

Cyber Threat Intelligence Analysis Report

***Vulnerd* Security Assessment**

Cyber Threat Intelligence (CTI) Analyst - Luisa Faria

Overview

As part of the Vulnerd security assessment, a vulnerability scan was conducted using Tenable Nessus Essentials to identify potential vulnerabilities present on the target system.

Although the scan primarily identified informational findings rather than critical vulnerabilities, it still revealed valuable intelligence about the system's technology stack and exposed services.

From a cyber threat intelligence perspective, this information could be used by adversaries to perform reconnaissance, identify attack vectors, and prepare targeted exploitation attempts.

The goal of this analysis is to interpret the scan results from a threat intelligence perspective, identifying how attackers could potentially leverage the discovered technologies and services.

I. System Intelligence Identified

- A. The Nessus scan revealed several important technologies and services running on the host system.

Host Information	
Attribute	Value
IP Address	100.105.13.32
Operating System	Linux Kernel 2.6
Web Server	Apache HTTP Server
Web Technology	PHP
Remote Access	SSH
Scan Duration	38 Minutes

- B. These findings provide insight into the system architecture and exposed services, which are commonly used by attackers during the reconnaissance phase of cyber attacks.

II. Key Findings from Scan

- A. The Nessus scan results primarily consisted of Informational severity findings, meaning they reveal system configuration details rather than confirmed exploitable vulnerabilities.

Examples of identified items include:

- 1) HTTP Service Detection;
- 2) SSH Service Detection;
- 3) Apache HTTP Server Identification;
- 4) PHP Version Detection;
- 5) OS Fingerprinting;
- 6) TCP/IP Timestamp Detection;
- 7) Traceroute Information;
- 8) Device Type Identification;

Although these findings are categorized as informational, they reveal valuable intelligence regarding the system's technology stack, given that attackers often rely on this type of information to profile a system and identify potential weaknesses before launching an attack.

III. Threat Intelligence Analysis

From a cyber threat intelligence perspective, the technologies discovered during the scan expose several potential attack surfaces.

A. Web Application Attack Surface

The scan confirmed that the system is running an Apache web server with PHP support, indicating the presence of a web application.

Web applications are commonly targeted by attackers through techniques such as:

- 1) SQL Injection
- 2) Cross-Site Scripting (XSS)
- 3) File Inclusion Attacks
- 4) Remote Code Execution

If the web application contains vulnerabilities, attackers could potentially exploit these weaknesses to gain unauthorized access to the system.

B. SSH Remote Access Exposure

The scan also detected an active SSH service, which enables remote administrative access to the system. If not properly secured, SSH services may be targeted through:

- 1) Brute force login attempts
- 2) Credential stuffing attacks
- 3) Exploitation of weak passwords

Threat actors frequently attempt to compromise SSH services in order to gain initial system access.

C. Outdated Operating System Risk

The scan identified the system as running Linux Kernel 2.6, which is an older kernel version. Older operating system versions are commonly associated with known vulnerabilities and may be susceptible to privilege escalation exploits. Once an attacker gains initial access to the system, they may attempt to exploit these vulnerabilities in order to obtain higher levels of system access.

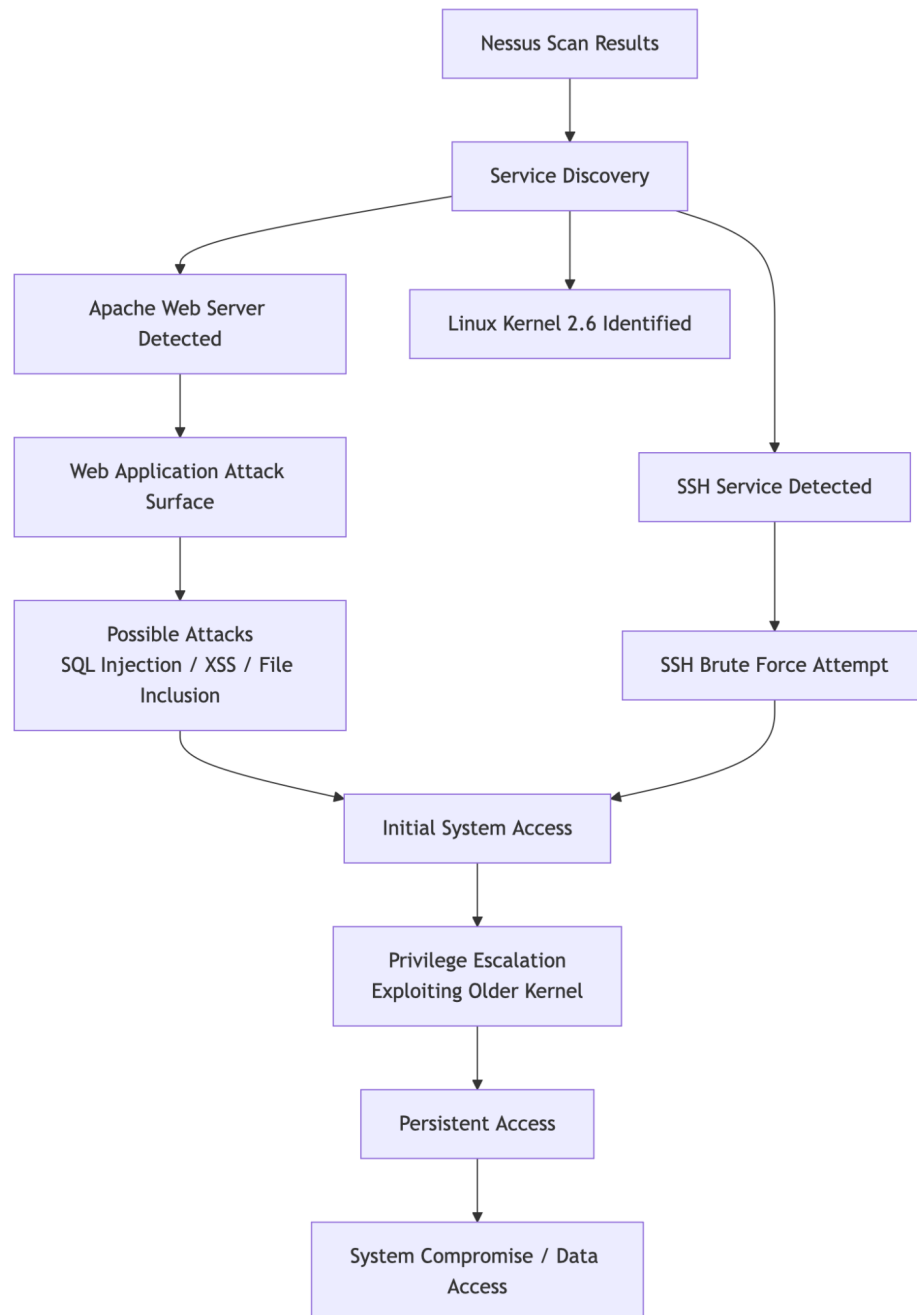
IV. MITRE ATT&CK Mapping

- A. The observed technologies and potential attack paths align with several techniques from the MITRE ATT&CK framework.

MITRE Technique	Description
T1595	Active Scanning
T1046	Network Service Discovery
T1190	Exploit Public Facing Application
T1110	Brute Force Authentication
T1059	Command Execution

- B. Mapping vulnerabilities and services to MITRE ATT&CK techniques helps defenders understand how attackers may progress through different stages of a cyber attack.

V. Example of Threat Scenario



Based on the intelligence gathered from the Nessus scan, we created an attack chain diagram that shows how an attacker could move from reconnaissance to system compromise using the services identified during the scan.

- A. Based on the intelligence gathered from the vulnerability scan, a potential attack scenario may unfold as follows:

01. An attacker performs network scanning and identifies the target system's open services.
02. The attacker discovers the presence of an Apache web server running PHP.
03. The attacker attempts to exploit vulnerabilities within the web application, such as SQL injection or file inclusion vulnerabilities.
04. After gaining initial access, the attacker attempts privilege escalation by exploiting weaknesses associated with the outdated Linux kernel.
05. The attacker establishes persistence on the system and potentially uses SSH access for continued control.

This attack chain reflects common techniques used in real-world cyber intrusions.

VI. Intelligence Based Security Recommendations

A. Based on the intelligence gathered from the scan, the following security measures are recommended:

- 1) Update the Linux kernel and system packages to supported versions.
- 2) Regularly patch Apache and PHP services.
- 3) Restrict SSH access using firewall rules or VPN access.
- 4) Implement multi-factor authentication (MFA) for remote administrative access.
- 5) Monitor system logs for repeated login attempts or reconnaissance activity.
- 6) Deploy web application security controls, such as input validation and web application firewalls.

These security improvements should be performed to help reduce the system's attack surface and improve overall security posture.

VII. CTI Contribution

The Cyber Threat Intelligence role focuses on transforming raw technical findings into actionable security insights. By analyzing the scan results and identifying how attackers might leverage discovered technologies and services, the CTI analysis helps the team understand potential attacker behavior, likely attack techniques, risk exposure of the system, and security improvements needed to reduce threats.

This intelligence-driven perspective allows the team to move beyond simple vulnerability detection and better understand the real-world security implications of the scan results.

VIII. Intelligence-Driven Detection Mapping

To strengthen the threat intelligence analysis, the findings from the Nessus scan were mapped to detection logic defined in the Vulnerd Detection Playbook. While the CTI analysis

identifies how attackers may behave, the detection playbook defines how these behaviors can be identified and responded to in practice.

For instance:

1. Web Application Exposure & Reconnaissance

The scan identified an Apache web server running PHP on a non-standard port, which provides attackers with detailed system fingerprinting information.

According to the detection playbook, this aligns with:

- Detection 4 — Web Application Exposed on Non-Standard Port
- Detection 5 — OS and Network Fingerprinting Exposure

Detection Logic:

- IF HTTP services are detected on non-standard ports AND version information is exposed
- THEN raise an alert for reconnaissance exposure

This directly supports the CTI assessment that attackers can use this information during the reconnaissance phase to prepare targeted attacks.

2. SSH Service Exposure

The scan revealed an active SSH service, which introduces risk of unauthorized access attempts.

This aligns with:

- Detection 10 — Brute Force Authentication Attack (manual testing)

Detection Logic:

- IF multiple failed login attempts occur followed by a successful login
- THEN raise a brute-force alert

This supports the CTI analysis that SSH is a common entry point for attackers using credential-based attacks.

3. Web Application Attack Surface

The CTI analysis identified risks such as SQL injection, XSS, and command injection.

These are directly supported by multiple detections in the playbook:

- Detection 11 — Command Injection
- Detection 12 — SQL Injection
- Detection 13 — Cross-Site Scripting (XSS)
- Detection 14 — File Inclusion
- Detection 15 — Unrestricted File Upload

Detection Logic:

- IF user input contains SQL keywords or script tags
- THEN raise a critical or high alert

This reinforces that the identified attack surfaces are not only theoretical but were successfully exploited during testing, highlighting a significant security risk.

4. Key Insight — Detection Gap

A critical insight from the detection playbook is that:

- 6 out of 15 detections (40%) were only identified through manual testing and not by Nessus scans

This includes major vulnerabilities such as:

- Command Injection
- SQL Injection
- Unrestricted File Upload

This demonstrates that while vulnerability scanners provide valuable system-level visibility, they may fail to detect application-layer vulnerabilities.

5. CTI + Detection Integration

By combining CTI analysis with detection playbook logic:

- CTI explains what attackers will do
- Detection playbook defines how to catch them
- Together, they enable faster and more effective response

This integration reflects a real-world security approach where threat intelligence, detection engineering, and response planning work together.