SKILLS FOR CYBER SECURITY ANALYST L2 TO STEP UP THEIR GAME

BY IZZMIER IZZUDDIN

TABLE OF CONTENTS

BREAKDOWN OF SKILL NEEDED

1. **Forensic Investigation**
   - **Memory and Disk Analysis**:
     - Investigate volatile data from RAM, process dumps, and disk images to uncover malicious activity.
     - Master tools like Volatility for memory forensics and EnCase or FTK for disk imaging and analysis.
     - Skills in carving out specific files, recovering deleted data, and reconstructing events from file metadata.
   - **Network Traffic Forensics**:
     - Analyse network packets to trace attacks, identify lateral movements, and reconstruct connections using tools like Wireshark and Zeek.
     - Understand how attackers use protocols like DNS, HTTP, or encrypted traffic to evade detection.
     - Ability to capture and parse network traffic for exfiltration patterns, unusual traffic spikes, or suspicious connections.
2. **Vulnerability Management**
   - **In-Depth Vulnerability Assessments**:
     - Perform thorough vulnerability scans using tools like Nessus, OpenVAS, and Qualys to identify security gaps.
     - Assess how discovered vulnerabilities could be exploited and prioritise them using threat intelligence.
     - Regularly assess systems, applications, and networks to identify misconfigurations or unpatched software.
   - **CVSS and Risk Assessment**:
     - Use the Common Vulnerability Scoring System (CVSS) to evaluate the severity of vulnerabilities.
     - Understand the contextual factors such as ease of exploitation, potential impact, and exposure to threat actors.
     - Apply risk-based prioritisation to recommend remediation measures in line with business priorities.
3. **Advanced Malware Analysis**
   - **Reverse Engineering Malware**:
     - Use reverse engineering techniques to deconstruct malware, examining its code, structure, and behaviour.
     - Identify Indicators of Compromise (IOCs) like file hashes, domain names, and registry changes left by malware.
     - Master tools like IDA Pro, Ghidra, and OllyDbg for reverse engineering executable files.
   - **Sandboxing**:
     - Use sandbox environments (e.g., Cuckoo, Joe Sandbox) to safely execute and observe malware in a controlled setting.
     - Analyse malware behaviour dynamically to determine how it operates, communicates, and spreads.

    - Extract behavioural signatures to create detection and mitigation strategies for future threats.

4. **Cloud Security**
    - **Securing Cloud Platforms**:
        - Secure various cloud environments (AWS, Azure, Google Cloud) by implementing identity management, encryption, and monitoring.
        - Use tools like AWS CloudTrail, Azure Security Centre, and Google Cloud Security Command Centre for real-time monitoring and alerting.
        - Ensure security policies, such as encryption of data at rest and in transit, and least privilege access controls, are enforced.
    - **Compliance and Traffic Monitoring**:
        - Monitor cloud traffic for unusual patterns, data exfiltration, or policy violations.
        - Understand shared responsibility models and configure security settings accordingly to ensure regulatory compliance.

5. **Compliance and Governance**
    - **Industry-Specific Regulatory Frameworks**:
        - Understand how regulations like GDPR (General Data Protection Regulation), HIPAA (Health Insurance Portability and Accountability Act), PCI DSS (Payment Card Industry Data Security Standard), and others apply to your organisation's data handling and security practices.
        - Implement controls such as encryption, access restrictions, and regular audits to ensure compliance.
        - Stay updated with changing laws and standards and adjust security policies and procedures to maintain compliance.
    - **Controls Implementation**:
        - Develop and maintain policies and procedures that align with regulatory frameworks.
        - Audit systems and workflows to ensure that data privacy and security requirements are met across the entire organisation.

SKILLS DEVELOPMENT

1. **Forensic Investigation**
   - **Courses**: Consider taking advanced courses like SANS FOR508 (Advanced Incident Response, Threat Hunting, and Digital Forensics) or FOR500 (Windows Forensic Analysis).
   - **Hands-on Practice**: Set up a lab environment with forensic images to practice disk and memory analysis. Use tools like EnCase, FTK, and Volatility for practical experience.
   - **Certifications**: Aim for certifications like GCFA (GIAC Certified Forensic Analyst) or CFCE (Certified Forensic Computer Examiner).
2. **Vulnerability Management**
   - **Training**: Enrol in courses focused on vulnerability management, such as the Offensive Security Certified Professional (OSCP) or the Certified Vulnerability Assessor (CVA).
   - **Practical Application**: Use tools like Nessus, Qualys, and OpenVAS to conduct vulnerability scans, then prioritise vulnerabilities using CVSS and threat intelligence.
   - **Reading**: Stay updated with the latest vulnerabilities by reading CVE reports and studying their CVSS scores.
3. **Advanced Malware Analysis**
   - **Courses**: Consider enrolling in SANS FOR610 (Reverse-Engineering Malware) or taking the Practical Malware Analysis course from Offensive Security.
   - **Tools**: Gain proficiency in reverse engineering with IDA Pro, OllyDbg, and use sandboxing tools like Cuckoo or Joe Sandbox to analyse malware behaviour.
   - **Practice**: Set up a dedicated lab to analyse real-world malware samples. Study the behaviour, extract IOCs, and create threat reports.
4. **Cloud Security**
   - **Certifications**: Obtain cloud security certifications like AWS Certified Security – Specialty, Microsoft Certified: Azure Security Engineer Associate, or Google Cloud Professional Cloud Security Engineer.
   - **Practical Skills**: Set up and secure cloud environments on AWS, Azure, and Google Cloud. Use tools like CloudTrail and Security Hub for monitoring.
   - **Compliance**: Learn how to apply security policies in cloud environments, ensuring compliance with frameworks like CIS Benchmarks and NIST.

5. **Compliance and Governance**
    - **Training**: Take courses on GDPR, HIPAA, and other regulatory frameworks. The Certified Information Systems Auditor (CISA) and Certified Information Privacy Professional (CIPP) are valuable certifications.
    - **Implementation**: Work on creating and implementing compliance frameworks within your organisation. Use tools like Archer or ServiceNow GRC for managing governance.
    - **Stay Updated**: Regularly review updates to regulations and how they impact your industry.

EXAMPLES AND SIMULATIONS

1. Forensic Investigation

Scenario: Data Exfiltration Detected from a Corporate Server

Background: The SOC team notices that large amounts of sensitive corporate data have been transferred out of the network to an unknown external IP address (192.168.200.50) from a server (Server A) between 2:00 AM and 3:00 AM. This is flagged as suspicious because data transfer from this server is normally restricted to internal use, and no authorised personnel were working during this time.

Step 1: Network Traffic Forensics

- **Tools**: Wireshark, Zeek (formerly Bro)

1. **Capture Network Traffic**: The SOC team captures a packet capture (PCAP) file containing traffic between Server A and the external IP address during the suspicious timeframe.
    - **PCAP File**: network_traffic.pcap
2. **Analyse the PCAP**: Open the PCAP file in Wireshark to analyse the traffic between Server A (192.168.1.10) and the suspicious IP (192.168.200.50).
    - **Filter**: Apply the filter ip.addr == 192.168.200.50 in Wireshark.
    - **Findings**:
        - The traffic consists of multiple large file transfers using an encrypted FTP (FTPS) protocol.
        - The destination IP belongs to an unknown third-party server located in another country.
        - The data transferred appears to be over 10GB in size.

Step 2: Disk Analysis

- **Tools**: FTK Imager, EnCase

1. **Create a Disk Image**: The forensic investigator creates a disk image of Server A using FTK Imager to preserve the integrity of the evidence.
    - **Disk Image File**: server_a_disk_image.e01
2. **Review File Access Logs**: The investigator loads the disk image into EnCase for deeper analysis.
    - **Findings**:

        - Logs show that several files in the /confidential_data/ directory were accessed and compressed into a .zip archive (e.g., confidential_backup.zip) at 1:45 AM.

        - The archive was then moved to a temporary folder /tmp/transfer/ and subsequently deleted shortly after the FTPS transfer.

3. **Recover Deleted Files**: Using FTK, the investigator recovers the deleted .zip file and reviews its contents.

    - **Recovered File**: confidential_backup.zip

    - **Contents**: The ZIP file contains multiple sensitive documents including financial statements (financial_report.pdf), employee records (employees.xlsx), and business contracts (contracts.docx).

Step 3: Memory Forensics

- **Tools**: Volatility

1. **Memory Dump**: The investigator takes a memory dump of Server A at the time of the incident for further analysis.

    - **Memory Dump File**: server_a_memory.dmp

2. **Analyse Active Processes**: Using Volatility, the investigator looks for suspicious processes running during the time of the data transfer.

    - **Command**: volatility pslist -f server_a_memory.dmp

    - **Findings**:

        - A suspicious process named malicious_exfil.exe was running, disguised as a legitimate system process.

        - The process was spawned by a service that usually does not interact with external networks.

3. **Analyse Network Connections**: Further analysis with Volatility reveals that malicious_exfil.exe established the connection with the external IP address (192.168.200.50).

    - **Command**: volatility netscan -f server_a_memory.dmp

    - **Findings**:

        - The process was responsible for initiating the FTPS connection, suggesting it was involved in the data exfiltration.

4. **Extract IOCs**: The investigator extracts the following Indicators of Compromise (IOCs) from the memory and disk analysis:
    - **File Path**: /system32/malicious_exfil.exe
    - **IP Address**: 192.168.200.50
    - **Hash of Malicious File**: SHA256: 3b5b6f8e9f4f3d60d5b8f95e6e61de1c72c6a7b2d9fcb104807c5b4e792f4cd9

Step 4: Investigate User Activity

- **Tools**: EnCase, Log Analysis

1. **Review User Login History**: The investigator reviews the login records and activity logs on Server A.
    - **Findings**:
        - The user account izzmier was logged in during the time of the incident, despite the user being off-duty. The login originated from an internal IP address (192.168.1.45).
2. **Correlate User Activity**: Logs indicate that the izzmier account was used to execute the malicious process. However, after contacting the user, it was discovered that their credentials had been compromised through a phishing email received earlier in the week.

Step 5: Reporting and Remediation

- **Document Findings**:
    - A detailed incident report is created that documents all the steps taken during the investigation, including the evidence gathered from network traffic, disk analysis, memory forensics, and user activity logs.
- **Remediate the Issue**:
    - **User Account Security**: The compromised account is locked, and the user undergoes security training.
    - **Malware Removal**: The malicious executable is deleted from the server, and a scan is run across the network to ensure no other systems are infected.
    - **Firewall and Network Segmentation**: The firewall rules are updated to prevent unauthorised outbound connections, and additional network segmentation is implemented to isolate sensitive servers.

- **IOC Sharing**: The extracted IOCs are shared with the SOC team to update detection rules in the SIEM for monitoring future attempts.

2. Vulnerability Management

Scenario: Vulnerability Management in a Corporate Network

Background: A cybersecurity analyst is responsible for managing vulnerabilities across a company's IT infrastructure. The company has recently undergone a regular vulnerability scan and identified several security gaps that need to be addressed. Some of the vulnerabilities are high-risk, while others are low-risk. The analyst must assess these vulnerabilities, prioritise remediation efforts, and ensure that the organisation is protected from potential attacks.

Step 1: Conduct a Vulnerability Assessment

- **Tools**: Nessus, Qualys, OpenVAS

1. **Scan the Network**: The analyst performs a vulnerability scan on the corporate network, which includes 50 servers, 100 workstations, and network devices (firewalls, routers, etc.). The scan is done using Nessus and reveals a number of vulnerabilities.
    - **Findings from the Scan**:
        - **Critical Vulnerabilities**:
            - CVE-2023-12345: Remote Code Execution (RCE) vulnerability in a web application running on Server A.
            - CVE-2023-67890: SQL Injection vulnerability in an internal database on Server B.
        - **Medium-Risk Vulnerabilities**:
            - Outdated software version on workstations (e.g., outdated versions of Adobe Reader and Java).
            - Weak SSH configurations on network devices (e.g., SSH version 1 enabled).
        - **Low-Risk Vulnerabilities**:
            - Insecure cookie settings in internal applications.

Step 2: Assess the Risk

- **Tools**: CVSS Calculator, Threat Intelligence Feeds (e.g., Recorded Future, MISP)

1. **Calculate CVSS Scores**: The analyst uses the Common Vulnerability Scoring System (CVSS) to assign severity scores to the vulnerabilities found.
    - **Critical Vulnerabilities**:

        - CVE-2023-12345 (RCE) receives a CVSS score of **9.8 (Critical)** because it can allow an attacker to take full control of Server A remotely.
        - CVE-2023-67890 (SQL Injection) receives a score of **8.7 (High)** due to the potential for data breaches and access to sensitive information in the internal database.
    - **Medium-Risk Vulnerabilities**:
        - Outdated software versions receive a score of **6.5 (Medium)** because while they can be exploited, they do not provide direct access to critical systems.
        - Weak SSH configurations receive a score of **5.8 (Medium)** since they increase the risk of brute-force attacks, but do not provide immediate access to systems.
    - **Low-Risk Vulnerabilities**:
        - Insecure cookie settings receive a score of **4.0 (Low)** because while they can lead to session hijacking, they are not directly exploitable without other vulnerabilities.

2. **Leverage Threat Intelligence**: The analyst looks up current threat intelligence to determine if any of the vulnerabilities are actively being exploited in the wild.
    - **Findings**:
        - Threat intelligence indicates that CVE-2023-12345 (RCE vulnerability) is actively being targeted by attackers and that a proof-of-concept (PoC) exploit is publicly available.
        - CVE-2023-67890 (SQL Injection) is not actively exploited, but threat intelligence suggests it could be easily weaponised in the near future.

Step 3: Prioritise Remediation

- **Tools**: Vulnerability Management Dashboard, Risk Matrix

1. **Prioritise Based on Impact and Exploitability**: Based on the CVSS scores, threat intelligence, and business impact, the analyst creates a priority list for remediation:
    - **Highest Priority**:
        - CVE-2023-12345 (RCE vulnerability) on Server A: Since this is actively being exploited and can lead to full system compromise, this vulnerability must be patched immediately.

- **High Priority**:
    - CVE-2023-67890 (SQL Injection vulnerability) on Server B: Although not actively exploited, it poses a significant risk if left unpatched.
- **Medium Priority**:
    - Outdated software versions on workstations: While these vulnerabilities are not critical, they should be addressed within the next few weeks to prevent exploitation.
    - Weak SSH configurations on network devices: These should be updated to stronger configurations (e.g., SSH version 2) as part of the ongoing security maintenance.
- **Low Priority**:
    - Insecure cookie settings: These can be addressed during the next routine security update for the web applications.

Step 4: Remediation and Validation

- **Tools**: Patch Management Systems, Configuration Management Tools

1. **Remediate Critical Vulnerabilities**:
    - **Server A (CVE-2023-12345)**: Apply the security patch provided by the web application vendor to fix the RCE vulnerability.
    - **Server B (CVE-2023-67890)**: Apply secure coding practices to prevent SQL injection and update the database to the latest, secure version.
2. **Remediate Medium-Risk Vulnerabilities**:
    - **Outdated Software**: Use the organisation's patch management system to push updates to all workstations with outdated software.
    - **Weak SSH Configurations**: Update the SSH configurations on all network devices, disabling SSH version 1 and enforcing stronger authentication methods.
3. **Remediate Low-Risk Vulnerabilities**:
    - **Insecure Cookie Settings**: Update the web application's cookie settings to enable HttpOnly and Secure flags to mitigate session hijacking risks.

Step 5: Monitor and Report

- **Tools**: SIEM, Vulnerability Scanning Tools, Reporting Dashboard

1. **Re-scan the Network**: After applying the necessary patches and updates, the analyst re-runs a vulnerability scan across the network to ensure that all critical vulnerabilities have been remediated.

    - **Findings**: The re-scan shows that the critical vulnerabilities (CVE-2023-12345 and CVE-2023-67890) are no longer present, and the medium- and low-risk vulnerabilities have been significantly reduced.

2. **Update Security Metrics**: The analyst updates the security metrics and generates a report showing the improvement in the organisation's security posture as a result of the remediation efforts.

3. **Continuous Monitoring**: The organisation's SIEM is updated with detection rules to monitor for any attempted exploitation of similar vulnerabilities in the future, and regular vulnerability scans are scheduled to maintain a strong security posture.

3. Advanced Malware Analysis

Scenario: Analysis of Suspicious Malware Found in a Corporate Network

Background: A cybersecurity analyst receives an alert from the organisation's antivirus solution about suspicious malware found on several workstations. The malware has evaded detection for some time, and the analyst must perform an advanced analysis to understand its behaviour, identify Indicators of Compromise (IOCs), and create appropriate remediation strategies.

Step 1: Static Analysis

- **Tools**: IDA Pro, Strings, VirusTotal

1. **Obtain the Malware Sample**: The malware file is retrieved from one of the infected machines. The file is named malware_sample.exe.
    - **File Information**:
        - **File Hash (SHA256)**: 4e3d3e7b5f2ed60e78fb9278e214324b19e4e4e9d29e56b0947b3f9b5177f55b
        - **Size**: 512 KB
2. **Analyse the File Header**: The analyst opens the malware file in IDA Pro, a disassembler, to understand its structure and functionality.
    - **Findings**:
        - The malware appears to be packed (obfuscated) to evade detection. The analyst notes that unpacking is necessary for further analysis.
3. **Basic String Extraction**: Using the strings tool, the analyst attempts to extract readable text from the file. This can provide hints about its functionality.
    - **Findings**:
        - Some suspicious strings are identified:
            - http://malicious-domain.com/command
            - /tmp/secret/
            - cmd.exe /c delete_all_logs
            - RC4 encryption key: 12f78a9c2d

        - These strings suggest that the malware is communicating with a command-and-control (C2) server and might delete logs to cover its tracks.

4. **VirusTotal Check**: The analyst uploads the file's hash to VirusTotal to see if other security vendors have flagged the sample.
    - **Findings**:
        - The file is flagged as malicious by several antivirus engines, with descriptions such as "Remote Access Trojan (RAT)" and "Info-Stealer."

Step 2: Dynamic Analysis

- **Tools**: Cuckoo Sandbox, Process Monitor, Regshot

1. **Set Up the Environment**: The analyst places the malware sample in an isolated virtual machine (VM) running a sandbox like Cuckoo to safely observe its behaviour.
    - **Environment**: Windows 10 VM with monitoring tools like Process Monitor (ProcMon) and Regshot to track changes.
2. **Execute the Malware**: The analyst runs the malware in the sandbox and begins monitoring its behaviour.
    - **Findings**:
        - **Network Activity**: The malware attempts to establish a connection to http://malicious-domain.com/command. It sends encrypted HTTP POST requests and awaits responses from the C2 server.
        - **Process Creation**: The malware spawns a new process, malicious.exe, which continuously runs in the background.
        - **Registry Changes**: The malware modifies the Windows registry to establish persistence:
            - **Registry Key**: HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
            - **Value**: malicious.exe = "C:\Users\Public\malicious.exe"
        - **File Creation**: It creates several temporary files in the /tmp/secret/ directory, suggesting data collection.

        - **Log Deletion**: The malware attempts to delete system logs using cmd.exe /c delete_all_logs.

3. **Capture IOCs**:
    - **Network IOCs**:
        - C2 Domain: malicious-domain.com
        - C2 IP: 192.168.200.100
    - **File IOCs**:
        - Dropped Files: C:\Users\Public\malicious.exe
        - Temporary Files: /tmp/secret/data.tmp
    - **Registry IOCs**:
        - Persistence Mechanism: HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run

Step 3: Behavioural Analysis

- **Tools**: Process Hacker, Wireshark, API Monitor

1. **Monitor API Calls**: The analyst uses API Monitor to observe the system calls made by the malware. This can reveal how the malware interacts with the system's core components.
    - **Findings**:
        - The malware uses CryptEncrypt() API calls, indicating that it is encrypting the data before exfiltration.
        - It also uses the WinExec() function to execute shell commands, such as deleting system logs.
2. **Monitor Network Traffic**: Using Wireshark, the analyst monitors the malware's network traffic to confirm the C2 communication and analyse the data being transmitted.
    - **Findings**:
        - The traffic shows that the malware is using RC4 encryption (as indicated by the static analysis) to transmit data to the C2 server. The analyst is able to decrypt the traffic using the RC4 key (12f78a9c2d) extracted from the static analysis.

        - Decrypted payload: Contains sensitive information such as screenshots, keylogs, and system information sent to the remote C2 server.

3. **Reverse Engineer the Payload**: With the unpacked binary, the analyst uses IDA Pro to reverse engineer the malware's core functionality.
    - **Findings**:
        - The malware is a multi-stage RAT capable of keylogging, screen capturing, and exfiltrating sensitive data. It receives commands from the C2 server to perform specific actions on the infected system, such as stealing browser credentials or executing additional malicious code.

Step 4: Report and Remediation

- **Tools**: IOC Database, Incident Reporting Tools

1. **Document Findings**: The analyst documents the entire malware analysis process, including:
    - **File Hash**: SHA256: 4e3d3e7b5f2ed60e78fb9278e214324b19e4e4e9d29e56b0947b3f9b5177f55b
    - **Malware Behaviour**: Data exfiltration, persistence, log deletion, keylogging, screen capture.
    - **IOCs**:
        - **Network**: malicious-domain.com, IP: 192.168.200.100
        - **Registry**: HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
        - **Files**: C:\Users\Public\malicious.exe
        - **RC4 Key**: 12f78a9c2d
2. **Update SIEM and Firewalls**: The IOCs are shared with the SOC team to update SIEM rules and firewall rules, ensuring that future attempts to communicate with the C2 server are blocked.
3. **Remediate Infected Systems**: The analyst provides guidance to the IT team on removing the malware from all infected machines:
    - Remove the malicious executable and temporary files.

- Clean up the registry entries used for persistence.
- Patch vulnerabilities that allowed the malware to infiltrate the network in the first place.

4. **Improve Defences**: Based on the malware's behaviour, the organisation improves endpoint detection by enhancing monitoring of abnormal process executions, encryption APIs, and network traffic analysis for unusual patterns.

4. Cloud Security

Scenario: Securing a Cloud Environment on AWS

Background: An organisation has migrated its infrastructure to Amazon Web Services (AWS). It hosts several web applications, databases, and storage in the cloud. The cybersecurity team is responsible for securing the cloud environment, monitoring for suspicious activity, and ensuring compliance with industry security standards like ISO 27001 and SOC 2.

Step 1: Securing Cloud Infrastructure

- **Tools**: AWS IAM, Security Groups, AWS Config, AWS CloudTrail

1. **Review Identity and Access Management (IAM) Policies**:
    - The security team reviews IAM policies to ensure the principle of least privilege is followed. This limits access to only what is necessary for users and services to function.
    - **Findings**:
        - **Overly Permissive Roles**: The analyst discovers that some IAM roles assigned to developers have full access to all S3 buckets (s3:*) when they only need read-only access to one specific bucket.
    - **Remediation**:
        - The team updates IAM roles and policies to restrict access. For example:

```
{
  "Effect": "Allow",
  "Action": "s3:GetObject",
  "Resource": "arn:aws:s3:::specific-bucket/*"
}
```

2. **Configure Security Groups and Network ACLs**:
    - The security team ensures that Security Groups and Network Access Control Lists (ACLs) are properly configured to limit inbound and outbound traffic to the cloud environment.
    - **Findings**:

        - An EC2 instance hosting a web application was found to have open ports (e.g., 0.0.0.0/0 for SSH and RDP), exposing the server to the internet.

    - **Remediation**:

        - The team restricts the Security Group rules to allow access only from trusted IP addresses (e.g., the internal office IP):

```
"IpRanges": [
 {
  "CidrIp": "203.0.113.0/24",
  "Description": "Office IP Range"
 }
]
```

Step 2: Monitoring Cloud Traffic and Activity

- **Tools**: AWS CloudTrail, AWS GuardDuty, AWS CloudWatch

1. **Enable AWS CloudTrail for Activity Logging**:

    - The team ensures that AWS CloudTrail is enabled to log API calls and user activities across all AWS services. This helps track who did what and when within the cloud environment.

    - **Findings**:

        - **Suspicious API Calls**: The logs show multiple failed attempts to change IAM roles from an IP address that doesn't belong to the organisation. The attacker appears to be trying to escalate privileges.

    - **Remediation**:

        - The team adds the suspicious IP to a deny list and enables multi-factor authentication (MFA) for all administrative accounts to prevent unauthorised access.

2. **Use AWS GuardDuty to Detect Anomalous Behaviour**:

    - AWS GuardDuty is set up to monitor for anomalies such as unusual traffic patterns, unexpected behaviour, and potential threats.

    - **Findings**:

        - GuardDuty flags an instance for connecting to a known malicious IP address, indicating a potential compromise.

    - **Remediation**:

        - The team isolates the compromised instance by modifying its Security Group and terminates it after investigating. They also use forensic tools to analyse the instance and determine the cause of the compromise.

Step 3: Compliance and Governance

- **Tools**: AWS Config, AWS Artifact, Third-Party Compliance Tools

1. **Ensure Compliance with Security Policies**:

    - The organisation must comply with ISO 27001 and SOC 2 standards. The security team uses AWS Config to ensure that cloud resources comply with the company's security policies and industry regulations.

    - **Findings**:

        - **Non-Compliant Resources**: AWS Config flags several S3 buckets that are publicly accessible, violating the organisation's data protection policy.

    - **Remediation**:

        - The team uses AWS Config rules to automatically remediate non-compliant resources. For example, a rule is created to ensure all S3 buckets have encryption enabled and public access is blocked by default.

2. **Manage Compliance Documentation with AWS Artifact**:

    - The team uses AWS Artifact to access compliance reports and documentation from AWS, ensuring they meet external regulatory requirements for audits.

    - **Findings**:

        - Some areas of the cloud infrastructure, like database encryption and backup policies, need to be enhanced to align with SOC 2 Type II controls.

    - **Remediation**:

        - The team implements database encryption at rest using AWS KMS (Key Management Service) and establishes regular automated backups for compliance purposes.

Step 4: Incident Response and Remediation

- **Tools**: AWS Security Hub, AWS Lambda, Incident Response Playbooks

1. **Respond to Security Incidents**:
    - The team sets up AWS Security Hub to aggregate security findings across AWS services and implement automated responses using Lambda functions.
    - **Findings**:
        - A Lambda function is triggered after GuardDuty detects an instance communicating with a suspicious IP address. The function automatically quarantines the instance and notifies the security team.
    - **Remediation**:
        - The compromised instance is investigated, and a root cause analysis reveals that it was exploited due to a vulnerable third-party application running on the instance. The security team patches the application and applies additional controls to prevent similar incidents in the future.
2. **Refine Security Policies**:
    - After the incident, the team reviews their incident response playbooks and updates them to include lessons learned from the attack. They implement additional logging and monitoring for high-risk resources and adjust network ACLs to block access to known malicious IP ranges.

5. Compliance and Governance

Scenario: Ensuring Compliance and Governance for a Healthcare Cloud Environment

Background: A healthcare organisation is using Amazon Web Services (AWS) to store and process sensitive patient data, including electronic health records (EHRs). They must comply with GDPR (for EU patients), HIPAA (for US patients), and ISO 27001 for security management. The organisation needs to ensure that its cloud infrastructure is secure and compliant with these standards.

Step 1: Compliance Framework Assessment

- **Tools**: AWS Config, AWS Artifact, GDPR Compliance Checklist, HIPAA Security Rule

1. **Identify Compliance Requirements**:
   The organisation reviews the legal and regulatory requirements it must adhere to. For example:
   - **GDPR**: Protects personal data of EU citizens, requiring strong data protection measures and the ability to respond to data breaches within 72 hours.
   - **HIPAA**: Imposes strict rules on the handling of healthcare information, including the encryption of protected health information (PHI) in transit and at rest.
   - **ISO 27001**: Requires the implementation of an information security management system (ISMS) to protect sensitive information.
2. **Gap Analysis**:
   The cybersecurity team performs a gap analysis to assess the current state of the cloud infrastructure and identify areas that do not meet compliance requirements.
   - **Findings**:
     - Several S3 buckets containing patient data are not encrypted at rest, which violates both HIPAA and ISO 27001 requirements.
     - GDPR Article 17 (Right to Erasure) is not fully implemented, meaning there is no automated way to delete patient data upon request.
3. **Remediation Plan**:
   The team develops a remediation plan to address the gaps:
   - Encrypt all S3 buckets using AWS KMS.

- Implement a data deletion policy to comply with GDPR.

Step 2: Configuring and Automating Compliance Controls

- **Tools**: AWS Config, AWS IAM, AWS Key Management Service (KMS)

1. **Encrypting Sensitive Data**:
   All EHR data stored in AWS S3 buckets must be encrypted. The team configures server-side encryption using AWS KMS keys.
   - **Action**:
     - Enable encryption for all S3 buckets with the following policy:

```
{
 "Effect": "Deny",
 "Principal": "*",
 "Action": "s3:PutObject",
 "Resource": "arn:aws:s3:::ehr-bucket/*",
 "Condition": {
  "StringNotEquals": {
   "s3:x-amz-server-side-encryption": "aws:kms"
  }
 }
}
```

   - **Result**: All data in S3 is now encrypted at rest, ensuring HIPAA and ISO 27001 compliance.

2. **Automating Compliance Monitoring**:
   The team uses AWS Config to continuously monitor compliance with security policies.
   - **Action**:
     - Configure an AWS Config rule to check that all S3 buckets have encryption enabled:

```
{
 "ComplianceType": "NON_COMPLIANT",
```

"ConfigRuleName": "s3-bucket-encryption-enabled",

"RuleDescription": "Ensure all S3 buckets storing EHR data have encryption enabled"

}

- **Result**: AWS Config now automatically flags any S3 bucket that is non-compliant, providing real-time monitoring and alerts.

Step 3: Data Protection and Privacy (GDPR Compliance)

- **Tools**: AWS Lambda, AWS DynamoDB, GDPR Compliance Framework

1. **Implementing the Right to Erasure**:
   Under GDPR, patients have the right to request the deletion of their personal data. The team implements a solution to automate data deletion across the cloud environment.

   - **Action**:

     - The team creates an AWS Lambda function that listens to GDPR erasure requests and deletes the relevant patient data from DynamoDB and S3.

     - DynamoDB and S3 objects are tagged with patient IDs, allowing the Lambda function to identify and delete data linked to a specific individual:

```
def delete_patient_data(patient_id):

  dynamodb.delete_item(Key={'PatientID': patient_id})

  s3.delete_objects(Bucket='ehr-bucket', Prefix=f'{patient_id}/')
```

   - **Result**: The organisation is now compliant with GDPR Article 17, ensuring that patient data can be erased upon request.

2. **Data Access Logs (GDPR & HIPAA)**:
   The team sets up CloudTrail to log all access to patient data, ensuring that audit logs are available for compliance with GDPR and HIPAA.

   - **Action**:

     - Configure CloudTrail to log all data access events and store logs in an encrypted S3 bucket:

```
{

 "Effect": "Allow",

 "Action": "cloudtrail:WriteLogs",
```

```
  "Resource": "arn:aws:s3:::cloudtrail-logs-bucket"
}
```

- **Result**: Detailed logs are stored for every access request made to patient data, ensuring that the organisation can provide audit trails when required by regulators.

Step 4: Auditing and Reporting

- **Tools**: AWS Artifact, Compliance Reporting Tools, Audit Documentation

1. **Audit Preparation**:
   The organisation prepares for an external audit to demonstrate compliance with ISO 27001 and HIPAA. AWS Artifact is used to access audit reports, certifications, and other compliance documentation.
   - **Action**:
     - Use AWS Artifact to download the SOC 2 report and ISO 27001 certifications.
     - Generate internal reports on S3 bucket encryption, access logs, and GDPR deletion requests.
   - **Result**: The organisation gathers all necessary evidence to demonstrate compliance with external auditors, reducing audit preparation time and ensuring transparency.
2. **Conducting Internal Audits**:
   Regular internal audits are conducted to ensure that the implemented security controls are working as expected.
   - **Action**:
     - The team schedules quarterly internal audits using predefined checklists for HIPAA and GDPR compliance.
     - Key areas of focus include data encryption, access control, logging, and privacy controls.
   - **Result**: Continuous internal audits ensure ongoing compliance and reduce the risk of non-compliance with legal and regulatory requirements.

Step 5: Incident Response and Reporting

- **Tools**: Incident Response Playbook, AWS Security Hub, AWS CloudWatch

1. **Data Breach Response (GDPR & HIPAA)**:
   In the event of a data breach involving patient data, the organisation must respond swiftly to notify regulators and affected individuals, as required by GDPR and HIPAA.

   - **Action**:

     - The security team uses AWS Security Hub to detect any anomalies indicating a potential data breach.

     - Upon detecting unauthorised access to patient data, an incident response is triggered following the GDPR breach notification guidelines.

   - **Remediation**:

     - The team mitigates the breach by revoking access to compromised accounts, restoring affected systems, and working with AWS support to analyse the breach.

   - **Result**: A data breach report is compiled, and notifications are sent to the relevant authorities within 72 hours, as required by GDPR. Additionally, HIPAA breach notification rules are followed, ensuring that all parties are informed.

2. **Review and Improve Incident Response**:
   After the incident, the organisation reviews the incident response playbook to incorporate lessons learned and enhance the response process for future incidents.

   - **Action**:

     - Update incident response workflows to include faster breach detection and more detailed reporting processes.

   - **Result**: The organisation enhances its incident response capability, reducing the risk of future non-compliance due to delayed breach notifications.