Certified Ethical Hacker (CEH v13), Skill Labs

Course Specifications

Course Number: ACI76-033SL_rev1.0
Lab Length: Approximately 16 hours

Introduction to AI in Ethical Hacking (CEHv13)

Introduction
Objective

CEHv13 Domain

Information Security and Ethical Hacking Overview

CEHv13 Sub Domain

Introduction to Ethical Hacking

CEHv13 Objectives

- Information Security Overview
- Cyber Kill Chain Concepts
- Hacking Concepts
- Ethical Hacking Concepts
- Information Security Controls
- Information Security Laws and Standards

Overview

The Information Security and Ethical Hacking Overview lab is an introduction to the principles and practices of ethical hacking. It covers key concepts such as information security overview, Cyber Kill Chain, hacking and ethical hacking concepts, information security controls, and relevant laws and standards. This domain provides essential knowledge to grasp the importance of securing systems and the ethical approach to identifying vulnerabilities.

Learning Outcomes

In this lab, you will learn to:

- Simulate prompt injection attacks and evaluate their impact.
- Perform simple automated reconnaissance and learn various options using ShellGPT
- Learn how to use Kali GPT.

	Key Term	Description
1	Prompt Injection	A technique used to manipulate the behavior of a large language model (LLM) by inserting hidden or conflicting instructions into user inputs, often bypassing content filters or policies
2	ShellGPT	A command-line tool that integrates with OpenAI's GPT models to convert natural language prompts into shell commands, enhancing automation and efficiency in reconnaissance and testing tasks
3	AI-Powered Reconnaissance	The use of artificial intelligence tools to automate the gathering of public information about targets, including domain data, subdomains, open ports, and metadata
4	OWASP Top 10 for LLMs	A security framework identifying the top ten vulnerabilities found in applications that use LLMs, such as prompt injection, training data poisoning, and sensitive data leakage
5	AI Abuse Detection	The identification of malicious or unethical uses of AI, including the generation of phishing emails, fake media (deepfakes), or model misuse for social engineering attacks
6	Ethical AI Hacking	The practice of testing AI systems for vulnerabilities and weaknesses—such as prompt manipulation or unintended output—using legally and ethically responsible methods
7	Overreliance on AI	A security risk where users or systems blindly trust AI outputs without validation, potentially leading to exploitation, misinformation, or operational errors
8	Training Data Poisoning	The insertion of harmful or misleading content into the datasets used to train AI models, causing the model to produce biased, inaccurate, or insecure outputs

Footprinting and Reconnaissance Techniques with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Reconnaissance Techniques

CEHv13 Sub Domain

Footprinting and Reconnaissance

CEHv13 Objectives

- Footprinting Concepts
- Footprinting Methodology
- Footprinting through Search Engines
- Footprinting through Web Services

- Footprinting through Social Networking Sites
- Website Footprinting
- Email Footprinting
- Whois Footprinting
- DNS Footprinting
- Network Footprinting
- Footprinting through Social Engineering
- Footprinting Tools
- Footprinting Countermeasures

Overview

This lab is designed to teach students the foundational skills of reconnaissance and footprinting, enhanced by AI-powered tools. Through five structured exercises, learners will use both manual techniques and AI augmentation using ShellGPT and Kali GPT to simulate real-world penetration testing workflows. Each section targets a specific aspect of recon: search engines, web services, social media, website technologies, and DNS systems. The goal is to equip ethical hackers with modern, efficient methods of discovering publicly accessible information that may expose vulnerabilities, all while reinforcing legal and ethical boundaries in cybersecurity.

Learning Outcomes:

In this lab, you will learn to:

- Understand the foundational principles of reconnaissance and footprinting.
- Analyze ethical considerations and legal frameworks.
- Interpret the role of AI tools, such as ShellGPT and ChatGPT, in recon workflows.
- Prepare to apply both traditional and AI-augmented methods.

	Key Term	Description
1	Reconnaissance	The process of collecting information about a target system or organization prior to conducting an attack or security test
2	Footprinting	A specific type of reconnaissance aimed at gathering detailed information like IP ranges, domain names, and network infrastructure
3	Passive Reconnaissance	Gathering information without directly interacting with the target (e.g., WHOIS lookups, search engine queries)
4	Active Reconnaissance	Information gathering that involves interacting with the target (e.g., ping sweeps, port scanning)
5	Open Source Intelligence (OSINT)	Intelligence collected from publicly available sources such as websites, social media, and public records
6	DNS Footprinting	The process of gathering information about a domain's DNS records to understand its structure and services
7	Metadata	Hidden data embedded in documents and files that can reveal

	Key Term	Description
		author names, creation dates, software used, and other insights
8	WhatWeb/Nikto	Tools used to fingerprint websites by identifying their server software, plugins, and vulnerabilities
9	ShellGPT	An AI-driven assistant designed to provide educational guidance, step walkthroughs, and security advice within Kali Linux

Network Reconnaissance Techniques with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Reconnaissance Techniques

CEHv13 Sub Domain

Scanning Networks

CEHv13 Objectives

- Network Scanning Concepts
- Scanning Tools
- Host Discovery
- Port and Service Discovery
- OS Discovery (Banner Grabbing and OS Fingerprinting)
- Scanning Beyond IDS and Firewall
- Draw Network Diagrams

Overview

This lab introduces students to practical network scanning techniques in alignment with CEHv13's reconnaissance and scanning objectives. Leveraging Kali GPT and ShellGPT, learners gain real-time feedback, syntax suggestions, and AI-guided troubleshooting for each task.

Learning Outcomes:

In this lab, you will learn to:

- Understand and apply network scanning concepts.
- Use multiple scanning tools to detect hosts, services, and operating systems.
- Evade IDS and firewalls during scans.
- Visualize network topologies from collected data.

	Key Term	Description
1	Reconnaissance	The phase of collecting information about a target system
2	Host Discovery	Identifying live systems on a network
3	Port Scanning	Determining which network ports are open
4	Banner Grabbing	Technique for collecting service information
5	Stealth Scanning	Methods to scan without triggering detection systems
6	TCP Stealth Scan	Also called SYN scan, used to quietly identify open ports
7	Zenmap	GUI for Nmap with visualization capabilities
8	ShellGPT	Command-line AI assistant for helping with command syntax and debugging
9	Kali GPT	An AI assistant for Kali Linux that provides real-time guidance, tool explanations, and support for ethical hacking and penetration testing tasks

Enumeration Reconnaissance Techniques with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Reconnaissance Techniques

CEHv13 Sub Domain

Enumeration

CEHv13 Objectives

- Enumeration Concepts
- NetBIOS Enumeration
- SNMP Enumeration
- LDAP Enumeration
- NTP and NFS Enumeration
- SMTP and DNS Enumeration
- Other Enumeration Techniques (IPsec, VoIP, RPC, Unix and Linux, Telnet, FTP, TFTP, SMB, IPv6, and BGP enumeration)
- Enumeration Countermeasures

Overview

This lab explores enumeration techniques vital to ethical hacking and penetration testing, aligned with CEHv13. Students will learn how to gather information from various services using both manual and AI-assisted methods via Kali GPT and ShellGPT. These tools enable real-time guidance, command generation, contextual learning, and enhanced understanding of enumeration protocols.

Learning Outcomes:

In this lab, you will learn:

- Understand enumeration concepts and objectives in the ethical hacking lifecycle.
- Perform enumeration on NetBIOS, SNMP, LDAP, DNS, and other protocols.
- Utilize AI for guidance, scripting, and tool recommendations.
- Evaluate enumeration countermeasures and propose security defenses.

	Key Term	Description
1	Enumeration	Active information gathering post-network scanning
2	NetBIOS, SNMP, LDAP, and DNS	Protocols commonly exploited during enumeration
3	Kali GPT	AI assistant within Kali Linux for tool recommendation, MITRE ATT&CK alignment, and syntax generation
4	ShellGPT	AI assistant in the terminal for real-time scripting, command generation, and automation

Vulnerability Analysis Tools & Techniques with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

System Hacking Phases and Attack Techniques

CEHv13 Sub Domain

Vulnerability Analysis

CEHv13 Objectives

- Vulnerability Assessment Concepts
- Vulnerability Classification and Assessment Types
- Vulnerability Assessment Solutions and Tools
- Vulnerability Assessment Reports

Overview

In this lab for CEHv13, students will explore critical concepts and hands-on techniques in vulnerability assessment and management. Leveraging industry-standard tools like OpenVAS, Damn Vulnerable Web Application (DVWA), Kali GPT, and ShellGPT, students will learn to identify, analyze, and document vulnerabilities effectively. This practical exercise enhances essential cybersecurity skills required for robust system protection.

Learning Outcomes:

In this lab, you will learn to:

- Understand vulnerability assessment concepts and methodologies.
- Classify and differentiate between various types of vulnerability assessments.
- Deploy and configure vulnerability assessment solutions, such as OpenVAS.
- Conduct thorough vulnerability scans.
- Generate comprehensive vulnerability assessment reports.
- Use ChatGPT for vulnerability analysis and reporting.

	Key Term	Description
1	Vulnerability Assessment	The systematic process of identifying and classifying security weaknesses in a system
2	OpenVAS	An open-source vulnerability assessment scanner used to identify vulnerabilities across networks and systems
3	DVWA	A web application is intentionally designed with multiple vulnerabilities to aid learning and practicing web security
4	Wireshark	A network protocol analyzer tool for capturing and analyzing network traffic
5	Kali GPT	A Kali Linux-integrated AI tool leveraging GPT models for cybersecurity analysis and tasks
6	ShellGPT	A GPT-based command-line tool that assists users in generating scripts and automating cybersecurity tasks

System Hacking Methodologies with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

System Hacking Phases and Attack Techniques

CEHv13 Subdomain

System Hacking

CEHv13 Objectives

- System Hacking Concepts
- Gaining Access
- Cracking Passwords
- Vulnerability Exploitation
- Escalating Privileges
- Maintaining Access
- Executing Applications
- Hiding Files
- Clearing Logs

Overview

This lab immerses students in the CEH v13 "System Hacking Phases and Attack Techniques" domain. You will reenact a realistic breach against a Windows Active Directory environment, pivot to sensitive-file exfiltration, and finally erase your footprints. Throughout, you will reinforce the attack–defense cycle: enumerate → exploit → escalate → exfiltrate → cover tracks → restore normal user privileges.

Learning Outcomes:

In this lab, you will learn to:

- Review previous enumeration data and—with ChatGPT's context suggestions—spot credential-reuse opportunities.
- Enumerate Active Directory objects and trust relationships.
- Exploit a vulnerable service to gain an initial foothold.
- Escalate privileges from domain user to domain admin.
- Install and use a steganography tool to hide data in plain sight.
- Exfiltrate a protected file over an encrypted channel.
- Delete PowerShell command history and relevant Windows Event Logs.
- Remove attacker-deployed tools and restore legitimate permissions.
- Leverage ChatGPT to translate plain English intents into precise PowerShell, CMD, or Bash commands.

	Key Term	Description
1	Credential reuse	Leveraging identical passwords or hashes across multiple hosts or services to expand compromise
2	Steganography	Concealing information inside innocuous files (e.g., PNGs, WAVs) to evade detection
3	Active Directory	Active Directory (AD) remains a crown jewel for attackers because compromising a single privileged service account can yield domain-wide control. Modern exploits take advantage of protocol weaknesses such as New Technology LAN Manager (NTLM) relaying, misconfigured delegation, and improperly scoped Group Policy Objects.

	Key Term	Description
4	Privilege escalation	Privilege escalation leverages both unpatched vulnerabilities and insecure default configurations. Examples include exploiting service permissions, Dynamic Link Library (DLL) search-order hijacking, and token impersonation via SeImpersonatePrivilege.
5	Covering tracks	Covering tracks is the process attackers use to hide evidence of their activities on a system or network to avoid detection and maintain access.

Malware Threat Concepts with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

System Hacking Phases and Attack Techniques

CEHv13 Sub Domain

Malware Threats

CEHv13 Objectives

- Malware Concepts
- APT Concepts
- Trojan Concepts
- Virus and Worm Concepts
- Fileless Malware Concepts
- Malware Analysis
- Malware Countermeasures
- Anti-Malware Software

Overview

The Malware Threat Concepts with AI lab walks you through building safe proofs of concept and then detecting, analyzing, and remediating them.

Learning Outcomes

In this lab, you will learn to:

- Explain how different malware types propagate and persist.
- Launch a controlled fork bomb to illustrate virus behavior.
- Use ChatGPT and ShellGPT to assist with malware threats.
- Perform on-host antimalware scans.

- Summarize countermeasures for fileless and traditional malware.

	Key Term	Description
1	Malware	Malicious software designed to compromise confidentiality, integrity, or availability
2	APT	Advanced Persistent Threat—an adversary that gains long-term covert access.
3	Fork Bomb	Self-replicating process that exhausts system resources
4	Fileless Malware	Malware that resides primarily in memory or leverages legitimate tools (e.g., PowerShell)
5	Heuristic Analysis	Behavioral detection technique that flags unknown binaries
6	Signature	Unique byte pattern used by AV engines to identify known threats
7	EDR	Endpoint Detection and Response (EDR)—real-time host telemetry and automated response.

Network Sniffing Techniques with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Network and Perimeter Hacking

CEHv13 Sub Domain

Sniffing

CEHv13 Objectives

- Sniffing Concepts
- Sniffing Technique: MAC Attacks
- Sniffing Technique: DHCP Attacks
- Sniffing Technique: ARP Poisoning
- Sniffing Technique: Spoofing Attacks
- Sniffing Technique: DNS Poisoning
- Sniffing Tools
- Sniffing Countermeasures
- Sniffing Detection Techniques

Overview

In this hands-on lab, you will reproduce two common Layer-2 and Layer-3 sniffing attacks—Media Access Control (MAC)-table flooding and Domain Name System (DNS) poisoning—then pivot to a

detection scenario that shows how defenders discover and mitigate those attacks. You will conduct all tasks using Kali Linux, Windows 11 client, and a Windows Server 2025 victim VM, employing both traditional command-line tools and Kali Generative Pre-trained Transformer (GPT) and ShellGPT to accelerate research, scripting, and troubleshooting.

Learning Outcomes

By the end of the lab, you will be able to:

- Explain how packet sniffing leverages weaknesses in switching and name-resolution protocols.
- Flood a switch's CAM table with fake MAC addresses using macof.
- Redirect a victim's DNS queries with dnschef to serve malicious answers.
- Generate AI-assisted Bash one-liners and PowerShell payloads with Kali GPT and Shell GPT.
- Detect MAC flooding and DNS spoofing with Wireshark, ARPwatch, and Zeek.
- Recommend countermeasures—port security, DHCP snooping, Domain Name System Security Extensions (DNSSEC), and continuous monitoring.

	Key Term	Description
1	Sniffer	Software or hardware that copies network frames for analysis
2	MAC Flooding	Overwhelming a CAM table with bogus entries so the switch fails open and broadcasts traffic
3	ARP Poisoning	Injecting forged ARP replies to associate the attacker's MAC with another IP
4	DNS Poisoning (Spoofing)	Altering DNS responses to direct victims to attacker-controlled IPs
5	dnschef	Flexible DNS proxy for spoofing and logging queries
6	Port Security	Switch feature that limits the number of MAC addresses per port
7	Kali GPT / Shell GPT	AI assistants pre-tuned for penetration testing and shell command generation

Social Engineering Techniques and Exploits with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Network and Perimeter Hacking

CEHv13 Sub Domain

Social Engineering

CEHv13 Objectives

- Social Engineering Concepts
- Social Engineering Techniques

- Insider Threats
- Impersonation on Social
- Networking Sites
- Identity Theft
- Social Engineering Countermeasures

Overview

In this hands-on lab, you will explore how attackers manipulate human trust, gather intelligence, and pivot into technical compromise. You will use classic reconnaissance tools (DNS utilities, TheHarvester, Recon-ng, Nmap) along with AI helpers—Kali GPT and ShellGPT—to expedite command generation, result interpretation, and report writing. By the end, you will understand both the psychology and the packet traces behind modern social engineering campaigns and know how to defend against them.

Learning Outcomes:

By the end of this lab, you will be able to:

- Conduct multiphase reconnaissance.
- Clone and deliver a phishing site end to end.
- Exploit social media trust relationships.
- Analyze psychological triggers within crafted luresR.
- Recommend layered countermeasures.
- Document findings with AI assistance.

	Key Term	Description
1	Social Engineering (SE)	Tricking people so you can trick computers
2	Phishing	A fraudulent message that lures a user into revealing secrets or running malware
3	Typosquatting	Registering look-alike domains (gooogle.com) to catch mistyped traffic
4	Dumpster Diving	Searching discarded material for sensitive info
5	Elicitation	Drawing information out of someone through casual conversation
6	Reconnaissance/Footprinting	Collecting open-source information to prepare an attack
7	Kali/ShellGPT	A GPT-powered assistant built into Chat GPT and Kali that suggests commands and explains output; ShellGPT is a terminal-native GPT tool that writes one-liners, decodes errors, and drafts reports

Denial-of-Service Attacks w/AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Network and Perimeter Hacking

CEHv13 Sub Domain

Denial-of-Service

CEHv13 Objectives

- DoS/DDoS Concepts
- DoS/DDoS Attack Techniques
- Botnets
- DDoS
- Case Study
- DoS/DDoS Attack Tools
- DoS/DDoS Countermeasures
- DoS/DDoS Protection Tools

Overview

This lab introduces the student to Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS) attack concepts, tools, and defenses. Students will use Kali Linux, along with Kali GPT and Shell GPT, to automate and explain various attack techniques, including SYN floods, ICMP floods, and the Ping of Death. The lab also covers disabling a firewall and launching a SYN flood using the Metasploit Framework.

Learning Outcomes:

By the end of this lab, you will be able to:

- Understand DoS and DDoS fundamentals.
- Perform a SYN flood attack using hping3.
- Perform an ICMP flood using ping or nping.
- Launch a Ping of Death attack.
- Conduct an SYN flood using Metasploit.
- Disable the Windows 11 firewall to simulate attack success.
- Use Kali GPT for tool explanations.
- Use Shell GPT for command assistance and automation.

	Key Term	Description
1	DoS	Denial-of-Service: An attack that disrupts normal traffic of a targeted system or service.
2	DDoS	Distributed Denial-of-Service: A DoS attack using multiple systems.
3	SYN Flood	Attack exploiting the TCP handshake by flooding with SYN requests.
4	ICMP Flood	An attack sending massive ICMP Echo Requests (ping) to overload the system.
5	Ping of Death	An attack sending malformed or oversized ping packets.
6	Metasploit	A popular penetration testing framework used to exploit vulnerabilities.
7	Kali GPT / Shell GPT	Kali GPT is an AI-powered assistant in Kali Linux for tool explanations and usage help. Shell GP is CLI-based AI assistant to generate and run terminal commands effectively.

Session Hijacking Concepts with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Network and Perimeter Hacking

CEHv13 Sub Domain

Session Hijacking

CEHv13 Objectives

- Session Hijacking Concepts
- Application Level Session Hijacking
- Network Level Session Hijacking
- Session Hijacking Tools
- Session Hijacking Countermeasures

Overview

Session hijacking is a critical web application security vulnerability where an attacker gains unauthorized access to a user's active session. This lab explores the mechanics of session hijacking through hands-on activities, simulating how attackers can steal and reuse session cookies to impersonate users without ever needing to know their login credentials.

Throughout this lab, you step into the role of both the victim and the attacker. You begin by launching a vulnerable web application, monitoring network traffic with Wireshark, and extracting a valid session cookie transmitted in plaintext. You then transfer this session ID to a second machine and inject it into a browser to successfully hijack the original user's session.

This experience highlights the risks associated with transmitting session data over unsecured channels (such as HTTP), the ease with which attackers can exploit weak session management practices, and the importance of encryption and secure cookie handling. By completing this lab, you gain a deeper understanding of session security and the real-world consequences of its failure.

Learning Outcomes

In this lab, you will be able to:

- Explain the concept of session hijacking and identify the conditions under which it can occur.
- Launch and configure a Damn Vulnerable Web Application (DVWA) using XAMPP on a Windows server.
- Capture HTTP network traffic using Wireshark and apply filters to isolate relevant packets.
- Identify and extract session cookies (e.g., PHPSESSID) from captured HTTP requests.
- Transfer and manipulate session data between virtual machines using tools like smbclient.
- Inject a stolen session cookie into a web browser using developer tools to hijack a session.
- Demonstrate how attackers bypass authentication mechanisms using intercepted session identifiers.
- Evaluate the importance of HTTPS, secure cookies, and logout practices as countermeasures to session hijacking.
- Reflect on ethical considerations and responsible disclosure when testing session management vulnerabilities.

	Key Term	Description
1	Session Hijacking	The act of taking over a valid user session by stealing the session ID (usually stored in a cookie) to impersonate the user without their credentials
2	Session Cookie	A small piece of data (e.g., PHPSESSID) that stores the user's session ID and is sent to the server with each request to maintain a logged-in state
3	HTTP vs HTTPS	HTTP transmits data in plaintext, making it vulnerable to eaves
4	Wireshark	A network protocol analyzer that captures and displays packets in real time, used for monitoring and analyzing network traffic
5	GET Request	An HTTP request method used to retrieve data from a server; it often includes headers like cookies that can be intercepted
6	PHPSESSID	The default session identifier used by PHP applications to track users' sessions; it is typically stored as a cookie
7	On-Path Attack	A type of network attack where the attacker intercepts communication between two parties without either party knowing, previously called "man-in-the-middle"
8	XAMPP	A free and open-source cross-platform web server stack package that includes Apache, MySQL, PHP, and more—used here to host DVWA
9	Damn Vulnerable Web Application (DVWA)	A deliberately insecure web application designed for security training and penetration testing practice

	Key Term	Description
10	Developer Tools	Built-in browser tools (e.g., in Edge or Chrome) that allow developers to inspect and modify web page elements, cookies, and storage—used in this lab to inject the session cookie
11	Cookie Injection	The act of manually placing a cookie into a browser's memory or storage to manipulate or hijack a web session
12	Logout Hygiene	The practice of always logging out of applications to invalidate the session cookie and reduce the risk of session hijacking

Compromising Web Servers with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Web Application Hacking

CEHv13 Sub Domain

Hacking Web Applications

CEHv13 Objectives

- Web App Concepts
- Web App Threats
- Web App Hacking Methodology
- Footprint Web Infrastructure
- Analyze Web Applications
- Bypass Client-Side Controls
- Attack Authentication Mechanism
- Attack Authorization Schemes
- Attack Access Controls
- Attack Session Management Mechanism
- Perform Injection Attacks
- Attack Application Logic Flaws
- Attack Shared Environments
- Attack Database Connectivity
- Attack Web App Client
- Attack Web Services
- Web API, Webhooks and Web Shell
- Web App Security

Overview

This lab is designed to introduce you to the core concepts of compromising web servers. In this hands-on session, you'll work with multiple tools and techniques to exploit vulnerabilities in web servers. By the end of the lab, you will have the skills necessary to conduct basic reconnaissance, identify server versions, find and scan files, and perform various types of attacks on web servers.

Lab Objectives:

After completing this module, you should be able to:

- Footprint using the nc command.
- Find the web server version using the Metasploit Framework.
- Find files on a web server using Metasploit Framework.
- Check for WebDAV on a web server using Metasploit Framework.
- Perform vulnerability scanning using Nikto.
- Perform web application brute forcing using DirBuster.

	Key Term	Description
1	Footprinting	The process of gathering information about a target system. It typically involves gathering DNS information, IP addresses, domain names, and server configurations.
2	WebDAV	A protocol that allows users to manage and manipulate files over the HTTP protocol. It can be a potential vulnerability when improperly configured.
3	Metasploit Framework	A popular tool used for testing exploits and assessing vulnerabilities in systems. It contains a variety of modules, including ones for web application attacks, vulnerability scanning, and brute-forcing.
4	Nikto	A web server scanner that identifies potential vulnerabilities in web applications, such as outdated software, misconfigurations, and common security flaws.
5	DirBuster	A tool that performs brute-force directory and file busting on web servers to identify hidden directories or files.

Web Application Hacking with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Web Application Hacking

CEHv13 Sub Domain

Hacking Web Applications

CEHv13 Objectives

- Web App Concepts
- Web App Threats
- Web App Hacking Methodology
- Footprint Web Infrastructure
- Analyze Web Applications
- Bypass Client-Side Controls
- Attack Authentication Mechanism
- Attack Authorization Schemes
- Attack Access Controls
- Attack Session Management Mechanism
- Perform Injection Attacks
- Attack Application Logic Flaws
- Attack Shared Environments
- Attack Database Connectivity
- Attack Web App Client
- Attack Web Services
- Web API, Webhooks, and Web Shell
- Web App Security

Overview

The Web Application Hacking with AI practice lab simulates a real-world environment where you will identify and exploit vulnerabilities in web applications. You will explore different types of web app threats, gather intelligence about the target's infrastructure, and conduct attacks ranging from authentication bypass to injection flaws. The lab integrates ChatGPT for contextual guidance and vulnerability explanation, and ShellGPT for live command-line support during reconnaissance and exploitation.

This approach ensures you not only execute the attacks but also understand the theory and security implications behind each step.

Lab Objectives

After completing this lab, you will be able to:

- Identify common web application threats and their potential impacts.
- Enable and test a vulnerable web server environment.
- Sniff sensitive credentials transmitted over insecure channels.
- Perform exploitation techniques such as broken authentication, OS command injection, SSI injection, and cross-site scripting.
- Use reconnaissance tools such as wafw00f, nmap, host, and Legion to enumerate web application infrastructure.
- Apply both manual and automated techniques for gathering information and planning web app attacks.

	Key Term	Description
1	Broken Authentication	A vulnerability where flaws in the authentication process allow attackers to impersonate users
2	OS Command Injection	An attack that allows execution of arbitrary operating system commands through a vulnerable application
3	Server-Side Includes (SSI) Injection	Exploitation of web server directives to execute commands or disclose information
4	Cross-Site Scripting (XSS)	A vulnerability where attackers inject malicious scripts into trusted websites
5	Footprinting	The process of gathering information about a target system or network before an attack
6	Enumeration	The active collection of detailed information about a system, often revealing exploitable details
7	wafw00f	A tool for detecting and identifying web application firewalls (WAFs)

SQL Injection Methodologies with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Web Application Hacking

CEHv13 Sub Domain

SQL Injection

CEHv13 Objectives

- SQL Injection Concepts
- Types of SQL Injection
- SQL Injection Methodology
- SQL Injection Tools
- Evasion Techniques
- SQL Injection Countermeasures

Overview

This lab introduces learners to the methodologies of SQL injection (SQLi), one of the most prevalent and dangerous forms of web application attacks. The focus will be on understanding how malicious SQL statements can be injected into input fields to manipulate backend databases. Learners will work with the Damn Vulnerable Web Application (DVWA) environment to safely simulate a real-world SQLi exploit.

Through hands-on practice, participants will learn how to exploit SQL vulnerabilities, extract sensitive data such as usernames and password hashes, and apply password cracking techniques. More importantly, this lab emphasizes the defensive side—helping learners to recognize countermeasures that can protect systems from such attacks.

Lab Objectives

- After completing this lab, you should be able to:
- Verify that the DVWA environment is operational.
- Perform SQLi exploits on a vulnerable database.
- Capture and extract password hashes.
- Use John the Ripper to crack captured hashes.
- Demonstrate awareness of SQLi countermeasures.

	Key Term	Description
1	SQL Injection (SQLi)	A code injection technique where malicious SQL queries are inserted into input fields to manipulate backend databases
2	Damn Vulnerable Web Application (DVWA)	A deliberately insecure web application designed for security testing and training
3	Password Hash	A cryptographic representation of a password, often targeted by attackers for offline cracking
4	John the Ripper	A password-cracking tool used to test the strength of password hashes
5	Evasion Technique	A method used by attackers to bypass detection systems or security filters when conducting SQLi

Introduction to Cloud Computing with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Cloud Computing

CEHv13 Sub Domain

Cloud Computing

CEHv13 Objectives

- Cloud Computing Concepts
- Container Technology
- Serverless Computing
- Cloud Computing Threats
- Cloud Hacking

- Cloud Security

Overview

This lab orients you to foundational Amazon Web Services (AWS) and safe, hands-on management interfaces used by security practitioners. You will explore the AWS Management Console (graphical UI) and AWS CloudShell (browser-based command-line interface [CLI]) inside a pre-provisioned, restricted environment. Along the way, you'll connect CEHv13 theory to practice: how cloud building blocks (Elastic Compute Cloud [EC2], Relational Database Service [RDS], Simple Storage Service [S3], Identity and Access Management [IAM]) map to Infrastructure as a Service (IaaS)/Platform as a Service (PaaS) models, where misconfigurations typically occur, and which native controls (IAM, Key Management Service [KMS], CloudTrail) help prevent, detect, and respond to threats.

The emphasis is on read-only discovery, correct regional context, and least privilege. You will identify where to review identities and permissions (IAM), spot common risk areas (S3 public access, overly permissive roles), and run a few safe CLI commands in CloudShell to verify identity and region. This prepares you for later CEH labs that simulate attacks ethically and show the corresponding defenses.

Alert: A pre-populated AWS account will be used to complete the exercises in this module. It will not be necessary to sign up for a free account through Amazon Web Services. It is important to note that certain restrictions have been applied to the lab environment. These include that only specific resources can be created in the specified region. If the region is changed, the steps in the tasks will not function accordingly.

Lab Objectives

By the end of this lab, you will be able to:

- Access and navigate the AWS Management Console in the correct region.
- Locate and describe core services (EC2, RDS, S3, IAM) and their primary security touchpoints.
- Open AWS CloudShell and run basic, safe commands (e.g., aws sts get-caller-identity, aws configure list, aws s3 ls as permitted).
- Identify your effective identity and region and explain why region scoping matters for security.
- Recognize common cloud threats at a conceptual level (misconfiguration, exposed keys, over-privileged IAM, public S3) and the native controls that mitigate them (PoLP, KMS, CloudTrail, GuardDuty).

	Key Term	Description
1	AWS Management Console	Web UI for creating and managing AWS resources.
2	Root User	The original account owner with unrestricted permissions; should be protected and rarely used.
3	Identity and Access Management (IAM)	A user is a long-lived identity; a role is an assumable identity (preferred for workloads).
4	Policy (IAM)	JSON document defining allowed/denied actions on resources
5	Principle of Least Privilege (PoLP)	Grant only the minimum permissions required.
6	Virtual Private Cloud VPC (VPC)	Logically isolated network in AWS.
7	Lambda	Serverless functions with per-invocation IAM and

	Key Term	Description
		ephemeral runtime
8	GuardDuty/Config	Threat detection (GuardDuty) and configuration/compliance tracking (AWS Config).

Cryptography Techniques with AI (CEHv13)

Introduction
Objective

CEHv13 Domain

Cryptography

CEHv13 Sub Domain

Cryptography

CEHv13 Objectives

- Cryptography Concepts
- Encryption Algorithms
- Cryptography Tools
- Public Key Infrastructure (PKI)
- Email Encryption
- Disk Encryption
- Cryptanalysis
- Countermeasures

Overview

Welcome to the Cryptography Techniques practice lab. You will gain hands-on experience with fundamental cryptographic operations and platform-native disk encryption on Windows. The lab balances concepts with practical tasks: hashing and file encryption using utility tools, and securing data at rest with BitLocker on a fixed data drive.

Across two focused exercises, you will prepare sample files, generate and verify hashes for integrity, perform basic file encryption, and then configure and validate BitLocker—covering policy prerequisites, key-protectors, recovery keys, and user unlock flows. By the end, you should be able to articulate when to use each control (hashing vs. file encryption vs. full-disk encryption), how the controls differ, and how to harden their deployment.

Lab Objectives

After completing this module, you will be able to:

- Prepare files and folders for encryption.
- Use the Crypt4Free tool for file encryption/decryption.
- Use the HashCalc tool to compute and verify message digests.

- Enable the "Require additional authentication at startup" BitLocker policy (no TPM scenario).
- Encrypt a fixed data drive with BitLocker and validate recovery mechanisms.

	Key Term	Description
1	Confidentiality	Keeping information hidden from unauthorized parties
2	Integrity	Assurance that data has not been altered (often validated with hashes)
3	Availability	Ensuring authorized users can access data when needed
4	Symmetric Encryption	One key for both encryption and decryption (e.g., AES)
5	Asymmetric Encryption	Key pair model using public/private keys (e.g., RSA, ECC)
6	BitLocker	Windows FDE that supports TPM, PIN, and recovery keys; commonly uses XTS-AES
7	Trusted Platform Module (TPM)	Hardware chip that secures keys and measures boot integrity