|  |  |
| --- | --- |
| Name | Vinoth Kumar R |
| College | Vel Tech Rangarajan Dr.Sagunthala R&D Institute of Science and Technology |
| Project Title | Intelligent Threat Detection And Response: AI Integration In Cybersecurity Frameworks |
| Contact Number | 9600837447 |

**Report**

Stage:2

1. **Nessus Plugins & Threat Detections And Response**

**Vulnerability Name: HTTP Server Type and Version**

**plugin : 10107**

**Severity : Info**

**Description :**  This plugin attempts to determine the type and the version of the remote web server.

The Nessus plugin 10107 indicates that the Nessus vulnerability scanner has detected information about the HTTP server type and version running on a target system. This type of information disclosure can potentially pose security risks as it provides attackers with insights into the software versions and configurations, which could be exploited to identify and target known vulnerabilities.

**Business Impact:**

**Increased Exposure to Attacks:**

Knowledge of the HTTP server type and version may expose the system to targeted attacks, as attackers can specifically target known vulnerabilities associated with that server software.

**Potential for Exploitation:**

Attackers may use the disclosed server information to tailor attacks that are more likely to succeed based on known vulnerabilities in the identified server type and version.

**Risk of Unpatched Systems:**

If the identified server version is outdated or unpatched, there is an increased risk of exploitation since known vulnerabilities may not have been addressed.

**Reputation Damage:**

Customers and users may lose trust in a company's security practices if they find out that outdated or vulnerable server software is being used.

1. **Vulnerability Name: HyperText Transfer Protocol (HTTP) Information**

Nessus plugin ID: 24260

Severity : Info

Description : This test gives some information about the remote HTTP protocol - the version used, whether HTTP Keep-Alive and HTTP pipelining are enabled, etc...

As of my last knowledge update in January 2023, Nessus plugin 24260 is related to the detection of the TRACE/TRACK Methods Allowed vulnerability in the HTTP server. The TRACE and TRACK methods are used for diagnostic purposes in the HTTP protocol, but their presence can lead to security vulnerabilities.

1. Vulnerability Name: TRACE/TRACK Methods Allowed (Nessus Plugin 24260)

Business Impact:

**Cross-Site Tracing (XST) Attacks:**

The TRACE method can be exploited for Cross-Site Tracing attacks, potentially leading to the theft of sensitive information such as authentication credentials.

**Cross-Site Scripting (XSS):**

TRACE method vulnerabilities might be leveraged to perform Cross-Site Scripting attacks or other malicious activities.

1. **Vulnerability Name: Debian DSA-5020-1 : apache-log4j2 - security update**

Solution:

To address the TRACE/TRACK Methods Allowed vulnerability, consider the following solutions:

1. Disable TRACE and TRACK Methods:

Disable the TRACE and TRACK methods on the HTTP server. This can be done by configuring the server to reject or ignore these methods. This measure prevents potential exploitation of these methods for security attacks.

1. Web Server Configuration:

Review and adjust the server configuration to disallow TRACE and TRACK methods. The exact steps depend on the web server software in use (e.g., Apache, Nginx, IIS).

1. Web Application Firewall (WAF):

Implement a Web Application Firewall that can detect and block malicious traffic, including attempts to exploit TRACE/TRACK vulnerabilities.

1. Regular Security Audits:

Conduct regular security audits to identify and address vulnerabilities promptly. Automated tools like Nessus can be part of a regular scanning regimen to detect and report vulnerabilities.

1. Update and Patch:

Keep the web server software up to date with the latest patches. Ensure that the software is configured securely to minimize the risk of vulnerabilities.

1. Security Headers:

Implement security headers, such as the HTTP Strict Transport Security (HSTS) header, Content Security Policy (CSP), and X-Frame-Options, to enhance the overall security posture of the web application.

1. Network Segmentation:

Consider network segmentation to isolate critical servers from potential attackers. This can help contain and limit the impact of successful attacks.

1. ***Vulnerability Name: Service Detection (Nessus Plugin 22964)***

**Business Impact:**

1. Information Disclosure:

Service detection provides information about the services and their versions running on a system. While not a vulnerability on its own, this information can be used by attackers for targeted attacks.

1. Risk of Targeted Attacks:

Knowledge of specific services and versions allows attackers to focus on known vulnerabilities associated with those services. It increases the risk of targeted exploitation if the services are not properly configured or patched.

**Solution:**

To address the concerns related to service detection and reduce the associated risks:

1. Minimize Information Exposure:

Configure systems and services to minimize information exposure. Limit the details provided in service banners and responses to reduce the surface available for potential attackers.

1. Regularly Update and Patch Services:

Keep all services and software up to date with the latest patches and updates. Regularly check for security advisories related to the detected services and apply patches promptly.

1. Implement Network Segmentation:

Use network segmentation to isolate critical services and systems from less secure parts of the network. This helps contain potential compromises and limit lateral movement for attackers.

1. Network Monitoring:

Implement network monitoring solutions to detect unusual or suspicious activities related to the services. Anomaly detection and intrusion prevention systems can aid in identifying potential threats.

1. Access Controls:

Implement strong access controls to limit access to services. Ensure that only authorized users and systems can interact with critical services.

1. Security Awareness and Training:

Train personnel to be aware of the information that might be exposed through service banners and the potential risks associated with service detection. Foster a security-aware culture within the organization.

1. Security Configuration Reviews:

Conduct regular security configuration reviews to ensure that services are configured securely. Disable unnecessary services and features that may pose security risks.

1. Vulnerability Management:

Integrate service detection findings into a comprehensive vulnerability management program. Prioritize and remediate identified vulnerabilities based on their severity and potential impact.