prioritized by domain concentration (IAM, S3, cross-account trust) per Section 47.
- Re-validation of all four composite attack chains in Section 40 to confirm the chain is broken at every previously-exploitable step, not merely at the first step.
- A fresh, lightweight external attack-surface scan to confirm no new public exposure has been introduced since the original assessment.

## Retest Timing

TMG Security recommends scheduling the retest 30-45 days after AstraVault's fictional Phase 1 (Critical) and Phase 2 (High) remediation work is reported complete, allowing time for the fixes to reach production and for CloudTrail/GuardDuty telemetry to confirm no residual exposure.

## Retest Deliverable

A retest letter or addendum documenting, for each retested finding, the original finding, the remediation applied, the retest method used, and the confirmed status (Remediated, Partially Remediated, or Not Remediated), consistent with the format TMG Security uses across its full report family.

# 52 POSITIVE SECURITY CONTROLS

Consistent with TMG Security's standard reporting practice, this section documents fictional controls confirmed operating effectively during this assessment — an important counterweight to the findings sections, since a cloud security report that only lists problems gives an incomplete picture of AstraVault's actual posture.

| Control Area | Observation |
|---|---|
| Primary Database Network Isolation | The primary transactional RDS Aurora cluster is correctly placed in a private database subnet with security-group access scoped to the application tier only. |
| CloudTrail Core Configuration | The organization trail is correctly multi-region, covers all three fictional accounts, and enforces log-file validation with a tamper-resistant delivery-bucket policy. |
| S3 Baseline (Majority of Buckets) | 21 of 22 fictional production buckets correctly enforce Block Public Access, and 16 of 22 correctly enforce default SSE-KMS encryption. |
| OIDC-Federated CI/CD Deployment | The deployment pipeline uses OIDC/workload-identity federation rather than long-lived static credentials, and no hard-coded secrets were found in build configuration. |
| GuardDuty & Security Hub Enablement | Both services are enabled organization-wide with centralized delegated-administrator visibility from the Security account, even where finding-routing maturity has room to improve. |
| Network Segmentation Model | The four-tier subnet model (public / private-application / private-database / management) is correctly implemented for the vast majority of Production resources, and Transit Gateway routing correctly isolates Development from Production at the network layer. |
| IAM Boundary Enforcement (Partial) | Several documented IAM privilege-escalation patterns tested in Section 17 (7 of 10) were correctly blocked by existing permission boundaries. |
| EBS & RDS Storage Encryption | Encryption at rest was confirmed enabled fleet-wide for EBS volumes and on all 5 reviewed database resources. |
| Break-Glass Administrative Access | The fictional break-glass administrative role is correctly scoped to a named principal with hardware-MFA and approval logging — a strong pattern AstraVault should extend to its other cross-account roles. |
| API Authorization Correctness | Broken object-level authorization testing against account-data and funds-transfer endpoints found no exploitable authorization bypass. |

These positive observations do not offset the findings elsewhere in this report; they are provided for a complete and balanced picture of AstraVault's fictional security posture.

# 53 TESTING LIMITATIONS

As with any time-boxed security assessment, this fictional engagement is subject to inherent limitations that AstraVault's leadership should understand when interpreting this report.

- This assessment reflects AstraVault's fictional AWS environment configuration as observed during the 07-25 September 2026 assessment window; any configuration change made after that window is not reflected in this report.
- Testing was performed from the four assessment perspectives defined in Section 10; TMG Security did not have unrestricted AWS Management Console access and cannot claim complete visibility into every possible misconfiguration an administrator-level review might surface.
- No destructive or denial-of-service testing was performed; some classes of availability-impacting weakness may not have been fully validated.
- Source-code-level application security testing was out of scope for this cloud-infrastructure-focused engagement; application-layer vulnerabilities not visible at the cloud-configuration layer may exist and are better addressed by TMG Security's companion Web Application and API assessment services.

- Third-party vendor infrastructure (e.g., the fictional payment aggregator) was not tested directly; only AstraVault-side integration configuration was reviewed.
- This report represents a point-in-time assessment; cloud environments change continuously, and TMG Security recommends recurring assessments (at minimum annually, or after significant architecture changes) to maintain an accurate risk picture.
- As with all fictional content in this sample report, these limitations are illustrative and included to demonstrate the transparency TMG Security applies to its real client deliverables.

# 54 CONCLUSION

This fictional cloud penetration test of AstraVault Technologies, Inc.'s AWS environment identified 42 illustrative findings (3 Critical, 8 High, 13 Medium, 9 Low, 9 Informational) across 138 executed test cases spanning IAM, storage, network, workload, data, logging, and deployment-pipeline security. The three Critical findings, and the composite attack chains they enable, represent the most urgent fictional risk to AstraVault's environment and should be remediated first, using the roadmap in Section 49.

Identity and access management — specifically over-permissive cross-account trust and documented IAM privilege-escalation patterns — emerged as this assessment's dominant risk theme, underlying both the most severe individual finding (CLOUD-001) and the majority of this report's composite attack chains. AstraVault's cloud team has also demonstrated meaningful security maturity in several areas: correct network segmentation for its primary database, modern OIDC-based CI/CD credential handling, and organization-wide GuardDuty/Security Hub enablement, among the positive controls detailed in Section 52.

TMG Security thanks AstraVault Technologies, Inc. (fictional) for the opportunity to conduct this illustrative assessment, and recommends proceeding with the remediation roadmap in Section 49 followed by the retest engagement described in Section 51 to validate closure of the findings in this report.

# APPENDIX A — CLOUD TEST CASE REGISTER

*138 fictional test cases executed during this illustrative assessment*

The register below documents all 138 fictional test cases executed during this illustrative assessment, spanning the 25 testable cloud-security categories presented across Sections 16-40 of this report, with the testing technique/reference used, a result, and a cross-reference to any corresponding finding. Every "Fail" or "Exception Identified" result maps to a finding documented in Sections 42-46.

| Test Case ID | Category | Test Description / Objective | Technique / Reference | Result | Finding |
|---|---|---|---|---|---|
| CLOUD-T001 | IAM Assessment | Enumerate all IAM users, roles, and groups and map to fictional business owners | aws iam list-users / list-roles / list-groups | Observation | CLOUD-035 |
| CLOUD-T002 | IAM Assessment | Review IAM credential report for stale (180+ day inactive) console/API access | iam generate-credential-report | Fail | CLOUD-017 |
| CLOUD-T003 | IAM Assessment | Review IAM credential report for access keys exceeding 365-day rotation age | iam generate-credential-report | Fail | CLOUD-018 |
| CLOUD-T004 | IAM Assessment | Verify root account has no active access keys and hardware MFA is enabled | iam get-account-summary | Pass | — |
| CLOUD-T005 | IAM Assessment | Review IAM password policy for minimum complexity and age requirements | iam get-account-password-policy | Pass | — |
| CLOUD-T006 | IAM Assessment | Identify IAM users with directly-attached policies rather than group membership | iam list-attached-user-policies | Observation | — |
| CLOUD-T007 | IAM Assessment | Review permission boundaries applied to human vs. service identities | iam list-policies --scope Local | Observation | — |
| CLOUD-T008 | IAM Assessment | Test IAM Identity Center / federated access configuration for session duration and MFA | IAM Identity Center console review | Pass | — |
| CLOUD-T009 | IAM Assessment | Review service-linked role and instance-profile inventory for unused entries | iam list-instance-profiles | Pass | — |
| CLOUD-T010 | IAM Assessment | Verify MFA enforcement policy for all human IAM principals with console access | iam list-virtual-mfa-devices | Observation | CLOUD-005 |
| CLOUD-T011 | Privilege Escalation | Build IAM privilege-escalation matrix and test each documented escalation pattern against current policies | iam simulate-principal-policy | Fail | CLOUD-004 |
| CLOUD-T012 | Privilege Escalation | Test iam:PassRole + service:CreateFunction/CreateInstance combinations across all groups | iam simulate-principal-policy | Fail | CLOUD-004 |
| CLOUD-T013 | Privilege Escalation | Test iam:CreatePolicyVersion / SetDefaultPolicyVersion self-escalation paths | iam simulate-principal-policy | Fail | CLOUD-004 |
| CLOUD-T014 | Privilege Escalation | Test iam:AttachUserPolicy / AttachGroupPolicy self-escalation paths for low-privilege users | iam simulate-principal-policy | Pass | — |
| CLOUD-T015 | Privilege Escalation | Test cross-service confused-deputy escalation via Lambda/CloudFormation service roles | Manual policy review + simulation | Pass | — |
| CLOUD-T016 | Privilege Escalation | Verify permission boundaries are consistently applied to prevent escalation beyond intended scope | iam get-policy / boundary review | Fail | CLOUD-001 |
| CLOUD-T017 | S3 Assessment | Enumerate all S3 buckets and test Block Public Access configuration at bucket and account level | s3api get-public-access-block | Fail | CLOUD-002 |
| CLOUD-T018 | S3 Assessment | Test unauthenticated anonymous GetObject/ListBucket against all discovered bucket names | curl / aws s3 --no-sign-request | Fail | CLOUD-002 |
| CLOUD-T019 | S3 Assessment | Review bucket policies for wildcard Principal or overly broad cross-account grants | s3api get-bucket-policy | Fail | CLOUD-012 |
| CLOUD-T020 | S3 Assessment | Review default encryption (SSE-KMS) configuration on all production buckets | s3api get-bucket-encryption | Fail | CLOUD-013 |
| CLOUD-T021 | S3 Assessment | Review S3 Versioning configuration on buckets storing financial or regulated records | s3api get-bucket-versioning | Fail | CLOUD-014 |
| CLOUD-T022 | S3 Assessment | Review bucket ACLs and Object Ownership settings for legacy ACL-based grants | s3api get-bucket-acl | Pass | — |
| CLOUD-T023 | S3 Assessment | Test presigned URL generation and expiry enforcement on application upload/download flows | Application-level presigned URL review | Pass | — |
| CLOUD-T024 | S3 Assessment | Review S3 Access Point configuration for any cross-account or public access path | s3control list-access-points | Pass | — |
| CLOUD-T025 | S3 Assessment | Review S3 Server Access Logging enablement across all production buckets | s3api get-bucket-logging | Fail | CLOUD-026 |
| CLOUD-T026 | S3 Assessment | Review S3 lifecycle and backup/version-retention configuration for compliance-relevant buckets | s3api get-bucket-lifecycle-configuration | Observation | CLOUD-019 |
| CLOUD-T027 | EC2 Assessment | Enumerate EC2 instances, launch templates, and Auto Scaling groups for IMDS configuration | ec2 describe-instances / describe-launch-templates | Fail | CLOUD-003 |
| CLOUD-T028 | EC2 Assessment | Test SSRF-reachable application features for instance-metadata credential theft | Manual SSRF probe against 169.254.169.254 | Fail | CLOUD-003 |
| CLOUD-T029 | EC2 Assessment | Review instance-profile role scope attached to each EC2 workload tier | iam get-instance-profile | Fail | CLOUD-001 |
| CLOUD-T030 | EC2 Assessment | Review golden AMI build pipeline for image signing and SBOM generation | AMI pipeline configuration review | Fail | CLOUD-021 |
| CLOUD-T031 | EC2 Assessment | Review Systems Manager Patch Manager compliance status across the production fleet | ssm describe-instance-patch-states | Observation | CLOUD-038 |
| CLOUD-T032 | EC2 Assessment | Test for unencrypted EBS volumes attached to production instances | ec2 describe-volumes | Pass | — |
| CLOUD-T033 | VPC Assessment | Map VPC, subnet, and route-table topology across all three fictional accounts | ec2 describe-vpcs / describe-route-tables | Observation | — |
| CLOUD-T034 | VPC Assessment | Review Internet Gateway and NAT Gateway attachment for unintended public routing | ec2 describe-internet-gateways | Pass | — |
| CLOUD-T035 | VPC Assessment | Review VPC Endpoint configuration for S3/Secrets Manager/KMS private connectivity | ec2 describe-vpc-endpoints | Observation | CLOUD-016 |
| CLOUD-T036 | VPC Assessment | Review VPC peering and Transit Gateway attachments for unintended cross-environment routes | ec2 describe-vpc-peering-connections | Pass | — |
| CLOUD-T037 | VPC Assessment | Review VPC Flow Logs enablement across all VPCs in all three accounts | ec2 describe-flow-logs | Fail | CLOUD-027 |
| CLOUD-T038 | Network Segmentation | Verify public/private/management subnet tiering matches the documented architecture | Architecture-vs-deployed comparison | Pass | — |
| CLOUD-T039 | Network Segmentation | Test network-layer reachability from public subnet to private database subnet | Internal network scan (authorized perspective) | Fail | CLOUD-010 |
| CLOUD-T040 | Network Segmentation | Review Network ACL rules as a defense-in-depth layer behind security groups | ec2 describe-network-acls | Observation | — |

| Test Case ID | Category | Test Description / Objective | Technique / Reference | Result | Finding |
|---|---|---|---|---|---|
| CLOUD-T041 | Network Segmentation | Test management/security subnet isolation from application workload subnets | Internal network scan (authorized perspective) | Pass | — |
| CLOUD-T042 | Network Segmentation | Review Transit Gateway route-table segmentation between Production and Development | ec2 describe-transit-gateway-route-tables | Pass | — |
| CLOUD-T043 | Security Groups | Enumerate all security groups and flag inbound rules from 0.0.0.0/0 on administrative ports | ec2 describe-security-groups | Fail | CLOUD-006 |
| CLOUD-T044 | Security Groups | Test external reachability of flagged administrative ports from the internet | nmap / external port scan | Fail | CLOUD-006 |
| CLOUD-T045 | Security Groups | Review egress rule scope across all production security groups | ec2 describe-security-groups | Fail | CLOUD-016 |
| CLOUD-T046 | Security Groups | Review security-group-to-security-group referencing for tiered application architectures | ec2 describe-security-groups | Pass | — |
| CLOUD-T047 | Security Groups | Test for orphaned/unused security groups retained after resource decommission | ec2 describe-security-groups (unattached) | Observation | — |
| CLOUD-T048 | Security Groups | Review RDS/database security-group source scoping for direct internet exposure | ec2 describe-security-groups (RDS-attached) | Fail | CLOUD-010 |
| CLOUD-T049 | Lambda Assessment | Enumerate all Lambda functions and map each to its execution role | lambda list-functions / get-function | Fail | CLOUD-009 |
| CLOUD-T050 | Lambda Assessment | Review execution-role permission scope against each function's actual resource access needs | iam get-role-policy (per function) | Fail | CLOUD-009 |
| CLOUD-T051 | Lambda Assessment | Review function environment variables for plaintext secret material | lambda get-function-configuration | Pass | — |
| CLOUD-T052 | Lambda Assessment | Review function timeout/memory configuration against observed CloudWatch Insights metrics | CloudWatch Lambda Insights review | Fail | CLOUD-020 |
| CLOUD-T053 | Lambda Assessment | Test function resource-based policy for unintended public or cross-account invoke access | lambda get-policy | Pass | — |
| CLOUD-T054 | Lambda Assessment | Review Lambda function URL / API Gateway integration authorization configuration | lambda list-function-url-configs | Pass | — |
| CLOUD-T055 | API Gateway | Enumerate all API Gateway REST/HTTP APIs, stages, and authorization configuration | apigateway get-rest-apis | Observation | — |
| CLOUD-T056 | API Gateway | Test object/broken-object-level authorization on report-fetch and account-data endpoints | Manual authenticated API testing | Pass | — |
| CLOUD-T057 | API Gateway | Review request-validation model configuration across all public routes | apigateway get-model | Observation | CLOUD-039 |
| CLOUD-T058 | API Gateway | Test rate-limiting and throttling behavior on authentication and funds-transfer endpoints | Scripted request-volume test (non-destructive) | Fail | CLOUD-030 |
| CLOUD-T059 | API Gateway | Review API Gateway access logging configuration for audit-trail completeness | apigateway get-stage | Pass | — |
| CLOUD-T060 | API Gateway | Test the report-fetch feature's outbound URL handling for SSRF (metadata and internal-range redirection) | Manual SSRF probe suite | Fail | CLOUD-003 |
| CLOUD-T061 | Container Security | Review ECS task definitions for privileged mode and added Linux capabilities | ecs describe-task-definition | Fail | CLOUD-024 |
| CLOUD-T062 | Container Security | Review ECS task role and execution role scoping per service | iam get-role-policy (per task role) | Observation | — |
| CLOUD-T063 | Container Security | Test container runtime isolation and host-mount configuration for workload separation | ecs describe-task-definition (volumes/mounts) | Pass | — |
| CLOUD-T064 | Container Security | Review ECS service network mode and security-group assignment | ecs describe-services | Pass | — |
| CLOUD-T065 | Container Security | Review GuardDuty ECS Runtime Monitoring / Malware Protection coverage for container workloads | GuardDuty console/API review | Observation | CLOUD-029 |
| CLOUD-T066 | ECR | Enumerate all ECR repositories and review image-scanning (scan-on-push) configuration | ecr describe-repositories | Pass | — |
| CLOUD-T067 | ECR | Review ECR scan findings for currently-deployed production image tags | ecr describe-image-scan-findings | Fail | CLOUD-008 |
| CLOUD-T068 | ECR | Verify CI/CD pipeline gates deployment on Critical/High ECR scan findings | Pipeline configuration review | Fail | CLOUD-008 |
| CLOUD-T069 | ECR | Review ECR repository policy for unintended cross-account pull/push access | ecr get-repository-policy | Pass | — |
| CLOUD-T070 | ECR | Review ECR image lifecycle policy configuration for unbounded image retention | ecr get-lifecycle-policy | Fail | CLOUD-031 |
| CLOUD-T071 | RDS | Enumerate all RDS instances and review Publicly Accessible configuration | rds describe-db-instances | Fail | CLOUD-010 |
| CLOUD-T072 | RDS | Test external network-layer reachability of RDS endpoints (connection only, no auth bypass attempted) | nmap / external network test | Fail | CLOUD-010 |
| CLOUD-T073 | RDS | Review RDS storage and snapshot encryption configuration (KMS) | rds describe-db-instances (StorageEncrypted) | Pass | — |
| CLOUD-T074 | RDS | Review RDS automated backup retention and cross-region snapshot copy configuration | rds describe-db-instances (BackupRetentionPeriod) | Fail | CLOUD-019 |
| CLOUD-T075 | RDS | Review RDS IAM database authentication availability as an alternative to static passwords | rds describe-db-instances (IAMDatabaseAuthenticationEnabled) | Observation | CLOUD-032 |
| CLOUD-T076 | Secrets Management | Search ECS/Lambda/EC2 workload configuration for plaintext secret material | ecs/lambda configuration grep | Fail | CLOUD-007 |
| CLOUD-T077 | Secrets Management | Enumerate Secrets Manager secrets and review rotation configuration | secretsmanager list-secrets / describe-secret | Fail | CLOUD-032 |
| CLOUD-T078 | Secrets Management | Review Secrets Manager resource policies for unintended cross-account access | secretsmanager get-resource-policy | Pass | — |
| CLOUD-T079 | Secrets Management | Review Systems Manager Parameter Store SecureString usage for sensitive configuration values | ssm describe-parameters | Pass | — |
| CLOUD-T080 | Secrets Management | Test for secrets committed to the fictional application's build artifacts or container layers | Container layer / artifact secret scan | Pass | — |
| CLOUD-T081 | KMS | Enumerate all customer-managed KMS keys and review key policy scope | kms list-keys / get-key-policy | Fail | CLOUD-023 |
| CLOUD-T082 | KMS | Review KMS key rotation configuration for all customer-managed keys | kms get-key-rotation-status | Pass | — |
| CLOUD-T083 | KMS | Test kms:ViaService and kms:CallerAccount condition enforcement on sensitive key policies | Policy simulation / manual review | Fail | CLOUD-023 |
| CLOUD-T084 | KMS | Review use of AWS-managed vs. customer-managed keys across sensitive data stores | Cross-service encryption configuration review | Pass | — |
| CLOUD-T085 | KMS | Review CloudTrail coverage of kms:Decrypt calls for anomaly-detection readiness | CloudTrail event review (kms:Decrypt) | Observation | — |
| CLOUD-T086 | CloudTrail | Verify organization trail is multi-region and covers all three fictional accounts | cloudtrail describe-trails | Pass | — |

| Test Case ID | Category | Test Description / Objective | Technique / Reference | Result | Finding |
|---|---|---|---|---|---|
| CLOUD-T087 | CloudTrail | Review CloudTrail data-event selector configuration for sensitive S3 buckets | `cloudtrail get-event-selectors` | Fail | CLOUD-011 |
| CLOUD-T088 | CloudTrail | Review CloudTrail data-event selector configuration for Lambda function invocations | `cloudtrail get-event-selectors` | Fail | CLOUD-011 |
| CLOUD-T089 | CloudTrail | Test CloudTrail log-file validation (digest) enablement and log-file integrity | `cloudtrail get-trail-status` | Pass | — |
| CLOUD-T090 | CloudTrail | Review CloudTrail log-delivery S3 bucket policy for tamper-resistance and cross-account write-only access | `s3api get-bucket-policy (CloudTrail bucket)` | Pass | — |
| CLOUD-T091 | CloudWatch | Review CloudWatch metric-filter alarm coverage against the CIS AWS Foundations-aligned baseline | `cloudwatch describe-alarms` | Fail | CLOUD-015 |
| CLOUD-T092 | CloudWatch | Test alarm routing and escalation for root-account-usage and unauthorized-API-call alarms | `Fictional test-event alarm trigger` | Pass | — |
| CLOUD-T093 | CloudWatch | Review CloudWatch Log Group retention configuration across all log groups | `logs describe-log-groups` | Fail | CLOUD-040 |
| CLOUD-T094 | CloudWatch | Review CloudWatch Logs Insights query access scoping for sensitive log data | `IAM policy review (logs:*)` | Pass | — |
| CLOUD-T095 | CloudWatch | Review CloudWatch dashboard and composite alarm coverage for executive/on-call visibility | `cloudwatch list-dashboards` | Observation | — |
| CLOUD-T096 | GuardDuty | Verify GuardDuty is enabled with appropriate protection plans (S3, EKS, Malware, RDS) across all accounts | `guardduty get-detector` | Pass | — |
| CLOUD-T097 | GuardDuty | Review EventBridge routing of GuardDuty findings to an alerting/chat-ops channel | `events list-rules (GuardDuty pattern)` | Fail | CLOUD-029 |
| CLOUD-T098 | GuardDuty | Review GuardDuty finding suppression rules for overly broad exclusions | `guardduty list-filters` | Pass | — |
| CLOUD-T099 | GuardDuty | Test GuardDuty delegated-administrator configuration for centralized findings visibility | `guardduty list-organization-admin-accounts` | Pass | — |
| CLOUD-T100 | Security Hub | Verify Security Hub is enabled and subscribed to the AWS Foundational Security Best Practices standard in all accounts | `securityhub get-enabled-standards` | Fail | CLOUD-034 |
| CLOUD-T101 | Security Hub | Review Security Hub finding aggregation across GuardDuty, Config, and Inspector | `securityhub get-findings` | Observation | — |
| CLOUD-T102 | Security Hub | Review Security Hub custom insight configuration for high-risk finding tracking | `securityhub get-insights` | Observation | — |
| CLOUD-T103 | Security Hub | Test cross-account Security Hub delegated-administrator visibility from the Security account | `securityhub list-organization-admin-accounts` | Pass | — |
| CLOUD-T104 | WAF | Enumerate WAF Web ACLs and review AWS Managed Rule Group coverage | `wafv2 list-web-acls / get-web-acl` | Pass | — |
| CLOUD-T105 | WAF | Review rate-based rule coverage on authentication and funds-transfer endpoints | `wafv2 get-web-acl (RateBasedStatement)` | Fail | CLOUD-030 |
| CLOUD-T106 | WAF | Test WAF Bot Control / CAPTCHA coverage on the login and account-creation flows | `Manual scripted-traffic test` | Fail | CLOUD-030 |
| CLOUD-T107 | WAF | Review WAF logging configuration for blocked/allowed request visibility | `wafv2 get-logging-configuration` | Pass | — |
| CLOUD-T108 | WAF | Test WAF managed-rule coverage against common injection and known-bad-input payloads (non-destructive) | `Scripted WAF bypass test suite` | Pass | — |
| CLOUD-T109 | CloudFront | Enumerate CloudFront distributions and review origin access control configuration | `cloudfront list-distributions` | Pass | — |
| CLOUD-T110 | CloudFront | Review Response Headers Policy attachment for security-header enforcement at the edge | `cloudfront get-response-headers-policy` | Fail | CLOUD-022 |
| CLOUD-T111 | CloudFront | Capture and review actual response headers across a representative sample of application routes | `Manual HTTP header capture` | Fail | CLOUD-022 |
| CLOUD-T112 | CloudFront | Test CloudFront-to-origin traffic for TLS enforcement and origin custom-header validation | `Manual origin-bypass test` | Pass | — |
| CLOUD-T113 | CloudFront | Review CloudFront geo-restriction and signed-URL/cookie configuration for restricted content paths | `cloudfront get-distribution-config` | Pass | — |
| CLOUD-T114 | Cross-Account Security | Enumerate all cross-account IAM trust relationships across the three fictional accounts | `iam list-roles + trust-policy review (all accounts)` | Fail | CLOUD-001 |
| CLOUD-T115 | Cross-Account Security | Test AssumeRole reachability of the Security-account cross-account role from a Production low-privilege identity | `sts assume-role (fictional test identity)` | Fail | CLOUD-001 |
| CLOUD-T116 | Cross-Account Security | Test AssumeRole reachability of the Production support role from a Development-account identity | `sts assume-role (fictional test identity)` | Fail | CLOUD-005 |
| CLOUD-T117 | Cross-Account Security | Review ExternalId and MFA condition enforcement on all cross-account trust policies | `IAM trust policy condition review` | Fail | CLOUD-005 |
| CLOUD-T118 | Cross-Account Security | Review AWS Organizations SCP coverage restricting cross-account privilege escalation | `organizations list-policies` | Pass | — |
| CLOUD-T119 | Backup & Recovery | Review AWS Backup plan coverage and retention configuration for production database resources | `backup get-backup-plan` | Fail | CLOUD-019 |
| CLOUD-T120 | Backup & Recovery | Review AWS Backup Vault Lock configuration for immutability against deletion | `backup describe-backup-vault` | Fail | CLOUD-019 |
| CLOUD-T121 | Backup & Recovery | Test cross-region/cross-account backup copy configuration for disaster-recovery resilience | `backup get-backup-plan (copy actions)` | Fail | CLOUD-019 |
| CLOUD-T122 | Backup & Recovery | Review backup-vault access policy for overly broad delete permissions | `backup get-backup-vault-access-policy` | Observation | CLOUD-019 |
| CLOUD-T123 | Backup & Recovery | Review restore-testing cadence and documentation for production backup sets | `Process/documentation review` | Observation | — |
| CLOUD-T124 | CI/CD Security | Review CI/CD pipeline OIDC/workload-identity federation configuration for the deployment role | `iam get-role (OIDC trust)` | Pass | — |
| CLOUD-T125 | CI/CD Security | Review production deployment-role permission scope against least privilege | `iam get-role-policy (deployment role)` | Observation | — |
| CLOUD-T126 | CI/CD Security | Test whether ECR scan findings block the CI/CD deployment pipeline | `Pipeline configuration review` | Fail | CLOUD-008 |
| CLOUD-T127 | CI/CD Security | Review pipeline secret handling for hard-coded credentials in build configuration | `Pipeline/build-config review` | Pass | — |
| CLOUD-T128 | CI/CD Security | Review artifact integrity verification (image signing) between build and deployment | `AMI/container signing configuration review` | Fail | CLOUD-021 |
| CLOUD-T129 | CI/CD Security | Review Infrastructure-as-Code drift between templated and deployed resource configuration | `terraform plan (fictional dry-run)` | Observation | CLOUD-036 |
| CLOUD-T130 | Metadata Service Security | Verify IMDSv2 (HttpTokens: required) enforcement across all EC2 launch templates and ASGs | `ec2 describe-launch-template-versions` | Fail | CLOUD-003 |
| CLOUD-T131 | Metadata Service Security | Test HttpPutResponseHopLimit configuration as a defense-in-depth SSRF mitigation | `ec2 describe-instances (MetadataOptions)` | Fail | CLOUD-003 |
| CLOUD-T132 | Metadata Service Security | Test application-layer outbound-fetch features for link-local (169.254.0.0/16) destination validation | `Manual SSRF probe suite` | Fail | CLOUD-003 |

| Test Case ID | Category | Test Description / Objective | Technique / Reference | Result | Finding |
|---|---|---|---|---|---|
| **CLOUD-T133** | Metadata Service Security | Review ECS task metadata endpoint (v4) access scoping within container network namespace | `ecs task metadata endpoint review` | Pass | — |
| **CLOUD-T134** | Metadata Service Security | Test DNS-rebinding resistance of the report-fetch feature's outbound URL validation | `Manual DNS-rebinding test (non-destructive)` | Observation | — |
| **CLOUD-T135** | Attack Path Analysis | Construct and validate the SSRF-to-metadata-to-S3 composite attack chain against documented findings | `Chain validation via finding cross-reference (non-destructive)` | Fail | CLOUD-003 |
| **CLOUD-T136** | Attack Path Analysis | Construct and validate the cross-account-trust-to-RDS composite attack chain against documented findings | `Chain validation via finding cross-reference (non-destructive)` | Fail | CLOUD-005 |
| **CLOUD-T137** | Attack Path Analysis | Construct and validate the compromised-container-to-cloud-resource-modification composite attack chain | `Chain validation via finding cross-reference (non-destructive)` | Fail | CLOUD-008 |
| **CLOUD-T138** | Attack Path Analysis | Construct and validate the IAM-policy-chaining-to-full-account-compromise composite attack chain | `Chain validation via finding cross-reference (non-destructive)` | Fail | CLOUD-004 |

# APPENDIX B — AWS ACCOUNT REGISTER

The register below documents all 3 fictional AWS accounts in scope for this assessment, their purpose, environment classification, and fictional ownership.

| Account | Account ID (Fictional) | Purpose | Environment | Owner |
|---|---|---|---|---|
| **AstraVault Production** | `123456789012 (fictional)` | Primary production workloads, customer-facing API and application, transactional data | Production | VP of Cloud Infrastructure (Fictional Role) |
| **AstraVault Security** | `123456789013 (fictional)` | Centralized logging, CloudTrail/GuardDuty/Security Hub aggregation, break-glass administrative access | Security / Shared Services | Cloud Security Engineering Lead (Fictional Role) |
| **AstraVault Development** | `123456789014 (fictional)` | Engineering development and staging workloads, CI/CD build environment | Development / Staging | Platform Engineering Lead (Fictional Role) |

# APPENDIX C — IAM PRINCIPAL REGISTER

The register below documents all 18 fictional IAM principals reviewed in depth during this assessment (a representative subset of the full 41-user / 60-role fictional inventory summarized in Section 16), their type, account, key permissions, MFA status, and any corresponding finding.

| Principal | Type | Account | Key Permissions | MFA | Finding |
|---|---|---|---|---|---|
| `astravault-security-crossaccount-role` | Role | Security (123456789013) | Inline: iam:*, sts:AssumeRole(*), organizations:* (intended: logs:PutLogEvents only) | No | CLOUD-001 |
| `astravault-prod-devsupport-role` | Role | Production (123456789012) | secretsmanager:GetSecretValue (scoped), rds:DescribeDBSnapshots | No | CLOUD-005 |
| `astravault-platform-engineers` | Group | Production (123456789012) | lambda:CreateFunction/UpdateFunctionCode, iam:PassRole(*), iam:CreatePolicyVersion(*) | Yes (SSO) | CLOUD-004 |
| `astravault-ec2-app-role` | Instance Profile Role | Production (123456789012) | s3:GetObject (scoped to app buckets), dynamodb:Query | N/A (service role) | CLOUD-003 |
| `astravault-lambda-exec-role` | Role | Production (123456789012) | s3:*, dynamodb:*, secretsmanager:GetSecretValue (Resource: *) — shared across 14 functions | N/A (service role) | CLOUD-009 |
| `astravault-app-readonly` | User | Production (123456789012) | ReadOnlyAccess (AWS managed) — fictional TMG Security limited-privilege test identity | Yes | — |
| `astravault-vendor-integration-role` | Role | Production (123456789012) | s3:GetObject/PutObject (scoped to vendor-integration-drop bucket) | N/A (service role) | CLOUD-012 |
| `astravault-deploy-oidc-role` | Role | Production (123456789012) | ecs:UpdateService, ecr:PutImage, cloudformation:* (scoped to deploy stacks) | N/A (OIDC federated) | — |
| `astravault-analytics-team` | Group | Production (123456789012) | rds:Describe*, athena:*, quicksight:* (read-focused) | Yes (SSO) | — |
| `j.former-contractor-01 (fictional)` | User | Production (123456789012) | PowerUserAccess (AWS managed) — not deactivated at offboarding | No | CLOUD-017 |
| `j.former-contractor-02 (fictional)` | User | Development (123456789014) | AdministratorAccess (AWS managed) — not deactivated at offboarding | No | CLOUD-017 |
| `astravault-legacy-batch-svc (fictional)` | User | Production (123456789012) | s3:*, batch:* — access key not rotated since creation (>500 days) | N/A (service account) | CLOUD-018 |
| `astravault-security-auditor-role` | Role | Security (123456789013) | SecurityAudit (AWS managed), securityhub:Get*, guardduty:Get* | Yes (SSO) | — |
| `astravault-breakglass-admin-role` | Role | Security (123456789013) | AdministratorAccess — fictional break-glass role, hardware-MFA + approval-logged | Yes (Hardware) | — |
| `astravault-ecs-task-role-apigw (fictional)` | Role | Production (123456789012) | s3:GetObject (scoped), secretsmanager:GetSecretValue (scoped) | N/A (service role) | CLOUD-007 |
| `astravault-ci-build-role (fictional)` | Role | Development (123456789014) | ecr:PutImage, codebuild:*, s3:GetObject (build artifacts) | N/A (service role) | — |
| `astravault-config-recorder-role` | Role | All 3 Accounts | config:Put*, s3:PutObject (scoped to Config bucket) | N/A (service role) | — |
| `astravault-eng-developers` | Group | Development (123456789014) | PowerUserAccess (AWS managed) — broad default baseline | No (policy not enforced) | CLOUD-041 |

# APPENDIX D — S3 BUCKET REGISTER

The register below documents all 22 fictional production S3 buckets reviewed during this assessment, their purpose, public-access configuration, encryption, versioning, and any corresponding finding.

| Bucket | Purpose | Public Access | Encryption | Versioning | Finding |
|---|---|---|---|---|---|
| `astravault-prod-financial-statements` | Customer account statement PDFs | Block Public Access: Disabled | None (default) | Suspended | CLOUD-002, CLOUD-014 |
| `astravault-prod-analytics-exports` | Monthly transaction analytics exports (CSV) | Block Public Access: Disabled | None (default) | Enabled | CLOUD-002 |
| `astravault-prod-user-uploads` | Customer-uploaded documents (KYC, support attachments) | Block Public Access: Enabled | None (default) | Enabled | CLOUD-013 |
| `astravault-prod-vendor-integration-drop` | Legacy vendor data exchange (fictional) | Block Public Access: Enabled | SSE-KMS | Enabled | CLOUD-012 |
| `astravault-prod-backups` | Application-level export backups | Block Public Access: Enabled | SSE-KMS | Enabled | — |
| `astravault-prod-app-static-assets` | Front-end static assets (behind CloudFront) | Block Public Access: Enabled (OAC only) | SSE-S3 | Enabled | — |
| `astravault-prod-cloudtrail-logs` | Centralized CloudTrail log delivery | Block Public Access: Enabled | SSE-KMS | Enabled | — |
| `astravault-prod-access-logs` | S3/ALB/CloudFront access log delivery | Block Public Access: Enabled | SSE-S3 | Enabled | CLOUD-026 |
| `astravault-prod-terraform-state` | Terraform remote state (fictional) | Block Public Access: Enabled | SSE-KMS | Enabled | — |
| `astravault-prod-lambda-artifacts` | Lambda deployment packages | Block Public Access: Enabled | SSE-S3 | Enabled | — |
| `astravault-dev-uploads` | Development-environment test uploads | Block Public Access: Enabled | None (default) | Disabled | CLOUD-013 |
| `astravault-dev-ci-cache` | CI/CD build cache | Block Public Access: Enabled | None (default) | Disabled | — |
| `astravault-prod-report-exports-internal` | Internal ad-hoc report exports (lower traffic) | Block Public Access: Enabled | SSE-S3 | Enabled | CLOUD-026 |
| `astravault-security-config-snapshots` | AWS Config configuration snapshots | Block Public Access: Enabled | SSE-KMS | Enabled | — |
| `astravault-prod-ml-training-data (fictional)` | Fraud-model training data exports | Block Public Access: Enabled | SSE-KMS | Enabled | — |
| `astravault-prod-support-attachments` | Customer support ticket attachments | Block Public Access: Enabled | None (default) | Enabled | CLOUD-013 |
| `astravault-prod-invoice-archive` | Long-term invoice/statement archive | Block Public Access: Enabled | SSE-KMS | Enabled | — |
| `astravault-prod-partner-webhooks-log` | Inbound partner webhook payload log | Block Public Access: Enabled | SSE-S3 | Enabled | — |
| `astravault-security-guardduty-exports` | GuardDuty finding export archive | Block Public Access: Enabled | SSE-KMS | Enabled | — |
| `astravault-prod-ecs-task-logs-archive` | Long-term ECS/CloudWatch log export archive | Block Public Access: Enabled | SSE-S3 | Enabled | — |
| `astravault-dev-sandbox-scratch` | Developer sandbox scratch space | Block Public Access: Enabled | None (default) | Disabled | — |
| `astravault-prod-reconciliation-exports` | Daily reconciliation batch exports | Block Public Access: Enabled | SSE-KMS | Enabled | — |

# APPENDIX E — EC2 / WORKLOAD REGISTER

The register below documents the fictional EC2/ECS workload inventory reviewed during this assessment, its type, purpose, IMDS configuration, and any corresponding finding.

| Resource | Type | Purpose | Finding |
|---|---|---|---|
| **astravault-prod-api-worker (ASG, 6 instances)** | EC2 Auto Scaling Group | Public API backend workers (report-fetch feature) | CLOUD-003 |
| **astravault-prod-mgmt-bastion** | EC2 Instance | Fictional management/bastion host | CLOUD-006 |
| **astravault-prod-batch-processor (ASG, 3 instances)** | EC2 Auto Scaling Group | Nightly reconciliation batch jobs | — |
| **astravault-prod-ecs-cluster** | ECS Cluster (Fargate + EC2 capacity providers) | Container orchestration for API gateway and analytics workers | — |
| **astravault-golden-ami (v2026.08)** | AMI | Golden image for all fictional production EC2 launches | CLOUD-021 |
| **astravault-dev-api-worker (ASG, 2 instances)** | EC2 Auto Scaling Group | Development-environment API backend | — |
| **astravault-prod-vpn-endpoint** | EC2 Instance (Client VPN adjunct) | Fictional corporate VPN termination | — |
| **astravault-prod-jump-analytics** | EC2 Instance | Fictional analytics-team jump host | — |

*IMDS configuration detail for each workload appears in Section 19 and Section 39.*

# APPENDIX F — VPC / SUBNET / SECURITY GROUP REGISTER

The register below documents the fictional VPC, subnet, gateway, and security-group inventory across AstraVault's three fictional AWS accounts, reviewed during the Section 20-22 network assessment.

| Resource | Type | Scope / CIDR | Notes | Finding |
|---|---|---|---|---|
| **vpc-0fictprod01** | VPC | 10.20.0.0/16 (Production, us-east-1, fictional) | Public, Private-App, Private-DB, Management/Security subnet tiers | — |
| **vpc-0fictsec01** | VPC | 10.30.0.0/16 (Security, us-east-1, fictional) | Log aggregation, break-glass tooling | — |
| **vpc-0fictdev01** | VPC | 10.40.0.0/16 (Development, us-east-1, fictional) | Single-tier development/staging network | CLOUD-027 |
| **subnet-pub-a/b (Production)** | Public Subnet (x2, multi-AZ) | 10.20.0.0/22 | ALB, NAT Gateway, bastion | — |
| **subnet-app-a/b (Production)** | Private Application Subnet (x2, multi-AZ) | 10.20.16.0/20 | ECS/EC2 application workloads, Lambda ENIs | — |
| **subnet-db-a/b (Production)** | Private Database Subnet (x2, multi-AZ) | 10.20.32.0/20 | RDS primary/replica — astravault-analytics-reporting-db incorrectly placed in public subnet | CLOUD-010 |
| **subnet-mgmt-a (Production)** | Management/Security Subnet | 10.20.48.0/24 | Bastion, VPN endpoint, SSM-managed hosts | — |
| **igw-0fictprod** | Internet Gateway | Production VPC | Attached to public subnets only | — |
| **nat-0fictprod-a/b** | NAT Gateway (x2, multi-AZ) | Production VPC public subnets | Outbound-only path for private subnets | — |
| **tgw-0fictastravault** | Transit Gateway | Inter-account connectivity (fictional) | Connects Production, Security, Development VPCs | — |
| **vpce-s3-prod, vpce-secretsmanager-prod, vpce-kms-prod** | VPC Gateway/Interface Endpoints | Production VPC | Private connectivity for S3/Secrets Manager/KMS — not yet used to eliminate broad egress (CLOUD-016) | CLOUD-016 |
| **sg-0fictmgmt01 (astravault-prod-mgmt-sg)** | Security Group | Bastion + RDS-adjacent hosts | Inbound 22/3389/5432 from 0.0.0.0/0 | CLOUD-006 |
| **sg-0fictapp01 (astravault-prod-app-sg)** | Security Group | Application/API worker tier | Inbound 443 from ALB SG only; default all-traffic egress | CLOUD-016 |
| **sg-0fictdb01 (astravault-prod-db-sg)** | Security Group | RDS primary transactional database | Inbound 5432 from app-sg only (correctly scoped) | — |
| **sg-0fictanalyticsdb (astravault-analytics-db-sg)** | Security Group | Analytics reporting RDS instance | Inbound 5432 from 0.0.0.0/0 (public-subnet placement) | CLOUD-010 |
| **nacl-0fictprodapp** | Network ACL | Private Application Subnet | Default allow-all (no additional restriction beyond security groups) | — |

# APPENDIX G — LAMBDA / API REGISTER

The register below documents all 16 fictional Lambda functions and API Gateway resources reviewed during the Section 23-24 serverless assessment, their purpose, authorization model, and any corresponding finding.

| Function / API | Type | Purpose | Auth | Finding |
|---|---|---|---|---|
| `astravault-prod-statement-generator` | Lambda Function | Generates monthly customer account statements | IAM (invoked by Step Functions) | — |
| `astravault-prod-webhook-processor` | Lambda Function | Processes inbound partner/payment-aggregator webhooks | API Gateway + shared secret header | CLOUD-009, CLOUD-020 |
| `astravault-prod-fraud-check` | Lambda Function | Fraud-monitoring pipeline consumer (SNS subscriber) | IAM (SNS-invoked) | — |
| `astravault-prod-transaction-router` | Lambda Function | Routes transaction events between services (EventBridge target) | IAM (EventBridge-invoked) | CLOUD-037 |
| `astravault-prod-report-export-worker` | Lambda Function | Async generation of analytics export files | IAM (SQS-invoked) | — |
| `astravault-prod-user-notification` | Lambda Function | Sends transactional email/SMS notifications | IAM (SNS-invoked) | — |
| `astravault-prod-kyc-doc-processor` | Lambda Function | Processes uploaded KYC documents | IAM (S3-invoked) | — |
| `astravault-prod-session-cleanup` | Lambda Function | Scheduled cleanup of expired session records | IAM (EventBridge scheduled) | — |
| `astravault-prod-audit-log-shipper` | Lambda Function | Ships application audit logs to centralized pipeline | IAM (CloudWatch Logs subscription) | — |
| `astravault-prod-payment-reconciliation` | Lambda Function | Daily payment-aggregator reconciliation job | IAM (EventBridge scheduled) | — |
| `astravault-prod-pdf-render` | Lambda Function | Renders statement PDFs from templated data | IAM (Step Functions-invoked) | — |
| `astravault-prod-webhook-retry` | Lambda Function | Retries failed outbound partner webhook deliveries | IAM (SQS-invoked) | — |
| `astravault-prod-analytics-aggregator` | Lambda Function | Aggregates daily analytics rollups | IAM (EventBridge scheduled) | — |
| `astravault-prod-support-ticket-sync` | Lambda Function | Syncs support tickets with fictional CRM | IAM (EventBridge scheduled) | — |
| `api.astravault-example.com (REST API)` | API Gateway | Primary customer-facing / mobile-and-web backend API (27 routes) | Cognito-style bearer token (fictional) | CLOUD-030, CLOUD-039 |
| `console.astravault-example.com (Internal API)` | API Gateway | Internal operations/admin console API | IAM + SSO-federated session | — |

# APPENDIX H — CONTAINER / ECR REGISTER

The register below documents the fictional ECR repository and container-image inventory reviewed during the Section 25-26 container assessment, including the most recent fictional scan status and lifecycle-policy configuration.

| Repository / Task | Latest Tag | Scan Status | Lifecycle Policy | Finding |
|---|---|---|---|---|
| **astravault/api-gateway-service** | 2026-09-10 | Pass (0 Critical/High) | Configured | CLOUD-024 |
| **astravault/analytics-worker** | 2026-08-12 | Fail (1 Critical, 4 High) | Configured | CLOUD-008 |
| **astravault/webhook-processor-container** | 2026-09-08 | Pass (0 Critical/High) | Not Configured | CLOUD-031 |
| **astravault/statement-generator** | 2026-09-01 | Pass (0 Critical/High) | Not Configured | CLOUD-031 |
| **astravault/fraud-detection-model-server (fictional)** | 2026-08-28 | Pass (2 Medium) | Not Configured | CLOUD-031 |
| **astravault/internal-admin-console** | 2026-09-05 | Pass (0 Critical/High) | Not Configured | CLOUD-031 |
| **astravault/reconciliation-batch** | 2026-09-02 | Pass (1 Medium) | Not Configured | CLOUD-031 |
| **astravault/notification-service** | 2026-08-30 | Pass (0 Critical/High) | Not Configured | CLOUD-031 |
| **astravault/kyc-doc-processor** | 2026-09-03 | Pass (0 Critical/High) | Not Configured | CLOUD-031 |
| **astravault/support-sync-service** | 2026-08-25 | Pass (0 Critical/High) | Not Configured | CLOUD-031 |
| **astravault/legacy-batch-runner (fictional, deprecated)** | 2025-11-14 | Fail (3 High — unscanned since deprecation) | Not Configured | CLOUD-031 |
| **astravault-api-gateway-service (ECS task def :41)** | Task Definition | privileged: false; capabilities.add: SYS_ADMIN, NET_ADMIN | N/A | CLOUD-024 |

# APPENDIX I — DATABASE REGISTER

The register below documents the fictional relational and NoSQL database inventory reviewed during the Section 27 data-security assessment, their public-accessibility and encryption configuration, backup retention, and any corresponding finding.

| Database | Engine | Publicly Accessible | Encryption | Finding |
|---|---|---|---|---|
| `astravault-prod-financial-db` | Aurora PostgreSQL 15 (fictional) | No | SSE-KMS (StorageEncrypted: true) | CLOUD-019 |
| `astravault-analytics-reporting-db` | RDS PostgreSQL 15 (fictional) | Yes | SSE-KMS (StorageEncrypted: true) | CLOUD-010 |
| `astravault-prod-sessions-cache (fictional)` | ElastiCache Redis (fictional) | No | Encryption in transit + at rest enabled | — |
| `astravault-dev-financial-db-replica` | Aurora PostgreSQL 15 (fictional, Development account) | No | SSE-KMS (StorageEncrypted: true) | CLOUD-041 |
| `astravault-prod-audit-log-db (fictional)` | DynamoDB (fictional) | No (private VPC endpoint only) | SSE (AWS owned key) | — |

*Backup retention detail for each database resource appears in Appendix I's underlying register and is summarized in Section 37.*

# APPENDIX J — CLOUD SECURITY CONTROL MATRIX

The matrix below maps this assessment's coverage against all 27 Cloud Security Domains introduced in Section 06 and tested across Sections 13-40, confirming every domain in scope received at least one dedicated testing method.

| Domain | Description | Tested? | Method | Result | Related Findings |
|---|---|---|---|---|---|
| **01 · External Cloud Attack Surface** | Internet-facing DNS, endpoints, and services reachable without authentication | Yes | External black-box reconnaissance and enumeration | **Weaknesses Identified** | `CLOUD-002, CLOUD-028` |
| **02 · IAM & Identity Security** | IAM users, roles, groups, policies, and credential lifecycle | Yes | Authenticated configuration review + IAM credential report analysis | **Weaknesses Identified** | `CLOUD-001, CLOUD-017, CLOUD-018` |
| **03 · Privilege Escalation** | Documented IAM privilege-escalation pattern analysis | Yes | IAM policy simulation against known escalation patterns | **Weaknesses Identified** | `CLOUD-004` |
| **04 · S3 & Object Storage Security** | Bucket policy, encryption, versioning, logging, and public-access configuration | Yes | External black-box + authenticated configuration review | **Weaknesses Identified** | `CLOUD-002, CLOUD-012, CLOUD-013, CLOUD-014, CLOUD-026` |
| **05 · EC2 & Workload Security** | Instance configuration, IMDS, AMI pipeline, patch compliance | Yes | Authenticated review + assumed-breach workload testing | **Weaknesses Identified** | `CLOUD-003, CLOUD-021, CLOUD-038` |
| **06 · VPC & Network Security** | VPC topology, subnet tiering, routing, endpoints, peering | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-010, CLOUD-027` |
| **07 · Security Groups & Network ACLs** | Inbound/outbound rule scope across all security groups | Yes | Authenticated review + external port scanning | **Weaknesses Identified** | `CLOUD-006, CLOUD-016` |
| **08 · Serverless / Lambda Security** | Execution role scope, configuration, resource-based policies | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-009, CLOUD-020, CLOUD-037` |
| **09 · API Gateway & Cloud APIs** | Authorization, request validation, rate limiting | Yes | Authenticated API testing + configuration review | **Weaknesses Identified** | `CLOUD-030, CLOUD-039` |
| **10 · Container Security** | ECS task definition privilege and isolation configuration | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-024` |
| **11 · ECR / Image Security** | Image scanning, lifecycle policy, deployment gating | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-008, CLOUD-031` |
| **12 · RDS & Database Security** | Public accessibility, encryption, backup configuration | Yes | Authenticated review + external network testing | **Weaknesses Identified** | `CLOUD-010` |
| **13 · Secrets Management** | Secrets Manager and Parameter Store usage, rotation, plaintext exposure | Yes | Authenticated configuration review + workload configuration search | **Weaknesses Identified** | `CLOUD-007, CLOUD-032` |
| **14 · KMS & Encryption** | Key policy scope, rotation, encryption enforcement | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-013, CLOUD-023` |
| **15 · Logging & Monitoring** | CloudWatch alerting, log retention, coverage completeness | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-015, CLOUD-026, CLOUD-040` |
| **16 · CloudTrail** | Trail configuration, data-event coverage, log integrity | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-011` |
| **17 · WAF & Edge Security** | Managed rules, rate-based rules, bot protection | Yes | Authenticated configuration review + scripted traffic testing | **Weaknesses Identified** | `CLOUD-030` |
| **18 · CloudFront / CDN Security** | Origin access control, security-header enforcement | Yes | External testing + authenticated configuration review | **Weaknesses Identified** | `CLOUD-022` |
| **19 · Cross-Account Trust** | Trust policy scope, MFA/ExternalId enforcement | Yes | Authenticated review + limited-privilege AssumeRole testing | **Weaknesses Identified** | `CLOUD-001, CLOUD-005` |
| **20 · Backup & Recovery Security** | Retention, cross-region copy, immutability (Vault Lock) | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-019` |
| **21 · Cloud Misconfiguration** | Tagging, naming, general hygiene | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-025, CLOUD-033, CLOUD-035, CLOUD-041` |
| **22 · SSRF / Metadata Service Exposure** | IMDS configuration and application-layer SSRF resistance | Yes | Assumed-breach testing + manual SSRF probes | **Weaknesses Identified** | `CLOUD-003` |
| **23 · Container Escape / Workload Isolation** | Linux capability minimization, runtime isolation | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-024` |
| **24 · CI/CD & Cloud Deployment Security** | Pipeline permissions, scan gating, artifact integrity | Yes | Authenticated pipeline configuration review | **Weaknesses Identified** | `CLOUD-008, CLOUD-021, CLOUD-036` |
| **25 · Cloud Persistence & Detection** | GuardDuty/Security Hub routing and response readiness | Yes | Authenticated configuration review | **Weaknesses Identified** | `CLOUD-029, CLOUD-034, CLOUD-042` |
| **26 · Attack Path Analysis** | Composite multi-finding attack-chain validation | Yes | Cross-referenced finding-chain validation (non-destructive) | **Weaknesses Identified** | `CLOUD-001, CLOUD-003, CLOUD-004, CLOUD-005, CLOUD-008` |
| **27 · Business Impact Analysis** | Translation of technical findings into fictional business risk | Yes | Analyst assessment against fictional business context | **Reviewed** | `All Critical/High findings` |

# APPENDIX K — FINDING REGISTER

The register below lists all 42 fictional findings in this report (CLOUD-001 through CLOUD-042), generated directly from this report's canonical finding data so that Sections 42-46 and this appendix can never drift out of sync. Full technical detail for each finding appears in Sections 42-46; this appendix serves as a single-page cross-reference.

| Finding | Severity | Title | Domain | CVSS | Status |
|---|---|---|---|---|---|
| CLOUD-001 | CRITICAL | Overly Permissive IAM Role Enables Cross-Account Administrative Privilege Escalation | 02 · IAM & Identity Security / 19 · Cross-Account Trust | 9.6 | Remediation In Progress |
| CLOUD-002 | CRITICAL | Publicly Accessible S3 Bucket Exposes Sensitive Financial Data | 04 · S3 & Object Storage Security | 9.1 | Remediation In Progress |
| CLOUD-003 | CRITICAL | EC2 Workload Metadata Exposure Enables Credential Theft and Privilege Escalation | 05 · EC2 & Workload Security / 22 · SSRF & Metadata Service Exposure | 9.3 | Remediation In Progress |
| CLOUD-004 | HIGH | Privilege Escalation via IAM Policy Chaining (iam:PassRole + iam:CreatePolicyVersion) | 03 · Privilege Escalation | 8.4 | Remediation Planned |
| CLOUD-005 | HIGH | Excessive Trust Relationship Between Production and Development AWS Accounts | 19 · Cross-Account Trust | 7.9 | Remediation Planned |
| CLOUD-006 | HIGH | Security Group Permits Unrestricted Administrative Access to Management Ports | 07 · Security Groups & Network ACLs | 8.1 | Remediation Planned |
| CLOUD-007 | HIGH | Secrets Exposed Through Workload Environment Configuration | 13 · Secrets Management | 7.6 | Remediation In Progress |
| CLOUD-008 | HIGH | Vulnerable Container Image With Exploitable Package in Production ECR Repository | 11 · ECR / Image Security | 8.2 | Remediation In Progress |
| CLOUD-009 | HIGH | Lambda Execution Role Grants Excessive IAM Permissions Beyond Function Requirements | 08 · Serverless / Lambda Security | 7.7 | Remediation Planned |
| CLOUD-010 | HIGH | RDS Database Instance Publicly Reachable From the Internet | 12 · RDS & Database Security | 8.0 | Remediation Planned |
| CLOUD-011 | HIGH | CloudTrail Logging Coverage Gap on Security-Sensitive API Activity | 16 · CloudTrail | 7.3 | Remediation Planned |
| CLOUD-012 | MEDIUM | Overly Broad S3 Bucket Resource Policy Grants Unnecessary Cross-Account Access | 04 · S3 & Object Storage Security | 6.5 | Remediation Planned |
| CLOUD-013 | MEDIUM | Missing Server-Side Encryption Enforcement on S3 Buckets Storing Sensitive Data | 04 · S3 & Object Storage Security / 14 · KMS & Encryption | 5.9 | Remediation Planned |
| CLOUD-014 | MEDIUM | S3 Bucket Versioning Disabled on Buckets Storing Financial Records | 04 · S3 & Object Storage Security / 20 · Backup & Recovery Security | 5.3 | Remediation Planned |
| CLOUD-015 | MEDIUM | Insufficient CloudWatch Alerting on High-Risk IAM and Security Events | 15 · Logging & Monitoring | 5.7 | Remediation Planned |
| CLOUD-016 | MEDIUM | Permissive Security Group Egress Rules Allow Unrestricted Outbound Access | 07 · Security Groups & Network ACLs | 5.4 | Remediation Planned |
| CLOUD-017 | MEDIUM | Stale IAM Users With Long-Unused Console and API Access | 02 · IAM & Identity Security | 5.0 | Remediation In Progress |
| CLOUD-018 | MEDIUM | Unused and Unrotated IAM Access Keys Exceeding Recommended Age | 02 · IAM & Identity Security | 4.9 | Remediation Planned |
| CLOUD-019 | MEDIUM | Weak AWS Backup Configuration for Production Database Resources | 20 · Backup & Recovery Security | 5.2 | Remediation Planned |
| CLOUD-020 | MEDIUM | Lambda Function Configured With Excessive Timeout and Memory Beyond Operational Need | 08 · Serverless / Lambda Security | 4.6 | Remediation Planned |
| CLOUD-021 | MEDIUM | Missing Workload Integrity Controls on EC2 AMI Build Pipeline | 05 · EC2 & Workload Security / 24 · CI/CD & Cloud Deployment Security | 5.6 | Remediation Planned |
| CLOUD-022 | MEDIUM | Insufficient Security-Header Controls at CloudFront Edge | 18 · CloudFront / CDN Security | 5.1 | Remediation Planned |
| CLOUD-023 | MEDIUM | KMS Key Policy Grants Overly Broad Decrypt Permissions | 14 · KMS & Encryption | 6.1 | Remediation Planned |
| CLOUD-024 | MEDIUM | ECS Task Definition Runs Containers With Elevated Linux Capabilities | 10 · Container Security / 23 · Container Escape & Workload Isolation | 5.8 | Remediation Planned |
| CLOUD-025 | LOW | Incomplete Resource Tagging Undermines Cost and Security Governance | 21 · Cloud Misconfiguration | 3.1 | Remediation Planned |
| CLOUD-026 | LOW | S3 Access Logging Not Enabled on Selected Buckets | 04 · S3 & Object Storage Security / 15 · Logging & Monitoring | 3.4 | Remediation Planned |
| CLOUD-027 | LOW | VPC Flow Logs Not Enabled on Selected VPCs | 06 · VPC & Network Security / 15 · Logging & Monitoring | 3.6 | Remediation Planned |
| CLOUD-028 | LOW | Route 53 DNS Zone Lacks Query Logging | 01 · External Cloud Attack Surface / 15 · Logging & Monitoring | 2.8 | Remediation Planned |
| CLOUD-029 | LOW | GuardDuty Findings Not Routed to a Centralized Alerting Channel | 25 · Cloud Persistence & Detection | 3.9 | Remediation Planned |
| CLOUD-030 | LOW | WAF Rule Set Missing Rate-Based Rule on Public API Gateway | 17 · WAF & Edge Security | 3.8 | Remediation Planned |
| CLOUD-031 | LOW | ECR Repository Missing Image Lifecycle Policy (Unbounded Image Retention) | 11 · ECR / Image Security | 2.6 | Remediation Planned |
| CLOUD-032 | LOW | Secrets Manager Rotation Not Configured for Long-Lived Database Credentials | 13 · Secrets Management | 3.7 | Remediation Planned |
| CLOUD-033 | LOW | SNS Topic Policy Permits Overly Broad Publish Access | 21 · Cloud Misconfiguration | 3.2 | Remediation Planned |
| CLOUD-034 | INFORMATIONAL | Security Hub Standard Subscriptions Incomplete Across Accounts | 15 · Logging & Monitoring | N/A | Remediation Planned |
| CLOUD-035 | INFORMATIONAL | Inconsistent Resource Naming Convention Complicates Governance | 21 · Cloud Misconfiguration | N/A | Remediation Planned |
| CLOUD-036 | INFORMATIONAL | Infrastructure-as-Code Drift Observed Between Deployed and Templated Resources | 24 · CI/CD & Cloud Deployment Security | N/A | Remediation Planned |
| CLOUD-037 | INFORMATIONAL | EventBridge Rule Lacks Dead-Letter Queue for Failed Invocations | 08 · Serverless / Lambda Security | N/A | Remediation Planned |
| CLOUD-038 | INFORMATIONAL | Systems Manager Patch Compliance Reporting Gaps on EC2 Fleet | 05 · EC2 & Workload Security | N/A | Remediation Planned |
| CLOUD-039 | INFORMATIONAL | API Gateway Missing Request Validation on Selected Public Endpoints | 09 · API Gateway & Cloud APIs | N/A | Remediation Planned |
| CLOUD-040 | INFORMATIONAL | CloudWatch Log Group Retention Set to Indefinite on Non-Critical Logs | 15 · Logging & Monitoring | N/A | Remediation Planned |
| CLOUD-041 | INFORMATIONAL | Development Account Lacks Equivalent Security Baseline to Production | 21 · Cloud Misconfiguration | N/A | Remediation Planned |
| CLOUD-042 | INFORMATIONAL | Documentation Gap in Incident Response Runbook for Cloud-Specific Scenarios | 25 · Cloud Persistence & Detection | N/A | Remediation Planned |

# APPENDIX L — EVIDENCE REGISTER

The register below indexes all 47 fictional evidence items (EV-CLOUD-001 through EV-CLOUD-047) referenced across the finding cards in Sections 42-46. All evidence is synthetic and illustrative, constructed for this sample report; no real AWS environment was accessed.

| Evidence ID | Type | Description | Finding |
|---|---|---|---|
| EV-CLOUD-001 | IAM Policy Export | Fictional cross-account role trust policy export (astravault-security-crossaccount-role) | CLOUD-001 |
| EV-CLOUD-002 | STS Trace | Fictional sts:AssumeRole probe and resulting inline-policy enumeration | CLOUD-001 |
| EV-CLOUD-003 | Network Capture | Fictional anonymous S3 bucket listing capture (astravault-prod-financial-statements) | CLOUD-002 |
| EV-CLOUD-004 | S3 Configuration Export | Fictional bucket policy export showing wildcard Principal grant | CLOUD-002 |
| EV-CLOUD-005 | HTTP Request/Response | Fictional SSRF request/response capture via report-fetch feature | CLOUD-003 |
| EV-CLOUD-006 | Credential Payload | Fictional recovered IMDS temporary-credential JSON payload | CLOUD-003 |
| EV-CLOUD-007 | IAM Policy Simulation | Fictional iam simulate-principal-policy output confirming escalation pattern EP-04 | CLOUD-004 |
| EV-CLOUD-008 | IAM Trust Policy Export | Fictional cross-account trust policy export (astravault-prod-devsupport-role) | CLOUD-005 |
| EV-CLOUD-009 | Port Scan Result | Fictional external nmap-style scan of the management/bastion security group | CLOUD-006 |
| EV-CLOUD-010 | Security Group Export | Fictional security-group rule export showing 0.0.0.0/0 on 22/3389/5432 | CLOUD-006 |
| EV-CLOUD-011 | ECS Configuration Export | Fictional ecs describe-task-definition output showing plaintext environment secrets | CLOUD-007 |
| EV-CLOUD-012 | ECR Scan Result | Fictional ECR image scan finding export (astravault/analytics-worker) | CLOUD-008 |
| EV-CLOUD-013 | IAM Policy Export | Fictional shared Lambda execution-role policy export and function-mapping | CLOUD-009 |
| EV-CLOUD-014 | Network Reachability Test | Fictional external TCP connectivity confirmation against RDS endpoint | CLOUD-010 |
| EV-CLOUD-015 | RDS Configuration Export | Fictional rds describe-db-instances output (PubliclyAccessible: true) | CLOUD-010 |
| EV-CLOUD-016 | CloudTrail Configuration Export | Fictional CloudTrail event-selector configuration (empty DataResources) | CLOUD-011 |
| EV-CLOUD-017 | S3 Policy Export | Fictional vendor-integration bucket policy export | CLOUD-012 |
| EV-CLOUD-018 | S3 Configuration Audit | Fictional default-encryption configuration audit across 22 production buckets | CLOUD-013 |
| EV-CLOUD-019 | S3 Configuration Export | Fictional bucket versioning configuration export (Suspended) | CLOUD-014 |
| EV-CLOUD-020 | CloudWatch Configuration Audit | Fictional CloudWatch alarm inventory vs. 12-alarm recommended baseline | CLOUD-015 |
| EV-CLOUD-021 | Security Group Audit | Fictional egress-rule audit across 24 production security groups | CLOUD-016 |
| EV-CLOUD-022 | IAM Credential Report | Fictional IAM credential report filtered to 180+ day inactivity | CLOUD-017 |
| EV-CLOUD-023 | IAM Credential Report | Fictional IAM credential report, access-key-age analysis (365+ days) | CLOUD-018 |
| EV-CLOUD-024 | AWS Backup Configuration Export | Fictional backup plan and vault-policy configuration export | CLOUD-019 |
| EV-CLOUD-025 | CloudWatch Metrics Export | Fictional Lambda Insights metric export vs. configured resource limits | CLOUD-020 |
| EV-CLOUD-026 | Pipeline Configuration Review | Fictional AMI build pipeline and launch-template configuration review | CLOUD-021 |
| EV-CLOUD-027 | HTTP Header Capture | Fictional response-header capture across sampled application routes | CLOUD-022 |
| EV-CLOUD-028 | KMS Policy Export | Fictional customer-managed KMS key policy export | CLOUD-023 |
| EV-CLOUD-029 | ECS Configuration Export | Fictional task-definition linuxParameters.capabilities export | CLOUD-024 |
| EV-CLOUD-030 | Tag Audit | Fictional organization-wide tag-coverage audit export | CLOUD-025 |
| EV-CLOUD-031 | S3 Configuration Audit | Fictional bucket logging configuration audit across 22 production buckets | CLOUD-026 |
| EV-CLOUD-032 | VPC Configuration Audit | Fictional VPC Flow Logs configuration audit across three accounts | CLOUD-027 |
| EV-CLOUD-033 | Route 53 Configuration Review | Fictional public hosted-zone query-logging configuration review | CLOUD-028 |
| EV-CLOUD-034 | GuardDuty/EventBridge Review | Fictional GuardDuty detector and EventBridge routing-rule review | CLOUD-029 |
| EV-CLOUD-035 | WAF Configuration Export | Fictional Web ACL rule configuration export | CLOUD-030 |
| EV-CLOUD-036 | ECR Configuration Audit | Fictional lifecycle-policy audit across 11 production ECR repositories | CLOUD-031 |
| EV-CLOUD-037 | Secrets Manager Configuration Review | Fictional secret rotation-configuration review | CLOUD-032 |
| EV-CLOUD-038 | SNS Policy Export | Fictional topic policy export (astravault-prod-transaction-notifications) | CLOUD-033 |
| EV-CLOUD-039 | Security Hub Configuration Review | Fictional standard-subscription review across three accounts | CLOUD-034 |
| EV-CLOUD-040 | Resource Inventory Sample | Fictional resource-naming-pattern sample from Appendix B-I inventory build | CLOUD-035 |
| EV-CLOUD-041 | Terraform Drift Summary | Fictional terraform plan dry-run drift summary | CLOUD-036 |
| EV-CLOUD-042 | EventBridge Configuration Review | Fictional EventBridge rule target/DLQ configuration review | CLOUD-037 |
| EV-CLOUD-043 | SSM Configuration Export | Fictional Patch Compliance dashboard export | CLOUD-038 |
| EV-CLOUD-044 | API Gateway Configuration Review | Fictional route/request-validation-model configuration review | CLOUD-039 |
| EV-CLOUD-045 | CloudWatch Configuration Audit | Fictional Log Group retention-configuration audit | CLOUD-040 |
| EV-CLOUD-046 | Cross-Reference Summary | Fictional Development-account baseline-gap cross-reference summary | CLOUD-041 |
| EV-CLOUD-047 | Document Review Notes | Fictional Incident Response runbook review notes | CLOUD-042 |

# APPENDIX M — ATTACK PATH REGISTER

The register below summarizes all 4 fictional composite attack chains documented in Section 40, each of which was validated conceptually against real finding IDs in this report using non-destructive analysis only.

## AP-01 — External Attack Surface → SSRF → Instance Metadata → S3 Financial Data

1) Attacker reaches the internet-facing report-fetch feature. 2) Supplies a link-local metadata URL, exploiting IMDSv1 and missing outbound-URL validation. 3) Recovers temporary IAM credentials for the EC2 instance profile. 4) Uses the over-permissive role to access the publicly-adjacent financial-statements S3 bucket.

**Findings Involved: CLOUD-003, CLOUD-001, CLOUD-002**

*Business Impact: Unauthenticated internet access converted directly into theft of live cloud credentials and exposure of sensitive customer financial data — the most severe fictional chain identified in this assessment.*

## AP-02 — Compromised Developer Identity → Cross-Account Trust → Secrets Manager → RDS Access

1) Attacker compromises a fictional Development-account developer credential (the weaker of the two account baselines). 2) Assumes the overly-broad astravault-prod-devsupport-role trust relationship into Production. 3) Reads Production Secrets Manager values reachable by that role. 4) Uses recovered database credentials to reach the RDS analytics-reporting instance, itself already publicly reachable at the network layer.

**Findings Involved: CLOUD-005, CLOUD-032, CLOUD-010**

*Business Impact: A foothold in the lower-assurance Development account escalates directly into Production database credential theft, defeating the intended account-boundary security model.*

## AP-03 — Compromised Container → Task Role Credentials → Excessive IAM Permissions → Cloud Resource Modification

1) Attacker exploits the known-vulnerable package in the astravault/analytics-worker container image. 2) Achieves code execution inside the container; elevated Linux capabilities (SYS_ADMIN/NET_ADMIN) increase escape feasibility. 3) Retrieves the ECS task role's temporary credentials via the task metadata endpoint. 4) Uses the shared, over-permissioned Lambda/task execution role pattern to modify cloud resources (S3, DynamoDB) beyond the compromised service's legitimate need.

**Findings Involved: CLOUD-008, CLOUD-024, CLOUD-009**

*Business Impact: A single unpatched container dependency cascades into broad cloud-resource-modification capability due to shared, over-permissioned execution roles.*

## AP-04 — IAM Policy Chaining → Role Assumption → Full Account Compromise

1) Attacker compromises a standard fictional engineering-tier credential (e.g., via phishing). 2) Uses the group's iam:CreatePolicyVersion and iam:PassRole permissions to create a new policy version granting broader access and pass a higher-privileged role to a new Lambda function. 3) Executes arbitrary AWS API calls under the elevated role's identity. 4) Reaches the same cross-account administrative path documented in AP-01/CLOUD-001, achieving full fictional Security-account compromise.

**Findings Involved: CLOUD-004, CLOUD-001**

*Business Impact: Demonstrates that even without directly compromising the most over-permissioned identity, a documented IAM policy-chaining technique reaches the same critical impact from a much lower-privileged starting point.*

# APPENDIX N — REMEDIATION PRIORITY MATRIX

The matrix below sequences all 42 fictional findings by remediation phase (generated directly from this report's canonical finding data, consistent with the roadmap in Section 49), each paired with its recommended AWS control focus and fictional owner.

| Finding | Severity | Priority Phase | Recommended Control Focus | Owner |
|---|---|---|---|---|
| CLOUD-001 | CRITICAL | Immediate — 0-14 Days | IAM permission boundaries on all cross-account roles; SCP (Service Control Policy) at the AWS Organizations level denying iam:CreateUser / iam:AttachUserPolicy for non-administrative OUs; IAM Access Analyzer cross-account findings review; mandatory ExternalId and MFA condition on every cross-account trust policy. | Cloud Security Engineering Lead (Fictional Role) |
| CLOUD-002 | CRITICAL | Immediate — 0-14 Days | S3 Block Public Access (account-level SCP enforcement), bucket policies scoped to VPC endpoints or specific role ARNs, default SSE-KMS encryption, AWS Config rule s3-bucket-public-read-prohibited, GuardDuty S3 protection, Macie for sensitive-data discovery on remaining buckets. | Cloud Platform Engineering Lead (Fictional Role) |
| CLOUD-003 | CRITICAL | Immediate — 0-14 Days | IMDSv2 enforcement via SCP (ec2:MetadataHttpTokens condition) and AWS Config rule ec2-imdsv2-check; VPC egress controls limiting outbound fetch destinations; instance-profile least-privilege scoping; GuardDuty EC2 finding types for IMDS credential exfiltration. | Platform Engineering Lead (Fictional Role) |
| CLOUD-004 | HIGH | 0-30 Days | IAM Access Analyzer, permission boundaries on all human IAM principals, separation of deployment-time IAM permissions from day-to-day engineering permissions, scheduled privilege-escalation-path scanning (e.g., PMapper/Cloudsplaining-style analysis) as a recurring control. | IAM & Identity Lead (Fictional Role) |
| CLOUD-005 | HIGH | 0-30 Days | Named-principal trust policies, mandatory MFA condition on cross-account AssumeRole, session-duration capping, CloudTrail alerting on this specific role's assumption events routed to the Security account. | Cloud Security Engineering Lead (Fictional Role) |
| CLOUD-006 | HIGH | 0-30 Days | Systems Manager Session Manager for host access, security-group rule restricted to VPN/bastion CIDR only, AWS Config rule restricted-ssh / restricted-common-ports, Network ACL as defense-in-depth. | Network Engineering Lead (Fictional Role) |
| CLOUD-007 | HIGH | 0-30 Days | ECS secrets integration with Secrets Manager, IAM least-privilege scoping of ecs:DescribeTaskDefinition, Secrets Manager automatic rotation, pre-deployment secret-scanning in the CI/CD pipeline. | Platform Engineering Lead (Fictional Role) |
| CLOUD-008 | HIGH | 0-30 Days | ECR scan-on-push with deployment-pipeline gating, scheduled base-image rebuilds, Amazon Inspector for continuous container scanning, image signing/attestation (e.g., cosign) verified at deploy time. | DevOps / Platform Engineering Lead (Fictional Role) |
| CLOUD-009 | HIGH | 0-30 Days | Per-function IAM roles, IAM Access Analyzer, AWS Config rule for Lambda function public access and role scope review, scheduled least-privilege audits. | Serverless Engineering Lead (Fictional Role) |
| CLOUD-010 | HIGH | 0-30 Days | RDS private-subnet placement, AWS Config rule rds-instance-public-access-check, VPC-only security-group scoping, IAM database authentication where supported. | Data Engineering Lead (Fictional Role) |
| CLOUD-011 | HIGH | 0-30 Days | CloudTrail advanced event selectors for S3/Lambda data events, centralized log aggregation to the Security account, CloudWatch Logs Insights or a SIEM-equivalent for correlation, AWS Config rule cloudtrail-s3-dataevents-enabled. | Security Operations Lead (Fictional Role) |
| CLOUD-012 | MEDIUM | 31-60 Days | IAM Access Analyzer external-access findings review, quarterly resource-policy audit, S3 Access Points for fine-grained per-consumer access scoping. | Cloud Platform Engineering Lead (Fictional Role) |
| CLOUD-013 | MEDIUM | 31-60 Days | S3 default encryption, AWS Config rule s3-bucket-server-side-encryption-enabled, SCP denying unencrypted PutObject, KMS customer-managed keys with rotation enabled. | Cloud Platform Engineering Lead (Fictional Role) |
| CLOUD-014 | MEDIUM | 31-60 Days | S3 Versioning, S3 Object Lock for compliance-retention data, AWS Config rule s3-bucket-versioning-enabled, MFA Delete for the highest-sensitivity buckets. | Cloud Platform Engineering Lead (Fictional Role) |
| CLOUD-015 | MEDIUM | 31-60 Days | CloudWatch metric filters and alarms per the CIS AWS Foundations-aligned baseline, EventBridge rules for GuardDuty/Security Hub finding routing, a defined on-call runbook. | Security Operations Lead (Fictional Role) |
| CLOUD-016 | MEDIUM | 31-60 Days | Scoped security-group egress rules, VPC endpoints (S3, DynamoDB, Secrets Manager, KMS), AWS Network Firewall or a NAT Gateway with flow-log-based anomaly detection, GuardDuty for anomalous outbound traffic detection. | Network Engineering Lead (Fictional Role) |
| CLOUD-017 | MEDIUM | 31-60 Days | IAM credential reports reviewed on a recurring schedule, automated inactive-credential deactivation (e.g., via a scheduled Lambda function), HR-system-to-IAM offboarding integration, AWS IAM Identity Center for centralized lifecycle management. | IAM & Identity Lead (Fictional Role) |
| CLOUD-018 | MEDIUM | 31-60 Days | IAM Access Analyzer unused-access findings, automated key-age alerting, IAM Identity Center federated access, scheduled key-rotation enforcement via a Config rule (access-keys-rotated). | IAM & Identity Lead (Fictional Role) |
| CLOUD-019 | MEDIUM | 31-60 Days | AWS Backup Vault Lock, cross-region/cross-account backup copies, scheduled restore testing, AWS Config rule backup-plan-min-frequency-and-min-retention-check. | Data Engineering Lead (Fictional Role) |
| CLOUD-023 | MEDIUM | 31-60 Days | Scoped KMS key policies, kms:ViaService and kms:CallerAccount conditions, separate KMS keys per data-sensitivity tier, CloudTrail monitoring of kms:Decrypt calls for anomaly detection. | Cloud Security Engineering Lead (Fictional Role) |
| CLOUD-024 | MEDIUM | 31-60 Days | ECS task definition capability minimization, AWS Config custom rule flagging added Linux capabilities, runtime-security tooling (e.g., GuardDuty ECS Runtime Monitoring) for anomalous syscall detection. | Platform Engineering Lead (Fictional Role) |
| CLOUD-020 | MEDIUM | 61-90 Days | Lambda Power Tuning for right-sizing, AWS Budgets cost-anomaly alerts, reserved concurrency limits to cap invocation-based cost abuse. | Serverless Engineering Lead (Fictional Role) |
| CLOUD-021 | MEDIUM | 61-90 Days | EC2 Image Builder with signing, SBOM generation (e.g., via Syft or EC2 Image Builder components), AWS Config custom rule for AMI provenance verification. | Platform Engineering Lead (Fictional Role) |
| CLOUD-022 | MEDIUM | 61-90 Days | CloudFront Response Headers Policy, AWS WAF managed rule groups for common web exploits, periodic header-configuration testing as part of routine change validation. | Web Platform Engineering Lead (Fictional Role) |
| CLOUD-025 | LOW | 61-90 Days | AWS Config required-tags rule, SCP-level tag enforcement on resource creation, Terraform-enforced tagging module reused across all fictional provisioning. | Cloud Platform Engineering Lead (Fictional Role) |
| CLOUD-026 | LOW | 61-90 Days | S3 Server Access Logging, AWS Config rule s3-bucket-logging-enabled. | Cloud Platform Engineering Lead (Fictional Role) |
| CLOUD-027 | LOW | 61-90 Days | VPC Flow Logs (all traffic), centralized log aggregation, AWS Config rule vpc-flow-logs-enabled applied organization-wide via Config aggregator. | Network Engineering Lead (Fictional Role) |
| CLOUD-028 | LOW | 61-90 Days | Route 53 query logging, CloudWatch Logs Insights for DNS query analysis. | Network Engineering Lead (Fictional Role) |
| CLOUD-029 | LOW | 61-90 Days | EventBridge GuardDuty finding routing, SNS/chat-ops integration, Security Hub as a finding-aggregation layer across GuardDuty/Config/Inspector. | Security Operations Lead (Fictional Role) |
| CLOUD-030 | LOW | 61-90 Days | AWS WAF rate-based rules, AWS WAF Bot Control for the login/authentication surface, API Gateway usage plans and per-client throttling. | Web Platform Engineering Lead (Fictional Role) |
| CLOUD-031 | LOW | 61-90 Days | ECR lifecycle policies, ECR repository scanning on a recurring schedule for retained images, tagging immutability for production-promoted images. | DevOps / Platform Engineering Lead (Fictional Role) |
| CLOUD-032 | LOW | 61-90 Days | Secrets Manager automatic rotation, RDS IAM database authentication as a longer-term alternative to password-based credentials entirely. | Data Engineering Lead (Fictional Role) |
| CLOUD-033 | LOW | 61-90 Days | Scoped SNS topic policies, IAM Access Analyzer external/cross-principal access review, CloudTrail alerting on sns:Publish calls from unexpected principals. | Platform Engineering Lead (Fictional Role) |
| CLOUD-034 | INFORMATIONAL | N/A — Informational | Security Hub central configuration across the AWS Organization, consistent standard subscription policy. | Security Operations Lead (Fictional Role) |
| CLOUD-035 | INFORMATIONAL | N/A — Informational | Documented naming-convention standard, Terraform module enforcement, periodic naming-compliance audit. | Cloud Platform Engineering Lead (Fictional Role) |
| CLOUD-036 | INFORMATIONAL | N/A — Informational | Scheduled Terraform drift detection, change-management policy requiring all infrastructure changes to flow through IaC, break-glass process with mandatory reconciliation follow-up. | Platform Engineering Lead (Fictional Role) |
| CLOUD-037 | INFORMATIONAL | N/A — Informational | EventBridge DLQ configuration, CloudWatch alarm on DLQ depth, scheduled DLQ review process. | Serverless Engineering Lead (Fictional Role) |
| CLOUD-038 | INFORMATIONAL | N/A — Informational | SSM Agent baked into all golden AMIs, Patch Manager maintenance windows, scheduled compliance-dashboard review. | Platform Engineering Lead (Fictional Role) |
| CLOUD-039 | INFORMATIONAL | N/A — Informational | API Gateway request-validation models, AWS WAF as a complementary edge-layer control. | Web Platform Engineering Lead (Fictional Role) |
| CLOUD-040 | INFORMATIONAL | N/A — Informational | CloudWatch Logs retention policy, AWS Config rule cw-loggroup-retention-period-check, log-scrubbing for sensitive data in debug output. | Security Operations Lead (Fictional Role) |
| CLOUD-041 | INFORMATIONAL | N/A — Informational | AWS Control Tower or equivalent account-baseline automation, Organization-wide SCPs, consistent Security Hub/Config/GuardDuty enablement across all accounts. | Cloud Security Engineering Lead (Fictional Role) |
| CLOUD-042 | INFORMATIONAL | N/A — Informational | Cloud-specific incident-response playbooks, scheduled tabletop exercises, AWS Incident Detection and Response engagement or equivalent for critical accounts. | Security Operations Lead (Fictional Role) |

# APPENDIX O — TEST ACCOUNTS / ASSUMED ROLES

The register below documents every fictional identity, assumed role, or access level TMG Security used across the four assessment perspectives defined in Section 10, so that the origin of every piece of evidence in this report can be traced to a specific, scoped fictional access level. TMG Security did not have unrestricted AWS Management Console access at any point during this engagement.

| Fictional Identity | Perspective | Access Level | Purpose |
|---|---|---|---|
| `(Unauthenticated)` | External Black-Box | None | Internet-facing reconnaissance: DNS/certificate-transparency enumeration, public S3/CloudFront/API discovery, external port scanning |
| `astravault-tmg-config-review-role (fictional, cross-account)` | Authenticated Cloud Configuration Review | SecurityAudit + ViewOnlyAccess (AWS managed, read-only) | Full-environment configuration review across all three fictional accounts via a scoped, time-limited cross-account role |
| `astravault-app-readonly` | Limited-Privilege IAM Assessment | ReadOnlyAccess (AWS managed), Production account only | Simulates a standard low-privilege application/engineering identity to test IAM privilege-escalation and cross-account trust paths |
| `(EC2 instance profile: astravault-ec2-app-role, obtained via IMDS)` | Assumed-Breach / Compromised Workload | As granted to the astravault-ec2-app-role instance profile | Simulates an attacker who has obtained live workload credentials via the CLOUD-003 SSRF-to-metadata technique, used only to confirm reachable resources — no destructive action taken |
| `(ECS task role, obtained via task metadata endpoint, fictional)` | Assumed-Breach / Compromised Workload | As granted to the compromised container's task role | Simulates an attacker who has achieved container-level code execution, used only to confirm the Attack Chain 3 (AP-03) resource-modification path was reachable in principle — no destructive action taken |

# APPENDIX P — METHODOLOGY NOTES

This appendix supplements the methodology described in Section 14 with implementation-level notes on how TMG Security's fictional cloud testing methodology was applied in this specific illustrative engagement.

## Framework Alignment

This fictional assessment's structure draws general inspiration from publicly available cloud security guidance, including the AWS Well-Architected Framework's Security Pillar, the Center for Internet Security (CIS) AWS Foundations Benchmark, and the MITRE ATT&CK Cloud Matrix for attack-path construction. TMG Security is not affiliated with, endorsed by, reviewed by, or certified by AWS, CIS, MITRE, or any other standards organization, and this report does not represent an official benchmark audit or certification against any of these frameworks.

## Tooling Notes (Illustrative)

- Fictional use of AWS CLI and IAM policy simulation for authenticated configuration review and privilege-escalation-pattern validation.
- Fictional use of open-source cloud security posture tooling categories (IAM graph analysis, public-exposure scanning, container image scanning) representative of tooling TMG Security uses in real client engagements — no real tool output is reproduced in this sample report.
- Fictional manual verification of every automated finding before inclusion, consistent with TMG Security's standard false-positive-elimination practice across its full report family.

## Severity Scoring Notes

Severity scores in this report are illustrative and CVSS v3.1-inspired only; they are not submitted to, registered with, or certified by any official CVSS scoring authority, and the specific numeric values are provided for illustrative prioritization purposes within this sample report only.

## Sample Report Notice

As with every section of this report, the methodology notes above describe a fictional, illustrative engagement structure created to demonstrate TMG Security's cloud penetration testing reporting standard. No real AWS environment, client, or engagement is described.

# APPENDIX Q — FINAL DISCLAIMER & CONTACT

**FICTIONAL SAMPLE REPORT — NOT AN ACTUAL CLIENT CLOUD PENETRATION TEST**

AstraVault Technologies, Inc. is a fictional company created for demonstration purposes. Every AWS account ID, IAM identity, S3 bucket, EC2 instance, Lambda function, IP address, domain, credential, configuration, finding, and item of evidence in this report is synthetic and was constructed solely to illustrate TMG Security's cloud penetration testing report format and depth.

No real AWS environment was accessed, scanned, or tested in the production of this report. No real credentials were used or generated. No real cloud resource was queried, modified, or exploited. All evidence panels, screenshots, and diagrams are illustrative mock-ups labeled as such.

TMG Security is not affiliated with, endorsed by, reviewed by, or certified by Amazon Web Services (AWS), the Center for Internet Security (CIS), MITRE, NIST, PTES, OWASP, or any other standards organization referenced anywhere in this report, unless explicitly stated otherwise. References to AWS service names are used descriptively to identify the fictional cloud platform this sample report is modeled on and do not imply any partnership, certification, or endorsement.

This document, including its fictional company name, findings, evidence, and narrative, is the intellectual property of TMG Security and is provided for demonstration and portfolio purposes only. It may not be represented as an actual client deliverable.

## Contact

For questions about this sample report, or to discuss a real cloud penetration testing engagement, contact TMG Security through the channels listed on the TMG Security website's Security Assessment Reports library, where this sample report is published alongside TMG Security's Web Application, API, Android, iOS, and Network sample reports.

**— End of Report —**