Shadow Sentry

SOC Project

Creating a Security Operations Center (SOC) project to install Elastic Cloud and Honeypot on DigitalOcean to detect and analyze malicious activity on your network, and develop pentest scripts to attack the honeypot and monitoring the alerts.

Project Objectives:

1. **Deploy Elastic Cloud on DigitalOcean:**
    - Set up Elastic Cloud on DigitalOcean.
    - Use Referral code: https://m.do.co/c/2099826f2433
        - $200 for 60 days
    - Configure the Elastic Stack components: Elasticsearch, Logstash, Kibana (ELK Stack).
    - Establish connectivity and ensure proper communication between the components.
2. **Choose Honeypot Solution(s) and Install on Digital Ocean:**
    - Choose a honeypot solution.
    - Possible Honeypots: Cowrie, Honeyd, Glastopf, Kippo
    - Deploy honeypot servers on DigitalOcean instances.
    - Configure network settings
        - Network Configuration: Ensure that the honeypot server is configured to listen on the desired network interfaces and ports. You may want to simulate common services such as SSH, Telnet, FTP, HTTP, or SMB.
        - Logging Configuration: Configure logging settings to capture all incoming traffic, interactions, and attempted exploits.
    - Simulate a realistic environment with web servers, databases, and other network services.
3. **Harden the Honeypot Server (You don't want this to be compromised)**
    - Disable unnecessary services and ports.
    - Apply security updates regularly.
    - Implement strong passwords and access controls.
    - Configure firewall rules using UFW or iptables to restrict incoming and outgoing traffic.

4. **Create Attacks Scripts:**
    - Create attack scripts that can simulate at least three (3) different attack types using functions.
    - Each attack should have a description to display once chosen
    - The system should display the IP addresses on the network
    - Display a list of all possible attacks with descriptions
    - The user can choose a specific attack or random from the list
    - If the user enters a different key, display a message and exit
    - For each attack, allow the user to choose a target or random from the found Ips
    - Everything other than the user input should be automated.
    - Use functions.
    - Possible tools: Nmap, Hydra, Masscan, Msfconsole, Hping3, Arpspoof etc.
5. **Integration with Elastic Stack:**
    - Configure Logstash to ingest logs from the sample infrastructure.
    - Integrate Elasticsearch for storing and indexing logs.
6. **Alerting and Monitoring:**
    - Define alerting rules based on security best practices and known attack patterns.
    - Configure Kibana dashboards to visualize real-time security events and alerts.
7. **Testing, Validation and Logging:**
    - Execute penetration testing scripts against the sample infrastructure.
    - Monitor the Elastic Stack for generated alerts and security events.
    - Validate the effectiveness of the alerting and monitoring system.
    - On attack selection, save it into a log file in /var/log
    - The log should hold the kind of attack, time of execution, and IP addresses
8. **Documentation and Reporting:**
    - Document the setup process, including configurations and settings.
    - Create a report detailing the findings, including detected security events and alerts.
    - Provide recommendations for improving the security posture based on the observed vulnerabilities.

Project Deliverables:

1. **Installation Guide:**
    - Step-by-step guide for deploying Elastic Cloud on DigitalOcean.
    - Instructions for configuring the Elastic Stack components.
2. **Pentest Scripts:**
    - Scripts for simulating various security attacks.
    - Documentation explaining the purpose and usage of each script.
3. **Configuration Files:**
    - Logstash configuration files for parsing and enriching logs.
    - Alerting rules and configurations.
4. **Dashboard and Visualizations:**
    - Kibana dashboards for visualizing security events and alerts.
    - Custom visualizations to track key security metrics.
5. **Testing Results:**
    - Report on the execution of pentest scripts and observed security events.
    - Analysis of the effectiveness of the alerting and monitoring system.
6. **Final Report:**
    - Comprehensive report summarizing the project objectives, methodologies, findings, and recommendations.
7. **Video Recording of Individual Presentation** (optional)
8. **Comments**
    - Use comments in your code to explain what you did. If you are using code from the internet, add credit and links. In the script, write the student's name and code, the class code, and the lecturer's name.
9. **Submission**
    - Submit the source code (.sh) and a PDF file with screenshots proving the functions work.
    - Send the project to the trainer’s email. In the email subject type project.