# Security Automation & SIEM Integration Report

Nathalia Bertol

2024

# Content

# 1 Executive Summary

This report proposes improvements to strengthen security visibility, access control, and asset management. The automated access workflow ensures managerial approval, temporary access expiration, and full logging to the SIEM. Logs from Google Workspace and Slack are integrated into Wazuh to enable centralized monitoring and detection of suspicious activity.

The solution was designed to support a multi-OS scenario (macOS, Windows, and Linux), ensuring flexibility and consistent enforcement of security controls. Standardizing device profiles by department further improves onboarding efficiency and configuration consistency.

Together, these measures enhance audit readiness and create a scalable, secure, and well-governed operational environment.

# 2 Automated Access Request Workflow

To ensure consistent and auditable access management across internal systems, I propose an automated access request workflow using a central request form, manager approval, and Python-based provisioning through system APIs (Google Workspace, Slack, IAM).

**Objectives:**

- Eliminate manual provisioning
- Enforce least-privilege and temporary access controls
- Maintain auditable logs of all operations
- Standardize manager accountability

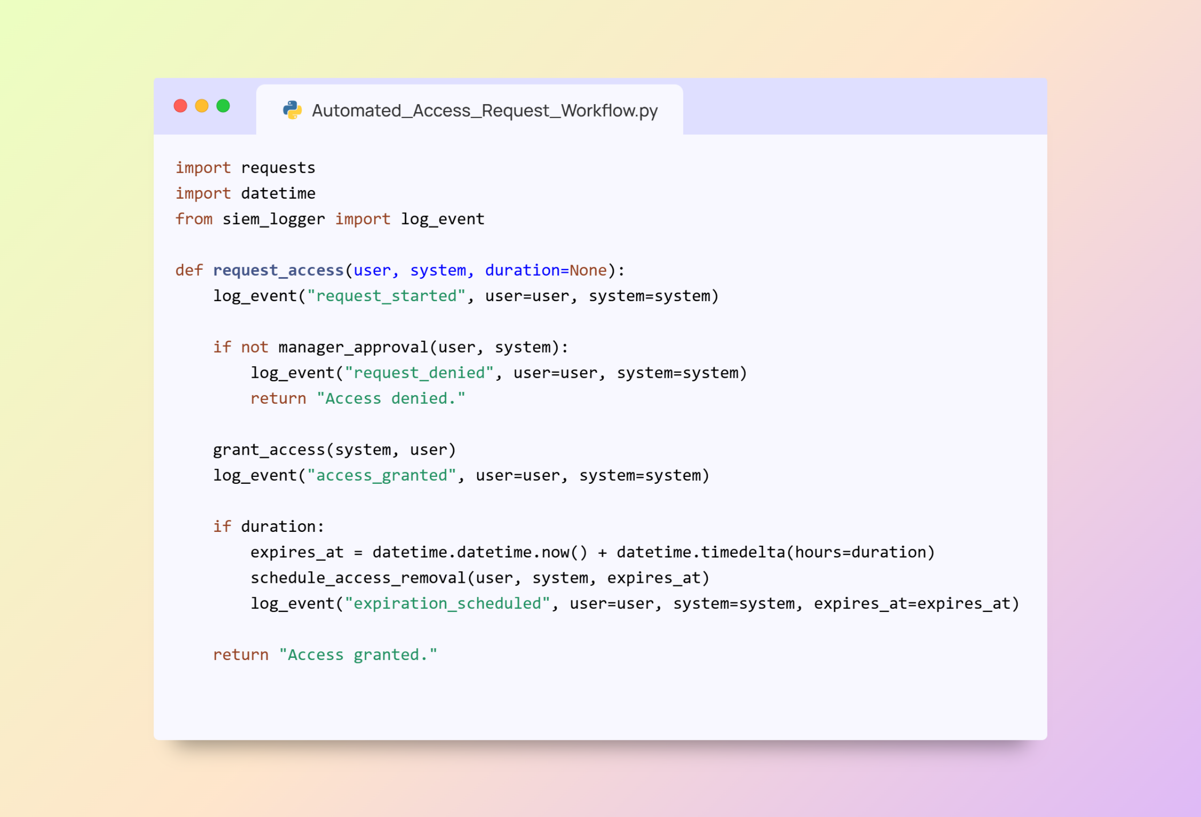
## 2.1 Workflow Description

| **Workflow Step** | **Description** |
|---|---|
| **1.** Employee Submits Access Request (Form includes:) | – Employee ID<br>– System requested (e.g., Slack, GitHub, AWS)<br>– Justification<br>– Temporary or permanent access |
| **2.** Manager Receives Approval Request | – Auto-notified through Slack or email.<br>– If approved → provision access.<br>– If rejected → request closes with justification. |
| **3.** Automated Provisioning (Python Script) | – Uses service APIs + secure credentials.<br>– If temporary access → scheduled auto-revocation. |
| **4.** Logging (All actions are sent to Wazuh SIEM:) | – Request timestamp<br>– Approver identity<br>– System modified<br>– Success or failure<br>– Expiration timestamp (if temporary) |

Table 1: Access Request Workflow Description

## 2.2 Python Automation

The function below handles access provisioning requests. It logs the request, verifies that the user's manager has approved the access, grants the access if authorized, and optionally schedules automatic access removal for temporary entitlements. All actions are logged to a SIEM for audit and compliance. The automation code is stored in a private internal Git repository and executed as a backend service in AWS, triggered whenever a new access request is submitted.



```python
import requests
import datetime
from siem_logger import log_event

def request_access(user, system, duration=None):
    log_event("request_started", user=user, system=system)

    if not manager_approval(user, system):
        log_event("request_denied", user=user, system=system)
        return "Access denied."

    grant_access(system, user)
    log_event("access_granted", user=user, system=system)

    if duration:
        expires_at = datetime.datetime.now() + datetime.timedelta(hours=duration)
        schedule_access_removal(user, system, expires_at)
        log_event("expiration_scheduled", user=user, system=system, expires_at=expires_at)

    return "Access granted."
```

Figure 1: Access Request Workflow Description

# 3 Log Integration and Monitoring

This section describes the implementation of centralized log collection and monitoring for Google Workspace and Slack, integrating both platforms into the existing Wazuh SIEM environment. Wazuh automatically normalizes logs collected from macOS, Windows, and Linux endpoints, ensuring that events are searchable and correlatable regardless of the operating system. The objective of this integration is to enhance security visibility, allow real-time detection of anomalous behavior, and support incident response.

## 3.1 Architecture Overview

| **System** | **Purpose** |
|---|---|
| Google Workspace | Identity, authentication, and activity logs |
| Slack | Collaboration and administrative action logs |
| SIEM Target: Wazuh | Centralized log analysis and alerting |

| **The integration workflow** |
|---|
| Google Workspace → Log Forwarder → Wazuh SIEM |
| Slack → Log Forwarder → Wazuh SIEM |

Table 2: Systems Involved

## 3.2 Google Workspace Integration

| **Item** | **Details** |
|---|---|
| Log Sources | Admin Activity, Login Audit, Drive Audit |
| Format | JSON (via Reports API) |
| Collection Frequency | Every 5–15 minutes (pull-based) |
| Secure Transmission | OAuth 2.0 and HTTPS TLS 1.2+ |
| Collection Method | Wazuh google_workspace module |

Table 3: Google Workspace Integration

**Example Collected Event:**

```
{
”event”: ”login_failure”,
”email”: ”employee@company.com”,
”ip”: ”185.21.33.90”,
”timestamp”: ”2025-01-17T02:41:05Z”
}
```

### 3.3 Slack Integration

| Item | Details |
|---|---|
| Log Sources | Audit Logs API + Event Subscriptions |
| Format | JSON |
| Collection Frequency | Near real time (webhook push) |
| Secure Transmission | HTTPS + Request signature validation (Slack Signing Secret) |
| Collection Method | Webhook → Fluentd/Wazuh Agent → Wazuh SIEM |

Table 4: Slack Integration

**Example Log Event:**

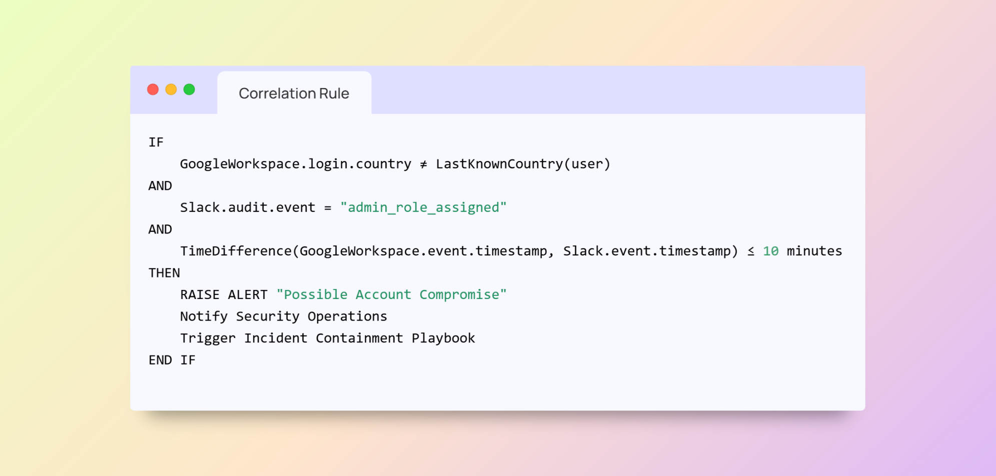
```
{
”action”: ”added_user_to_workspace”,
”actor”: ”admin_user”,
”target”: ”new_employee”,
”timestamp”: ”2025-01-17T02:45:22Z”
}
```

### 3.4 Correlation Rule and Alert

**Scenario:** Suspicious login followed by a privileged action.
**Goal:** Detect possible account takeover.

- In Google Workspace, the user logs in from an unusual country
- Within 10 minutes, in Slack, a new admin is added or privileges are elevated

**Correlation Rule (Pseudocode):**



```
IF
    GoogleWorkspace.login.country ≠ LastKnownCountry(user)
AND
    Slack.audit.event = "admin_role_assigned"
AND
    TimeDifference(GoogleWorkspace.event.timestamp, Slack.event.timestamp) ≤ 10 minutes
THEN
    RAISE ALERT "Possible Account Compromise"
    Notify Security Operations
    Trigger Incident Containment Playbook
END IF
```

Figure 2: Correlation Rule

### 3.5 Expected Impact

This integration significantly strengthens the organization's security posture by enabling:

- Early detection of suspicious lateral movement
- Reduced mean time to response (MTTR) through automated alerting
- Improved traceability and audit support for compliance requirements

# 4 Asset Management and Inventory Control

The asset inventory will be maintained using the MDM, SIEM, and a CMDB. The main idea is to use tools that remain flexible across macOS, Windows, and Linux, ensuring consistent visibility and compliance regardless of the operating system.

## 4.1 Tools and Integrations

– **MDM (JumpCloud):** Used to automatically enroll devices, collect hardware identifiers, check compliance policies, and report the device's current status.

– **SIEM (Wazuh):** Receives security and activity logs from devices to confirm whether the endpoint is active and compliant.

– **Asset Inventory (Jira Assets CMDB):** Centralizes asset lifecycle data, including device ownership, delivery and return dates, usage status, and compliance information.

## 4.2 Essential Fields in the Inventory

| **Field** | **Description** |
|---|---|
| Device ID / Serial Number | Unique identifier for the asset |
| Device Name | How the device appears in MDM / SIEM dashboards |
| Owner / Assigned User | Person currently responsible for the device |
| Device Type / Model | Laptop, workstation, mobile device |
| Status | Active, Returned, Offboarded, Repair, Missing |
| Date of Assignment | When the device was issued |
| Last Connection Timestamp | Last check-in reported by MDM |
| Compliance State | Compliant / Non-compliant (based on security policies) |

Table 5: Essential Fields in the Inventory

## 4.3 Automation to Keep Inventory Accurate

– **Automatic Enrollment:** Every new device must automatically enroll into the MDM before being used.

– **Daily Sync with SIEM:** A scheduled job compares the list of known devices in the SIEM with the inventory to detect:

Unlisted devices → Flag as Unknown Device.
Devices inactive for more than 30 days → Flag for review or collection.

– **Offboarding Automation:** When HR initiates an offboarding:

The device is remotely locked via MDM.
The status in the inventory changes to Pending Return.

SIEM monitors whether the device reconnects before returning.

### 4.4 (Extra) Suggested Operational Improvement: Standardized Device Profiles by Department

To improve efficiency in device troubleshooting, onboarding, and offboarding, the company can adopt standardized device profiles based on each department's needs. Different teams rely on different operating systems and software stacks, which means aligning hardware and OS selection with role requirements can significantly reduce support load and setup time.

**Proposed Standard**

| **Department** | **Device Type / OS** | **Justification** |
|---|---|---|
| Engineering Oriented | macOS (MacBook) | Better compatibility with development tools, UNIX shell, containerization, SDKs, and DevOps workflows. |
| Business / Administrative | Windows Laptop | More familiar and user-friendly for non-technical users. Better compatibility with corporate productivity tools, finance applications, and general office workflows. |

Table 6: Proposed Standard

**Benefits**

- Faster Troubleshooting: Support teams can maintain pre-built troubleshooting guides specific to each device profile.
- Consistent Setup and Security Baseline: Each device type can have a predefined configuration, compliance checks, and monitoring policies.
- Reduced Training Time: Support and onboarding processes become uniform within each department.
- Minimized Software Misalignment: Ensures employees receive tools aligned with their job functions.

## 5 Conclusion

In summary, the proposed solution strengthens security visibility, access governance, and asset management while ensuring compatibility across macOS, Windows, and Linux. By selecting tools and workflows that operate consistently in a multi-OS environment, the organization avoids platform fragmentation and achieves scalable, repeatable, and centralized control. This approach improves operational efficiency, supports audit readiness, and provides a more manageable and secure security posture.

# Lists

## List of Figures

## List of Tables