_# PRJ-CISA-047: Automated Evidence Collection for Audits

Certification: Certified Information Systems Auditor (CISA) **Domain:** Domain 1 - The Process of Auditing Information Systems

---

1. Project Overview

This project directly addresses a core task for any information systems auditor: the collection of evidence. Manually gathering audit evidence from a cloud environment is time-consuming, prone to human error, and difficult to perform on a continuous basis. This project demonstrates how to build a fully automated, event-driven system on AWS to collect, secure, and store audit evidence, significantly improving the efficiency and reliability of the audit process.

We will create a solution that periodically gathers configuration snapshots, compliance checks, and security findings from various AWS services. The system will package this evidence, encrypt it for integrity and confidentiality, store it in a secure, immutable location, and notify the relevant compliance or audit team. This “auditor-in-a-box” approach ensures that evidence is collected consistently and is readily available for review, streamlining preparations for audits like CISA, SOC 2, or ISO 27001.

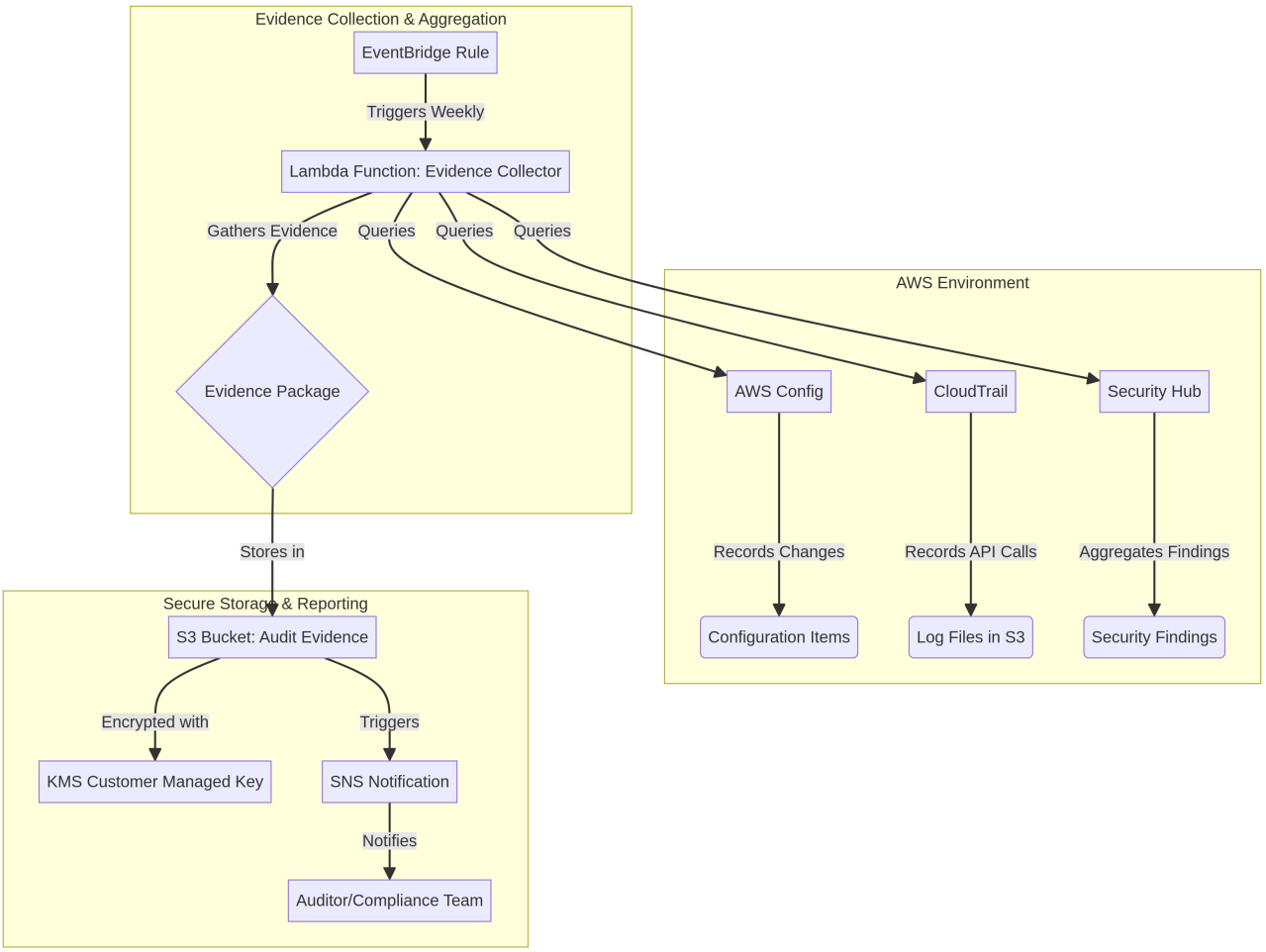
Key Objectives

- Understand the principles of collecting sufficient, reliable, and relevant audit evidence.
- Automate the collection of evidence from key AWS governance services: **AWS Config**, **AWS CloudTrail**, and **AWS Security Hub**.
- Use **AWS Lambda** and **Amazon EventBridge** to create a scheduled, serverless collection mechanism.
- Implement strong security controls for storing evidence, including encryption at rest with **AWS KMS** and write-once, read-many (WORM) principles using **S3 Object Lock**.
- Create an automated notification system using **Amazon SNS** to alert auditors when new evidence is available.

---

2. Architecture

The architecture is a serverless, event-driven pipeline that automates the entire evidence lifecycle from collection to notification.



Architectural Flow:

1. **Evidence Sources:**
    - **AWS Config:** Continuously tracks resource configurations and evaluates them against defined rules (e.g., “Are all S3 buckets encrypted?”). It provides a point-in-time snapshot of compliance.
    - **AWS CloudTrail:** Records all API activity in the account, providing a detailed log of who did what, and when. This is crucial for testing access controls and investigating incidents.
    - **AWS Security Hub:** Aggregates security findings from various AWS services (like GuardDuty, Inspector) and third-party tools into a standardized format.

2. **Automated Collection:**
    - An **Amazon EventBridge Rule** is configured to run on a schedule (e.g., weekly or monthly).
    - The rule triggers an **AWS Lambda function**, the "Evidence Collector."
    - The Lambda function is granted specific, read-only IAM permissions to query AWS Config, CloudTrail, and Security Hub. It programmatically gathers the required reports and logs.
3. **Secure Storage and Notification:**
    - The Lambda function packages the collected evidence (e.g., into a compressed ZIP file).
    - It uploads this evidence package to a dedicated **S3 bucket** designated for audit evidence.
    - This S3 bucket is configured with strict security settings:
        - **Encryption:** All objects are encrypted using a **Customer-Managed Key (CMK)** in AWS KMS, giving auditors control and visibility over key usage.
        - **S3 Object Lock (Optional but Recommended):** The bucket can be configured in "Compliance Mode" to make the uploaded evidence immutable for a specified retention period, preventing tampering or deletion.
    - The S3 `PutObject` event triggers an **Amazon SNS notification**.
    - The SNS topic sends an email or SMS to a distribution list for the **Auditor/Compliance Team**, informing them that a new evidence package is ready for review, including a link to the S3 object.

---

3. Prerequisites

- An AWS account with administrative permissions.
- AWS Config, AWS CloudTrail, and AWS Security Hub enabled and configured in your region.
- An email address to receive SNS notifications.

---

4. Step-by-Step Implementation Guide

Step 4.1: Create the Secure S3 Bucket

1. **Create a KMS Customer-Managed Key (CMK):**
    - Go to the **KMS Console -> Customer managed keys -> Create key**.
    - Create a symmetric key and give it an alias (e.g., `audit-evidence-key`).
2. **Create the S3 Bucket:**
    - Go to the **S3 Console -> Create bucket**.
    - **Bucket name:** `your-company-audit-evidence-` followed by your account ID.
    - **Enable S3 Object Lock:** This is optional but highly recommended for audit integrity. You must enable it at bucket creation.
    - **Enable default encryption:** Choose **AWS Key Management Service key (SSE-KMS)** and select the `audit-evidence-key` you just created.
    - **Block all public access.**

Step 4.2: Create the SNS Notification Topic

1. Go to the **SNS Console -> Topics -> Create topic**.
2. Choose **Standard** type and give it a name (e.g., `Audit-Evidence-Notification`).
3. Once created, go to the **Subscriptions** tab and create a new subscription.
4. **Protocol:** Email.
5. **Endpoint:** Enter the email address of the audit team.
6. You will receive a confirmation email. Click the link to confirm the subscription.

Step 4.3: Create the Evidence Collector Lambda Function

1. **Create an IAM Role:**
    - Go to the **IAM Console -> Roles -> Create role**.
    - **Trusted entity:** AWS service -> Lambda.

- **Permissions:** Attach the following AWS managed policies:
  - `AWSLambdaBasicExecutionRole`
  - `AWSConfigUserAccess`
  - `CloudTrailReadOnlyAccess`
  - `AWSSecurityHubReadOnlyAccess`
- Create a custom inline policy to allow the Lambda to write to the S3 bucket and publish to the SNS topic.

2. **Create the Lambda Function:**
   - Go to the **Lambda Console -> Create function**.
   - **Name:** `Evidence-Collector`
   - **Runtime:** Python 3.9.
   - **Role:** Choose the IAM role you just created.
   - **Code:** Paste the following Python code. This is a simplified example; a real-world script would be more robust.

```
import boto3
import datetime


def lambda_handler(event, context):
    s3 = boto3.client('s3')
    config = boto3.client('config')
    securityhub = boto3.client('securityhub')


    # 1. Get Security Hub findings
    findings = securityhub.get_findings()
    s3.put_object(
        Bucket='YOUR_BUCKET_NAME',
        Key=f'evidence-{datetime.date.today()}/security-hub-
findings.json',
        Body=str(findings)
    )


    # 2. Get non-compliant AWS Config rules
    non_compliant_rules = config.describe_compliance_by_config_rule(
        ComplianceTypes=['NON_COMPLIANT']
    )
    s3.put_object(
        Bucket='YOUR_BUCKET_NAME',
        Key=f'evidence-{datetime.date.today()}/config-non-
compliant.json',
        Body=str(non_compliant_rules)
    )


    # In a real-world scenario, you would also pull CloudTrail logs,
    # package everything into a zip, and then upload the single zip
file.


    return {'status': 'SUCCESS'}
```

- **Replace** `YOUR_BUCKET_NAME` with the name of your audit evidence bucket.
- Increase the **Timeout** under General configuration to 1 minute.

Step 4.4: Schedule the Lambda with EventBridge

1. Go to the **EventBridge Console -> Rules -> Create rule**.
2. **Name:** `Weekly-Evidence-Collection`
3. **Define pattern:** Schedule.
4. **Cron expression:** `0 12 ? * 1 *` (This runs every Monday at 12:00 PM UTC. Adjust as needed).
5. **Select targets:**
   - **Target:** Lambda function.
   - **Function:** Select the `Evidence-Collector` function.
6. Create the rule.

Step 4.5: Configure S3 Event Notification

1. Go to your audit evidence **S3 bucket -> Properties -> Event notifications**.
2. **Create event notification:**
   - **Name:** `Notify-Auditors-On-Upload`
   - **Event types:** Select `s3:ObjectCreated:Put`.
   - **Destination:** SNS Topic.
   - **SNS topic:** Select the `Audit-Evidence-Notification` topic.
   - Save changes.

---

5. How to Test

1. Go to the **Lambda Console**, select your `Evidence-Collector` function, and click **Test**.
2. Configure a test event (the content doesn't matter for this test) and run it.
3. Check your **S3 audit evidence bucket**. You should see a new folder with the current date containing the JSON files.
4. Check your **email inbox**. You should receive an SNS notification about the new objects created in S3.

---

6. Cleanup

1. **Delete the EventBridge rule.**
2. **Delete the Lambda function.**
3. **Delete the IAM role** for the Lambda function.
4. Go to the **S3 bucket**, delete the event notification, and then empty and delete the bucket.
5. Go to the **SNS console** and delete the topic and subscription.
6. Go to the **KMS console** and schedule the `audit-evidence-key` for deletion.

Business Context

The Problem

Organizations need to demonstrate compliance with audit requirements but lack automated evidence collection. Manual audit preparation is time-consuming and error-prone. Auditors struggle to verify security controls and compliance in dynamic cloud environments.

The Solution

Automated compliance monitoring and audit evidence collection system. Continuously assesses AWS environment against compliance frameworks, generates audit reports, and maintains evidence repository. Provides real-time compliance dashboards and automated remediation for non-compliant resources.

Business Value

- **Audit Efficiency:** Reduces audit preparation time from months to days
- **Continuous Compliance:** Real-time monitoring vs. point-in-time assessments
- **Cost Reduction:** Eliminates manual evidence collection, saving 200+ hours per audit
- **Risk Visibility:** Executive dashboards show compliance posture at a glance

Risk Mitigation

Prevents compliance violations, identifies control gaps before audits, ensures evidence availability, and reduces audit findings and remediation costs.

GRC Mapping

Compliance Frameworks

- **COBIT 2019:** APO13 (Manage security), DSS05 (Manage security services)
- **ISO 27001:** A.18.1 (Compliance with legal requirements), A.18.2 (Information security reviews)
- **NIST CSF:** ID.GV-3 (Legal and regulatory requirements), PR.IP-1 (Baseline configuration)
- **ITIL v4:** Service validation and testing

Security Controls Implemented

- Automated compliance assessments
- Continuous control monitoring
- Audit trail and evidence collection
- Configuration compliance checking
- Automated remediation workflows

Audit Evidence

- Compliance assessment reports
- Control effectiveness evidence
- Configuration snapshots and change logs
- Remediation records and timelines

Regulatory Alignment

- **SOX:** Section 404 (Internal controls assessment)

- **PCI DSS:** Requirement 11.3 (Penetration testing), Requirement 12.9 (Service provider compliance)
- **HIPAA:** § 164.308(a)(8) (Evaluation of security measures)
- **SOC 2:** All trust service criteria