UWE CTF Falcons Ethical Hacking and CTF Guide

Contents

1. Introduction

Welcome to the UWE CTF Falcons Ethical Hacking Guide! This guide is designed to provide members with a comprehensive understanding of ethical hacking and prepare them for Capture The Flag (CTF) competitions. Ethical hacking involves testing and evaluating the security of systems and networks to identify vulnerabilities and improve overall security.

2. Ethical Hacking Principles

Ethical hacking is founded on the following principles:

Permission: Always obtain explicit permission before performing any hacking activities on systems or networks.

Confidentiality: Respect the privacy and confidentiality of the data and systems you are testing.

Integrity: Ensure that your actions do not negatively impact the integrity or availability of the systems and data.

Accountability: Be accountable for your actions and maintain detailed documentation of your activities and findings.

Legal Compliance: Adhere to all relevant laws and regulations.

3. Legal Considerations

Understanding the legal landscape is crucial for ethical hackers. Here are some key points:

Computer Misuse Act 1990: In the UK, this act outlines offenses related to unauthorized access to computer systems.

GDPR (General Data Protection Regulation): Compliance with GDPR is essential when handling personal data.

University Policies: Adhere to UWE's IT policies and guidelines.

NDAs (Non-Disclosure Agreements): Respect any NDAs you sign when participating in external engagements or CTF competitions.

4. Essential Skills and Tools

To be effective ethical hackers, you need a strong foundation in various technical skills and tools. Here are some essentials:

Skills

Networking: Understanding TCP/IP, DNS, HTTP, and other protocols.

Programming: Proficiency in languages such as Python, JavaScript, and C/C++.

Web Security: Knowledge of web vulnerabilities like XSS, SQL injection, and CSRF.

System Security: Familiarity with operating systems (Windows, Linux) and their security mechanisms.

Cryptography: Basic understanding of encryption, hashing, and digital signatures.

Tools

Kali Linux: A popular penetration testing operating system.

Nmap: Network scanning and enumeration tool.

Wireshark: Network protocol analyser.

Burp Suite: Web vulnerability scanner.

Metasploit: Penetration testing framework.

John the Ripper: Password cracking tool.

5. CTF (Capture The Flag) Overview

5. CTF (Capture The Flag) Overview

Capture The Flag (CTF) competitions are designed to challenge participants with a variety of security-related tasks. They come in different formats, each emphasizing different aspects of cybersecurity. Here is an expanded overview of the main types of CTF competitions:

Jeopardy-Style CTFs

Description: In Jeopardy-style CTFs, teams or individuals solve challenges across various categories to earn points. Each challenge is independent and has a different point value based on its difficulty.

Categories:

Web Exploitation: Identifying and exploiting vulnerabilities in web applications.

Cryptography: Decrypting messages, cracking hashes, and solving puzzles related to cryptographic algorithms.

Forensics: Analysing digital artifacts such as memory dumps, disk images, and network traffic to uncover hidden information.

Reverse Engineering: Decompiling and analysing binaries to understand their functionality and find hidden features or vulnerabilities.

Binary Exploitation: Exploiting vulnerabilities in compiled programs, such as buffer overflows and format string vulnerabilities.

Miscellaneous: Challenges that don't fit into the other categories, including steganography, trivia, and puzzles.

Attack-Defence CTFs

Description: In Attack-Defence CTFs, teams are given identical systems or networks to defend while simultaneously attacking the systems of other teams. These competitions emphasize both offensive and defensive skills.

Key Elements:

Service Protection: Ensuring the availability and security of provided services.

Exploitation: Finding and exploiting vulnerabilities in the opponent's systems.

Patching: Quickly identifying and fixing vulnerabilities in your own systems.

Incident Response: Detecting and mitigating attacks in real-time.

King of the Hill (KoTH)

Description: In KoTH CTFs, participants compete to gain and maintain control of a vulnerable system. Points are awarded based on the duration of control over the system

Key Elements:

Privilege Escalation: Gaining higher-level access within the system.

Persistence: Maintaining access and control over the system while defending against other attackers.

Defence: Implementing security measures to prevent others from taking control.

Boot-to-Root CTFs

Description: In Boot-to-Root CTFs, participants start with minimal access to a system and must escalate their privileges to achieve root or administrator-level access. These are typically found in virtual machines or isolated environments

Key Elements:

Initial Access: Finding a way to gain an initial foothold on the target system.

Privilege Escalation: Exploiting vulnerabilities to gain higher levels of access.

Full Compromise: Achieving root or administrative access to the system.

Red Team vs. Blue Team Exercises

Description: These exercises simulate real-world attack and defence scenarios where one team (Red Team) acts as the attackers, and another team (Blue Team) acts as the defenders.

Key Elements

Red Team: Focuses on simulating sophisticated attack techniques to breach the Blue Team's defences.

Blue Team: Concentrates on monitoring, defending, and responding to the Red Team's attacks.

Purple Team: Sometimes included to facilitate collaboration and learning between Red and Blue Teams, enhancing overall security posture.

Mixed Format CTFs

Description: These CTFs combine elements from multiple formats to create a more comprehensive competition. Participants might solve Jeopardy-style challenges to gain tools or information useful in an Attack-Defence scenario.

Key Elements:

Variety of Challenges: Incorporating tasks from different categories to test a broad range of skills.

Integration of Skills: Combining offensive and defensive techniques in a single competition.

6. Common CTF Challenges

CTF challenges are designed to test a wide range of cybersecurity skills. Each category focuses on different aspects of security, providing participants with opportunities to learn and apply various techniques. Here's an expanded overview of the most common types of CTF challenges:

Web Exploitation

Description: Web exploitation challenges involve finding and exploiting vulnerabilities in web applications. These challenges are designed to test participants' understanding of web security principles.

Common Vulnerabilities:

SQL Injection: Exploiting flaws in SQL queries to access or manipulate the database.

Cross-Site Scripting (XSS): Injecting malicious scripts into web pages viewed by other users.

Cross-Site Request Forgery (CSRF): Forcing a user to execute unwanted actions on a web application in which they are authenticated.

File Inclusion: Including local or remote files on the server through vulnerabilities like Local File Inclusion (LFI) or Remote File Inclusion (RFI).

Command Injection: Executing arbitrary commands on the host system through vulnerable web applications.

Cryptography

Description: Cryptography challenges involve decrypting messages, cracking encryption algorithms, and solving puzzles related to cryptographic techniques

Common Tasks:

Cipher Decryption: Breaking classical ciphers (Caesar cipher, Vigenère cipher) or modern encryption algorithms.

Hash Cracking: Finding the original data from hashed values using techniques like brute force or rainbow tables.

Steganography: Hiding or uncovering information embedded within images, audio, or other files.

Digital Signatures: Understanding and exploiting weaknesses in digital signature algorithms.

RSA Attacks: Exploiting vulnerabilities in RSA encryption, such as small key sizes or poor implementations.

Forensics

Description: Forensics challenges require participants to analyse digital artifacts, such as memory dumps, disk images, or network traffic, to uncover hidden information.

Common Tasks:

File Analysis: Examining file metadata, extracting hidden data, and recovering deleted files.

Memory Analysis: Analysing memory dumps to extract useful information, such as passwords, encryption keys, or running processes.

Network Traffic Analysis: Interpreting data captured from network traffic to identify patterns, extract credentials, or reconstruct sessions.

Log Analysis: Reviewing and analysing log files to detect suspicious activities or reconstruct events.

Malware Analysis: Analysing malicious software to understand its behaviour and impact.

Reverse Engineering

Description: Reverse engineering challenges involve decompiling and analysing binaries to understand their functionality and find hidden features or vulnerabilities.

Common Tasks:

Disassembly: Converting binary code back into assembly code to understand program logic.

Decompilation: Transforming compiled binaries into higher-level source code.

Binary Analysis: Identifying functions, data structures, and control flow within a binary.

Patching: Modifying binaries to change their behaviour or bypass security checks.

Cracking: Removing software protections, such as license checks or encryption.

Binary Exploitation (Pwn)

Description: Binary exploitation challenges require participants to find and exploit vulnerabilities in compiled programs to achieve unintended behaviour or gain unauthorized access.

Common Vulnerabilities:

Buffer Overflow: Overwriting memory adjacent to a buffer to execute arbitrary code.

Format String Vulnerability: Exploiting improper use of format strings to read or write arbitrary memory.

Use-After-Free: Accessing memory after it has been freed, leading to undefined behaviour and potential code execution.

Return-Oriented Programming (ROP): Exploiting buffer overflows by chaining together small sequences of instructions already present in memory.

Heap Exploitation: Manipulating the heap memory allocator to corrupt data structures and execute arbitrary code.

Miscellaneous

Description: Miscellaneous challenges encompass a wide range of tasks that don't fit neatly into the other categories. These can include puzzles, trivia, and unique problem-solving scenarios.

Common Tasks:

Steganography: Hiding and uncovering information within images, audio, or other files.

OSINT (Open Source Intelligence): Gathering information from publicly available sources to solve challenges.

Programming: Writing scripts or programs to solve specific problems or automate tasks.

Trivia: Answering questions related to cybersecurity history, concepts, or notable figures.

Real-World Scenarios: Solving challenges based on realistic attack or defense scenarios.

Social Engineering

Description: Social engineering challenges test participants' ability to manipulate people into divulging confidential information or performing actions that compromise security

Common Tasks:

Phishing: Crafting convincing emails or messages to trick users into revealing credentials.

Impersonation: Pretending to be someone else to gain access to restricted areas or information.

Baiting: Leaving malicious devices, such as USB drives, in public places to tempt users into connecting them to their systems.

Programming

Description: Programming challenges require participants to write code to solve specific problems or automate tasks.

Common Tasks:

Script Writing: Developing scripts to automate repetitive tasks or solve challenges.

Algorithm Implementation: Implementing algorithms to solve complex problems efficiently.

Debugging: Identifying and fixing bugs in provided code.

Code Optimization: Improving the performance or security of existing code.

8. Continuous Learning and Resources

Ethical hacking is a constantly evolving field. Stay updated with these resources:

Books

"The Web Application Hacker's Handbook" by Dafydd Stuttard

"Hacking: The Art of Exploitation" by Jon Erickson

"Metasploit: The Penetration Tester's Guide" by David Kennedy

Online Courses

Hacktivity

TryHackMe

Hack The Box

Websites and Blogs

OWASP

Exploit Databases

Forums and Communities

Reddit (r/netsec, r/hacking)

Discord and Slack groups for cybersecurity enthusiasts

By following this guide, members of the UWE CTF Falcons will be well-equipped to practice ethical hacking responsibly and excel in CTF competitions. Remember, the goal is not just to find vulnerabilities but to learn and help improve the security landscape. Happy hacking!