Sig•Mal

Meeting 2

Howdy !

Let’s do a quick Intro

- Name \ Nickname
- Major
- Year
- Interests / Fun Facts about you

Shell

Goal for the Attacker:

Compromise System → Gain Shell Access → Elevate privileges → $$ Clout $$

Gain Shell Access

Most systems will have a firewall so direct shell connections are impossible

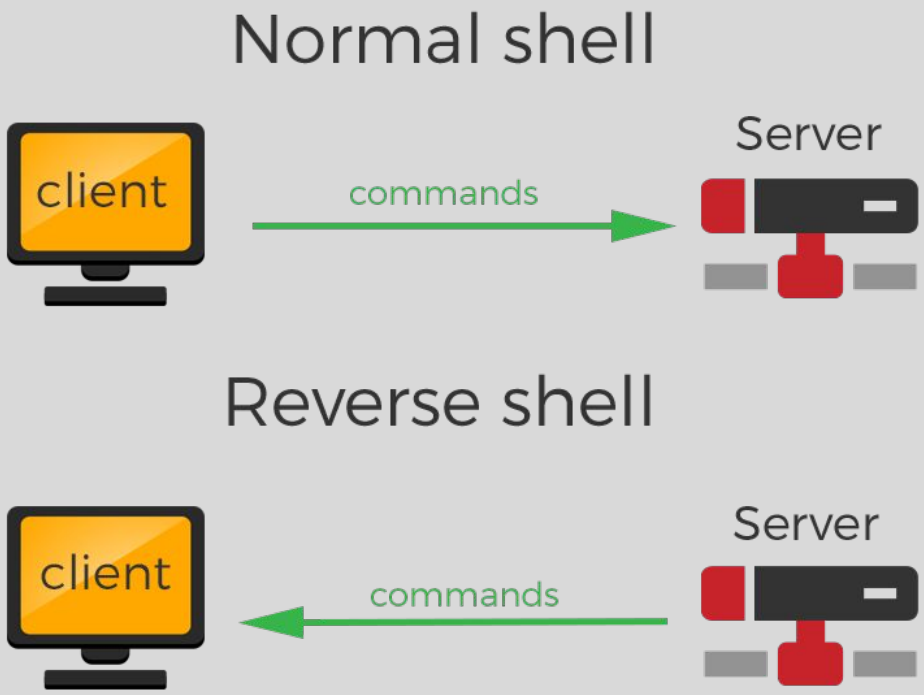
→ Solution: Reverse Shell

Reverse Shell

In a remote system access, the user is the client and the target machine is the server.

Reverse Shell, is basically that but in reverse.
Client is the target machine and the Server is the user.

Most malicious reverse shells work on port 80 + 443


Normal shell
client
commands
Server
Reverse shell
client
commands
Server

Reverse Shell

Why Reverse Shells?

→ Normally to combat the default configuration of firewalls

Ex. A server will only accept connections on certain ports. A web server, might only have 2 ports open. Port 80 + 443 (http and https, respectively)

→ On the other hand, firewalls might not limit outgoing traffic.

∴ Attacker establish server on their own machine +
And create a reverse connection.

Hacker voice I'm in

LIVE DEMO TIME

(Simple)

Hypothetical Hacks

CRON JOB EXPLOITATION
(Cron Privilege Escalation)

Disclaimer: Savage Way to get in but isn’t a novel method

I don’t full know if they fixed/patched it so this is purely hypothetical

CRON JOBS

What is a cron job?

Cron allows system administrators to automate/schedule tasks with ease and precision.

Why it works?

Cron runs commands as root.

What it looks like (typically)?

\/ \/ \/


minute (0-59)
hour (0-23)
day of month (1-31)
month (1-12)
day of week (0-6) (Sunday is 0)
* * * * *
<command to execute>

https://www.armourinfosec.com/linux-privilege-escalation-by-exploiting-cronjobs/