Vantalect

AWS SECURITY, COMPLIANCE & GOVERNANCE · SAMPLE DELIVERABLE

Identity Rebuilt

Week 4 of 6: Identity · prepared for StrataForge Robotics, Inc.

Prepared by Charles Sieg · charles.sieg@vantalect.com · vantalect.com

August 7, 2026 · AWS Security, Compliance & Governance Audit: The Hardening (six weeks)

SAMPLE DELIVERABLE

This is a sample deliverable. StrataForge Robotics, Inc. is a fictional company: every account, resource, finding, figure, and person in this report is simulated to show the form and depth of a Vantalect engagement. AWS prices are list prices from the AWS Price List API on October 13, 2026.

Contents

Summary **2**

People: everyone through Identity Center **4**

The policy that keeps it closed 5

Access keys: from 1,139 to 404 ..... **6**

Keeping the count down 7

Root and break-glass: one root, two sealed identities, every sign-in alarmed **7**

The calls 8

The management root, break-glass, and the alarms 8

Per account 9

Roles cut to what each workload uses **12**

The replacement policies 14

Escalation paths: 12 of 12 removed **15**

The boundary and the policy 17

Cross-account and OIDC trusts **19**

The trust policies 22

What is not finished **23**

Pull request ledger **25**

Summary

In week 4 we moved every person at StrataForge who touches AWS onto IAM Identity Center and Okta, and took the human IAM users, the CI users, and every member account's root credentials out of the organization. **Identity Center coverage went from 70.1% to 100.0%** of the 1,420 people with access. IAM users fell from 1,394 to 403, and **none of the 544 human IAM users or 247 CI users remain**. Console users without MFA fell from 108 to 0. Access keys older than 90 days fell from 1,009 to 57.

7 findings closed (10 pull requests, merged August 3, 2026 through August 7, 2026), and 2 more went partway. The ones that matter most for the guarantee are already behind us: 9 of the 10 critical findings were closed in week 3, and the remaining one waits on the acceptance in that report. This week also closed **12 of the 12 privilege-escalation paths** and the remaining **3 of the 7 risky cross-account and federation trusts** the Audit found (the other 4, the GitHub trusts, closed

in week 3). What this week removes is the way an attacker turns a foothold in one account into control of many: long-lived keys that never expire, root passwords that every member account carried, roles that can write their own permissions, and trust relationships with accounts StrataForge no longer controls.

Two things are not finished, and both have an owner and a date: the 57 remaining keys older than 90 days belong to factory gateways that can only change in their maintenance windows (`SF-SEC-012`), and the Lambda functions in `fleet-dev` still have secret-named environment variables that need to become Secrets Manager references (`SF-SEC-015`). They are in What is not finished.

Where the code is. This report says what changed, why, and how we proved it. The complete code for each fix (policies, CDK, commands, and rollback) is in the Audit Report Appendix, the sample report `strataforge-security-2-appendix`, under the same finding ID. Each section below names the ID to open.

The week-3 and week-4 identity posture, from the credential reports of the 48 accounts and Identity Center.

Identity, before and after	Week 3	Week 4	Change
People with AWS access	1,468	1,420	−48
In IAM Identity Center	1,029	1,420	+391
Identity Center coverage	70.1%	100.0%	+29.9 points
Permission sets	23	31	+8
IAM users	1,394	403	−991
Console users	553	11	−542
Console users without MFA	108	0	−108
Active access keys	1,139	404	−735
Keys older than 90 days	1,009	57	−952
Keys never used	161	7	−154
Accounts with root credentials	48	1	−47
Accounts with a root access key	3	0	−3
Accounts alarmed on root and break-glass sign-in	0	48	+48
Legacy Okta SAML accounts outside Identity Center	6	0	−6

What the sponsor needs to know

Marcus Oyelaran: nothing needs your decision this week. One thing deserves your attention. 48 of the 439 people who had only an IAM user no longer work at StrataForge. Their IAM users still existed. We deleted those users without replacement, and this is how we found them: by matching every IAM user to the HR roster. Elena Vasquez's team can repeat that match every month, and People: everyone through Identity Center says how. An IAM user nobody can match to a person is the signal.

People: everyone through Identity Center

Before this week, 439 people reached AWS only through an IAM user that someone created for them. Some had a console password with no MFA, some had access keys on a laptop, and nobody could say with confidence who still worked at the company. Now every person is a user in Identity Center, federated from Okta, with multi-factor sign-in, a session that expires, and access granted through permission sets that name the job.

How many people reach AWS through Identity Center, before and after week 4.

Measure	Before	After
People with AWS access	1,468	1,420
In Identity Center	1,029	1,420
With an IAM user only	439	0
Human IAM users	544	0
Permission sets	23	31
Legacy Okta SAML apps granting administrator in accounts outside Identity Center	6	0

How we did it, in the order we did it:

1. **Match every IAM user to a person.** For the 439 people with only an IAM user, we matched the user to the HR roster with Grace Liu's team, account by account. The two administrators in the management account were matched first.
2. **Add the people who still work here.** 391 people were added to Identity Center with the permission set their job needs. 48 were no longer employed, and no user was created for them.

3. **Cut 8 new permission sets** for the groups the existing 23 did not cover: Kestrel engineers, research, SAP Basis, and vendors.
4. **Remove the old paths in.** Delete the human IAM users, remove the 6 legacy Okta SAML apps that granted administrator outside Identity Center, and keep new ones from appearing: the `NoNewIamUsers` statement of `sf-protect-account-defaults`, attached in week 3, denies `iam:CreateUser`, `iam:CreateAccessKey`, and `iam:CreateLoginProfile` to everyone but the `PlatformIdentityAdmin` role.

The policy that keeps it closed

The `sf-protect-account-defaults` statement named above is what keeps the door closed: only the platform identity role can create a console password. The Identity Center assignment commands and the policy text are in the Audit Report Appendix under SF-SEC-013: "Find users with a password and no MFA from the credential report"; "Assign an Okta group to a permission set, then delete the login profile"; "sf-protect-account-defaults, statement NoNewIamUsers: console passwords only through the identity administrator role (the whole policy is in the guardrail SCPs)". Verify it with the three checks in the register: a new credential report shows no user with a password and no MFA (Security Hub CSPM IAM.5); `iam:CreateLoginProfile` from an administrator who is not the platform identity role is denied; and the Identity Center sign-in log shows each moved person arriving through Okta.

Rollback. Revert the pull request and the permission sets return to the previous assignments. We would not restore any person's IAM user. If someone lacks access they need, change their assignment in `identity/permission-sets.yaml`, with a reviewer; the sign-in log shows who tried and failed.

Access keys: from 1,139 to 404

A long-lived access key is a password that never expires, never asks for MFA, and gets copied. Before this week StrataForge had 1,139 active keys, 1,009 of them older than 90 days and 161 that had never been used at all. Now there are 404, 57 of them older than 90 days, and 7 unused.

Active keys by age and by kind of owner. After week 4 no human or CI user holds a key, so those two columns are zero and are left out of the after columns.

Age of key	Human, before	CI, before	Service, before	Vendor, before	Total, before	Total, after	Service, after	Vendor, after
Under 90 days	28	26	62	14	130	347	303	44
90 days to 1 year	38	16	30	3	87	3	3	0
1–2 years	68	53	102	20	243	13	13	0
2–3 years	67	67	121	24	279	16	16	0
Over 3 years	59	93	193	55	400	25	25	0

What happened to each kind of key:

- **Human keys (260) were deleted with their users.** People use Identity Center sessions and `aws sso login` for the command line.
- **CI keys (255) were replaced by GitHub OIDC deploy roles.** `fleet-infra#4362` adds the roles; the CI users were then deleted. No CI user holds a key now.
- **Unused keys were deleted.** A key with no use in 45 days was removed, which clears IAM.22 and most of the 161 that had never been used.
- **Service and vendor keys in use were rotated** the day we touched them, which is why the count under 90 days rose from 130 to 347. Each now has an owner tag, a creation date, and an entry in the retirement register.
- **The factory gateway keys were left, on purpose.** The remaining 57 keys older than 90 days belong to gateway users that authenticate machines on the factory floor, and changing them mid-shift stops a production line. They move to IAM Roles Anywhere during factory maintenance windows. Owner and date are in What is not finished.

Keeping the count down

The triage we ran against the credential report in every account is: deactivate first, wait, then delete. The `NoNewIamUsers` statement in `sf-protect-account-defaults` and a Config rule that reports keys older than 90 days are what stop the count from growing back. The identity pipeline's own `PlatformIdentityAdmin` role is the one place a new user or key may still be created. The commands, the policy, and the Config rule are in the Audit Report Appendix under SF-SEC-012: "Triage keys by age and last use from the credential report"; "Deactivate the exact key, delete it after the hold, and save then detach user policies"; "sf-protect-account-defaults, statement NoNewIamUsers: no new IAM users or access keys outside the identity administrator role (the whole policy is in the guardrail SCPs)"; "Config rule that reports keys older than 90 days".

Root and break-glass: one root, two sealed identities, every sign-in alarmed

Before this week all 48 accounts had a root password, 40 protected only by a virtual MFA app on someone's phone and 7 with no MFA at all, and 3 had a root access key. Now **47 member accounts have no root credentials at all**, and the management account's root has a hardware key kept at two sites. Nothing needs the member accounts' root users any more. When one is needed, it can be recovered on demand through a short, logged root session, which pages the security team.

The root posture of the 48 accounts, before and after week 4.

Root and break-glass	Before	After
Accounts with a root password	48	1
Accounts with a root access key	3	0
Root users with no MFA	7	0
Root users on virtual MFA	40	0
Member accounts with root credentials deleted	0	47
Accounts alarmed on root or break-glass sign-in	0	48

The calls

Centralized root access management does the work, run from the management account: enable the service access and the two root-management features, then, one member account at a time, start a short root session with the `IAMDeleteRootUserCredentials` task policy against a Regional STS endpoint (the global endpoint does not support `sts:AssumeRoot`) and delete the credentials. An audit session with the `IAMAuditRootUserCredentials` task policy comes first, so the deletion is never a surprise. The commands, in order, are in the Audit Report Appendix under SF-SEC-010 and SF-SEC-029: “Enable centralized root access management (management account)”; “Audit, then delete one member account’s root credentials”; “Page on every root session started through sts:AssumeRoot”.

The management root, break-glass, and the alarms

The management account keeps its root user, because an organization needs one. It has a hardware FIDO2 key registered, the virtual device removed, and a second key at the second site. The two break-glass identities, `breakglass-01` and `breakglass-02`, have hardware MFA in a sealed envelope, and every sign-in by either of them, by the management root, or by any `sts:AssumeRoot` call raises a page. The test that proves it runs every quarter: someone signs in as `breakglass-01`, and the page time goes in the test log. `SF-SEC-040` had last been tested in April 2025.

The alarm template, the event pattern that pages on every sign-in, and the command that deploys it to all 17 Regions are in the Audit Report Appendix under SF-SEC-040: “Alarm template for the stack set (one instance per Region)”; “The same pattern, for a test with `aws events test-event-pattern`”; “Deploy to all 17 Regions from the management account”.

Per account

Root credentials and alarms in every account, before to after. The management account keeps its root, with a hardware key.

Account	OU	Root password	Root MFA	Root keys	Last root sign-in	Credentials deleted	Alarmed
sf-management	Root	yes to yes	hardware to hardware	0 to 0	February 11, 2026	no	no to yes
sf-log-archive	Security	yes to no	virtual to none	0 to 0	January 29, 2025	yes	no to yes
sf-security-tooling	Security	yes to no	virtual to none	0 to 0	never recorded	yes	no to yes
sf-forensics	Security	yes to no	virtual to none	0 to 0	May 7, 2024	yes	no to yes
sf-network-hub	Infrastructure	yes to no	virtual to none	0 to 0	never recorded	yes	no to yes
sf-shared-services	Infrastructure	yes to no	virtual to none	0 to 0	never recorded	yes	no to yes
sf-identity	Infrastructure	yes to no	virtual to none	0 to 0	never recorded	yes	no to yes
sf-cicd	Infrastructure	yes to no	virtual to none	0 to 0	October 18, 2024	yes	no to yes
sf-backup	Infrastructure	yes to no	virtual to none	0 to 0	never recorded	yes	no to yes
fleet-ingest-prod	Workloads/Fleet/Prod	yes to no	virtual to none	0 to 0	July 1, 2024	yes	no to yes
fleet-api-prod	Workloads/Fleet/Prod	yes to no	virtual to none	0 to 0	never recorded	yes	no to yes
fleet-data-prod	Workloads/Fleet/Prod	yes to no	virtual to none	0 to 0	August 13, 2025	yes	no to yes
fleet-analytics-prod	Workloads/Fleet/Prod	yes to no	virtual to none	0 to 0	never recorded	yes	no to yes
fleet-ota-prod	Workloads/Fleet/Prod	yes to no	virtual to none	0 to 0	never recorded	yes	no to yes
fleet-ota-staging	Workloads/Fleet/NonProd	yes to no	virtual to none	0 to 0	August 29, 2025	yes	no to yes
fleet-remote-ops-prod	Workloads/Fleet/Prod	yes to no	virtual to none	0 to 0	never recorded	yes	no to yes
fleet-eu-prod	Workloads/Fleet/Prod	yes to no	virtual to none	0 to 0	February 7, 2024	yes	no to yes
fleet-jp-prod	Workloads/Fleet/Prod	yes to no	virtual to none	0 to 0	never recorded	yes	no to yes

Account	OU	Root password	Root MFA	Root keysLast root sign-in	Credentials deleted	Alarmed
customer-portal-prod	Workloads/Fleet/Prod	yes to no	virtual to none	0 to 0January 30, 2026	yes	no to yes
digital-twin-prod	Workloads/Fleet/Prod	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
fleet-staging	Workloads/Fleet/NonProd	yes to no	virtual to none	0 to 0December 5, 2025	yes	no to yes
fleet-dev	Workloads/Fleet/NonProd	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
fleet-perf	Workloads/Fleet/NonProd	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
fleet-eu-staging	Workloads/Fleet/NonProd	yes to no	virtual to none	0 to 0November 29, 2025	yes	no to yes
customer-portal-dev	Workloads/Fleet/NonProd	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
digital-twin-dev	Workloads/Fleet/NonProd	yes to no	virtual to none	0 to 0December 16, 2025	yes	no to yes
crucible-sim-prod	Workloads/Crucible	yes to no	virtual to none	0 to 0December 8, 2025	yes	no to yes
crucible-training-prod	Workloads/Crucible	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
crucible-data	Workloads/Crucible	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
crucible-research	Workloads/Crucible	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
crucible-dev	Workloads/Crucible	yes to no	virtual to none	0 to 0August 29, 2025	yes	no to yes
crucible-models	Workloads/Crucible	yes to no	virtual to none	0 to 0August 12, 2025	yes	no to yes
erp-sap-prod	Workloads/Enterprise	yes to no	virtual to none	0 to 0September 26, 2024	yes	no to yes
erp-sap-nonprod	Workloads/Enterprise	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
plm-prod	Workloads/Enterprise	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
mfg-mes-prod	Workloads/Enterprise	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
corp-data-warehouse	Workloads/Enterprise	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
corp-apps	Workloads/Enterprise	yes to no	virtual to none	0 to 0never recorded	yes	no to yes

Account	OU	Root password	Root MFA	Root keysLast root sign-in	Credentials deleted	Alarmed
kestrel-prod	Workloads/Kestrel	yes to no	none to none	1 to 0never recorded	yes	no to yes
kestrel-staging	Workloads/Kestrel	yes to no	none to none	0 to 0never recorded	yes	no to yes
kestrel-dev	Workloads/Kestrel	yes to no	none to none	0 to 0never recorded	yes	no to yes
kestrel-data	Workloads/Kestrel	yes to no	none to none	1 to 0never recorded	yes	no to yes
kestrel-legacy-mgmt	Workloads/Kestrel	yes to no	none to none	1 to 0December 8, 2025	yes	no to yes
sandbox-platform	Sandbox	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
sandbox-robotics	Sandbox	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
sandbox-ml	Sandbox	yes to no	virtual to none	0 to 0never recorded	yes	no to yes
sandbox-hackathon	Sandbox	yes to no	none to none	0 to 0December 5, 2024	yes	no to yes
suspended-legacy-ci	Suspended	yes to no	none to none	0 to 0January 29, 2026	yes	no to yes

Rollback. Deleted root credentials cannot be restored, and nothing needs them. A member account that someone must open as root can be recovered on demand: lift the member-root deny for that account, start a root session with `aws sts assume-root` and the `IAMCreateRootUserPassword` task policy, and complete the reset. Starting a root session pages the security team. Disabling the alarm rule is the only rollback and we do not recommend it. If the page is too noisy, tighten the pattern to the named identities; do not mute it.

Roles cut to what each workload uses

The Audit's SF-SEC-017: 14 customer managed policies grant Action * on Resource * to 17 workload roles and 46 IAM users, across 14 accounts. The EKS node role `fleet-api-node-role` was the sharpest case: it is attached to every `fleet-api` node, and those nodes still allowed IMDSv1 with a hop limit of 2, so a server-side request forgery bug in any pod could have returned administrator credentials. We replaced each with a policy generated from 90 days of CloudTrail by IAM Access Analyzer, and capped each role with the one `sf-workload-boundary` permissions boundary, which allows every service action and takes away only identity and organization actions.

The wildcard-administrator policies replaced in week 4, and who carried them.

Policy	Account	Roles it was attached to
`fleet-api-node-admin`	fleet-api-prod	`fleet-api-node-role`, `fleet-api-batch-role`
`ingest-ops-admin`	fleet-ingest-prod	`telemetry-replay-role`
`crucible-batch-admin`	crucible-sim-prod	`crucible-sim-batch-role`, `crucible-sim-node-role`
`research-batch-admin`	crucible-research	`crucible-batch-admin-profile`
`sap-backup-admin`	erp-sap-prod	`sap-backup-role`; 1 IAM users
`mes-integration-admin`	mfg-mes-prod	`mes-gateway-role`
`kestrel-api-admin`	kestrel-prod	`perception-api-node-role`, `perception-api-task-role`; 9 IAM users
`kestrel-ci-admin`	kestrel-legacy-mgmt	`jenkins-agent-role`; 14 IAM users
`kestrel-data-admin`	kestrel-data	`dataset-loader-role`; 4 IAM users
`jenkins-deployer`	suspended-legacy-ci	`jenkins-controller-role`; 11 IAM users
`platform-automation`	sf-shared-services	`platform-automation-role`
`hackathon-admin`	sandbox-hackathon	`hackathon-ec2-role`; 7 IAM users
`dev-environments-admin`	fleet-dev	`fleet-dev-admin-exec`
`sandbox-power`	sandbox-robotics	`sandbox-robotics-exec`

The replacement policies

The generated policy for `fleet-api-node-role`, the scoped node role with its permissions boundary, and the `sf-workload-boundary` document itself (the same one as in the escalation-path entry below) are in the Audit Report Appendix under SF-SEC-017: “Generate a policy from 90 days of CloudTrail for fleet-api-node-role”; “Scoped node role with a permissions boundary, imported rather than created”; “Import the existing role, deploy, and prove the node role still works in non-production”; “sf-workload-boundary: the one document, shared with the privilege-escalation entry”.

How to verify: Security Hub CSPM IAM.1 and IAM.21 show no failed resource in `fleet-api-prod`; the service ran a full release cycle with no new AccessDenied in CloudTrail; and `aws iam get-role --role-name fleet-api-node-role` shows a `PermissionsBoundary`. **Rollback.** If a workload fails with AccessDenied after its policy was replaced, read the denied action from CloudTrail and add that action, on that resource, to the role in the CDK app. Do not reattach the old `Action: *` policy: it is attached to nothing now, and the pull request that detached it is the record.

The Access Analyzer-generated policy is a starting point, not a finished design. It reflects what the role did in 90 days, so an action used once a quarter, such as a disaster-recovery job, may be missing. We listed the quarterly jobs with each owner before the change. If a role you own fails with AccessDenied on a rare action, add that action to the CDK app; it is a one-line pull request.

The roles that nobody uses are a separate problem. 1,922 of the 5,730 workload roles have not been used in 90 days, and 460 have never been used. They carry no key and cannot be signed into, so they are a cleanup, not a risk, and `SF-SEC-048` holds them in the backlog with an owner and a date.

Escalation paths: 12 of 12 removed

An escalation path is a way for a role to give itself more power than it was granted: pass an administrator role to a new function, edit its own policy, or rewrite a trust policy. The Audit found 12, each leading to administrator in a different account. We closed 12 of them with one permissions boundary that denies those edits, a scoped `iam:PassRole`, and a service control policy that requires the boundary on every new role and stops anyone but the platform roles from editing it. The people behind Identity Center permission sets cannot take a boundary through `put-role-permissions-boundary` (their `AWSReservedSSO_` roles reject it), so their permission sets carry it as a permission-set boundary; none remains.

The 12 privilege-escalation paths found in the Audit; 12 are closed by pull request `platform-landing-zone#667`.

ID	Technique	Who can use it	WhoAccount	What it reaches	Fixed in week 4
PE-01	`iam:PassRole + lambda:CreateFunction + lambda:InvokeFunction`	FleetDeveloper permission set	212fleet-dev	fleet-dev-admin-exec (Action * on *)	yes
PE-02	`iam:PassRole + lambda:CreateFunction + lambda:InvokeFunction`	FleetDeveloper permission set	212fleet-staging	cdk-hnb659fds-cfn-exec-role (AdministratorAccess)	yes
PE-03	`iam:PassRole + ec2:RunInstances`	crucible-research IAM users	58crucible-research	crucible-batch-admin-profile (Action * on *, read on crucible-data)	yes
PE-04	`iam:CreatePolicyVersion`	jenkins-deployer user	1suspended-legacy-ci	Account administrator	yes
PE-05	`iam:PutUserPolicy on self, then sts:AssumeRole into Kestrel accounts`	Kestrel CI users	34kestrel-legacy-mgmt	OrganizationAccountAccessRole in four Kestrel accounts (AdministratorAccess)	yes
PE-06	`iam:UpdateAssumeRolePolicy`	platform-automation-role	1sf-shared-services	Any role in sf-shared-services, including OrganizationAccountAccessRole	yes
PE-07	`iam:CreateAccessKey on other users`	sap-basis-admins group	9erp-sap-prod	sap-backup-admin user (Action * on *)	yes
PE-08	`cloudformation:CreateStack + iam:PassRole`	FleetDeveloper permission set	212fleet-staging	cdk-hnb659fds-cfn-exec-role (AdministratorAccess)	yes

ID	Technique	Who can use it	WhoAccount	What it reaches	Fixed in week 4
PE-09	`ssm:SendCommand to admin-profiled nodes`	FleetOnCall permission set	64fleet-api-prod	fleet-api-node-role (Action * on *)	yes
PE-10	`codebuild:CreateProject + iam:PassRole`	ci-maintainers group	7sf-cicd	ci-admin-service-role (AdministratorAccess)	yes
PE-11	`glue:CreateJob + iam:PassRole`	DataAnalyst permission set	38fleet-data-prod	glue-etl-admin (Action * on *)	yes
PE-12	`iam:AttachUserPolicy on self`	hackathon IAM users	26sandbox-hackathon	Account administrator	yes

The boundary and the policy

The one boundary `sf-workload-boundary` (every service action except identity and organization actions, a scoped `iam:PassRole`, no self-escalation, and a deny on any change to the boundary itself), the service control policy that requires it on new roles and users, and the commands that attach it are in the Audit Report Appendix under SF-SEC-018: “sf-workload-boundary: allow everything except identity and organization actions, scoped PassRole, no self-escalation, the boundary protected”; “The boundary as a stack set template (the same text goes in a stack in sf-management)”; “sf-require-workload-boundary: new roles and users must carry the boundary, and only the platform roles may edit it”; “Publish the boundary, attach it, and prove the result”. The boundary is the part to read:

sf-workload-boundary: allow everything except identity and organization actions, scoped PassRole, no self-escalation, the boundary protected

```JSON
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowEverythingExceptIdentityAndOrganization",
      "Effect": "Allow",
      "NotAction": [
        "iam:*",
        "organizations:*",
        "account:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "PassOnlyWorkloadRolesToWorkloadServices",
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::*:role/sf-workload-*",
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": [
            "lambda.amazonaws.com",
            "ecs-tasks.amazonaws.com",
            "states.amazonaws.com",
            "sagemaker.amazonaws.com",
            "iot.amazonaws.com",
            "events.amazonaws.com",
            "scheduler.amazonaws.com",
            "firehose.amazonaws.com"
          ]
        }
      }
    },
    {
```

```
      "Sid": "DenyEscalation",
      "Effect": "Deny",
      "Action": [
        "iam:CreatePolicyVersion",
        "iam:SetDefaultPolicyVersion",
        "iam:PutUserPolicy",
        "iam:PutRolePolicy",
        "iam:PutGroupPolicy",
        "iam:AttachUserPolicy",
        "iam:AttachRolePolicy",
        "iam:AttachGroupPolicy",
        "iam:UpdateAssumeRolePolicy",
        "iam:PutRolePermissionsBoundary",
        "iam:PutUserPermissionsBoundary",
        "iam:DeleteRolePermissionsBoundary",
        "iam:DeleteUserPermissionsBoundary"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyChangesToTheBoundaryPolicy",
      "Effect": "Deny",
      "Action": [
        "iam:CreatePolicyVersion",
        "iam:SetDefaultPolicyVersion",
        "iam:DeletePolicy",
        "iam:DeletePolicyVersion"
      ],
      "Resource": "arn:aws:iam::*:policy/sf-workload-boundary"
    }
  ]
}
```

We proved each path closed by asking IAM. `aws iam simulate-principal-policy` returns `implicitDeny` or `explicitDeny` for every escalation action on each flagged role, an `aws iam create-role` without the boundary is denied by the service control policy, and the same call with the boundary succeeds for the platform workflow. **Rollback.** Revert the pull request to lift the boundary requirement on one role, then fix the specific permission that needed it. The SCP that requires the boundary on new roles is detachable per OU. Detaching it reopens the paths; we do not recommend it.

Cross-account and OIDC trusts

A trust policy decides who may assume a role. We reviewed all 17 trust relationships that cross an account or identity boundary, and found that 7 were findings and the rest were what they should be. We fixed 7 of the 7. Of the 7 GitHub and Kubernetes OIDC trusts, 4 accepted any token from the provider because they checked the audience and not the subject (`sub`), and 4 of those are now pinned to a repository and environment.

Every cross-account and OIDC trust, the assessment from the identity review, and what changed.

Role	Accounts	Trusted principal	External ID	Conditions	Permissions	Assessment	What we did
`kettleworth-mes-integration`	1	Kettleworth Integration, LLC (`<contractor-account-id>`)	no	none	AdministratorAccess	finding	Role deleted (`SF-SEC-020`, `enterprise-infra#911`)
`OrganizationAccountAccessRole (Kestrel, pre-acquisition)`	4	kestrel-legacy-mgmt (…9120), Kestrel's former management account	no	none	AdministratorAccess	finding	Trust re-pointed to the management account (`SF-SEC-019`, `platform-landing-zone#664`)
`gha-artifact-publisher`	1	token.actions.githubusercontent.com	no	aud only (no sub)	ECR push and artifact bucket write for every workload	finding	`sub` pinned to repository and environment in week 3 (`SF-SEC-003`, `fleet-infra#4340`)
`gha-deploy-fleet-api`	1	token.actions.githubusercontent.com	no	aud only (no sub)	EKS and CDK deploy	finding	`sub` pinned to repository and environment in week 3 (`SF-SEC-003`, `fleet-infra#4340`)

Role	Accounts	Trusted principal	External ID	Conditions	Permissions	Assessment	What we did
`gha-deploy-portal`	1	token.actions.githubusercontent.com	no	aud only (no sub)	S3, CloudFront, and Lambda deploy	finding	`sub` pinned to repository and environment in week 3 (`SF-SEC-003`, `fleet-infra#4340`)
`gha-deploy-staging`	1	token.actions.githubusercontent.com	no	aud only (no sub)	PowerUserAccess	finding	`sub` pinned to repository and environment in week 3 (`SF-SEC-003`, `fleet-infra#4340`)
`okta-admin / okta-developer (legacy Okta SAML app)`	6	Okta SAML provider (pre-Identity Center federation)	no	SAML:aud	AdministratorAccess / PowerUserAccess	finding	Federation removed; people use Identity Center (`SF-SEC-013`)
`backup-vendor-restore`	1	backup vendor (external)	yes	sts:ExternalId	AWS Backup restore operator	expected	No change
`observability-vendor-integration`	24	observability SaaS vendor (external)	yes	sts:ExternalId	ReadOnly metrics and logs	expected	No change
`vantalect-audit-readonly`	48	Vantalect, LLC (external)	yes	sts:ExternalId	SecurityAudit + ViewOnlyAccess	expected	No change
`AWSControlTowerExecution`	40	sf-management (…0007)	no	none	AdministratorAccess	expected	No change
`OrganizationAccountAccessRole`	42	sf-management (…0007)	no	none	AdministratorAccess	expected	No change
`cdk-hnb659fds-deploy-role (CDK bootstrap, --trust)`	29	sf-cicd (…1356)	no	none	CDK deploy (passes the CloudFormation execution role)	expected	No change

Role	Accounts	Trusted principal	External ID	Conditions	Permissions	Assessment	What we did
`sf-security-automation`	48	sf-security-tooling (…3374)	no	none	SecurityAudit plus remediation actions	expected	No change
`EKS IAM roles for service accounts (96 roles)`	10	oidc.eks.`<region>`.amazonaws.com (16 cluster providers)	no	aud + sub pinned to namespace and service account	workload roles	expected	No change
`gha-* (33 roles)`	9	token.actions.githubusercontent.com	no	aud + sub pinned to repository and environment	deploy roles	expected	No change
`gha-cache-writer`	1	token.actions.githubusercontent.com	no	sub = repo:strataforge/* (any repository in the organization)	build-cache bucket write	expected	No change; see the note below

Three things to say plainly about the ones we left alone.

- **`gha-cache-writer` is correct, and we would still tighten it.** It trusts `repo:strataforge/*`, so a workflow in any StrataForge repository can write the shared build cache. The cache holds no deployable artifact, which is why it is not a finding. A poisoned cache is still a supply-chain route, so the next time the CDK construct is touched, pin it to the repositories that use the cache.
- **The vendor roles are the model for new ones.** `vantalect-audit-readonly`, `observability-vendor-integration`, and `backup-vendor-restore` each carry an external ID and a narrow policy. Vantalect's own role is read-only, requires its external ID, and is removed at the close of the engagement; the register records the day.
- **`OrganizationAccountAccessRole` and `AWSControlTowerExecution` trust the management account and carry AdministratorAccess.** They have to: Control Tower and Organizations use them. They are protected by the management account's controls, not by the role, which is why the management account has a hardware-key root, alarms on every root and break-glass sign-in, and no human IAM users.

The trust policies

The trust policy for the four Kestrel `OrganizationAccountAccessRole` roles and the command that updates each account are in the Audit Report Appendix under SF-SEC-019: "OrganizationAccountAccessRole trusting the StrataForge management account"; "Update the trust in each Kestrel member account"; "Page when anyone outside the management account assumes the role".

OrganizationAccountAccessRole trusting the StrataForge management account

```JSON
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::<management-account-id>:root"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "Bool": {
          "aws:MultiFactorAuthPresent": "true"
        }
      }
    }
  ]
}
```

The idle-role proof, the vendor role pattern (external ID, scoped policy, one-hour session), and the resulting trust policy are under SF-SEC-020: “Prove the role is idle, then remove it”; “Vendor role pattern: ExternalId, scoped policy, one-hour sessions”; “Resulting trust policy shape for a vendor role”. The vendor role shape is the model for any new third-party access:

Resulting trust policy shape for a vendor role

```JSON
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::<vendor-account-id>:root"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:ExternalId": "<vendor-external-id>"
        }
      }
    }
  ]
}
```

How to verify: the four Kestrel roles show the real management account and no `<kestrel-legacy-mgmt-account-id>`; an assume-role attempt from `kestrel-legacy-mgmt` returns AccessDenied; `aws iam get-role --role-name kettleworth-mes-integration` returns NoSuchEntity in `mfg-mes-prod`; and IAM Access Analyzer shows no external-access finding for a principal outside the organization. **Rollback.** There is no reason to trust the former management account again. If a Kestrel pipeline still needs one of the four accounts, give it an OIDC role through the Kestrel CDK app. A revert would restore the old trust, and we do not recommend it. The role was deleted, not disabled. A future integrator receives a new role with an external ID, a one-hour maximum session, and a scoped policy, approved by the application owner and the security manager. It never receives AdministratorAccess.

What is not finished

2 findings moved but did not close. Both have owners, a reason, and a date.

Week-4 work that did not close its finding.

ID	Severity	Finding	What remains	Owner	Due
SF-SEC-012	High	1,009 active access keys are older than 90 days; 400 are over three years old	Replace the factory gateway users' keys with IAM Roles Anywhere during factory maintenance windows	Grace Liu	Nov 20
SF-SEC-015	High	144 Lambda functions have secret-named environment variables	Move the fleet-dev functions' secrets to Secrets Manager	Sam Okafor	Sep 30

The functions in `SF-SEC-015` outside `fleet-dev` moved to Secrets Manager this week and the exposed values were rotated (`fleet-infra#4367`). The `fleet-dev` functions stay for the owner to finish, because their secrets are shared with a test harness that Sam Okafor's team is still moving. The same pattern applies; the inventory, the secret and function code, and the rotation commands are in the Audit Report Appendix under SF-SEC-015: "List the secret-named variables, without their values"; "Secret, function, and read grant"; "Create the secret from a file, set rotation, and remove the old variable".

Pull request ledger

Author for every pull request: Charles Sieg of Vantalect. Reviewers are the engineers who own the account or the system.

The week-4 pull requests that merged first: repository and number, files, reviewer, merge date, rollout, and the finding each addresses.

Pull request	Title	Files	Reviewer	Merged	Rollout	Finding
`platform-landing-zone#661`	Identity Center: permission sets for Kestrel, research, SAP Basis, and vendors; remove IAM console users	`identity/permission-sets.yaml`; `identity/assignments.yaml`; `policies/scp/protect-account-defaults.json`	Grace Liu	Aug 3	Deployed to sf-management first and verified, then to production.	Closes SF-SEC-013
`platform-landing-zone#662`	Identity: retire CI, human, and unused keys; SCP denying new IAM users and keys	`identity/iam-user-retirement.csv`; `policies/scp/protect-account-defaults.json`; `ops/retire-keys.sh`	Grace Liu	Aug 3	Deployed to sf-management first and verified, then to production.	Addresses SF-SEC-012; not yet closed
`enterprise-infra#911`	MES: delete the former integrator's admin role	`apps/mes/lib/integration-roles.ts`	Mei Chen	Aug 4	Deployed to the account's non-production twin first where one exists, then to production.	Closes SF-SEC-020
`fleet-infra#4362`	CI: replace legacy CI users with GitHub OIDC deploy roles	`apps/ci/lib/deploy-roles.ts`	Grace Liu	Aug 4	Deployed to sf-management first and verified, then to production.	Addresses SF-SEC-012; not yet closed
`platform-landing-zone#664`	Kestrel accounts: re-point OrganizationAccountAccessRole to the management account	`stacksets/org-access-role/template.yaml`; `stacksets/org-access-role/targets/kestrel.json`	Jonah Reyes	Aug 4	Deployed to kestrel-staging first and verified, then to production.	Closes SF-SEC-019

The remaining week-4 pull requests, in merge order.

Pull request	Title	Files	Reviewer	Merged	Rollout	Finding
`fleet-infra#4367`	Lambda: production secrets to Secrets Manager; rotate the exposed values	`apps/fleet-api/lib/functions.ts`; `apps/ingest/lib/functions.ts`; `lib/constructs/secret-env.ts`	Sam Okafor	Aug 5	Deployed to fleet-dev first and verified, then to production.	Addresses SF-SEC-015; not yet closed
`platform-landing-zone#667`	Identity: permissions boundary for developer roles; scoped iam:PassRole	`identity/boundaries/developer-boundary.json`; `identity/permission-sets.yaml`; `policies/scp/require-workload-boundary.json`	Grace Liu	Aug 5	Deployed to sf-shared-services first and verified, then to production.	Closes SF-SEC-018
`fleet-infra#4368`	IAM: replace Action * policies with generated least-privilege policies	`apps/fleet-api/lib/node-role.ts`; `apps/ingest/lib/replay-role.ts`; `lib/constructs/least-privilege.ts`	Elena Vasquez	Aug 6	Deployed to sf-shared-services first and verified, then to production.	Closes SF-SEC-017
`platform-landing-zone#673`	Root: centralized root access management; delete member root credentials	`org/root-access.yaml`; `ops/delete-member-root-credentials.sh`	Grace Liu	Aug 6	Deployed to sf-log-archive first and verified, then to production.	Closes SF-SEC-029
`platform-landing-zone#679`	Alarms: root and break-glass sign-in to PagerDuty in all Regions	`stacksets/security-alarms/template.yaml`; `runbooks/break-glass.md`	Nadia Petrova	Aug 7	Deployed to the account's non-production twin first where one exists, then to production.	Closes SF-SEC-040

© 2026 Vantalect, LLC. Sample deliverable; StrataForge Robotics is fictional.

AWS, Amazon Web Services, AWS Graviton, and the AWS service names are trademarks of Amazon.com, Inc. or its affiliates. Kubernetes is a registered trademark of The Linux Foundation. GitHub is a trademark of GitHub, Inc. NVIDIA and Isaac Sim are trademarks of NVIDIA Corporation. SAP and SAP S/4HANA are trademarks of SAP SE. Okta is a trademark of Okta, Inc. PagerDuty is a trademark of PagerDuty, Inc. SOC 2 is a registered trademark of the American Institute of Certified Public Accountants. CIS Benchmarks is a trademark of the Center for Internet Security, Inc. Vantalect is not affiliated with or endorsed by any of them. StrataForge Robotics is fictional.