Hacktales

# SHELLS & PAYLOADS

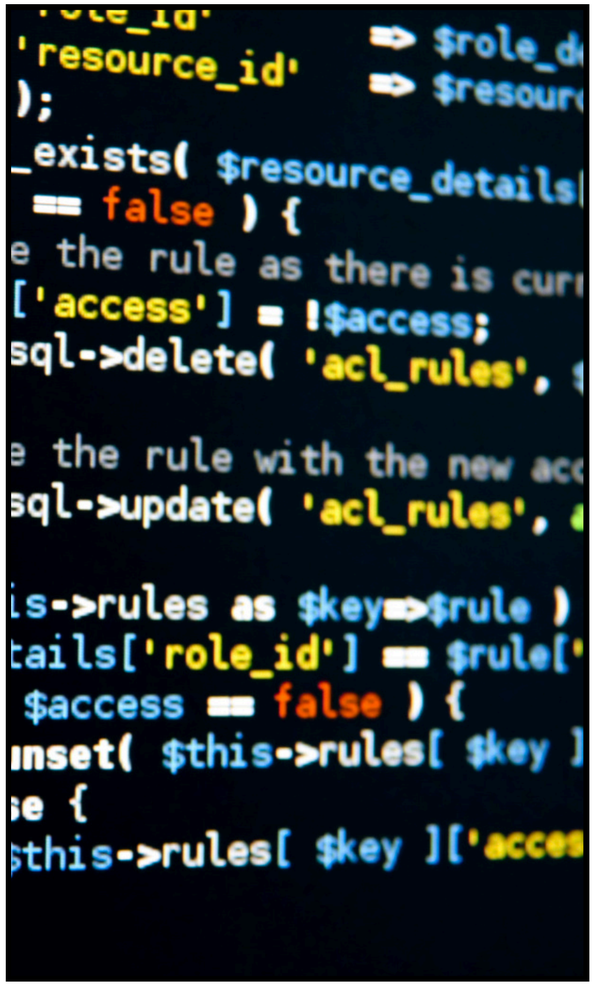
RED TEAM: WEEK 6

# SHELLS

SHELLS & PAYLOADS

**WHAT IS A SHELL?**

A shell is a program that provides a computer user with an interface to input instructions into the system and view text output (Bash, Zsh, cmd, and PowerShell, for example) A shell is received when an attacker forces a server to execute arbitrary code that sends a reverse connection providing a CLI session to the attacker OR opens up a port on the target server through which the attacker can connect to and establish a CLI session on the target. The above phrases translate to establishing a remote CLI session on a target computer, this is also known as Remote Code Execution (RCE)

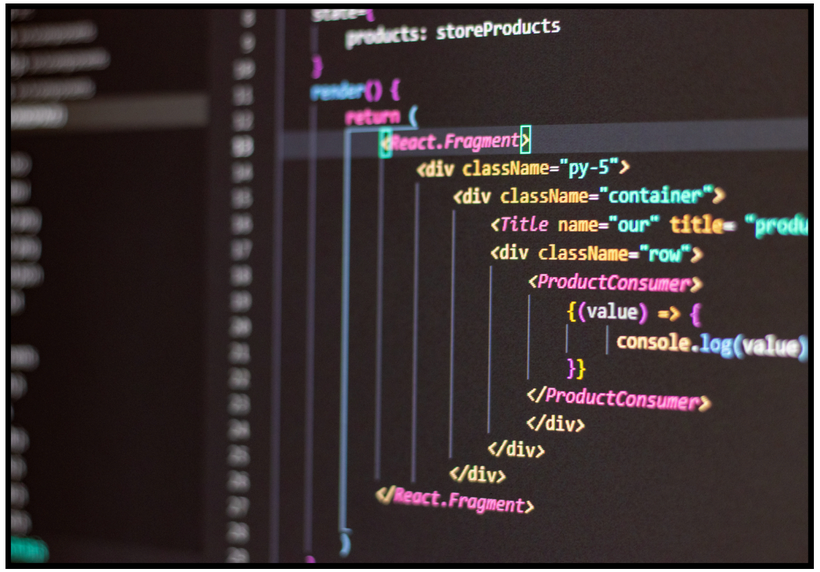


# SHELLS

Overview

Every operating system has a shell, and to interact with it, we must use an application known as a terminal emulator. The shell gives us direct access to the OS, system commands, and file system.

Pentesters primarily use a network utility called “Netcat” for interacting, and receiving shells that provide us a CLI connection to the target. However a shell can be unstable and these shells do not have a TTY (a teletype), stable shells have a TTY or PTY(pseudoteletype)



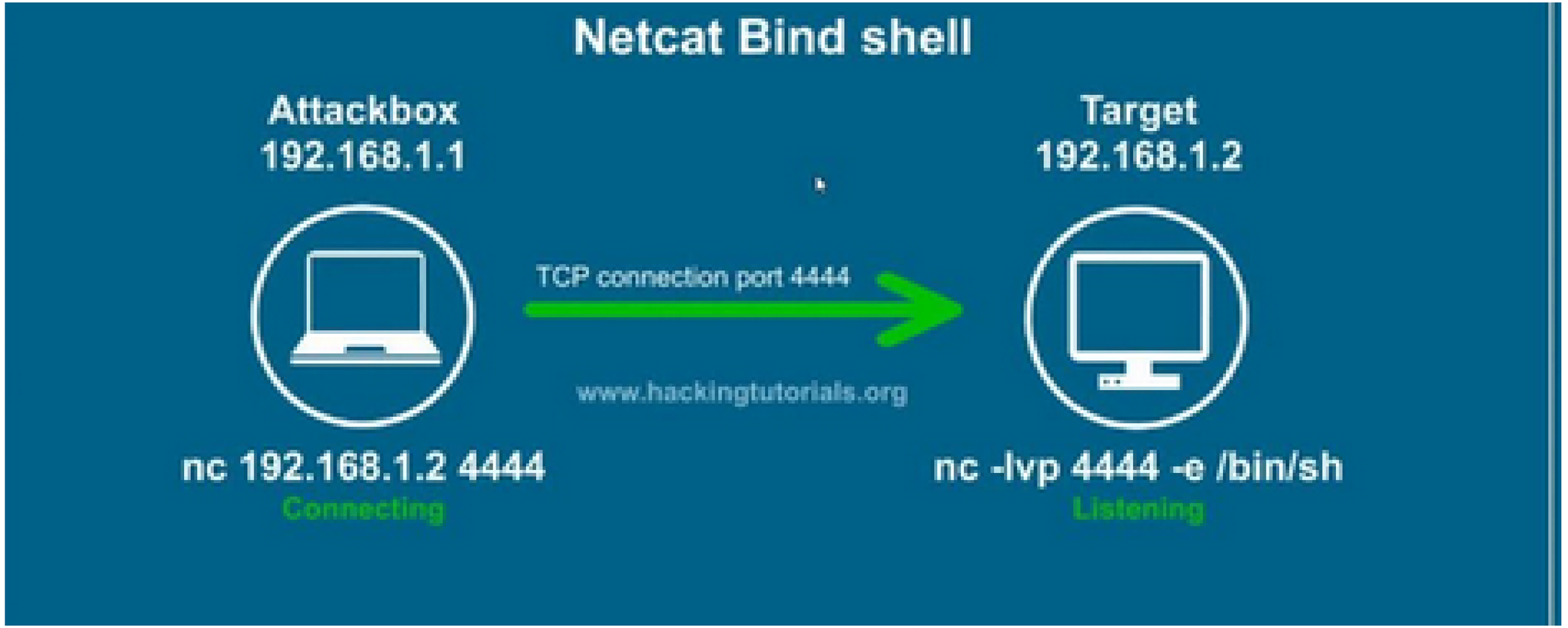
# TYPES OF SHELLS

BIND SHELLS

## REVERSE VS BIND SHELLS

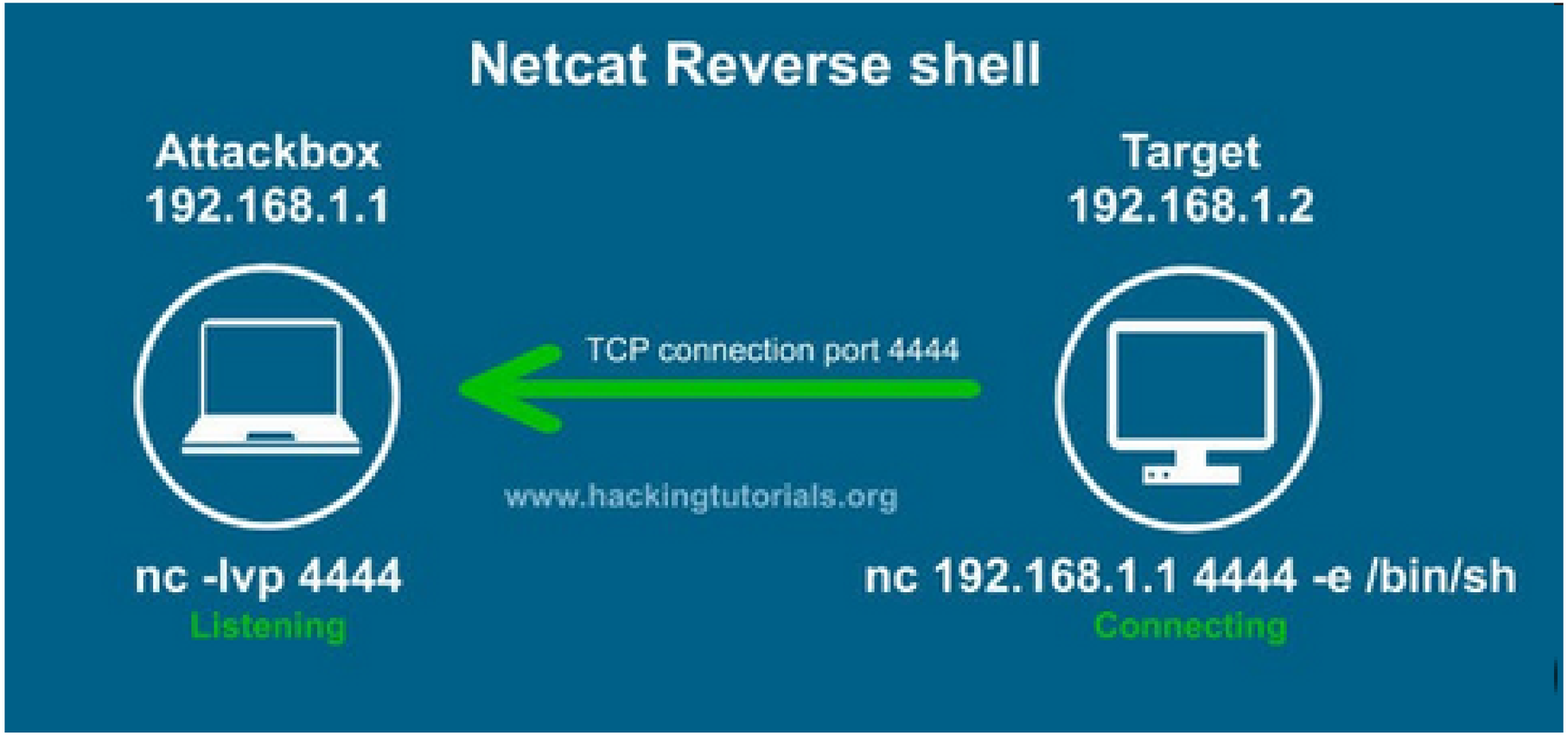
In many cases, we will be working to establish a shell on a system on a local or remote network. This means we will be looking to use the terminal emulator application on our local attack box to control the remote system through its shell. This is typically done by using a Bind or Reverse shell.

A bind shell is gained by an attacker when he is able to force a vulnerable server to execute arbitrary code that opens up a port for an attacker to connect to, to establish a CLI session. With a bind shell, the target system has a listener started and awaits a connection from a pentester's system (attack box).




**Bind Shell Connection**

A bind shell is gained when you force a server to execute code that opens up a local port through which we can connect to (with nc) and establish a CLI session on




**Reverse shell connection**

# Demo: Reverse & Bind Shells

# EXPLOITS & PAYLOADS

EXPLOITS

## WHAT IS AN EXPLOIT

An Exploit is a piece of code, software, or a sequence of commands that takes advantage of a vulnerability, bug, or misconfiguration in a system, application, or network to cause unintended or malicious behavior. It is a technique through which the payload is delivered on the vulnerable target.

## TYPES OF EXPLOITS

Local Exploits

- These require prior access to the target system (either physical or via a user session).
- The attacker is already on the system in some capacity — maybe as a low-privileged user.
- Goal: Privilege escalation, lateral movement, or access to restricted data.
- Example: Exploiting SUID binaries in Linux, or kernel privilege escalation.

# Remote Exploits

- These are launched from a separate system over a network.
- The attacker doesn’t need any prior access to the target.
- Goal: Initial access, code execution, or DoS.
- Example: Exploiting a vulnerable web app (e.g., RCE via a PHP vulnerability), or a buffer overflow in a network service.

# EXPLOITS & PAYLOADS

PAYLOADS

## WHAT IS A PAYLOAD

A payload is code crafted with the intent to exploit a vulnerability on a computer system. The term payload can describe various types of malware, including but not limited to ransomware. It is the significant part of an exploit that exploits a vulnerability to send us a remote CLI session.

# EXPLOITS & PAYLOADS

PAYLOADS

## STAGED PAYLOAD

A payload delivered in multiple stage, usually a small initial payload (called a stager) is sent first, which then downloads and executes a larger payload (called the stage).

### How it works:

- Exploit sends a small stager (e.g., reverse_tcp).
- The stager connects back to the attacker's machine.
- The attacker sends the full payload (e.g., Meterpreter).
- The payload is loaded into memory and executed.

Pros:

- Smaller initial payload size (good for fitting into buffer limits).
- More flexible and modular.
- Harder to detect all at once.

Cons:

- Relies on a stable connection for second-stage delivery.
- Can fail if firewall/AV blocks outbound connection.

# EXPLOITS & PAYLOADS

PAYLOADS

## UNSTAGED PAYLOAD

A single, complete payload sent all at once. No separate stager or external download.

### How it works:

- Exploit sends the full payload directly.
- It gets executed immediately without needing a callback for more data.

Pros:

- More stable (doesn’t depend on a second connection).
- Easier to debug or analyze.
- Doesn’t leave a stager hanging around in memory.

Cons:

- Larger size (harder to inject in some exploits).
- Less stealthy—everything’s exposed in one go.

# TYPES OF PAYLOADS

| Non-staged | Staged |
| --- | --- |
| Sends exploit shellcode all at once<br>Larger in size and won't always work<br>Example: windows/meterpreter_reverse_tcp | Sends payload in stages<br>Can be less stable<br>Example: windows/meterpreter/reverse_tcp |

# DEMO: INTRO TO METASPLOIT & EXPLOITING ETERNAL BLUE


Q
&
A