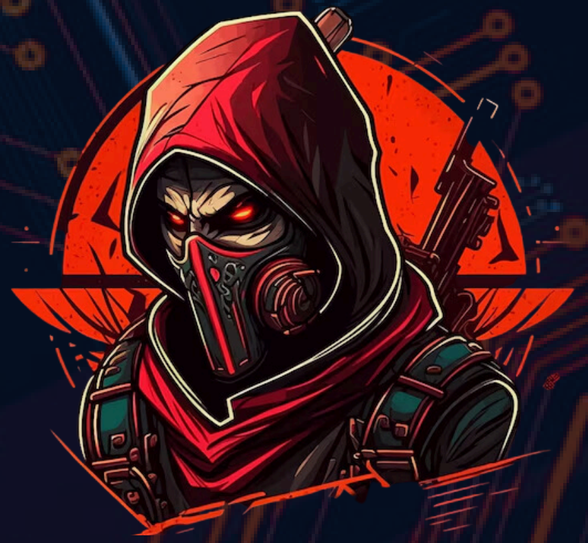
WHITESEC

CYBER SECURITY CONSULTANCY

COMPLETE INFRASTRUCTURE HACKING AND PENETRATION TESTING



trainings.whitesec.org

This training program, offered by Whitesec Cyber Security, provides hands-on experience in Vulnerability Assessment and Penetration Testing, tailored to meet industry needs. It covers a detailed, step-by-step approach to penetration testing that is widely recognized in the field. Through practical workshops and challenges, learners will develop new skills and improve their ability to apply them effectively.

The course focuses on manual penetration testing to uncover logical threats that automated tools might miss. It is designed around the most common penetration testing services, including network, application, Android, database, API, web, and cloud penetration testing, as well as Capture the Flag (CTF) challenges.

WHITESEC

CYBER SECURITY CONSULTANCY

HERE'S WHAT YOU CAN ACCOMPLISH

- Bring all candidates to the same level across different areas to ensure the curriculum is well-rounded.
- Learn about the technical and commercial aspects of securely setting up servers, network devices, and applications.
- Get practical experience with real-time penetration testing.
- Ensure the reports meet business standards and keep security strong by following industry best
- Follow OWASP and NIST guidelines for responding to attacks.

PREREQUISITES

To start the Vulnerability Assessment & Pen-Testing Training, the candidate should have a basic understanding of Ethical Hacking. This course is designed for beginners and requires knowledge of how to set up VMware and Kali Linux.

Training Duration 140 Hrs

CURRICULUM (SECTION - 1)

In the first section, you'll find 8 important modules that you need to learn before diving deeper into infrastructure penetration testing. These modules are just the starting point. After completing them, we'll explore each module in more detail.

- The modules are:
- Rules of Engagement
- Internal and External Network Penetration Testing
- Web Application Penetration Testing
- Windows Penetration Testing
- Linux Penetration Testing
- Network Device Security Audits
- Android Penetration Testing
- Automating Penetration Testing

CHAPTER - 1

Rules of Engagement

The Rules of Engagement (RoE) is a document that explains how a penetration test (a type of security test) should be done. Before starting the test, it's important to agree on certain guidelines.

Internal and External Network Penetration Testing

Your company might need both internal and external network security tests to protect your information. The main goal is to reduce or prevent any damage to your data. If a hacker has enough time, tools, and skills, they can find and exploit a weakness in your network.

A penetration test is a detailed process carried out by experts to find any weak spots in your network that a hacker might use to break in. This test not only identifies vulnerabilities but also shows the potential damage they could cause.

Key Learning Points

- Setting Goals, Objectives, and Expected Results for the Penetration Test
- Defining Technical Details and Completing Questionnaires
- Determining the Scope, Including Third Parties and Compensation Structures
- Establishing Communication and Engagement Plans
- Following Standard Operating Procedures
- Creating a Testing Checklist
- Preparing a Standard Business Report (Both Executive Summary and Technical Details)

CHAPTER - 2

Internal and External Network Penetration Testing

There are several reasons why your company might need an internal and external network penetration test. The main purpose of this test is to protect your valuable information by identifying and minimizing potential risks. If a hacker has enough time, tools, and skills, they can find and exploit weaknesses in your network.

A penetration test is a thorough process led by experts to identify all the ways an attacker could potentially break into your network. It not only finds these weaknesses but also shows how much damage they could cause.

Key Learning Points

- Strategic Approach to Network Vulnerability Assessment and Penetration Testing (VAPT)
- Collecting Information
- Mapping the Internal Network
- Using Nmap in Advanced Mode
- Testing Well-Known Ports for Vulnerabilities
- Identifying Operating Systems & Services
- Understanding Man-in-the-Middle Attacks
- Hands-On Practice with Top Vulnerability Assessment Tools
- Writing Network Reports that Clearly Define Risks Using CVSS and CWE

CHAPTER - 3

Web Application Penetration Testing

Web application security testing involves a step-by-step process to understand a web system, find its weaknesses or problems, and explore ways these issues could be exploited to compromise the application.

Key Learning Points

- Understand key security guidelines like OWASP Top 10 and WSTG Testing Guide, and learn testing methods and strategies.
- Perform manual security checks using tools like Burp Suite and OWASP ZAP.
- Collect information and identify the web application's characteristics.
- Test how well the application is configured and managed.
- Check for common code vulnerabilities such as OS, SQL Injection, XSS, LFI, etc.
- Use automated tools for vulnerability scanning and testing.
- Assess the security of SSL/TLS connections.
- Write detailed security reports, including risk analysis using CVSS and CWE.
- Test the security of Content Management Systems (CMS).
- Hack into a web server to test its security.

CHAPTER - 4

Windows Penetration Testing

In the threat modeling phase, a company might hire a pentester to perform authorized tests. This helps verify if accounts have the right permissions and spot internal risks from unauthorized access. In this training, the focus will be on identifying internal threats and using Capture The Flag (CTF) challenges to build skills.

Key Learning Points

- Connect remotely using different SMB methods.
- Use PowerShell Empire for pentesting.
- Find ways to bypass whitelisting programs.
- Report dangerous security misconfigurations.
- Automate privilege escalation with scripts.
- Perform manual privilege escalation.
- Move laterally within the network.

CHAPTER - 5

Linux Penetration Testing

In the threat modeling phase, the organization might hire a security tester to perform authenticated testing. This helps verify if user accounts are properly authorized on a Linux system and to spot internal risks from unauthorized access. The focus of the lesson will be on internal threats, using Linux-based Capture The Flag (CTF) challenges to improve skills.

Key Learning Points

- Basics of Linux permissions and user rights
- Creating and using reverse shell commands
- Methods for transferring files
- Exploiting network shares
- Overcoming restricted shell environments
- Gaining higher-level access (Privilege Escalation)
- Misusing sudo commands
- Identifying and exploiting misconfigured Suid permissions
- Exploiting misconfigured network file shares
- Techniques for pivoting and tunneling within networks

CHAPTER - 6

Network Device Security Audits

Organizations hire a security expert to check the safety of their network devices, such as computers, routers, switches, and printers. This expert might use automated tools to scan the network and review the device settings to ensure they are secure.

If the security expert isn't very experienced, they might find it challenging to meet the organization's needs. This guide will walk you through the best practices for conducting a Network Security Audits.

Key Learning Points

- Assess and review the security of routers
- Assess and review the security of switches
- Assess and review the security of firewalls
- Assess and review the security of printers
- Check for secure device settings
- Understand how to prevent data leaks

CHAPTER - 7

Android Penetration Testing

Just like with other platforms, many organizations need Android penetration testing because the privacy and security of Android users can be at risk from unsafe apps. These risky apps might even cause financial harm. This issue mostly comes from the open nature of the Android system. Mobile apps are now more likely to be targeted by cyberattacks than ever before. Android penetration testing is one of the best ways to boost the security of an Android app.

Key Learning Points

- Basics of the Android System
- Setting Up a Lab with a Simulator
- Using Genymotion
- Testing the Top 10 OWASP Risks
- Analyzing Secure Code

CHAPTER - 8

Automate Penetration Testing

In this chapter, you'll learn how to use top industry tools essential for automated penetration testing. Our instructor-led training will guide you in mastering these tools, helping you streamline manual pen-testing tasks through automation.

Key Learning Points

- Nmap NSE Scripts: Techniques for network scanning and vulnerability detection.
- Metasploit Framework and Workspace: How to use this powerful tool for exploiting security flaws.
- Powershell Empire: Tools and methods for red team operations.
- Responder: How to use this tool for network attacks and defenses.
- Impacket Python Libraries: Essential libraries for network communication and security.
- BurpSuite: Techniques for web application security testing.
- OWASP ZAP: Tools for identifying vulnerabilities in web applications.

WINDOWS PENETRATION TESTING – SECTION 2

SECTION 2

WINDOWS PENETRATION TESTING

Penetration testers need to learn Windows penetration testing because Windows is the most widely used operating system in corporate environments, making it a prime target for cyberattacks. Understanding how to identify and exploit vulnerabilities in Windows systems helps testers assess the security of critical infrastructure, from desktop machines to Windows servers. Companies rely on these systems for daily operations, and a breach can lead to data loss, financial damage, and reputational harm. By mastering Windows penetration testing, professionals can proactively uncover weaknesses, ensure compliance with security standards, and help organizations fortify their defenses

CHAPTER - 1

Introduction to Privilege Escalation

- Overview of Privilege Escalation Techniques
- Key Concepts and Architecture of Privilege Management

Understanding Windows Privilege Architecture

- User Permissions and Roles
- Windows Privilege Levels Explained

CHAPTER - 2

User Access Control (UAC)

- Fundamentals of UAC in Windows
- Bypassing UAC for Privilege Escalation

Enumeration Techniques

- Manual Enumeration Methods
- Tools for Automatic Enumeration
  - Using Windows PrivEsc Check
  - WinPEAS for Automated Enumeration

CHAPTER - 3

Windows Services Exploitation

- Introduction to Windows Services and Vulnerabilities
- Exploiting Weak Service Executables and Permissions
- Unquoted Service Paths and DLL Hijacking
- Practical Labs on Service Abuse and Exploitation

Sensitive Credentials Exploitation

- Identifying Sensitive Credentials in Windows
- Exploiting Weak Credentials via Registry Attacks

CHAPTER - 4

Windows Registry Attacks

- Introduction to Windows Registry and Its Role in Security
- Exploiting Autorun and Weak Registry Permissions
- AlwaysInstallElevated Vulnerability

Token Impersonation Exploits

- Understanding SeImpersonate Privilege
- Exploiting with JuicyPotato, RoguePotato, and PrintSpoofer

CHAPTER - 5

Other Privilege Exploits

- SeBackup and SeTakeOwnership Exploits
- Windows Kernel Exploitation Techniques (Windows 10)

Exploiting Scheduled Tasks

- Using Scheduled Tasks for Privilege Escalation
- Practical Exploitation Scenarios

CHAPTER - 6

Startup Apps Exploitation

- Bypassing Security via Startup Applications

Insecure GUI Apps

- Identifying and Exploiting Insecure Graphical User Interface Applications

Windows UAC Bypass Techniques

- Various Methods to Bypass Windows UAC

Vulnerable Software Exploitation

- Identifying and Exploiting Vulnerabilities in Installed Software

LINUX PENETRATION TESTING – SECTION 3

SECTION 3

LINUX PENETRATION TESTING

Penetration testers need to learn Linux because it is one of the most widely used operating systems in enterprise environments, especially for servers and critical infrastructure. Understanding Linux penetration testing allows security professionals to identify vulnerabilities in Linux-based systems, from misconfigurations to privilege escalation exploits. Many modern technologies, such as cloud platforms and DevOps environments, rely heavily on Linux, making it crucial to know how to secure these systems. For companies, having penetration testers skilled in Linux ensures they can detect and mitigate potential threats before attackers exploit them, safeguarding sensitive data and maintaining system integrity.

CHAPTER - 1

Getting Started with Linux Privilege Escalation

- Introduction to Linux Privilege Escalation Methods
- Understanding User and Root Permissions

Exploring Linux System Information

- Manual Techniques for System Enumeration
- Hands-On Lab: Exploring System Info (Part 1 & Part 2)
- Automated Tools for Gathering System Data

CHAPTER - 2

Finding and Using Sensitive Credentials

- Identifying Sensitive Information in Linux Systems
- Practical Lab: Exploiting Credentials for Access

Exploiting Weak File Permissions

- How to Identify and Exploit Files with Weak Permissions
- Practical Lab: Real-World File Exploitation

CHAPTER - 3

Hacking Cron Jobs

- Introduction to Cron Job Scheduling and Security Issues
- Exploiting Weak Permissions in Cron Jobs
- Manipulating PATH Variables in Cron Jobs
- Using Wildcard Characters to Escalate Privileges

Breaking SUID/SGID Permissions

- Understanding SUID and SGID Files in Linux
- Hands-On Labs: Common Exploits for SUID Files
- Using Shared Objects and Environment Variables for Exploitation

CHAPTER - 4

Hacking SUDO Commands

- Introduction to SUDO Vulnerabilities
- Bypassing SUDO Restrictions with Escape Sequences
- Using ld_preload and ld_library_path for Privilege Escalation
- Practical Labs: Hacking SUDO for Privilege Gain

Exploiting the Linux Kernel

- Basics of Kernel Exploits in Linux
- Hands-On Lab: Exploiting the Kernel for Root Access

CHAPTER - 5

Breaking into Network File System

- Understanding NFS and Its Security Flaws
- Exploiting NFS Misconfigurations for Elevated Access

Exploiting Linux Services

- Targeting Common Linux Services for Privilege Escalation
- Exploiting MySQL User Defined Functions (UDF)
- Hands-On Lab: Attacking Linux Services for Escalation

CHAPTER - 6

Linux Capabilities Exploitation

- Understanding Linux Capabilities and Security Risks
- Using getcap to Find and Exploit Capabilities
- Practical Lab: Privilege Escalation via Capabilities

NETWORK PENETRATION TESTING – SECTION 4

SECTION 4

NETWORK PENETRATION TESTING

Network penetration testing is a smart way to find and fix weaknesses in a network before anyone can take advantage of them. By simulating attacks on an organization's network, businesses can spot vulnerabilities and create strategies to prevent and recover from security issues.

This course is designed to help IT professionals improve their security skills, covering both basic and advanced topics in network security and infrastructure. One of the great things about this course is its flexible structure, making it accessible even for those with little or no technical background.

CHAPTER - 1

Network Basics

- Understanding TCP/IP and how data travels through packets
- Introduction to network security concepts
- Learning about ports and protocols
- Setting up a Windows lab environment
- Setting up a Linux lab environment
- Key services and commands for Linux
- Key services and commands for Windows

CHAPTER - 2

Testing for Security Weaknesses

- Using Kali Linux Toolkit

CHAPTER - 3

Understanding Network Traffic

- Why Packet Analysis Matters
- How to Capture Network Data
- Using Promiscuous Mode
- Getting Started with Wireshark
- Filtering and Decoding Data
- Physical Layer Basics
- Internet Layer Basics
- Transport Layer Basics
- Application Layer Basics

CHAPTER - 4

Traffic Analysis with Tshark

- Getting to Know Tshark
- Capturing Network Data
- Using Promiscuous Mode
- Counting Packets
- Saving Data to Files
- Different Output Formats
- Using Display Filters
- Analyzing Endpoints

CHAPTER - 5

Finding Active Systems and Analyzing Data

- Identifying Live Systems Using ICMP
- Identifying Live Systems Using TCP
- Analyzing ICMP Packets
- Using Traceroute for Network Paths

CHAPTER - 6

Advanced Port Scanning with Nmap

- Fragmented Scanning: Breaking packets into smaller pieces for stealthy scanning.
- Data Length Scanning: Checking packet sizes to gather information.
- TTL Scanning: Using the Time to Live value to discover hosts.
- Source Port Scanning: Scanning from different source ports for better results.
- Decoy Scanning: Masking your scan by using fake IP addresses.
- TCP and UDP Scanning: Checking both types of ports to find open ones.
- Combining Nmap with Wireshark: Analyzing Nmap results in real-time.
- Reading Nmap Output: Understanding the scan results and what they mean.
- Operating System Detection: Identifying the OS of the target machine.
- Spoofing IP Addresses: Faking your IP for anonymity in scans.
- Spoofing MAC Addresses: Changing your device's MAC address for privacy.
- Data String Scanning: Searching for specific text in responses.
- Hexadecimal String Scanning: Looking for patterns in hex format.
- IP Options Scanning: Checking for extra options in IP packets.

CHAPTER - 7

Hands-on with Metasploit

- Metasploit Basics: Learn the core functions of Metasploit for testing security vulnerabilities.
- Creating Payloads with Msfvenom: Create malicious payloads for testing purposes.
- Network Scanners: Use Metasploit to scan for vulnerabilities.
- Windows TCP Attack: Execute a reverse shell attack on Windows.
- Windows HTTPS Tunnel: Set up encrypted connections for attacks.
- Hidden TCP Shell: Create a hidden communication channel on TCP.
- Macro-based Payloads: Deliver payloads through Microsoft Office macros.
- Dynamic Shell Access: Gain real-time shell access on a target machine.
- Bypassing User Account Control: Disable security prompts to gain full access.
- Pass-the-Hash Attack: Use password hashes to gain unauthorized access.
- Post-Exploitation Actions: Explore advanced actions after successful entry.

CHAPTER - 8

Cracking Passwords & Attacks

- Hydra Tool: Automate brute force attacks on various services.
- Medusa Tool: Use Medusa for fast password cracking.
- Generate Wordlists with Crunch: Create custom wordlists for password attacks.
- CeWL Tool: Build targeted wordlists from websites.
- cUPP Tool: Generate personalized password lists.
- Online Brute Force: Test passwords remotely via brute force attacks.

CHAPTER - 9

FTP Testing

- FTP Setup: Set up an environment to test FTP vulnerabilities.
- Banner Grabbing: Collect information about an FTP server.
- Hiding Server Information: Conceal FTP details to prevent attacks.
- FTP Exploitation: Find and exploit FTP weaknesses.
- Brute Force FTP Logins: Crack FTP passwords using brute force.
- Defend FTP from Brute Force: Learn how to protect FTP servers.
- Remote Port Forwarding: Access internal networks through FTP.
- Pivoting through FTP: Use FTP access to attack other systems.

CHAPTER - 10

SSH Security Testing

- SSH Setup: Prepare an SSH testing environment.
- SSH Info Gathering: Collect SSH server details.
- Hiding SSH Info: Conceal SSH information to protect servers.
- SSH Port Redirection: Redirect ports for access control.
- Crack SSH Passwords: Use brute force to break into SSH accounts.
- Secure SSH from Brute Force: Learn how to prevent brute force attacks.
- SSH User Key Enumeration: Discover user keys in SSH.
- Stealing SSH Keys: Capture SSH keys for unauthorized access.
- SSH Persistence: Maintain ongoing access through SSH.
- Port Forwarding with SSH: Securely forward network traffic.
- SSH Tunneling: Encrypt and route traffic through SSH tunnels.

CHAPTER - 11

Telnet Testing

- Telnet Setup: Create a testing lab for Telnet vulnerabilities.
- Collect & Hide Telnet Info: Gather and conceal Telnet server details.
- Telnet Port Redirection: Redirect Telnet traffic to control access.
- Brute Force Telnet Logins: Crack Telnet credentials.
- Telnet Port Forwarding: Securely forward Telnet ports.
- Pivoting via Telnet: Use Telnet to access other systems.

CHAPTER - 12

SMTP Security Testing

- SMTP Setup: Create a lab to test email (SMTP) vulnerabilities.
- SMTP Info Gathering: Collect information from SMTP servers.
- Concealing SMTP Banners: Hide server information to reduce attack risks.
- SMTP User Enumeration: Identify valid SMTP users.

CHAPTER - 13

DNS & DHCP Testing

- DNS & DHCP Lab: Set up a lab for testing DNS and DHCP vulnerabilities.
- DNS Information Gathering: Collect details from DNS servers.
- Analyzing DHCP Traffic: Monitor DHCP activity with Wireshark.
- DHCP Starvation Attack: Launch an attack to exhaust DHCP resources.
- Fake DHCP Server Attack: Set up a rogue DHCP server to manipulate traffic.

CHAPTER - 14

NetBIOS & SMB Testing

- NetBIOS/SMB Lab: Test vulnerabilities in NetBIOS and SMB protocols.
- SMB Information Gathering: Collect details from SMB servers.
- SMB Null Sessions: Exploit anonymous access in SMB.
- Enum4Linux Tool: Gather SMB data using Enum4Linux.
- SMB Password Cracking: Brute force SMB passwords.
- SMB Denial of Service: Launch DoS attacks on SMB services.
- Exploiting EternalBlue: Use the EternalBlue vulnerability to attack SMB.
- Remote SMB Logins: Access SMB shares remotely.

CHAPTER - 15

MySQL Testing

- MySQL Lab: Set up a MySQL environment to test vulnerabilities.
- MySQL Password Cracking: Brute force MySQL credentials.
- MySQL Data Gathering: Collect database information.
- Remote MySQL Commands: Execute queries on a MySQL server.
- MySQL Password Hashes: Dump and analyze password hashes.
- File Enumeration: Discover writable directories and system files.

CHAPTER - 16

Remote Desktop Testing

- RDP Lab: Set up a lab for Remote Desktop testing.
- RDP Information Gathering: Collect details about RDP sessions.
- RDP Man-in-the-Middle Attack: Intercept RDP traffic.
- Brute Force RDP: Crack RDP passwords.
- Hijack RDP Sessions: Take over an active RDP session.
- Remote RDP Port Forwarding: Forward RDP ports securely.
- RDP Denial of Service: Perform a DoS attack on RDP services.

CHAPTER - 17

VNC Testing

- VNC Lab: Create a lab to test VNC vulnerabilities.
- VNC Info Gathering: Collect and hide VNC details.
- Redirect VNC Ports: Control VNC traffic via port redirection.
- Brute Force VNC Logins: Crack VNC passwords.
- VNC Port Forwarding: Securely forward VNC ports.
- SSH Tunneling with VNC: Route VNC traffic through SSH.

CHAPTER - 18

Credential Dumping

- Dump Auto-login Passwords: Extract passwords from auto-login setups.
- Dump Application Credentials: Steal passwords stored in applications.
- Use Fake Services: Set up fake services to capture passwords.

CHAPTER - 19

Testing Socks Proxy

- Socks Proxy Lab: Create an environment to test Socks proxies.
- Socks Proxy with SSH: Use SSH over a Socks proxy.
- Socks Proxy with FTP: Test FTP over a Socks proxy.
- Socks Proxy with HTTP: Test HTTP over a Socks proxy.

CHAPTER - 20

Sniffing & Spoofing

- Intro to Sniffing & Spoofing: Learn how to capture and fake network traffic.
- ARP Poisoning: Intercept network traffic using ARP spoofing.
- MAC Snooping: Collect MAC addresses from network devices.
- DNS Spoofing: Redirect DNS traffic to malicious sites.
- ICMP Redirect Attacks: Manipulate routing with ICMP redirects.
- Capture NTLM Hashes: Steal NTLM password hashes.

CHAPTER - 21

Denial of Service (DoS) Attacks

- DoS Attack Basics: Learn how to overwhelm and shut down systems.
- Using Botnets for DoS: Conduct large-scale attacks using botnets.
- DDoS Attacks: Launch distributed denial-of-service attacks.
- SYN Flood Attack: Overwhelm servers with SYN requests.
- UDP Flood Attack: Crash servers with UDP packets.
- Smurf Attack: Use spoofed ICMP packets to flood networks.
- Craft Malicious Packets: Create customized packets for DoS attacks.
- Other DoS Tools: Explore various tools for DoS.

CHAPTER - 22

Hiding Tracks & Maintaining Access

- Service-based Persistence: Stay hidden by installing services.
- Executable-based Persistence: Maintain access using executables.
- Registry-based Persistence: Hide within the Windows registry.
- Persistence via Netcat: Use Netcat to maintain backdoor access.
- Clear Logs: Delete event logs to erase traces of activity.

CHAPTER - 23

Honeypots

- Honeypot Basics: Learn what honeypots are and how they work.
- Setting Up Honeypots: Install and configure honeypots to catch attackers.

CHAPTER - 24

Firewalls

- Firewall Overview: Learn what firewalls do and their types.
- Windows Firewall: Configure firewall settings on Windows.
- Linux Firewall: Set up firewall rules on Linux systems.
- Untangle Firewall: Use Untangle to manage network security.

CHAPTER - 25

Honeypots

- IDS Overview: Understand what IDS is and how it detects attacks.
- Setting Up Snort IDS: Install and configure Snort for attack detection.
- Detect ICMP Alerts: Capture alerts from ICMP traffic.
- Detect TCP Alerts: Monitor TCP traffic for suspicious activity.
- Detect Malicious Attacks: Identify and capture harmful attacks.

CHAPTER - 26

Network Vulnerability Tools

- Nessus Tool: Scan networks for vulnerabilities using Nessus.
- Nmap Scanning: Use Nmap to find network weaknesses.

WEB PENETRATION TESTING (BASIC) – SECTION 5

SECTION - 5

WEB Penetration testing basic

The Web Penetration Testing Program, also known as the Bug Bounty Program, is a collaborative effort where organizations invite security experts and ethical hackers (also called white hat hackers) from around the world to find and report vulnerabilities in their systems. This helps companies identify and fix security issues before malicious attackers can exploit them.

Participants in the program look for security weaknesses in a company's infrastructure. Based on how serious the vulnerability is and the quality of the report, hackers can receive rewards such as money or public recognition. The rewards vary depending on the severity of the issue found and the quality of the provided Proof of Concept (POC).

Most bug bounty programs are open to the public, allowing anyone to participate. However, some are private and by invitation only, often with strict rules that participants must follow closely.

During this course, you will learn the basics of application security, how to identify vulnerabilities, and the techniques used in penetration testing to find these flaws in web applications.

This version simplifies the language to make it more accessible for trainees while maintaining the essential details of the bug bounty program and the course focus.

CHAPTER - 1

Introduction to Web Application Security Testing

1. What is Penetration Testing?

Testing a system's security by simulating real-world cyberattacks to find weaknesses.

2. What is a Web Application?

A program that runs on a web server and can be accessed through a browser.

3. How Do Web Applications Work?

Web applications operate by processing user requests through the internet, interacting with databases and servers.

4. Web Application Architecture

The structure of a web application, including its frontend, backend, and database components.

5. What are Web Services?

Services that allow applications to communicate and share data over the web.

6. What is Web Application Penetration Testing?

Simulating attacks on a web application to uncover and fix security vulnerabilities.

7. Tools for Web Application Security Testing

A list of tools used to find security flaws in web applications, like BurpSuite and OWASP ZAP.

CHAPTER - 2

Steps in Web Application Security Testing

8. Web Application Testing Process
A step-by-step method for identifying and fixing security issues in web applications.

9. Footprinting Web Infrastructure
Gathering information about the target website's network setup.

10. Discovering the Server
Finding and identifying the server that hosts the web application.

11. WHOIS Lookup
A tool to collect ownership and registration information of a domain.

12. DNS Lookup
Tools that reveal a website’s DNS records and domain details.

13. Port Scanning
Scanning for open ports on a web server to find entry points.

14. Service Discovery
Identifying services running on open ports for further analysis.

15. Banner Grabbing
Getting details about a server and its services through server banners.

16. Detecting Web Firewalls and Proxies
Identifying any firewalls or proxies that might block attacks on the target site.

17. Discovering Hidden Content
Finding hidden files and directories on the web server that might contain sensitive information

18. Detecting Load Balancers
Finding out if a web application uses load balancers to distribute traffic.

CHAPTER - 3

Analyzing Web Application Security

19. Analyzing Web Applications

Understanding how the web application is built to find weak points.

20. Identifying Server Technologies

Finding out what programming languages and technologies the web server uses.

21. Identifying Files and Directories

Mapping files and directories to find potential vulnerabilities.

22. Tools for Finding Vulnerabilities

Using specialized tools to automatically discover weak spots in the web app.

23. Mapping the Attack Surface

Creating a complete overview of potential vulnerabilities for further testing.

CHAPTER - 4

Understanding OWASP Security Risks

24. What is OWASP?
A non-profit organization that sets web security standards.

25. OWASP Top 10 Security Risks
A list of the 10 most critical security risks for web applications.

26. Injection Flaws
Attacks that insert malicious code (like SQL or commands) into a web application.

27. SQL Injection
Injecting SQL code to manipulate the database.

28. Command Injection
Executing system-level commands through a web application.

29. Shell Injection
Gaining shell access to a server by injecting commands.

30. HTML Injection
Inserting HTML code to change or disrupt web pages.

31. File Injection
Uploading or injecting malicious files into the server.

CHAPTER - 5

Other Injection Attacks

32. LDAP Injection
Injecting commands into LDAP queries to access sensitive information.

33. Server-Side JavaScript Injection
Running malicious JavaScript code on the server.

34. Server-Side Template Injection
Injecting commands into server-side templates to manipulate content.

35. Log Injection
Inserting code into server logs to compromise the system.

36. CRLF Injection
Manipulating HTTP headers with malicious code.

CHAPTER - 6

Authentication and Session Security

37. Broken Authentication
Exploiting flaws in the authentication system to bypass login security.

38. Session ID in URL
Storing session details in URLs, making it easier to steal user sessions.

39. Password Exploitation
Cracking weak passwords or using password vulnerabilities.

40. Timeout Exploitation
Exploiting poor session timeout settings to maintain unauthorized access.

41. Attack Authentication Flaws
Finding weaknesses in the login system to bypass authentication.

CHAPTER - 7

Sensitive Data Exposure

42. Sensitive Data Exposure
Finding unsecured sensitive data such as personal information and passwords.

43. XML External Entities (XXE)
Exploiting XML parsers to read sensitive server files.

44. Broken Access Control
Taking advantage of flaws that allow unauthorized users to access restricted data.

CHAPTER - 8

Security Misconfigurations

45. Security Misconfiguration
Weaknesses caused by improper settings in the application's configuration.

46. Unvalidated Inputs
Failing to check user input, which can lead to attacks.

47. Form and Parameter Tampering
Manipulating forms and parameters to bypass security checks.

48. Improper Error Handling
Exposing system information through error messages.

CHAPTER - 9

Cross-Site Scripting (XSS)

49. Cross-Site Scripting (XSS)
Injecting malicious scripts into a web page viewed by other users.

50. Insecure Deserialization
Exploiting weaknesses in how the server processes serialized data.

51. Using Vulnerable Components
Using software components with known security flaws.

CHAPTER - 10

Other Web Application Threats

52. Insufficient Logging and Monitoring
Failing to track user activities, which makes it hard to detect attacks.

53. Directory Traversal
Accessing restricted directories on the server to view sensitive files.

54. Unvalidated Redirects
Redirecting users to untrusted sites without proper checks.

55. Cross-Site Request Forgery (CSRF)
Tricking users into making unwanted actions on a web application.

56. Cookie/Session Poisoning
Manipulating cookies or session data to hijack user accounts.

CHAPTER - 11

Practical Tools and Attacks

57. Introduction to BurpSuite
Learn how to use BurpSuite for web application testing.

58. Setting Up DVWA
Install Damn Vulnerable Web App (DVWA) on Kali Linux for practice.

59. Perform Brute Force Attacks
Simulate brute force attacks on web applications with different security levels.

60. Command Injection Attacks
Practice injecting system commands into a vulnerable web application.

61. CSRF Attacks
Simulate cross-site request forgery attacks on DVWA.

62. File Inclusion Attacks
Perform file inclusion attacks to access server files.

63. File Upload Attacks
Upload malicious files to gain access to the web server.

64. XSS (Cross-Site Scripting) Attacks
Test and practice DOM-based, reflected, and stored XSS attacks.

WEB PENETRATION TESTING (INTERMEDIATE) - SECTION 6

COMPLETE SECTION 6

- Bug Hunting Basics: Learn the fundamentals of finding security flaws in websites.
- Understanding Websites: Overview of how websites are built and function.
- Finding Database Logins: Techniques for uncovering database login credentials.
- Discovering Hidden Data & Endpoints: How to find hidden web pages and sensitive information.
- Understanding HTTP Response Codes: Learn what different website response codes mean.
- Thinking Like a Hacker to Find Admin Logins: Use a hacker's approach to discover admin login pages.
- Controlling Website Behavior with GET Requests: Change how websites behave using GET requests.
- Controlling Website Behavior with POST Requests: Modify website functionality using POST requests.
- Intercepting Web Traffic with Burp Proxy: Capture and analyze web traffic to find vulnerabilities.
- Accessing Restricted User Information: Methods for getting access to private user data.
- Identifying IDOR (Insecure Direct Object Reference) Issues: Find vulnerabilities that expose unauthorized data.
- Escalating Privileges with Burp Repeater: Gain higher access levels using Burp Suite.
- Using HTTP TRACE to Debug and Access Admin Features: Use TRACE requests to understand data flow and gain admin access.
- Introduction to File Path Vulnerabilities: Learn how attackers access restricted files on a server.

COMPLETE SECTION

- Bypassing File Path Security: Techniques to bypass file path restrictions and access sensitive files.
- Avoiding File Type Restrictions: Methods to bypass file type limitations for uploading or accessing files.
- Getting Around Advanced Security Filters: Bypass strong security filters meant to block attacks.
- Exploiting CSRF (Cross-Site Request Forgery) Flaws: Learn how attackers trick users into performing unintended actions.
- Hacking OAuth 2.0 for User Control: Exploit OAuth flows to take over user sessions.
- Finding Command Injection Weaknesses: Discover how attackers can run unauthorized commands on servers.
- Exploiting Hidden Command Injection: Attack servers when the result of a command isn't visible to the attacker.
- Understanding XSS (Cross-Site Scripting): Learn how attackers inject harmful scripts into websites.
- Finding HTML Injection Issues: Identify ways attackers can inject dangerous HTML into web applications.
- Exploring XSS Flaws (Reflected & Stored): Discover how attackers inject code that harms users.
- Discovering DOM-based XSS in JavaScript: Find XSS vulnerabilities within JavaScript on a website.
- Bypassing XSS Filters: Techniques to bypass XSS protections on a website.

COMPLETE SECTION

- Introduction to SQL Injection (SQLi): Learn how attackers manipulate database queries to exploit websites.
- Bypassing Logins with SQL Injection: Use SQL injection to bypass website logins.
- Extracting Data with SQL Injection: Steal sensitive data from a vulnerable database.
- Finding Blind SQL Injection Flaws: Identify SQL injection issues without direct feedback from the website.
- Using Delays to Extract Data with Blind SQL Injection: Extract data using time-based SQL injection techniques.
- SSRF (Server-Side Request Forgery) Attacks: Learn how attackers can make a server perform unintended actions.
- Getting Around SSRF Protections: Techniques to bypass SSRF protections like blacklists and whitelists.
- Turning SSRF Into Full Control (RCE): Escalate SSRF vulnerabilities into full server control.
- XML Basics: Understand the role of XML in web applications.
- Exploiting XXE (XML External Entity) Vulnerabilities: Learn how attackers exploit weaknesses in XML parsing.
- Discovering SSRF Through XXE Vulnerabilities: Use XXE vulnerabilities to exploit SSRF flaws.

API PENETRATION TESTING – SECTION 7

SECTION 7

API PENETRATION TESTING

Large companies are expanding their reach by offering online services and platforms that automate tasks, often using APIs (Application Programming Interfaces). However, APIs are rarely tested thoroughly for security flaws, and even when they are, the testing is often not in-depth enough.

API penetration testing involves examining the functions of an API to find weaknesses and figure out how hackers could exploit them. This type of testing requires a highly skilled security professional with a strong understanding of how APIs work.

In this course, you will learn about different API functions and how to perform security testing on them, starting with the basics and gradually moving to advanced techniques.

CHAPTER - 1

1. Getting Started with Postman
Overview of Postman and its basic features.

2. Installing Postman
A step-by-step guide to installing Postman.

3. Exploring Postman Features
Learn to navigate Postman's user interface.

4. Making Simple API Requests
How to create and send basic API requests using Postman.

5. Handling API Authentication
Overview of how to handle authentication in API testing.

6. Using OAuth 2.0 for Secure API Testing
Learn how to implement OAuth 2.0 for secure API authentication.

7. Working with JWT Tokens
Understand how JWT tokens are used in API requests.

8. Simplifying API Testing with Swagger
How to use Swagger files for easier API testing.

9. Transforming API Data in Postman
Learn how to import and transform API data using JSON.

10. Using OpenAPI Parser in Burpsuite
Integrate OpenAPI with Burpsuite for better API analysis.

CHAPTER - 2

11. Intercepting API Requests with Burpsuite
How to use Burpsuite to capture and modify API requests.

12. Transforming WADL XML Files in Burpsuite
Learn to capture and transform XML files for API testing.

13. Finding Hidden API Endpoints for Bug Bounties
Discovering API endpoints from a bug bounty hunter's perspective.

14. Setting Up the Lab and Sharing Postman Documents
How to set up your testing environment and share Postman files.

15. Understanding OWASP API Top 10 Security Risks
Overview of the most common API security vulnerabilities.

16. Broken Object Level Authorization (BOLA)
Learn how broken object-level authorization can expose data.

17. API Authentication Flaws
Discover common issues with API authentication.

18. Data Leaks in APIs
Understand how sensitive data can be unintentionally exposed.

19. Rate Limiting and Resource Management
Learn why rate limiting is important for API security.

20. Function-Level Authorization Issues
How broken function-level authorization can be exploited.

CHAPTER - 3

21. Mass Assignment Vulnerabilities
Learn about the dangers of mass assignment in APIs.

22. Security Misconfiguration in APIs
Common mistakes in API security configuration.

23. API Injection Attacks
How hackers use SQL and command injection vulnerabilities to exploit APIs.

24. Poor Asset Management in APIs
The risk of not managing API assets properly.

25. Logging and Monitoring Failures in APIs
Why logging and monitoring are crucial for API security.

26. SQL Injection in APIs
A deep dive into different types of SQL injection, including blind SQL injection and admin bypass.

27. Command Injection in APIs
Learn how command injection can be used to exploit APIs.

28. Exploiting XXE Vulnerabilities in APIs
A step-by-step look at XML External Entity (XXE) attacks and how they are used to exploit APIs.

29. Server-Side Request Forgery (SSRF) in APIs
Explore basic and advanced SSRF techniques in API testing.

30. Cross-Site Scripting (XSS) in APIs
Discover how XSS vulnerabilities can be exploited in APIs.

CHAPTER - 4

31. Transport Layer Security (TLS) Issues in APIs
Common SSL/TLS issues, clear-text password submission, and missing security headers in APIs.

32. Mass Assignment Attacks in APIs
A practical guide on how mass assignment attacks can occur in APIs.

33. User Enumeration in APIs
How attackers can enumerate users through API vulnerabilities.

34. Information Disclosure through APIs
The risk of sensitive information being exposed via APIs.

35. JSON Web Token (JWT) Security in APIs
How JWT can be exploited in API security.

36. Unauthorized Password Changes via API Calls
How attackers can change passwords through insecure API calls.

37. Excessive Data Exposure in Debug Endpoints
How APIs can accidentally expose too much data through debug endpoints.

38. Lack of Resource and Rate Limiting in APIs
Demonstrations of how resource exhaustion can be exploited in APIs.

39. Regex Denial of Service (DoS) Attacks on APIs
How regular expressions can be used for denial of service attacks in APIs.

40. Broken Authentication in APIs
Learn how broken authentication can lead to security issues.

41. Billion Laugh Attack via XXE
A detailed look at the billion laugh attack and how it exploits XML entities.

42. Hidden API Functionality Exposure
How attackers can discover hidden API functionalities and exploit them.

43. Remote Code Execution (RCE) through Deserialization in APIs
Learn how deserialization vulnerabilities can lead to remote code execution.

PHYSICAL PENETRATION TESTING – SECTION 8

SECTION 8

PHYSICAL PENETRATION TESTING

Penetration testers need to learn physical penetration testing because many security breaches occur through physical access, which bypasses even the most sophisticated digital defenses. By understanding how attackers exploit physical vulnerabilities, such as insecure entry points, weak access controls, or human manipulation (social engineering), testers can provide a more comprehensive assessment of a company's security posture. For companies, physical penetration testing is crucial as it helps identify weaknesses in their physical infrastructure, such as buildings, servers, and equipment, ensuring that sensitive data and assets are protected from unauthorized access. Without physical security, digital security measures alone may not be enough to prevent a full-scale breach, making this skill essential for a well-rounded security strategy.

CHAPTER - 1

Introduction to Physical Security Audits

- Getting Started with Security Audits
- An overview of what physical security audits involve and why they are important.
- Core Concepts of Security Auditing
- Learn the basic principles behind assessing physical security measures.
- Key Terminology in Security Audits
- Understanding the common terms used during security assessments.
- Why Perform a Physical Security Audit?
- Discover the reasons why organizations conduct these audits and their benefits.
- Recognizing Security Threats
- Learn to identify potential physical threats that could compromise security.
- Legal and Ethical Considerations
- Understand the legal requirements and ethical concerns tied to performing security audits.

CHAPTER - 2

Preparing for a Security Audit

- Starting the Audit Process and Defining Rules
- Steps for initiating a security audit and establishing clear rules of engagement.
- Mitigating Potential Risks During Audits
- Explore strategies to minimize risks when conducting a physical security audit.
- Information Collection, Equipment Preparation, and Teamwork
- How to gather crucial data, organize necessary tools, and ensure effective communication within the team.
- Building and Leading an Audit Team
- Insights into forming a capable team for a comprehensive audit.

CHAPTER - 3

Data Collection Techniques

- Effective Methods for Collecting Information
- Techniques to gather relevant data about the security of a location.
- Visual and Electronic Monitoring Techniques
- Use of imagery, electronic surveillance, and reconnaissance for gathering information.
- Hands-On Data Collection Exercises
- Practical labs to enhance your skills in gathering information for security audits.

CHAPTER - 4

Essential Tools for Security Audits

- Introduction to Security Testing Tools
- Get familiar with various tools used for auditing physical security.
- Lock Picking and Entry Tools
- Overview of tools that can be used to bypass locks and restricted entries.
- Advanced Tools for Specialized Auditing
- Introduction to high-tech gadgets designed for more advanced security testing.
- Other Useful Equipment for Security Audits
- A look at additional tools that can aid in assessing physical security measures.

CHAPTER - 5

Human Vulnerability in Security – Social Engineering

- Understanding Social Engineering in Security
- Learn about how attackers exploit human psychology to breach physical security.
- Psychological Tactics Used in Social Engineering
- Delve into the mental tricks attackers use to manipulate people.
- Principles of Influence Used by Attackers
- Explore the core methods used by social engineers to influence their targets.
- Advanced Techniques in Social Engineering
- Learn about additional principles and methods used in social engineering attacks.

CHAPTER - 6

Lock Manipulation and Bypassing Locks

- Basics of Lock Picking
- Learn how pin locks and other common locking mechanisms work and how they can be bypassed.
- Tools and Techniques for Lock Picking
- Discover the tools and techniques required to pick various types of locks.
- Advanced Lock Picking Techniques
- Gain insights into picking high-security pins and other advanced methods for bypassing locks.

CHAPTER - 7

Advanced Bypass Techniques

- Overview of Bypass Tools and Strategies
- Learn the various tools and strategies used to bypass different security measures.
- Bypassing Door Security Systems
- Techniques for getting past locked doors using specialized tools.
- Methods for Disabling Security Sensors
- Explore the ways to bypass or disable security sensors and alarms.
- Breaking Through Physical Access Controls
- Learn how attackers bypass access control systems, such as badge readers and locks.

CHAPTER - 8

Conducting a Security Audit in Real Time

- Carrying Out a Physical Security Audit
- Explore how to execute a real-world security audit on a facility or location.
- Different Approaches to Auditing
- Learn about various approaches, such as overt, covert, and stealth audits, and how they differ.
- Surveying and Mapping Target Areas
- Techniques for exploring and understanding the layout of a target location.
- Real-World Examples of Entry and Access
- Case studies and examples of how physical security audits have been carried out in different scenarios.

CHAPTER - 9

Strengthening Security Post-Audit

- Reducing Information Exposure
- Tips for limiting the amount of sensitive information available to potential attackers.
- Defending Against Social Engineering
- Best practices to prevent social engineering attacks from compromising security.
- Securing Data Against Monitoring and Surveillance
- Methods to protect data from being monitored or captured during a security breach.

CHAPTER - 10

Real-Life Scenarios and Key Takeaways

- Reflecting on Penetration Testing Practices
- A final recap of the key concepts and practices in physical security testing.
- Success Stories and Challenges in Security Audits
- Hear real-life stories of successful security audits and the challenges faced during the process.

VOIP PENETRATION TESTING – SECTION 9

SECTION 9

VOIP PENETRATION TESTING

Penetration testers need to learn VoIP penetration testing because Voice over Internet Protocol (VoIP) systems, like any other network infrastructure, are vulnerable to various cyber threats. With the increasing reliance on VoIP for business communication, these systems have become a prime target for attackers seeking to exploit weaknesses such as poor authentication, misconfigurations, or unencrypted data. For companies, securing VoIP systems is crucial to prevent data breaches, call interception, eavesdropping, and denial of service (DoS) attacks that can disrupt operations and compromise sensitive information. By understanding how to test and secure VoIP networks, penetration testers help organizations protect their communication infrastructure and ensure business continuity.

CHAPTER - 1

Introduction to VoIP Security Testing

- Course Overview: An introduction to the essential components and skills for testing the security of VoIP systems.
- Understanding Switching Methods: A deep dive into packet-switching and circuit-switching, comparing their uses in communication networks.
- Public Switched Telephone Network (PSTN): Exploring how traditional phone systems operate and how they relate to VoIP technology.
- Layered Approach of the OSI Model: Understanding the OSI model to see how VoIP fits within various network layers.
- Internet Protocols from a Data Perspective: Examining the fundamental role of IP in data communication and its importance in VoIP.
- Overview of TCP/IP: An introduction to TCP/IP, focusing on its role in facilitating data exchange over networks.

CHAPTER - 2

Voice Over Internet Protocol (VoIP)

- VoIP Basics: An overview of VoIP technology, explaining its purpose and the transition from traditional phone systems.
- VoIP Architecture: A look into the components and structure of a VoIP system, covering servers, phones, and protocols.
- Understanding VoIP Protocols: A comprehensive explanation of the protocols that govern VoIP communication, such as SIP, RTP, and others.
- VoIP Media Protocols: Discussing the specific protocols used for handling media (voice and video) within VoIP systems.

CHAPTER - 3

Lab Setup: VoIP Hacking Environment

- Introduction to Asterisk: An overview of the Asterisk platform, which is commonly used for setting up VoIP servers.
- VoIP Servers Explained: Exploring different types of VoIP servers and their roles in a secure VoIP environment.
- Creating a VoIP Server with Trixbox: A step-by-step guide to setting up a VoIP server using Trixbox.
- Setting Up AsteriskNow IP PBX Server: Detailed instructions on creating a VoIP server using AsteriskNow.
- Trixbox Server Configuration: Guidance on configuring Trixbox to ensure a functional VoIP system.
- AsteriskNow Server Configuration: Instructions for setting up and configuring AsteriskNow for VoIP services.
- Configuring Softphones (Linphone & Zoiper): How to set up softphones, which are used to make calls over VoIP systems, for testing purposes.

CHAPTER - 4

VoIP Network Exploitation Techniques

- VoIP Information Gathering & Enumeration: Techniques for identifying and mapping out components within a VoIP network, focusing on vulnerabilities.
- Advanced Enumeration Tactics: Continuation of enumeration strategies for finding user extensions and network assets.
- User Extension Enumeration: Methods for discovering user accounts and extensions within the VoIP system.
- VoIP Authentication Cracking: Exploring how VoIP authentication works and strategies for cracking these credentials.
- Man-in-the-Middle Attacks (Passive and Active): Theoretical and practical aspects of intercepting VoIP traffic through both passive and active methods.
- VoIP Registration Hijacking & Spoofing: Understanding how attackers can take over VoIP registrations and spoof calls for malicious purposes.
- Denial of Service (DoS) Attacks on VoIP Networks: Exploring how attackers can overwhelm a VoIP system and disrupt its services.

ICS/SCADA PENETRATION TESTING – SECTION 10

SECTION 10

ICS/SCADA PENETRATION TESTING

Penetration testers must learn Industrial Control Systems (ICS) and Supervisory Control and Data Acquisition (SCADA) penetration testing due to the increasing reliance of critical infrastructure on these technologies. As cyber threats evolve, the vulnerabilities in ICS and SCADA systems—often connected to essential services like water, electricity, and transportation—become more pronounced. By understanding the unique architecture and security challenges of these systems, penetration testers can identify weaknesses that could be exploited by malicious actors. This knowledge is crucial for companies to protect sensitive operational technology, ensure compliance with regulatory standards, and maintain the safety and reliability of their services. Ultimately, effective ICS and SCADA security measures help prevent costly disruptions and safeguard public safety, making it an essential area of focus for any organization operating in these sectors.

CHAPTER - 1

Introduction to Industrial Systems Security

- Understanding Vulnerabilities in Industrial Systems
- Learn why industrial systems are attractive targets for cyber attackers due to inherent weaknesses.
- Exploring Common Attack Surfaces in Industrial Environments
- Identify typical areas where industrial control systems (ICS) are most vulnerable.
- Risks of Default Passwords and Unsecured Interfaces
- Understand how default credentials and exposed web servers can compromise security.
- Real-World Scenarios in Operational Technology Security Testing
- Discover common situations where security assessments are crucial in industrial settings.
- Types of Security Assessments in Industrial Contexts
- Learn about different classifications of penetration tests specific to industrial environments.
- Aligning Security Goals Between IT and OT
- Understand the differences and overlaps in security objectives for Information Technology (IT) and Operational Technology (OT).

CHAPTER - 2

Preparing for Industrial Security Testing

- Basics of IP Addressing and Network Segmentation
- Refresh your knowledge on IPv4 addresses and subnetting to effectively map networks.
- Introduction to Offensive Open-Source Intelligence (OSINT)
- Learn how to gather publicly available information to aid in security testing.
- Common Default Credentials in Industrial Systems
- Explore how default passwords pose significant security risks in ICS.
- Leveraging Shodan for Reconnaissance
- Use Shodan to find internet-connected devices and potential vulnerabilities.
- Scanning Public Networks for Vulnerabilities
- Learn techniques to find and analyze public IP ranges for security flaws.
- Hunting for Vulnerabilities with Government Resources
- Utilize tools and databases from organizations like CISA to identify known issues.

CHAPTER - 3

Setting Up Your Security Testing Environment

- Building Your Industrial Security Lab
- Set up a controlled environment to safely conduct security assessments.
- Overview of Virtual Machines and Lab Setup
- Learn how to configure virtual machines that simulate industrial systems.
- Installing Virtualization Software
- Step-by-step guide to installing software like VirtualBox for your lab.
- Setting Up Kali Linux for Testing
- Install and configure Kali Linux, a popular platform for security professionals.
- Configuring a Linux Server for Simulations
- Set up an Ubuntu Server to emulate industrial control devices.
- Creating Industrial System Simulations
- Learn to simulate ICS environments for practical testing.
- Installing Essential Open-Source Tools
- Equip your Kali Linux setup with tools needed for security assessments.

CHAPTER - 4

Fundamentals of Security Testing Tools

- Getting Familiar with Your Testing Platform
- Explore the tools and interfaces you'll use during penetration testing.
- Launching Honeypots and Security Tools
- Learn how to start honeypots and other tools in Kali Linux.
- Discovering Network Devices with Scanning Tools
- Use tools like Netdiscover to identify devices on a network.
- Identifying Systems with Advanced Scanning
- Master Nmap for in-depth network exploration and vulnerability detection.
- Gathering System Information with SNMP Tools
- Extract valuable data from devices using SNMP enumeration techniques.
- Introduction to Metasploit Framework
- Learn how Metasploit serves as a comprehensive toolkit for testers.
- Utilizing Various Open-Source Security Tools
- Discover additional tools that can aid in your security testing efforts.

CHAPTER - 5

Simulating Attacks on Industrial Systems

- Setting Up and Exploring PLC Simulations
- Create simulations of Programmable Logic Controllers for hands-on practice.
- Advanced Reconnaissance Techniques
- Enhance your information-gathering skills using Shodan and specialized search queries.
- Exploiting Weak Passwords and Default Settings
- Understand how attackers take advantage of poor credential practices.
- Conducting Network Scans in Simulated Environments
- Practice scanning techniques to identify potential vulnerabilities.
- Detailed Device Enumeration Strategies
- Use specialized tools to gather comprehensive information about target devices.
- Finding and Utilizing Exploits
- Learn how to search for and apply exploits from databases and repositories.
- Adding Custom Exploits to Your Toolkit
- Integrate external exploit code into your testing framework.
- Executing Simulated Attacks Safely
- Apply your skills in a controlled environment to understand attack methodologies.

CHAPTER - 6

Advanced Industrial Security Testing

- Specialized Tools for Device Analysis
- Explore advanced tools designed for specific industrial equipment.
- Assessing Real Industrial Hardware
- Learn best practices for testing actual industrial devices ethically and safely.
- Advanced Exploitation Techniques
- Dive deeper into sophisticated methods for exploiting vulnerabilities.
- Simulating Attacks on Specialized Systems
- Practice attacking simulations like gas station controllers to broaden your experience.
- Mastering Advanced Scanning Methods
- Utilize advanced Nmap features for more effective scanning results.
- Enhancing Intelligence Gathering with OSINT
- Leverage open-source information to strengthen your testing approach.
- Applying Exploitation Skills in Various Scenarios
- Test your ability to exploit different types of simulated industrial systems.
- Understanding and Testing the Modbus Protocol
- Learn about Modbus communication and how to assess devices using this protocol.
- Advanced Techniques for Finding Vulnerable Systems
- Use Shodan and other tools to locate systems susceptible to specific attacks.
- Overcoming Security Through Default Configurations
- Identify how default settings can be a security weakness and how to address them.
- Executing Exploit Modules on Target Systems
- Practice running exploit code against simulated targets effectively.
- Interacting with Device Memory Safely
- Learn how to read and manipulate memory blocks without causing harm.

CHAPTER - 7 - SCADA

- The Importance of SCADA/ICS Security
- Explore the critical reasons behind the increasing focus on securing SCADA and ICS environments.
- Unique Challenges in SCADA/ICS Security
- Understand what sets SCADA and ICS security apart from traditional IT security concerns.
- Protocols Used in SCADA Systems
- Delve into the various communication protocols that underpin SCADA systems, including their functions and vulnerabilities.
- Components of SCADA Systems
- Examine the key components that make up SCADA systems and their roles in industrial operations.
- Visibility and Reconnaissance in SCADA Environments
- Learn about techniques to enhance visibility and conduct reconnaissance in SCADA systems for better security assessments.
- PLC Programming and Simulations
- Gain insights into programming languages like Modbus and DNP3, and explore PLC simulations for practical experience.
- Conducting Risk Assessments for SCADA/ICS
- Understand how to perform thorough risk assessments tailored to the specific threats faced by SCADA and ICS.
- Penetration Testing SCADA/ICS Systems
- Learn strategies for ethically hacking SCADA systems, focusing on protocols and Human-Machine Interfaces (HMIs).
- Researching Vulnerabilities and Threat Detection
- Discover methods for identifying vulnerabilities within SCADA systems and techniques for effective threat detection.

CHAPTER - 7 - SCADA

- Creating a SCADA/ICS Honeypot
- Learn how to set up honeypots to capture and analyze attacks targeting SCADA systems.
- Securing and Monitoring SCADA Infrastructure
- Explore best practices for securing SCADA systems and monitoring them for unusual activity.
- Detecting and Analyzing Attacks on SCADA Systems
- Understand the methodologies for recognizing and investigating attacks on SCADA environments.
- Adapting Intrusion Detection Systems for SCADA/ICS
- Learn how to modify existing intrusion detection systems to better protect SCADA and ICS environments.
- Establishing Security Zones and Access Control
- Discover how to create secure zones and implement access control measures within SCADA systems.
- Best Practices for SCADA/ICS Security
- Review recommended practices that enhance the security posture of SCADA and ICS environments.
- Regulatory Standards in SCADA/ICS Security
- Familiarize yourself with the standards and regulations that govern the security of industrial control systems.
- Analyzing Case Studies in SCADA/ICS Security
- Study real-world examples to understand the implications of security breaches and the lessons learned.

ANDROID APPLICATION PENETRATION TESTING - SECTION 11

SECTION 11

ANDROID APPLICATION PENETRATION TESTING

Penetration testers really need to dive into Android penetration testing because mobile devices are such a huge part of our daily business operations. Most of us rely on apps for everything, and those apps can be prime targets for attackers looking to steal sensitive information. By understanding how to test these apps for vulnerabilities, penetration testers can catch potential security flaws before they can be exploited. This not only helps protect company data but also builds trust with customers and partners. In a world where cyber threats are constantly evolving, mastering Android penetration testing is crucial for any organization wanting to stay secure and competitive.

CHAPTER - 1

Mobile Security Lab Setup and Analysis

- Setting Up Virtual Environments
- Introduction to creating virtual machines using VirtualBox and VMware to simulate mobile security scenarios.
- Configuring Android Emulators
- Steps for installing Genymotion and setting up a virtual Android device for testing.
- Installing Mobile Security Tools
- Overview of essential tools like Mobexler and Santoku for mobile application analysis.
- Understanding Android Fundamentals
- An exploration of Android history, architecture, file structure, permissions, and the boot process.
- Android Application Structure
- Insights into application packaging, file types, and components crucial for mobile app security.
- ADB (Android Debug Bridge) Overview
- Introduction to using ADB for device communication and application management.

CHAPTER - 2

Static Analysis Tools for Android Security

- Using Reverse Engineering Tools
- Introduction to tools like Jadx, JD-GUI, APK-Tool, Android Studio, Bytecode Viewer, and QARK for static analysis.

Dynamic Analysis and Vulnerability Assessment

- Identifying Common Vulnerabilities
- Discussion of insecure logging, hardcoding, data storage issues, input validation flaws, and access control weaknesses.
- Assessing Server-Side Security
- Examination of issues related to weak server controls and insufficient transport layer protection.
- Exploring Authentication and Cryptography Weaknesses
- Analysis of poor authentication methods, broken cryptography, and client-side injection vulnerabilities.

CHAPTER - 3

Advanced Testing with Security Tools

- Installing and Using Burp Suite
- Comprehensive guide to Burp Suite for capturing and analyzing application traffic, including setup instructions for different Android versions.
- Mobile Security Framework (MobSF)
- Installation and use of MobSF for both static and dynamic analysis of mobile applications.
- Application Testing with Drozer
- Hands-on testing of mobile applications using Drozer for identifying security flaws.

CHAPTER - 4

Hands-On Application Security Challenges

- Practical Application Testing
- Engaging in hands-on labs with various Android applications to identify vulnerabilities and implement security measures.
- Man-in-the-Middle (MitM) Setup
- Step-by-step guide to setting up MitM attacks for testing application security, including intercepting traffic and understanding certificate pinning.
- Capture the Flag (CTF) Challenges
- Explore challenges like LicenseValidator and Androgoat, focusing on techniques such as reversing, binary patching, and vulnerability exploitation.

IOS APPLICATION PENETRATION TESTING – SECTION 12

SECTION 12

IOS APPLICATION PENETRATION TESTING

Penetration testers need to learn iOS penetration testing because iOS devices are widely used in both personal and professional environments, making them attractive targets for cyber threats. Understanding the unique security architecture and potential vulnerabilities of iOS applications enables testers to identify weaknesses that could be exploited by attackers. For companies, ensuring the security of their iOS apps is crucial to protect sensitive data, maintain customer trust, and comply with regulatory standards. By proactively addressing vulnerabilities through iOS penetration testing, organizations can mitigate risks, prevent costly breaches, and enhance their overall security posture.

CHAPTER - 1

- Introduction to the Course

Explore the fundamentals of securing iOS applications, including the latest trends and techniques in application security.

- Revisiting iOS Application Security

Understand the importance of continuously updating your knowledge in iOS security practices to combat evolving threats.

- Setting Up Your iOS Testing Environment

Learn how to establish a robust penetration testing lab tailored for iOS applications.

- Recent Developments in iOS Security

Stay informed about the latest features and improvements in the iOS platform that impact application security.

- Overview of iOS Architecture

Gain insights into the core components of the iOS ecosystem and how they interact to provide security.

- iOS Security Frameworks

Dive into the security architecture of iOS, including mechanisms for code integrity and data protection.

- Exploring Advanced Security Features

Discover features like pointer authentication and the secure boot process that enhance iOS security.

- Data Protection Mechanisms

Learn about the encryption techniques and data protection strategies employed by iOS to safeguard user information.

CHAPTER - 2

- Basics of iOS Security

Review foundational security concepts, including sandboxing and the Mobile Content Store (MCS).

- Jailbreaking and Provisioning

Understand the implications of jailbreaking and how mobile provisioning profiles impact application security.

- Static Analysis of iOS Apps

Master the art of static analysis to uncover vulnerabilities within iOS applications.

- Network Traffic Interception

Set up and utilize tools for intercepting and analyzing network traffic to identify potential security risks.

- Dynamic Testing with Frida

Get acquainted with Frida, a powerful dynamic analysis tool for manipulating and testing iOS applications.

- Data Storage and Security

Explore local data storage practices, including how to identify sensitive information in plist and SQLite files.

- Automation and Penetration Testing

Learn to automate security testing processes using tools like Objection and conduct tests on non-jailbroken devices.

- Reverse Engineering iOS Applications

Delve into reverse engineering techniques to analyze and understand the inner workings of iOS apps, including those written in Swift.

- Advanced Techniques

Apply advanced tactics such as binary patching to bypass security measures like SSL pinning and enhance your testing methodologies.

AWS PENETRATION TESTING - SECTION 13

SECTION 13

AWS PENETRATION TESTING

Penetration testers must learn AWS penetration testing because cloud environments like AWS have unique security challenges that differ from traditional on-premises systems. As more companies migrate their operations to the cloud, understanding how to identify and exploit vulnerabilities within these infrastructures becomes crucial. AWS offers a vast array of services, each with its own security configurations and potential weaknesses. By mastering AWS penetration testing, professionals can help organizations safeguard sensitive data, ensure compliance with industry regulations, and protect against data breaches that could lead to significant financial and reputational damage. Moreover, as cyber threats continue to evolve, companies need skilled penetration testers who can proactively assess their cloud security posture and implement robust defenses, making this knowledge not just beneficial but essential for maintaining trust and security in today's digital landscape.

CHAPTER - 1

- Fundamental Principles

Explore the foundational concepts of cloud security, focusing on risks and vulnerabilities specific to cloud environments.

- Understanding Cloud Attack Vectors

Examine the various pathways attackers may exploit to compromise cloud infrastructure and data.

- Motivations Behind Cloud Attacks

Delve into the reasons attackers target cloud services, including financial gain, data theft, and disruption of services.

- Public Bucket Permissions

Investigate the risks associated with cloud storage buckets that have open access permissions for everyone.

- Authenticated User Vulnerabilities

Learn about the dangers of allowing any authenticated AWS user to access sensitive resources.

- Exposing Cloud Credentials

Understand how inadvertently leaking AWS keys through version control systems like Git can lead to significant security breaches.

- Snapshot Accessibility Risks

Discuss the implications of making EC2 snapshots accessible to all AWS users and how to mitigate this risk.

- Instance Metadata Exposure

Evaluate the security risks posed by exposed proxy access to instance metadata.

CHAPTER - 2

- Managing Excessive Permissions

Analyze the risks associated with granting excessive permissions to users and roles in cloud environments.

- Privilege Escalation Techniques

Learn about methods attackers use to escalate privileges, including rollback tactics and automation.

- Analyzing Cloud Breaches

Study real-world examples of AWS cloud breaches to understand vulnerabilities and defensive strategies.

- Understanding SSRF in EC2

Explore Server-Side Request Forgery (SSRF) vulnerabilities in EC2 instances and their potential impact.

- Managing Secrets in Codebuild

Examine how secrets are handled in AWS CodeBuild and the implications of improper management.

- Remote Code Execution (RCE) Vulnerabilities

Investigate RCE vulnerabilities in web applications hosted on cloud platforms and strategies for prevention.

- Input Validation Flaws

Discuss the importance of input validation in preventing vulnerabilities and securing cloud applications.

- Security Risks of Open Container Images

Learn about the vulnerabilities associated with using open container images in cloud deployments.

- Exploiting SSRF for Metadata Access

Understand how SSRF can be leveraged to gain unauthorized access to private IP instance metadata.

As an exciting addition to our curriculum, students will soon have the opportunity to explore firmware analysis and Azure penetration testing. This enhancement aims to equip learners with essential skills in evaluating firmware security and assessing vulnerabilities within Azure environments, further enriching their expertise in penetration testing. Stay tuned for updates on this valuable bonus content!

TRAINING FEES 950$ USD
IF YOU ENROLLED OUR RED TEAM
TRAINING THEN IT WILL BE (850$ USD)

CONTACT US

https://wa.me/918019263448