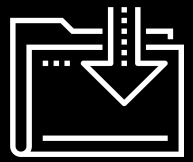
Post-Exploitation and Reporting

Cybersecurity Boot Camp

Lesson 16.4



Class Objectives

By the end of today's class, you will be able to:

Describe common tasks included in privilege post-exploitation.

Perform post-exploitation tasks, such as gathering password hashes.

Explain how password crackers work and perform password cracking.

Understand the importance of reporting and fill out a strong report.


Day 3
Recap

Day 3 Recap

Command and control (C2) is a framework that consists of tools and techniques that attackers use to maintain communication with compromised devices following initial exploitation.

The C2 architecture consists of:

C2 server

The attacker's server, where the attacker communicates with the compromised machines.

C2's agents

The payload that is run on the compromised machine in order to open up a connection back to the C2 server.

Day 3 Recap

Metasploit is a popular C2 framework that contains a suite of tools for enumerating and exploiting servers and other networked devices.

The main tools of Metasploit are:

01

MSFconsole:
The main interface for Metasploit, which runs on your local machine.

02

Meterpreter:
A Linux-style shell that runs on the machines you compromise.

Post Exploitation is the next phase after Exploitation and involves any actions performed after exploitation.

Reporting

Post Exploitation

Exploitation

Scanning

Planning and Reconnaissance

Post Exploitation

Post Exploitation includes three common tasks:

Enumeration and searching for useful data:	**Persistence:**	**Privilege Escalation:**
Similar to the enumeration process during the Reconnaissance phase, except we are enumerating and searching for data from inside the target after it has been exploited.	The method of attempting to maintain long-term access into your target.	The method of escalating privileges from a less-privileged user to a more-privileged user (or a low-privilege to high-privilege user).

Re-performing Enumeration After Privilege Escalation

Re-performing Enumeration After Privilege Escalation

We finished the previous lesson by performing the post-exploitation task of enumeration and utilizing that data to conduct privilege escalation.

Low-privileged

Means having limited access to the machine. In Linux, this typically means not being able to browse through configuration files or other home folders.

High-privileged

Means having a high amount of access. In the case of root, privileges are unrestricted.

We can now re-perform enumeration as a high-privilege user to see if there are any other files that may be useful.

High-Privilege Escalation

On Linux, there are several files that would now be accessible and could offer additional information to an attacker, such as:

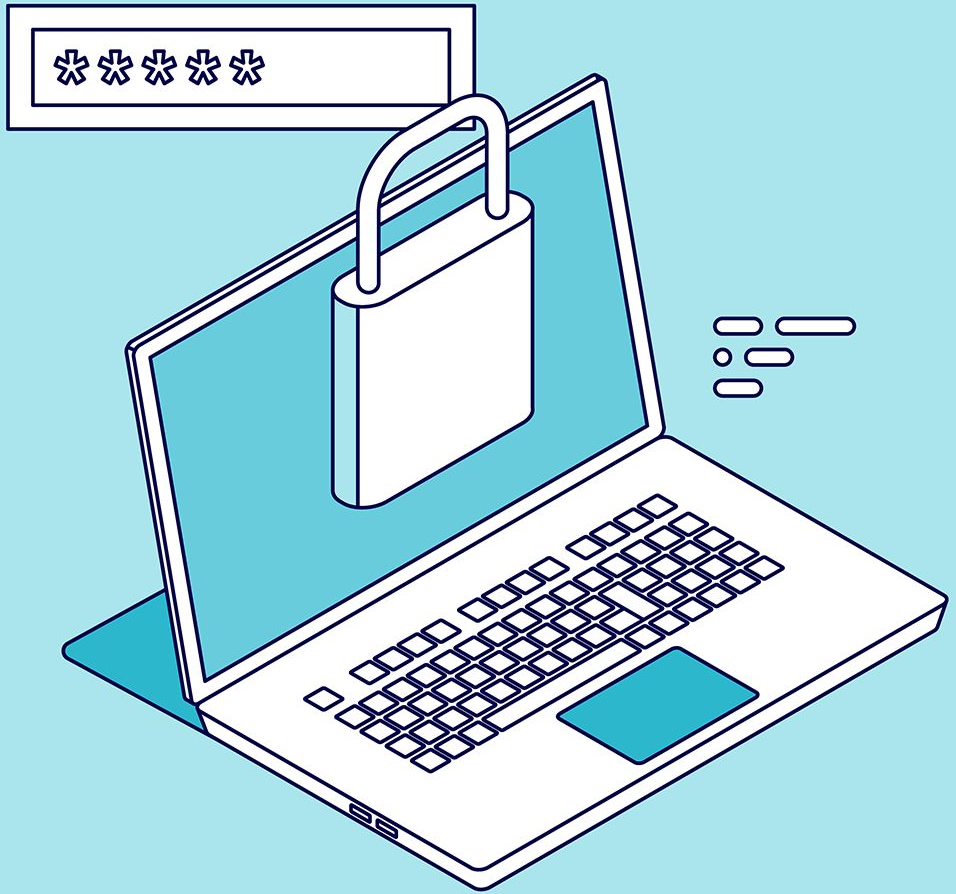
`/var/lib/mysql/mysql/user.MYD`	Is the MySQL database file, which may contain SQL login credentials.
`/home/[USER]/.bash_history`	Contains a history of a user's bash commands.
`/home/[USER]/.ssh/`	Contains private SSH keys.

Can you think of any other file in Linux that holds extremely sensitive information and would be fruitful for an attack?

High-Privilege Escalation

The **/etc/shadow** file is one of the most sensitive files on Linux, because it houses the password hashes for all users on the machine.

This file is typically limited to root-level access.



High-Priv Escalation

While the **/etc/shadow** file appears nearly identical to **/etc/passwd**, there is one **/etc/passwd:**ce.

```
timmy:x:1016:1019::/home/timmy:/bin/sh
```

/etc/shadow:

```
timmy:$6$6Y/fI1nx$zQJj6AH9asTNfhxV7NoVgxByJyE.rVKK6tKXiOGNCfWBsrTGY7wtC6Cep6co9eVNkRFrpK6koXs1NU3AZQF8v/
```

High-Privilege Escalation

Instead of an `x`, like in the **passwd** file, the second field in the **shadow** file (after the colon) contains a long, complex string:

```
$6$6Y/fI1nx$zQJj6AH9asTNfhxV7NoVgxByJyE.rVKK6tKXiOGNCfWBsrTGY7wtC6Cep6co9eVNkRFrpK6koXs1NU3AZQF8v/
```

- The string is a password hash.
- This hash cannot be used to log in, but can be cracked to retrieve the user's original password.
- If a row in **`/etc/shadow`** contains an **`!`** instead of a password hash, it means the account is locked.
- If a row in **`/etc/shadow`** contains an **`*`** instead of a password hash, it means the user is not allowed to log in.

High-Privilege Escalation

Now that we have sudo or root access, we're able to read this file and obtain the hashes, which we can crack offline.

In the next demonstration, we will continue our post-exploitation activities by revisiting the tool John the Ripper to crack the hashes from the shadow file.

https://www.openwall.com/john/

Instructor Demonstration

John the Ripper Refresher

Activity: Password Cracking

In this activity, you will use John the Ripper to conduct post-exploitation password cracking.

Suggested Time:

15 Mins

Time’s Up! Let’s Review.

Persistence

Now that we have completed the post-exploitation task of enumerating data after privilege escalation, we'll move on to another post-exploitation task: establishing persistence.

Per MITRE:

Persistence consists of techniques that adversaries use to keep access to systems across restarts, changed credentials, and other interruptions that could cut off their access.

Techniques used for persistence include any access, action, or configuration changes that let them maintain their foothold on systems, such as replacing or hijacking legitimate code or adding startup code.

Persistence

Persistence refers to an adversary maintaining their access to the target.
But there are several techniques that can be used to maintain persistence:

Technique 1
Utilizing Existing Services

Technique 2
Blending into the Environment

Technique 3
Maintaining Current Level of Access

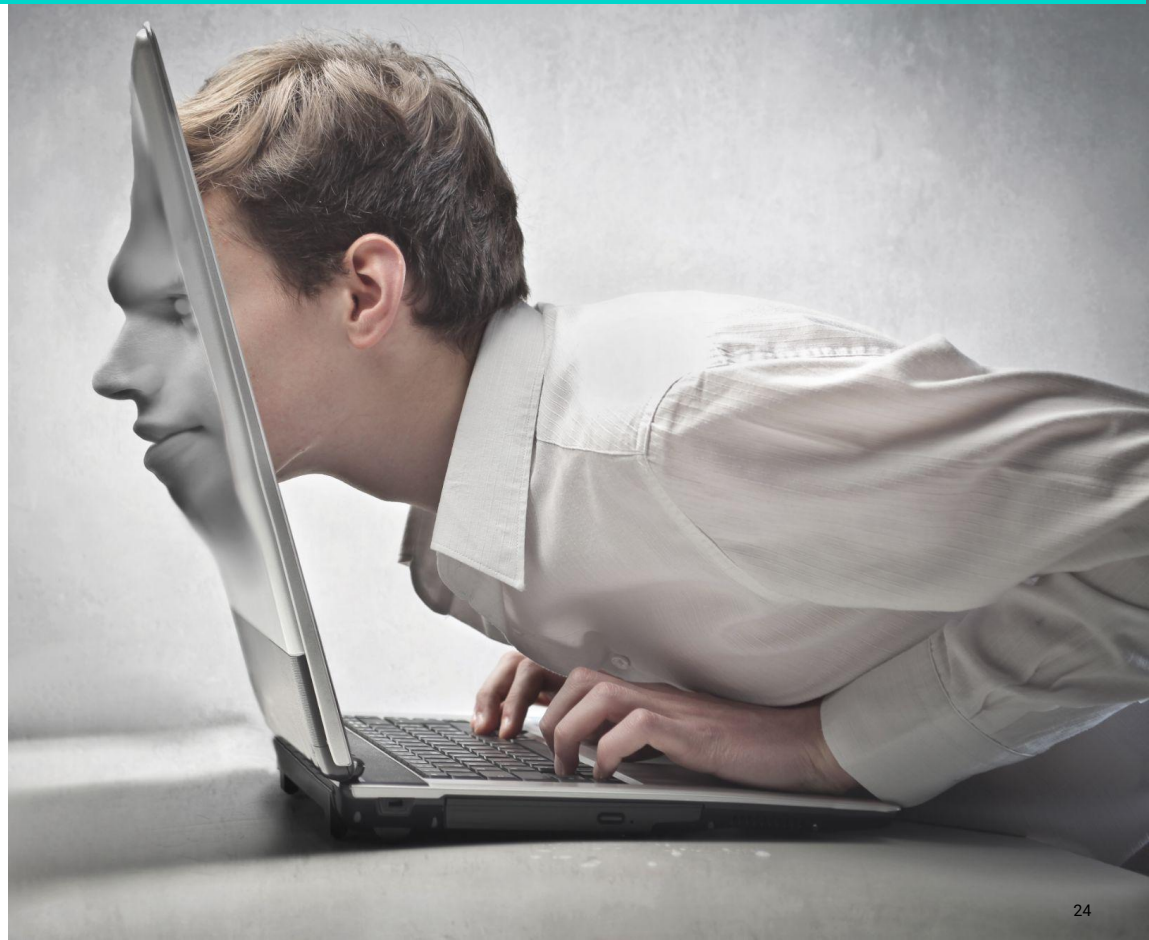
Technique 4
Privesc Persistence

Technique #1: Utilizing Existing Services

Persistence is less about exploitation of vulnerable software and services and more about utilizing existing system services.

For example:

Utilizing `useradd` to create a new user as a “backdoor.”



How else might you use existing system services or functions for persistence?

Technique #2: Blending into the Environment

An important part of persistence is making sure that the attacker's work blends in with the current environment and can't be detected.

Knowledge of the system you're on is crucial for this.

For example:

If creating a backdoor account, you want to blend in with current account names, instead of naming the account backdoor.

(E.g., Use sysadmin to pose as a system admin or systemd-ssh to pose as a service.)

Technique #3: Maintaining Current Level of Access

Another key aspect of persistence is maintaining your current level of access.

- You could, and typically should, establish persistence as a low-privilege user before privilege escalation, just in case you lose access to the machine.
- You should then re-establish persistence as an elevated user.
- It’s important to establish persistence and ensure persistence with the highest privileges possible, to avoid having to re-step through the privilege-escalation phase.

Technique #4: Privesc Persistence

Privesc persistence means purposely establishing opportunities for low-privilege users to escalate their privileges.

For example, the SUID bit technique:

```
https://attack.mitre.org/techniques/T1548/001/
```

- On Linux or macOS, when the setuid or setgid bits are set for an application, the application will run with the privileges of the owning user or group.
- Normally, an application is run in the current user's context, regardless of which user or group owns the application. However, there are instances where programs need to be run in an elevated context to function properly, but the user running them doesn't need the elevated privileges.
- As an attacker with root privileges, you could establish a privesc persistence script by creating a reverse shell in a Python script and setting the SUID bit to root.

Summary

In the Post Exploitation stage, after privilege escalation has been accomplished, we can now re-perform enumeration.

- You could, and typically should, establish persistence as a low-privilege user before privilege escalation, just in case you lose access to the machine.
- This is because we now have access to higher-privileged files, such as the shadow file.

Persistence is another post-exploitation method that an attacker can utilize to maintain access to their target. Techniques that can be used for persistence include:

- Utilizing existing services
- Maintaining current level of access
- Blending into the environment
- Privesc persistence

Activity: Persistence

In this activity, you will create a different form of persistence by configuring the SSH configuration file to log back in at a later time undetected.

Suggested Time:

15 Mins

Time’s Up! Let’s Review.


Break

Reporting

While we have completed all the technical steps of our penetration test, one important step remains.

The Five Stages of Pentesting

01A	Planning	Define the purpose and scope of the test, and sign all legal contracts.
01B	Reconnaissance	Obtain publicly available information about your target.
02	Scanning	Use tools to run a scan against your target to gather information, such as open ports, and run services to determine potential vulnerabilities.
03	Exploitation	Attack the vulnerabilities discovered in the previous steps in order to gain access to the target.
04	Post Exploitation	Gather valuable information from the compromised systems.
05	Reporting	Report on the previous five steps to provide a summary of actions taken, findings, and recommended mitigations.

Reporting is important because it's where the pentester describes their actions in words, which will eventually be communicated to stakeholders and leadership who will interpret the results.

Reporting

Post Exploitation

Exploitation

Scanning

Planning and Reconnaissance

Phase 5: Reporting

Communication is the key component in linking actions to results.

To the client, the quality of the report reflects the quality of the pentester's work.

A poorly written report could make the client believe the pentester is incompetent and the pentest was a failure, even if the pentester had several important findings.

A well-written report will help establish the pentester's reputation and perhaps even generate repeat business, while also taking a positive step toward remediating any findings within the client's network.

Report-Writing Best Practices

01

The report should be well balanced between technical and non-technical results.

02

The findings should be understandable by management, who are usually less technical.
But there should also be a section that gets more technical, so that system administrators can read the report and develop remediation strategies.

03

It's extremely important to document and log all actions taken, even if they were unsuccessful.

- Sometimes pentester attacks trigger alerts, and the pentester needs to be able to prove with timestamps that it was their actions that caused the alert to fire and not an actual adversary.

04

The report should also contain any notable failed actions that the pentester attempted.

- For example, if anti-virus software successfully stopped a payload from executing, that should be documented in the report. It shows that the current defensive stack for the customer is effective, which is a form of positive reinforcement.

Instructor Demonstration

Report Walkthrough

Reporting Activity

In this activity, you will begin writing a report that details the findings you've gathered throughout this week.

Suggested Time:

0:35

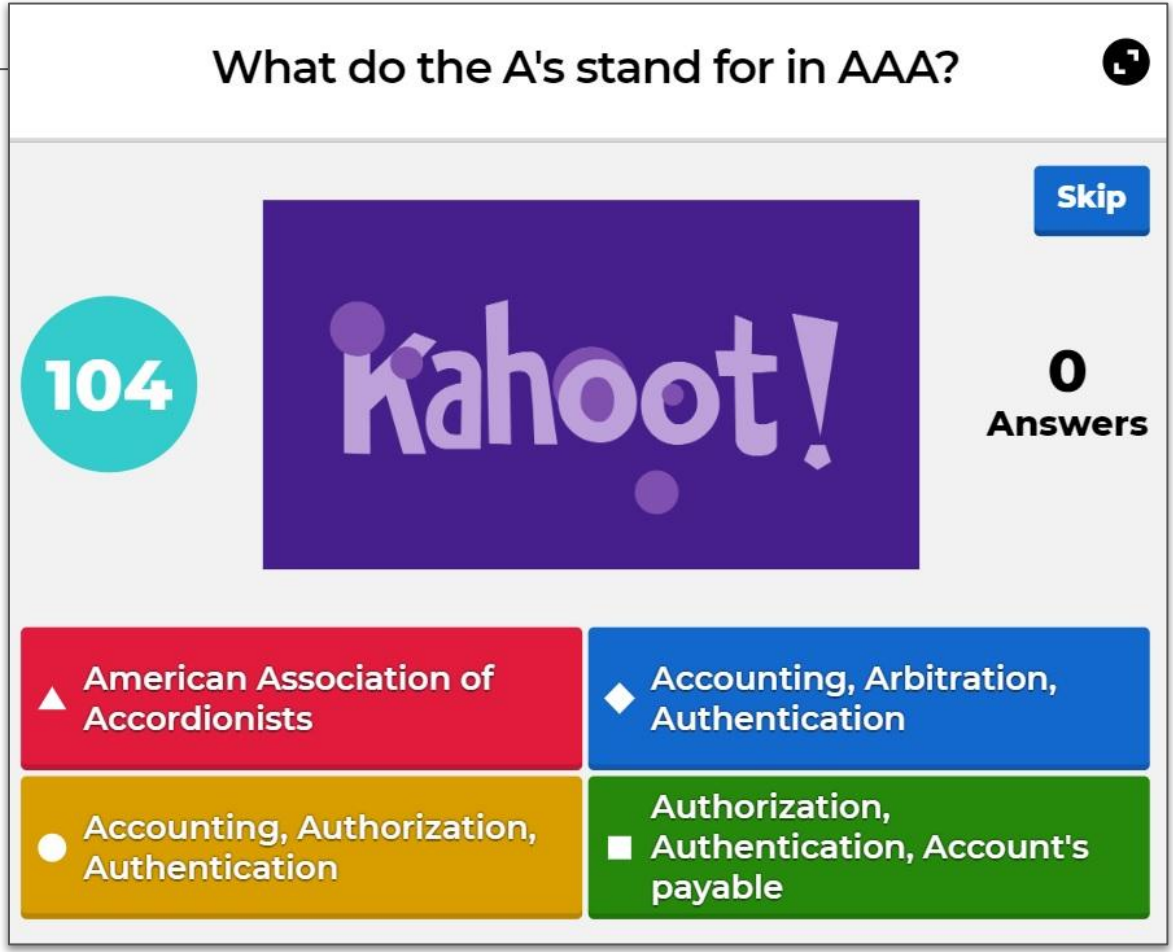
Week 1 Review

Now, we will conduct a review competition using:

What's Kahoot?

Kahoot is a web-based tool that:

- Displays questions and answers to select from in real time.
- Keeps track of individual and team scores.
- Keeps track of remaining time for each question.



(kahoot.com)

Kahoot! Rules and Guidelines

- There are a total of 29 Pentesting questions.
- Points are not deducted for incorrect answers.
- You will have two minutes to answer each question.
- If you are competing as a team, select a team captain to answer the questions.
- Points are awarded for correct answers and for how quickly you answer the questions compared to your classmates.
- **Note:** If your class is currently online, it will be easier if each student competes individually.

Kahoot! Rules and Guidelines

- The questions will come from the previous Pentesting Classes
- You can use all available resources: books, the internet, class notes, etc.
- Any issues will be decided by the judges (the TAs and/or instructor), e.g.:
  - Answer disputes
  - Frozen or lagging computers
  - Kahoot issues
- The team or individual with the most points at the end of 30 questions is the winner!

(kahoot.com)

Time’s Up! Let’s Review.


The End