Demonstrating SOC Fundamentals

In the **Security Operations Center (SOC)**, the primary focus is on defending digital assets, responding to threats in real-time, and ensuring organizational continuity. For this simulation, I set up a small-scale, open-source SOC environment to manage potential threats. This involved defining SOC roles, configuring monitoring tools, handling alerts, and detecting cyber threats using SOC protocols and tools.

SOC Functions and Operations

A SOC is essentially a hub where cybersecurity professionals monitor, detect, and respond to security incidents. In my implementation, I focused on defining three core SOC roles, each of which serves a specific function in safeguarding the organization's data and assets.

1. **SOC Analyst (Tier 1):**
    - **Responsibilities:** The SOC Analyst is the first line of defense and is primarily responsible for the initial analysis of security alerts, reviewing logs, and performing basic threat triage. They escalate critical threats to higher-tier analysts.
    - **My Role:** I assumed the role of a Tier 1 SOC Analyst, reviewing alerts from the monitoring system in real-time, analyzing their context, and escalating them as needed.
2. **SOC Engineer:**
    - **Responsibilities:** The SOC Engineer maintains the technical infrastructure that supports the SOC operations. This includes configuring and fine-tuning monitoring systems, implementing threat intelligence feeds, and ensuring seamless operation of the monitoring tools.
    - **My Role:** I configured **Wazuh**, an open-source SIEM tool, as the monitoring platform. This involved creating custom rules to capture network events and threats, such as SSH login failures and anomalous DNS queries.
3. **Incident Responder (Tier 2/Tier 3 Analyst):**
    - **Responsibilities:** The Incident Responder is responsible for investigating incidents flagged by the SOC Analysts, conducting deeper analysis, and coordinating responses. They also perform forensic analysis and threat hunting to uncover advanced attacks.
    - **My Role:** I acted as the Incident Responder during an alert concerning a potential malware infection. I followed up by isolating the affected system, conducting a forensic investigation, and taking the necessary remedial actions.

This detailed understanding of each role allowed me to effectively engage with the SOC's workflow and simulate real-world incident handling.

Monitoring Fundamentals

Monitoring is one of the most critical functions of a SOC. The objective is to identify and flag any abnormal or suspicious network activity that could indicate a security breach.

Tool Used:

I deployed **Wazuh** for the role of monitoring tool. Wazuh, a free open-source SIEM, integrates host-based intrusion detection with log analysis, providing real-time monitoring and alerting capabilities.

Two Types of Network Activity Monitored:

1. **SSH Login Activity:**
   - **Purpose:** SSH is a common vector for unauthorized access attempts, especially brute-force attacks.
   - **Monitoring Details:** Using Wazuh, I monitored SSH login attempts by tracking failed and successful login attempts. I configured custom alerts to detect more than five failed attempts within a short time frame, signaling a possible brute-force attack.
   - **Outcome:** The system successfully flagged an attempt from an external IP address that made repeated failed login attempts, which I later investigated.
2. **DNS Requests:**
   - **Purpose:** DNS requests can be indicative of malicious activity, such as Command-and-Control (C2) beaconing or malware traffic.
   - **Monitoring Details:** Wazuh tracked all DNS requests to external servers and flagged any requests to suspicious domains, particularly those identified in threat intelligence feeds.
   - **Outcome:** I detected a suspicious DNS request pattern targeting an unknown domain that had been linked to known malware-related activity.

Both activities provided key insights into the network's health and possible threats, showcasing how a monitoring tool can be leveraged for proactive security defense.

Alert Management

Effective alert management ensures that SOC teams can prioritize threats and manage them swiftly. I tested the alert system using two different scenarios to evaluate its efficiency in generating, investigating, and resolving incidents.

1. **Alert 1: Brute Force SSH Attempt**
   - **Generated by:** I manually simulated a brute-force attack by executing a script that attempted multiple login attempts using random credentials.
   - **Investigated by:** Upon receiving the alert in Wazuh, I reviewed the

logs and correlated the failed login attempts with the originating IP address. The repeated login failures from the same IP signaled the brute-force attempt.
    - **Resolved by:** To mitigate the risk, I blocked the attacking IP via a firewall rule and disabled password-based authentication for SSH, enforcing key-based authentication instead.
2. **Alert 2: DNS Request to Malicious Domain**
    - **Generated by:** I used a custom script to simulate DNS queries to domains listed in a threat feed.
    - **Investigated by:** The Wazuh alert flagged the suspicious DNS request. I cross-referenced the domain against threat intelligence sources like VirusTotal, which confirmed the domain was blacklisted for malicious activity.
    - **Resolved by:** I updated the internal DNS resolver to block requests to the domain and isolated the affected system to ensure no malware presence remained.

Both incidents were successfully investigated and resolved using systematic investigation and mitigation strategies, emphasizing the importance of comprehensive alert management.

Threat Detection and Analysis

Threat detection is the core of any SOC's activities. In this section, I highlight the detection and analysis of a potential malware attack based on network behavior.

- **Identified Threat:**
  A suspicious **DNS request pattern** flagged an attempt at malware command-and-control beaconing. The request came from a previously unknown domain, and the requests followed a regular interval — a hallmark of C2 communications.
- **Detection Method:**
  The DNS anomaly alert was generated by Wazuh, which flagged the pattern of repeated queries with no corresponding responses from the DNS server. Further analysis using internal network logs confirmed that no legitimate business applications used the domain.
- **Analysis:**
  Upon further inspection, the regular pattern of DNS requests indicated the system might be infected with malware attempting to communicate with a remote server. The constant, predictable DNS lookups were indicative of a **C2 beacon**.
- **Action Taken:**
  The domain was blocked using internal DNS policies, and the affected machine was isolated from the network. I then conducted a forensic analysis of the machine’s logs and reimaged the system after confirming

no further traces of malware.

This process highlighted how SOC tools like Wazuh can be used to detect anomalous activity and how quick response can mitigate the effects of a potential cyber attack.