Activity: Apply OS Hardening Techniques

Apply OS Hardening Techniques (Part 1)

Activity Overview

In this activity, I took on the role of a cybersecurity analyst working for a company hosting the cooking website, **yummyrecipesforme.com**. Visitors to the website experienced a security issue when loading the main webpage. My responsibilities included investigating, identifying, documenting, and recommending a solution to the security problem.

During the investigation, I analyzed a **tcpdump log** to identify network protocols involved in the connection between users and the website. This step was crucial, as malicious actors often exploit network protocols to invade networks. Understanding these protocols helped in determining how to secure the website against such attacks.

I documented the security incident and proposed a recommendation to prevent similar issues in the future, focusing on mitigating brute force attacks.

Activity Details

Scenario Summary

A disgruntled baker executed a brute force attack to gain access to the admin account of **yummyrecipesforme.com** by guessing the default password. Once logged in, the attacker modified the website's source code to embed malicious JavaScript. Visitors to the website were prompted to download and run a fake browser update, which redirected them to a counterfeit site, **greatrecipesforme.com**, where proprietary recipes were made available for free.

Steps Taken

1. **Network Analysis**
    - Conducted a sandboxed investigation using **tcpdump** to capture network traffic packets.
    - Observed the following sequence of events:
        1. DNS resolution for yummyrecipesforme.com provided the correct IP address.
        2. HTTP request established a connection to the server.
        3. A malicious file download was initiated and executed.
        4. DNS resolution for greatrecipesforme.com redirected traffic to a new IP.
        5. HTTP request redirected users to the fake site.
2. **Source Code Review**
    - Identified malicious JavaScript code prompting users to download and execute a malicious file.
    - Determined the attacker gained access through a brute force attack, exploiting the admin account's default password.
3. **Documentation**
    - Recorded all observed actions, affected users, and the network traffic behavior.
4. **Remediation Proposal**
    - Suggested measures to prevent brute force attacks and mitigate future risks.

Apply OS Hardening Techniques (Part 2)

Section 1: Identify the network protocol involved in the incident
The network protocol used in this incident was **HTTP** (Hypertext Transfer Protocol). The **tcpdump logs** confirmed that HTTP traffic facilitated both the initial website interaction and the download of the malicious file. This traffic operated at the application layer of the TCP/IP model.

Section 2: Document the incident

Incident Summary

Customers reported being prompted to download a file when visiting the website. After running the file, their systems slowed down, and they were redirected to a counterfeit website.
A sandboxed investigation with **tcpdump** confirmed the attack sequence, revealing that a brute force attack allowed the attacker to access the admin panel, embed malicious JavaScript, and compromise visitors' systems.

Key Observations

- Initial DNS resolution to yummyrecipesforme.com.
- Malicious JavaScript prompted the download of a file via HTTP.
- DNS resolution to greatrecipesforme.com redirected traffic to the counterfeit site.
- Logs showed unauthorized changes to the website's source code.

Cause of the Incident

- The admin account's password was set to the default, enabling a successful brute force attack.
- Lack of measures to detect or prevent such attacks.

The senior cybersecurity professional analyzed the source code for the websites and the downloaded file. The analyst discovered that an attacker had manipulated the website to add code that prompted the users to download a malicious file disguised as a browser update. Since the website owner stated that they had been locked out of their administrator account, the team believes the attacker used a brute force attack to access the account and change the admin password. The execution of the malicious file compromised the end users' computers.

Section 3: Recommend one or more remediations for brute force attacks
To mitigate future brute force attacks: 1. **Require Strong Passwords** o Disallow default or previously used passwords. 2. **Enforce Two-Factor Authentication (2FA)** o Add a second layer of security requiring a one-time passcode (OTP) sent to email or phone. 3. **Monitor Login Attempts** o Implement mechanisms to detect and limit multiple failed login attempts. 4. **Frequent Password Updates** o Require periodic password changes to minimize exposure to unauthorized access.

Conclusion

This project highlights the critical importance of proactive cybersecurity measures to safeguard against brute-force attacks and malicious website manipulations. By thoroughly investigating the security incident, analyzing network traffic using tools like **tcpdump**, and documenting the event in detail, I was able to identify vulnerabilities and recommend actionable steps to enhance security.

The remediation strategies—such as enforcing strong passwords, implementing two-factor authentication, and monitoring login attempts—serve as practical solutions to prevent similar incidents in the future. This project demonstrates my ability to approach cybersecurity challenges with a methodical, evidence-driven process, ensuring the protection of sensitive systems and user data.