**ScoutSuite Security Audit Report**

**Introduction to ScoutSuite**

According to NCCGroup (2024), ScoutSuite is an open-source tool for security auditing across multiple clouds, allowing for the assessment of security posture in cloud environments. By utilizing the APIs provided by cloud service providers, ScoutSuite collects configuration information for manual review and identifies potential risk areas. Instead of sifting through numerous pages in web consoles, ScoutSuite offers a straightforward view of the attack surface automatically.

In this report, we demonstrate the setup and execution of ScoutSuite against an AWS environment, including an EC2 instance. The process involved configuring the tool, setting up AWS credentials, running the audit, and analyzing the results to identify potential security vulnerabilities.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generated

**Steps to Set Up and Run ScoutSuite**

We followed a series of steps to configure and run ScoutSuite on a Windows environment, targeting an AWS account with an active EC2 instance.

**Step 1: Environment Setup**

- Cloned the ScoutSuite repository from GitHub.

- Created and activated a Python virtual environment.

- Installed required dependencies from requirements.txt.

- Verified installation by running `python scout.py --help`.

**Step 2: Key Permissions**

- Set permissions on the SSH key file (key-test.pem) to read-only using `attrib +r key-test.pem` on Windows, equivalent to `chmod 400` on Linux, to ensure secure access.

**Step 3: SSH into EC2 Instance**

- Connected to the EC2 instance using SSH: `ssh -i "key-test.pem" ec2-user@13.53.206.90`.

- This step confirmed access to the instance running Amazon Linux 2023.

**Step 4: AWS Credentials Configuration**

- Although ScoutSuite can use IAM roles or credentials, in this setup, it authenticated using configured AWS credentials (Access Key ID, Secret Access Key, Region, etc.).

- Verified credentials using `aws sts get-caller-identity` to ensure proper access.

**Step 5: Running ScoutSuite**

- Executed `python scout.py aws` from the ScoutSuite directory.

- ScoutSuite gathered data from various AWS APIs, fetching resources from services such as EC2, IAM, S3, VPC, and more.

- The process included fetching resources for ACM, Lambda, CloudFormation, CloudTrail, CloudWatch, CloudFront, CodeBuild, Config, Direct Connect, DynamoDB, EC2, EFS, ElastiCache, ELB, ELBv2, EMR, IAM, KMS, RDS, RedShift, Route53, S3, SES, SNS, SQS, VPC, and Secrets Manager.

- Errors were encountered for services not subscribed to (e.g., Direct Connect, Redshift, EMR), which is expected.

- After data collection, ScoutSuite ran pre-processing, rule engine, applied filters, and post-processing.

- Generated an HTML report at `scoutsuite-report\aws-287882045163.html`.

**Step 6: Reviewing the Report**

- The HTML report provides a dashboard view of the security posture.

- Services are color-coded: Green (secure), Orange (warnings), Red (critical issues).

- Key dashboards include the main testing dashboard, IAM dashboard, and EC2 dashboard.

**ScoutSuite Results**

Below are screenshots from the ScoutSuite report generated in our audit:

**Testing Dashboard**

This overview shows the overall security score and summary of findings across all services.

![Testing Dashboard](testing dashboard.jpeg)

**IAM Dashboard**

The IAM dashboard highlights issues related to Identity and Access Management, such as user permissions, roles, and policies.

![IAM Dashboard](IAM dashboard.jpeg)

**EC2 Dashboard**

The EC2 dashboard details findings for Elastic Compute Cloud instances, including security groups, key pairs, and instance configurations.

![EC2 Instance](EC2 instance.jpeg)

From the results, ScoutSuite successfully detected the EC2 instance and assessed its configuration. No critical red flags were noted in the provided screenshots, indicating a relatively secure setup, though further review of the full report is recommended for comprehensive analysis.

**How ScoutSuite Helps Harden Against Common Attacks**

ScoutSuite analyzes cloud configurations to identify misconfigurations that could lead to security vulnerabilities. While it does not scan application source code directly, it strengthens the infrastructure layer. Based on insights from CoPilot (2025), here's how it addresses common attacks:

- **Session Hijacking**: Identifies IAM users without Multi-Factor Authentication (MFA), reviews session policies for long idle durations, and verifies CloudTrail logging for session monitoring.

- **Clickjacking**: Ensures secure headers are enforced and audits Web Application Firewall (WAF) rules to prevent framing attempts.

- **SQL Injection**: Checks RDS exposure (e.g., public accessibility), verifies encryption at rest and in transit, and flags overly permissive IAM roles accessing databases.

- **Cross-Site Scripting (XSS)**: Reviews S3 bucket policies to prevent hosting of malicious scripts, examines WAF rules for script injection patterns, and evaluates CloudFront header configurations.

- **Man-in-the-Middle (MitM)**: Ensures TLS is enforced on all endpoints (ALB, API Gateway), audits AWS Certificate Manager (ACM) certificates for validity and expiration, and identifies unencrypted traffic routes.

- **DDoS Attacks**: Verifies AWS Shield and WAF are enabled, assesses rate limiting and throttling on API Gateway, and audits CloudWatch alarms for traffic anomalies.

By addressing these infrastructure-level issues, ScoutSuite helps mitigate risks associated with common web application attacks, leading to a more secure cloud environment.

**Conclusion**

ScoutSuite is a powerful tool for auditing cloud security postures, providing automated insights into potential vulnerabilities. In this exercise, we successfully set up the tool, configured access, and ran an audit on an AWS environment, detecting resources like the EC2 instance. The generated report, with its visual dashboards, allows for quick identification of security issues. While ScoutSuite focuses on infrastructure, its findings are crucial for implementing best practices in access controls, encryption, and monitoring, ultimately enhancing the overall security of cloud-based applications.

**Reference List**

OpenAI. 2025. ChatGPT.

[Large Language Model].

Available at:  
<https://chatgpt.com/share/68c87960-fad8-8012-a87c-eefb68d59ca3>   
[Accessed 13 September 2025].

NCCGroup. 2024. ScoutSuite.

[Online]. Available at: <https://github.com/nccgroup/ScoutSuite>

[Accessed 13 September 2025].

### Annexure

Title: Disclosure of AI Usage in My Assessment

Section within the assessment in which it was used: ScoutSuite Cloud Security Audit – AWS Environment

Name of AI tool: ChatGPT

Purpose of use: To generate a structured, professional report based on the ScoutSuite audit findings, summarizing methodology, key findings, risk analysis, and recommendations.

Date of use: 15 September 2025

**Link:** <https://chatgpt.com/share/68c87960-fad8-8012-a87c-eefb68d59ca3>