

Module 05

Vulnerability Analysis

Learning Objectives

01

Summarize Vulnerability Assessment Concepts

02

Use Vulnerability Assessment Tools

03

Analyze Vulnerability Assessment Reports

Objective 01

Summarize Vulnerability Assessment Concepts

Vulnerability Classification

Vulnerability Type	Description	Examples
Misconfigurations/Weak Configurations	<ul style="list-style-type: none"> Misconfigurations occur when systems, applications, or devices are not configured correctly, leaving them susceptible to exploitation It allows attackers to break into a network and gain unauthorized access to systems 	Network Misconfigurations <ul style="list-style-type: none"> Insecure protocols, open ports and services, errors, and weak encryption
		Host Misconfigurations <ul style="list-style-type: none"> Open permissions and unsecured root accounts
Application Flaws	<ul style="list-style-type: none"> Application flaws are vulnerabilities in applications that are exploited by attackers Flawed applications pose security threats such as data tampering and unauthorized access to configuration stores 	<ul style="list-style-type: none"> Buffer overflows, memory leaks, resource exhaustion, integer overflows, null pointer/object dereference, DLL injection, race conditions, improper input handling, improper error handling, and code signing weakness
Poor Patch Management	<ul style="list-style-type: none"> Software vendors provide patches that prevent exploitations and reduce the probability of threats exploiting a specific vulnerability Unpatched software can make an application, server, or device vulnerable to various attacks 	<ul style="list-style-type: none"> Unpatched servers, unpatched firmware, unpatched OS, and unpatched applications
Design Flaws	<ul style="list-style-type: none"> Logical flaws in the functionality of the system are exploited by the attackers to bypass the detection mechanism and acquire access to a secure system 	<ul style="list-style-type: none"> Incorrect encryption and poor validation of data
Third-Party Risks	<ul style="list-style-type: none"> Third-party services can have access to privileged systems and applications, through which financial information, customer and employee data, and processes in the enterprise's supply chain can be compromised 	<ul style="list-style-type: none"> Vendor management, supply-chain risks, outsourced code development, data storage, and cloud-based vs. on-premises risks

Vulnerability Scoring Systems and Databases

Common Vulnerability Scoring System (CVSS)

- CVSS helps capture the principal characteristics of a vulnerability and produces a numerical score to reflect its severity

CVSS Ratings

Severity	Base Score Range
None	0.0
Low	0.1-3.9
Medium	4.0-6.9
High	7.0-8.9
Critical	9.0-10.0

<https://www.first.org>

National Vulnerability Database (NVD)

- NVD is the U.S. government repository of standards-based vulnerability management data
- NVD performs an analysis on CVEs that have been published to the CVE Dictionary

<https://nvd.nist.gov>

Common Weakness Enumeration (CWE)

- CWE is a category system for software vulnerabilities and weaknesses
- CWE's over 600 categories of weaknesses provide an effective baseline for the community's identification, mitigation, and prevention efforts

<https://cwe.mitre.org>

Common Vulnerabilities and Exposures (CVE)

- CVE® is a publicly available and free-to-use list or dictionary of standardized identifiers for common software vulnerabilities and exposures

<https://cve.mitre.org>

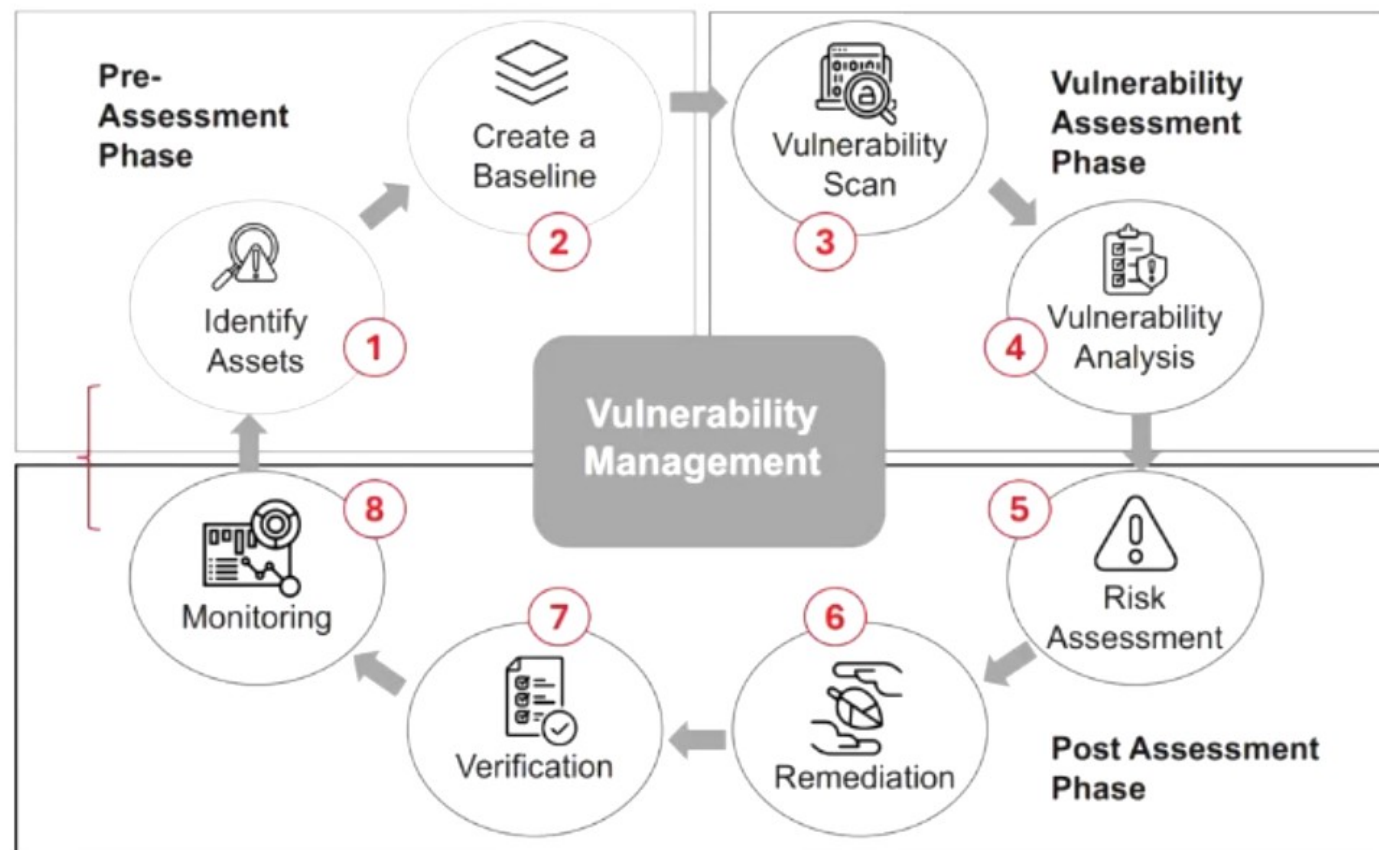


Search Results

There are 163 CVE Records that match your search.

Name	Description
CVE-2024-27632	** REJECT ** DO NOT USE THIS CANDIDATE NUMBER. ConsultIDs: CVE-2024-22939. Reason: This candidate is a duplicate of CVE-2024-22939. Notes: All CVE users should reference CVE-2024-22939 instead of this candidate.
CVE-2024-27646	langchain_experimental: (aka LangChain Experimental) in LangChain before 0.1.8 allows an attacker to bypass the CVE-2023-44467 fix and execute arbitrary code via the __import__, __subclass__, __builtins__, __globals__, __getattr__, __bases__, __mro__, or __base__ attribute in Python code. These are not prohibited by pal_chain/base.py.
CVE-2024-27316	An issue was discovered in pycryptlib 1.x before 1.0.23, 2.x before 2.0.47, and 3.x before 3.0.36. An attacker can construct a malformed certificate containing an extremely large prime to cause a denial of service (CPU consumption for an iPrime primality check). NOTE: this issue was introduced when attempting to fix CVE-2023-27569.
CVE-2024-27318	Versions of the package onnx before and including 1.15.0 are vulnerable to Directory Traversal as the external_data field of the tensor proto can have a path to the file which is outside the model current directory or user-provided directory. The vulnerability occurs as a bypass for the patch added for CVE-2022-25862.
CVE-2024-27215	** REJECT ** DO NOT USE THIS CANDIDATE NUMBER. ConsultIDs: CVE-2024-1709. Reason: This candidate is a duplicate of CVE-2024-1709. Notes: All CVE users should reference CVE-2024-1709 instead of this candidate. All references and descriptions in this candidate have been removed to prevent accidental usage.
CVE-2024-27089	** REJECT ** This candidate was withdrawn by its CNA. Further investigation showed that it was not in the allowed scope of that CNA's CVE ID assignments.

Vulnerability-Management Life Cycle



Vulnerability Research

An administrator needs vulnerability research:

- 1 To gather information concerning **security trends, threats, attack surfaces**, attack vectors and techniques
- 2 To discover **weaknesses** in the OS and applications, and alert the network administrator before a **network attack**
- 3 To **gather information** to aid in the prevention of security issues
- 4 To know **how to recover** from a network attack

Microsoft MSRC | Security Updates Acknowledgements

MSRC > Customer Guidance > [Security Update Guide](#) **Microsoft Security Response Center (MSRC)**

Security Update Guide [Subscribe](#) [RSS](#) [PowerShell](#) [API](#)

The Microsoft Security Response Center (MSRC) investigates all reports of security vulnerabilities affecting Microsoft products and services, and provides the information here as part of the ongoing effort to help you manage security risks and help keep your systems protected.

AB Deployments Vulnerabilities Advisories

Select date range [Edit columns](#) [Download](#) [Filters](#)

Keyword Product Fam. Max Severity Impact Platform Release notes

Release	Product	Platform	Impact	Max Severity	Article	Download
Feb 20, 2024	Microsoft Defender for Endpoint for Windows	Windows 10 Version 22H2 for x64-based Systems	Elevation of Privilege	Important	SO32189	Security Update
Feb 20, 2024	Microsoft Defender for Endpoint for Windows	Windows 11 Version 22H2 for ARM64-based Systems	Elevation of Privilege	Important	SO32190	Security Update
Feb 20, 2024	Microsoft Defender for Endpoint for Windows	Windows 11 Version 22H2 for ARM64-based Systems	Elevation of Privilege	Important	SO32190	Security Update
Feb 20, 2024	Microsoft Defender for Endpoint for Windows	Windows Server 2012 R2 (Server Core installation)	Elevation of Privilege	Important	SO34819	Monthly Rollup
Feb 20, 2024	Microsoft Defender for Endpoint for Windows	Windows Server 2012 R2	Elevation of Privilege	Important	SO34819	Monthly Rollup
Feb 20, 2024	Microsoft Defender for Endpoint for Windows	Windows Server 2016 (Server Core installation)	Elevation of Privilege	Important	SO32187	Security Update
Feb 20, 2024	Microsoft Defender for Endpoint for Windows	Windows Server 2016	Elevation of Privilege	Important	SO32187	Security Update
Feb 20, 2024	Microsoft Defender for Endpoint for Windows	Windows 10 Version 1807 for x64-based Systems	Elevation of Privilege	Important	SO32187	Security Update
Feb 20, 2024	Microsoft Defender for Endpoint for Windows	Windows 10 version 1807 for 32-bit Systems	Elevation of Privilege	Important	SO32187	Security Update

Vulnerability Scanning and Analysis

1

Vulnerability scanning involves analyzing protocols, services, and configurations to **discover vulnerabilities and design flaws** that may expose an operating system and its applications to exploitation, attack, or misuse

2

Vulnerability analysis is the systematic process of **identifying, evaluating, and prioritizing** security weaknesses in systems, networks, applications, or protocols

3

Vulnerabilities are classified based on **severity level** (low, medium, or high) and **exploit range** (local or remote)

4

The goal of this analysis is to understand the **nature of these vulnerabilities**, assess their **potential impact**, and develop strategies to mitigate or eliminate them

Types of Vulnerability Scanning

1 External Scanning

Scans the **network** from a hacker's perspective to discover exploits and vulnerabilities that are accessible to the outside world

4 Network-based Scanning

Determines possible **network security attacks** that may occur on the organization's systems

7 Non-Credentialed Scanning

A security testing method that assesses systems, networks, and applications without using valid credentials to log into the target system

2 Internal Scanning

Scans the **internal infrastructure** to discover exploits and vulnerabilities

5 Application Scanning

Tests and analyzes all elements of the **web infrastructure** for any **misconfiguration, outdated content, or known vulnerabilities**

8 Manual Scanning

Manually **identifying, evaluating, and validating** security vulnerabilities in systems, networks, and applications

3 Host-based Scanning

Conducts a **configuration-level check** to identify system configurations, user directories, file systems, registry settings, etc., to evaluate possibility of compromise

6 Credentialed Scanning

Scanner logs into the target system **using valid credentials** to perform a more thorough and comprehensive scan

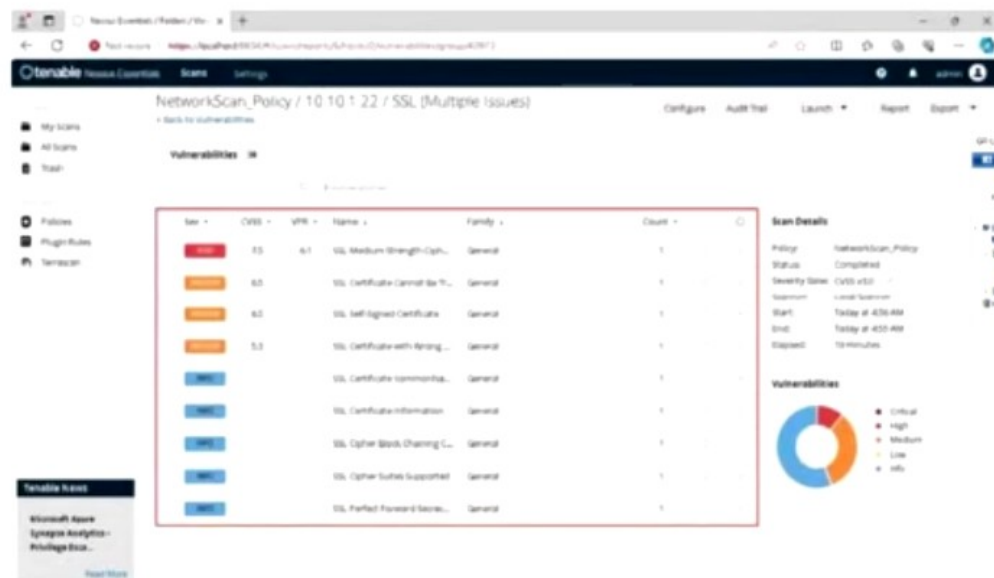
9 Automated Scanning

Uses automated software tools such as **Nessus, Qualys, and GFI LanGuard** to systematically identify, evaluate, and report security vulnerabilities

Objective **02**

Use Vulnerability Assessment Tools

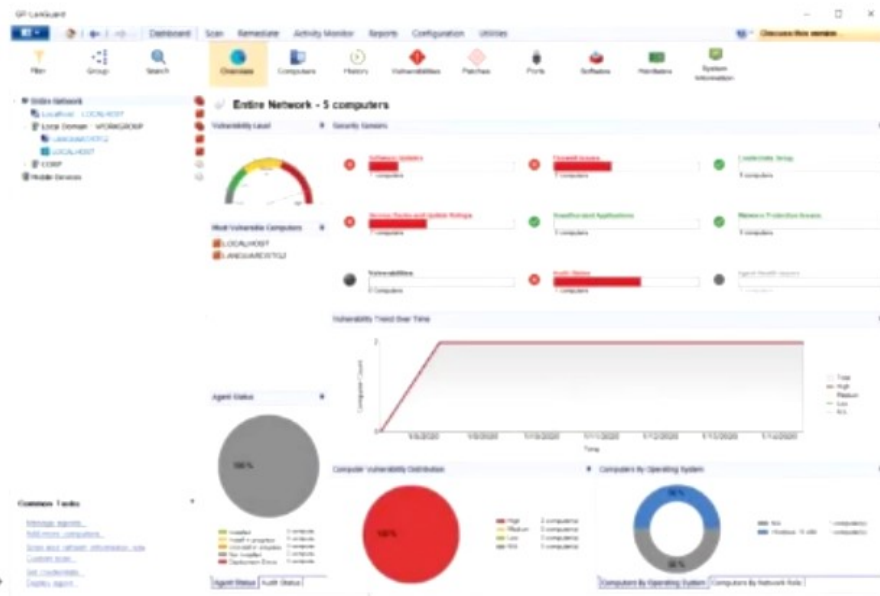
Vulnerability Assessment Tools: Nessus Essentials and GFI LanGuard



<https://www.tenable.com>

GFI LanGuard scans, detects, assesses, and rectifies **security vulnerabilities** in a network and connected devices

Nessus Essentials is an assessment solution for identifying **vulnerabilities, configuration issues, and malware**, which can be used to penetrate networks



<https://www.gfi.com>

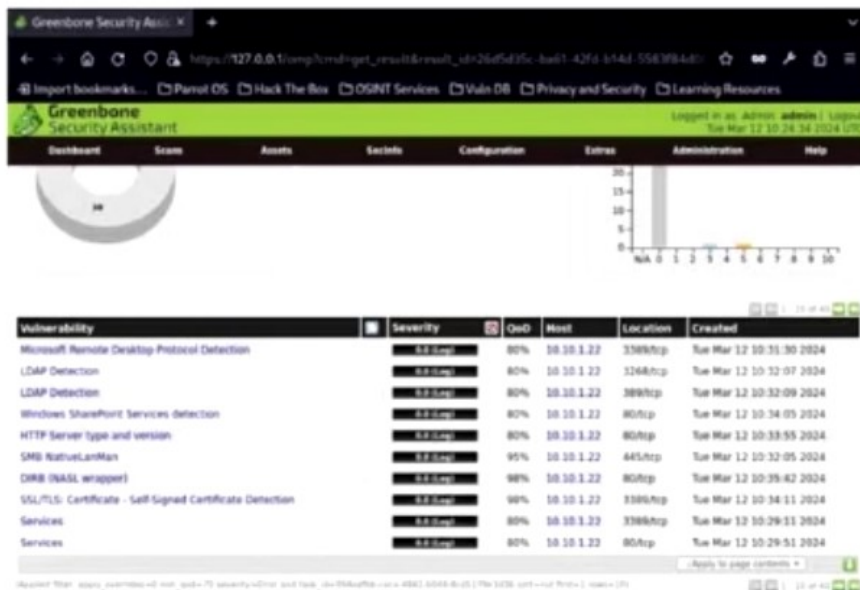
Vulnerability Assessment Tools: OpenVAS and Nikto

OpenVAS

A framework of several services and tools offering a comprehensive and powerful **vulnerability scanning** and **vulnerability management** solution

Nikto

A **web server assessment** tool that examines a web server to discover potential problems and security vulnerabilities



<https://www.openvas.org>

```

nikto -h https://www.certifiedhacker.com -o Nikto_Scan_Results -F.txt
nikto -h https://www.certifiedhacker.com -o Nikto_Scan_Results -F.txt
Nikto v2.5.0

+ Target IP: 162.241.216.11
+ Target Hostname: www.certifiedhacker.com
+ Target Port: 443

+ SSL Info: Subject: /CN=www.uyr.fvr.mybluehost.me
            Ciphers: TLS_AES_256_GCM_SHA384
            Issues: /C=US/O=Let's Encrypt/CN=R3
+ Start Time: 2024-03-12 07:20:56 (GMT-4)

+ Server: nginx/1.21.6
+ / The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/web/HTTP/Headers/X-Frame-Options
+ / Uncommon header 'host-header' found, with contents: c2hcnwKLMjsdwVub3N0aWVubQ==.
+ / Uncommon header 'x-server-cache' found, with contents: true
+ / Uncommon header 'x-proxy-cache' found, with contents: HIT.
+ / The site uses TLS and the Strict-Transport-Security HTTP header is not defined. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security
+ / The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ Server banner changed from 'nginx/1.21.6' to 'Apache'.
  
```

<https://cirt.net>

AI-Powered Vulnerability Assessment Tools

Equixly

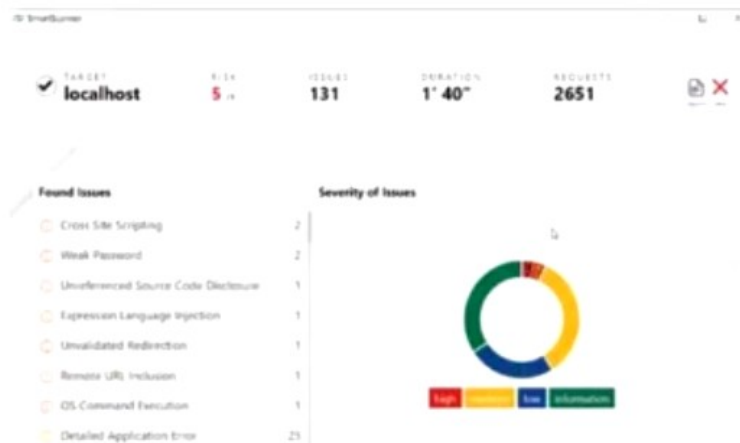
- Equixly is a SaaS platform that integrates API security testing into the development workflow
- It leverages machine learning to **automate vulnerability assessments** and provide developers with actionable remediation plans



<https://equixly.com>

SmartScanner

- SmartScanner an automated vulnerability scanner designed to **identify and mitigate potential vulnerabilities in websites** utilizes AI and machine learning (ML) to enhance its threat detection capabilities



<https://www.thesmartscanner.com>

Other Tools: CodeDefender
<https://codedefender.ro>

DryRun Security
<https://www.dryrun.security>

Hackules
<https://hackules.com>

Corgea
<https://corgea.com>

Pentest Copilot
<https://copilot.bugbase.ai>

Vulnerability Assessment using Python Script with AI

- An attacker can also leverage AI-powered ChatGPT or other generative AI technology to perform this task by using an appropriate prompt such as
sgpt --chat scancode --code "Create a python script to run a fast but comprehensive Nmap scan on the IP addresses in scan1.txt and then execute vulnerability scanning using nikto against each IP address in scan1.txt"

```

sgpt --chat scancode --code "Create a python script to run a fast but comprehensive Nmap scan on
File Edit View Search Terminal Help
root@parrot:~# sgpt --chat scancode --code "Create a python script to run a fast but comprehensive Nmap scan on the IP addresses in scan1.txt and then execute vulnerability scanning using nikto against each IP address in scan1.txt"
import subprocess

# Read the list of IP addresses from scan1.txt
with open('scan1.txt', 'r') as file:
    ip_addresses = file.readlines()

# Run Nmap scan on each IP address
for ip in ip_addresses:
    subprocess.run(['nmap', '-iL', ip])

# Run Nikto vulnerability scan on each IP address
subprocess.run(['nikto', '-h', ip])
root@parrot:~#
  
```

```

python3 vulscan.py - Parrot Terminal
File Edit View Search Terminal Help
Nikto v2.5.0
-----
+ Target IP: 10.10.1.9
+ Target Hostname: 10.10.1.9
+ Target Port: 80
+ Start Time: 2024-03-04 07:55:50 (GMT-5)
-----
+ Server: Apache/2.4.52 (Ubuntu)
+ / The anti-clickjacking X-Frame-Options header is not present
+ / The X-Content-Type-Options header is not set. This could allow a user agent to render the content of the site in a different fashion
+ type. See: https://www.netsparker.com/web-vulnerability-scanner/missing-content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible)
+ / Server may leak inodes via ETags, header found with file /, size: 61178ae134d96, mtime: gzip. See: http://cve.mitre.org/cve/2003-1418
+ Apache/2.4.52 appears to be outdated (current is at least Apache 2.2.34 is the EOL for the 2.x branch)
+ OPTIONS Allowed HTTP Methods: GET, POST, OPTIONS, HEAD
+ 8074 requests: 0 error(s) and 5 item(s) reported on remote host
+ End Time: 2024-03-04 07:56:06 (GMT-5) (16 seconds)
-----
  
```

```

python3 vulscan.py - Parrot Terminal
File Edit View Search Terminal Help
root@parrot:~# python3 vulscan.py
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-04 07:54 EST
NSE: Loaded 156 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 07:54
Completed NSE at 07:54, 0.00s elapsed
Initiating NSE at 07:54
Completed NSE at 07:54, 0.00s elapsed
Initiating NSE at 07:54
Completed NSE at 07:54, 0.00s elapsed
Initiating ARP Ping Scan at 07:54
Scanning 10.10.1.2 [1 port]
Completed ARP Ping Scan at 07:54, 0.00s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host at 07:54
Completed Parallel DNS resolution of 1 host at 07:54, 0.00s elapsed
Initiating SYN Stealth Scan at 07:54
Scanning 10.10.1.2 [1000 ports]
Discovered open port 53/tcp on 10.10.1.2
Discovered open port 80/tcp on 10.10.1.2
Completed SYN Stealth Scan at 07:54, 4.15s elapsed (1000 total ports)
Initiating Service scan at 07:54
Scanning 2 services on 10.10.1.2
  
```



Vulnerability Scan using Skipfish with AI (Cont'd)

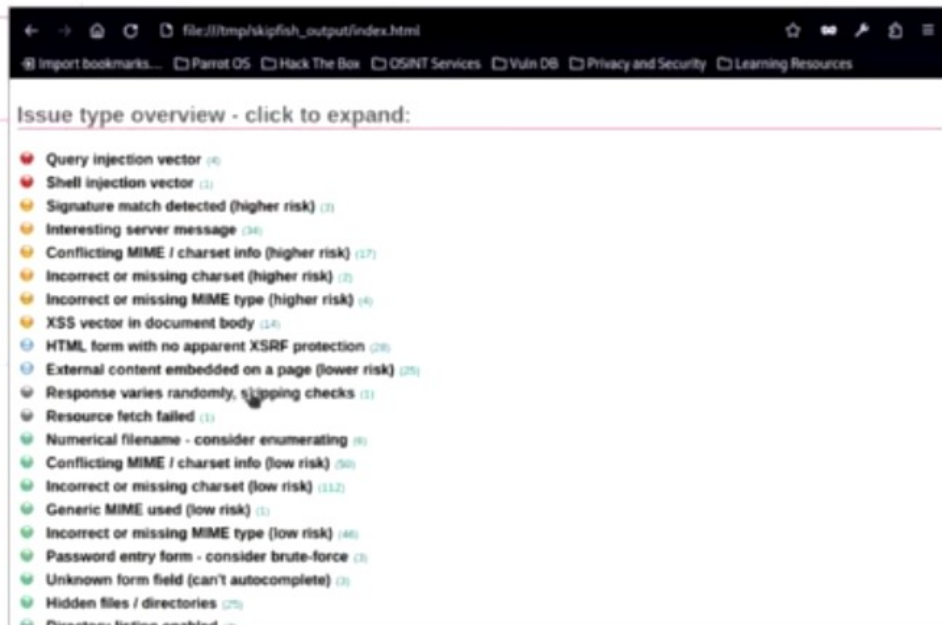


Crawl results - click to expand:

<http://testphp.vulnweb.com/> 5 74 53 2 235 132

Document type overview - click to expand:

application/binary (2)
application/xhtml+xml (14)
image/gif (2)
image/jpeg (14)
text/css (2)
text/html (10)
text/plain (6)



Objective 03

Analyze Vulnerability Assessment Reports

Vulnerability Assessment Reports

A vulnerability assessment report is a **comprehensive document** that details the findings of a vulnerability assessment

Executive Summary

- Assessment scope and objectives
- Testing narrative
- Findings summary
- Remediation summary
- Component compliance summary

Risk Assessment

- Classification of vulnerabilities based on the risk level
- Potential vulnerabilities that can compromise the system or application
- Critical hosts with severe vulnerabilities

Appendices and Supporting Information

- Additional information that supports the report's findings such as detailed logs, configuration files, or references to external resources

Assessment Overview

- Assessment methodology
- Scan information
- Target information
- Tools involved

Findings

- Scanned hosts
- Affected assets
- Types of vulnerabilities identified
- Detailed information on identified vulnerabilities
- Notes describing additional details of scan results

Recommendations

- Prioritization of remediation based on the risk ranking
- Action plan to implement the recommendations for each identified vulnerability
- Root-cause analysis
- Application of patches/fixes
- Lessons learned
- Awareness training
- Implementation of periodic vulnerability assessment
- Implementation of policies, procedures, and controls

Conclusion

- Summary of key findings and recommendations, reinforcing the importance of addressing the identified vulnerabilities

Follow-Up Actions and Timeline

- Timeline for re-assessment or follow-up actions to ensure vulnerabilities are addressed and to monitor the effectiveness of the remediation efforts

Glossary of Terms

- Definitions for technical terms used in the report

Module Summary



- In this module, we have discussed:
 - Various types of vulnerabilities, the CVSS vulnerability scoring system, and databases
 - The vulnerability-management life cycle and vulnerability research
 - Vulnerability scanning, vulnerability analysis, and various types of vulnerability scanning techniques
 - Various vulnerability assessment solutions, along with their characteristics
 - Various tools that are used to test a host or application for vulnerabilities, along with the criteria and best practices for selecting the tool
 - We concluded with a detailed discussion on how to analyze a vulnerability assessment report and how it discloses the risks detected after scanning the network
- In the next module, we will discuss the methods attackers, as well as ethical hackers and pen testers, utilize to hack a system based on the information collected about a target of evaluation; for example, footprinting, scanning, enumeration, and vulnerability analysis phases