ScoutSuite Mini-Report — AWS Account 441336784577

Date: 12 Sep 2025
Prepared for: Security & Architecture (CR Review)
Prepared by: <Your Name / Team>

---

1) Scope & Objective

Run **ScoutSuite** against the AWS lab account to identify misconfigurations and produce a concise remediation plan suitable for CR sign-off.

---

2) How We Ran It (Reproducible)

Runner: Docker
Command:

```
docker run -it --rm -v "${env:USERPROFILE}\.aws:/root/.aws:ro" -v "${PWD}\scout-report:/root/scout-report" rossja/ncc-scoutsuite:latest scout aws --no-browser --report-dir /root/scout-report
```

Output: HTML report in `./scout-report/` (file name includes `aws` and timestamp).

---

3) Summary Dashboard (High-Level)

- Most AWS services show **0 resources / 0 findings** (fresh/clean account).
- **IAM: 4 findings** across **37 checks** (account-level hygiene).

---

4) Key Findings (IAM)

> Copy the *exact* rule names from the report: **Security → IAM → Findings**.

1. **[F1: <paste exact rule name> — likely Root account MFA not enabled]**
   Risk: High — Unprotected root access can lead to total account compromise.
   Fix: Enable **MFA on the root user**; remove any root access keys.

2. **[F2: <paste exact rule name> — likely Weak or Missing Password Policy]**
   Risk: Medium — Increases brute-force and credential-stuffing risk.
   Fix: Set strict policy: length ≥ 14, complexity (ULNS), prevent reuse (≥24), optional rotation ≤ 90 days.

3. **[F3: <paste exact rule name> — likely MFA not enforced for IAM console users]**
   Risk: Medium — Stolen credentials can be used without a second factor.
   Fix: Enforce **MFA for all console users**; add conditional checks (e.g., `aws:MultiFactorAuthPresent`).

4. **[F4: <paste exact rule name> — likely Access Key Hygiene/Rotation]**
   Risk: Medium — Long-lived keys increase blast radius if leaked.
   Fix: Remove unused keys; rotate ≤ 90 days; prefer **roles** and short-lived credentials.

---

5) Prioritized Remediation Plan (Do Now → Next)

Do Now (24–48h): 1. Enable **root MFA**; verify no root access keys exist.
2. Enforce **account-wide password policy** (ULNS, ≥14 chars, reuse prevention).
3. Enroll **MFA for all IAM users** with console access.

Next (This Week): 4. Audit **access keys**; remove stale; rotate active; migrate to role-based auth.
5. Add CI guardrails: periodic **ScoutSuite** run; send deltas to Slack/Email.

---

6) Evidence to Attach

- **HTML report** from `./scout-report/` (zip the folder).
- **Screenshots** of dashboard and the 4 IAM findings pages.
- (Optional) CLI output of `aws sts get-caller-identity` for traceability.

---

7) Conclusion

The account is largely empty (minimal attack surface), but **IAM hygiene** needs attention. Applying the above remediations will likely reduce findings to **zero** on re-scan and meets baseline expectations for lab/ prototype environments.

Sign-off

- **Security Lead: ___ *Date:* _____**
- **App/Account Owner: ___ Date: ______**