Week 3

Day 11

Enumeration Techniques

Enumeration is a critical phase in the ethical hacking and penetration testing process that follows scanning. It involves actively extracting detailed information from a target system to identify users, system resources, network shares, services, and configurations. Unlike scanning, which only identifies open ports and services, enumeration establishes direct interaction with the target system to gather meaningful and exploitable data.

The primary objective of enumeration is to understand how a system is configured, what information is exposed, and where security weaknesses exist. This phase plays a vital role in identifying attack vectors and strengthening system defenses.

Tools Used

1. Nmap (Network Mapper)

Nmap is an advanced network exploration and security auditing tool widely used for enumeration and vulnerability analysis.

Enumeration Activities Performed Using Nmap

i. Identified live hosts within the network.
ii. Enumerated open TCP and UDP ports.
iii. Detected running services and their versions.
iv. Identified operating system details and system behavior.
v. Used Nmap Scripting Engine (NSE) scripts to perform:
    - SMB enumeration

- HTTP service enumeration
- FTP, SSH, and DNS information gathering

Through these tasks, Nmap helped in understanding how exposed services can leak system information and how attackers gather intelligence during early attack stages.

2. Enum4linux

Enum4linux is a specialized enumeration tool used to gather detailed information from Windows-based systems using SMB and NetBIOS protocols.

Enumeration Activities Performed Using Enum4linux

i. Enumerated local and domain user accounts.

ii. Retrieved information about system groups and privileges.

iii. Identified shared directories and access permissions.

iv. Extracted system policies and domain configuration details.

v. Gathered NetBIOS name tables and SMB service data.

This practical work demonstrated how improperly configured SMB services can expose sensitive organizational information to attackers.

Importance of Enumeration

i. Helps identify valid usernames and groups.

ii. Reveals shared resources and access rights.

iii. Assists in detecting misconfigured services.

iv. Provides valuable data for vulnerability exploitation.

v. Enables security professionals to strengthen access control and system hardening.

Work Description

The topic of Enumeration Techniques was studied in depth as part of the cybersecurity syllabus. All practical activities were performed on authorized systems within a controlled lab environment to ensure ethical and legal compliance. The hands-on exercises enhanced understanding of real-world enumeration methods, system information leakage risks, and preventive security measures.

Conclusion

Enumeration is a vital step in ethical hacking that bridges scanning and exploitation. By using tools like Nmap and Enum4linux, detailed insights into system configurations and exposed resources were obtained, helping to understand how attackers gather intelligence and how such threats can be effectively mitigated.

Day 12

Vulnerability Assessment

Vulnerability Assessment is the process of identifying, analyzing, and evaluating security weaknesses in a system, network, or application. The main goal is to detect potential vulnerabilities before attackers can exploit them and to improve the overall security posture of the system.

Tools Used

1. Nmap (Network Mapper)

Nmap is an open-source network scanning tool used to discover hosts, open ports, services, and potential vulnerabilities in a network.

Practical Work Performed using Nmap:

i. Scanned target systems to identify live hosts.

ii. Detected open and closed ports.

iii. Identified running services and their versions.

iv. Analyzed possible security risks based on open ports and services.

This helped in understanding how attackers gather information about network infrastructure.

2. OWASP ZAP (Zed Attack Proxy)

OWASP ZAP is a widely used web application security testing tool that helps in finding vulnerabilities in web applications.

Practical Work Performed using OWASP ZAP:

i. Configured ZAP as a proxy to intercept web traffic.

ii. Performed passive scanning to detect common security issues.

iii. Conducted active scanning on test web applications.

iv. Identified vulnerabilities such as:

- SQL Injection risks
- Cross-Site Scripting (XSS)
- Security misconfigurations
- Missing security headers

This provided hands-on experience in testing web application security.

Work Description

The topic was studied in detail and practical vulnerability assessment tasks were performed strictly according to the cybersecurity syllabus. All activities were carried out in a controlled lab environment using authorized systems only. The exercise helped in understanding real-world security testing techniques, ethical hacking practices, and the importance of vulnerability mitigation.

Day 13

System Hacking

System Hacking is the process of identifying security weaknesses in a computer system and understanding how attackers exploit these weaknesses to gain unauthorized access. This topic focuses on learning different phases of hacking, such as gaining access, maintaining access, and covering tracks, so that effective security measures can be implemented to prevent such attacks.

Tools Used

1. Metasploit Framework

The Metasploit Framework is a powerful penetration testing tool used to test system vulnerabilities in a controlled environment.

Practical Work Performed:

i. Studied different types of exploits, payloads, and auxiliary modules.

ii. Performed controlled exploitation of vulnerable systems to understand how attackers gain access.

iii. Used payloads to simulate remote access and privilege escalation.

iv. Observed post-exploitation techniques such as system information gathering and session management.

This helped in understanding how known vulnerabilities can be exploited and how systems can be secured against them.

2. Netcat

Netcat is a versatile networking tool often called the “Swiss Army knife” of networking.

Practical Work Performed:

i. Established client-server communication between systems.

ii. Simulated backdoor connections in a lab setup.

iii. Practiced file transfer and port listening techniques.

iv. Understood how attackers use Netcat for remote command execution.

This practical work improved understanding of network-based system compromise methods.

Work Description

The topic of System Hacking was studied in detail and all practical activities were conducted strictly according to the cybersecurity syllabus. Testing was performed only on authorized systems in a controlled lab environment to ensure ethical and legal compliance. This exercise provided hands-on knowledge of system attack techniques, system vulnerabilities, and effective defensive strategies to enhance system security.

Day 14

Web Application Attacks

Web Application Attacks are security threats that target vulnerabilities present in web-based applications. These attacks exploit weaknesses in application logic, input validation, authentication mechanisms, and configuration settings. Since web applications are widely used for online services, databases, and user interaction, they are common targets for attackers.

Studying web application attacks helps in understanding how attackers compromise websites and how secure coding and testing practices can prevent such attacks.

Tools Used

1. Burp Suite

Burp Suite is a popular web application security testing tool used for analyzing and exploiting web vulnerabilities.

Practical Work Performed Using Burp Suite

i. Configured Burp Suite as an intercepting proxy to capture HTTP and HTTPS requests.

ii. Intercepted and analyzed client-server communication.

iii. Modified request parameters to test input validation.

iv. Identified common web vulnerabilities such as:

- SQL Injection
- Cross-Site Scripting (XSS)
- Broken authentication

2. OWASP ZAP (Zed Attack Proxy)

OWASP ZAP is an open-source web application security testing tool widely used for automated and manual testing.

Practical Work Performed Using OWASP ZAP

- Set up ZAP as a proxy for web traffic interception.
- Performed passive scanning to identify security issues without affecting the application.
- Conducted active scanning on test web applications.
- Detected vulnerabilities such as:
    - Cross-Site Scripting (XSS)
    - SQL Injection risks
    - Missing security headers

OWASP ZAP provided hands-on experience in automated vulnerability detection and reporting.

Types of Web Application Attacks Studied

i. SQL Injection (SQLi)
ii. Cross-Site Scripting (XSS)
iii. Cross-Site Request Forgery (CSRF)
iv. Authentication and Session Attacks
v. Security Misconfiguration
vi. Input Validation Attacks

Work Description

The topic of Web Application Attacks was studied in detail as per the cybersecurity syllabus. Practical testing was performed using Burp Suite and OWASP ZAP on authorized applications in a controlled lab environment. All activities were conducted ethically and legally. The exercises improved understanding of real-world web threats and effective defensive security measures.

Conclusion

Web application attacks pose serious security risks, but by using security testing tools like Burp Suite and OWASP ZAP, vulnerabilities can be identified and mitigated at an early stage. This study provided practical knowledge of web attack techniques and secure application testing methods.

Day 15

SQL Injection

SQL Injection (SQLi) is one of the most critical web application vulnerabilities in which an attacker injects malicious SQL queries into input fields of an application. If proper input validation is not implemented, these injected queries are executed by the database, allowing attackers to access, modify, or delete sensitive data.

SQL Injection mainly occurs due to insecure coding practices and improper handling of user input in web applications.

Tools Used

1. Burp Suite

Burp Suite is a widely used web application security testing tool that helps in manual testing and analysis of SQL Injection vulnerabilities.

Practical Work Performed Using Burp Suite

- Configured Burp Suite as an intercepting proxy to capture HTTP requests.
- Analyzed input parameters in URLs, forms, and cookies.
- Modified intercepted requests to inject SQL payloads.
- Observed server responses to identify SQL error messages.
- Tested authentication and search fields for SQL Injection vulnerabilities.

This practical work helped in understanding how attackers manipulate user input to exploit databases.

2. SQLmap

SQLmap is an automated SQL Injection testing tool that detects and exploits SQL Injection vulnerabilities in web applications.

Practical Work Performed Using SQLmap

- Identified vulnerable parameters in target web applications.
- Automated detection of different SQL Injection techniques such as:
    - Error-based SQL Injection
    - Union-based SQL Injection
    - Boolean-based Blind SQL Injection
    - Time-based Blind SQL Injection
- Extracted database information like database names, tables, and columns in a controlled setup.

SQLmap demonstrated how automated tools can efficiently exploit database vulnerabilities if proper security controls are not in place.

Types of SQL Injection Studied

i. In-band SQL Injection

ii. Blind SQL Injection

iii. Error-based SQL Injection

iv. Time-based SQL Injection

Work Description

The topic of SQL Injection was studied in detail as per the cybersecurity syllabus. Practical testing was carried out using Burp Suite and SQLmap on authorized applications within a controlled lab environment. All activities were performed ethically and legally to understand vulnerabilities and preventive security measures.

Conclusion

SQL Injection remains a major threat to web application security. Through hands-on practice using Burp Suite and SQLmap, practical knowledge of identifying, exploiting, and preventing SQL Injection vulnerabilities was gained. This study emphasized the importance of secure coding, input validation, and database security.