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:
    - Fire 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

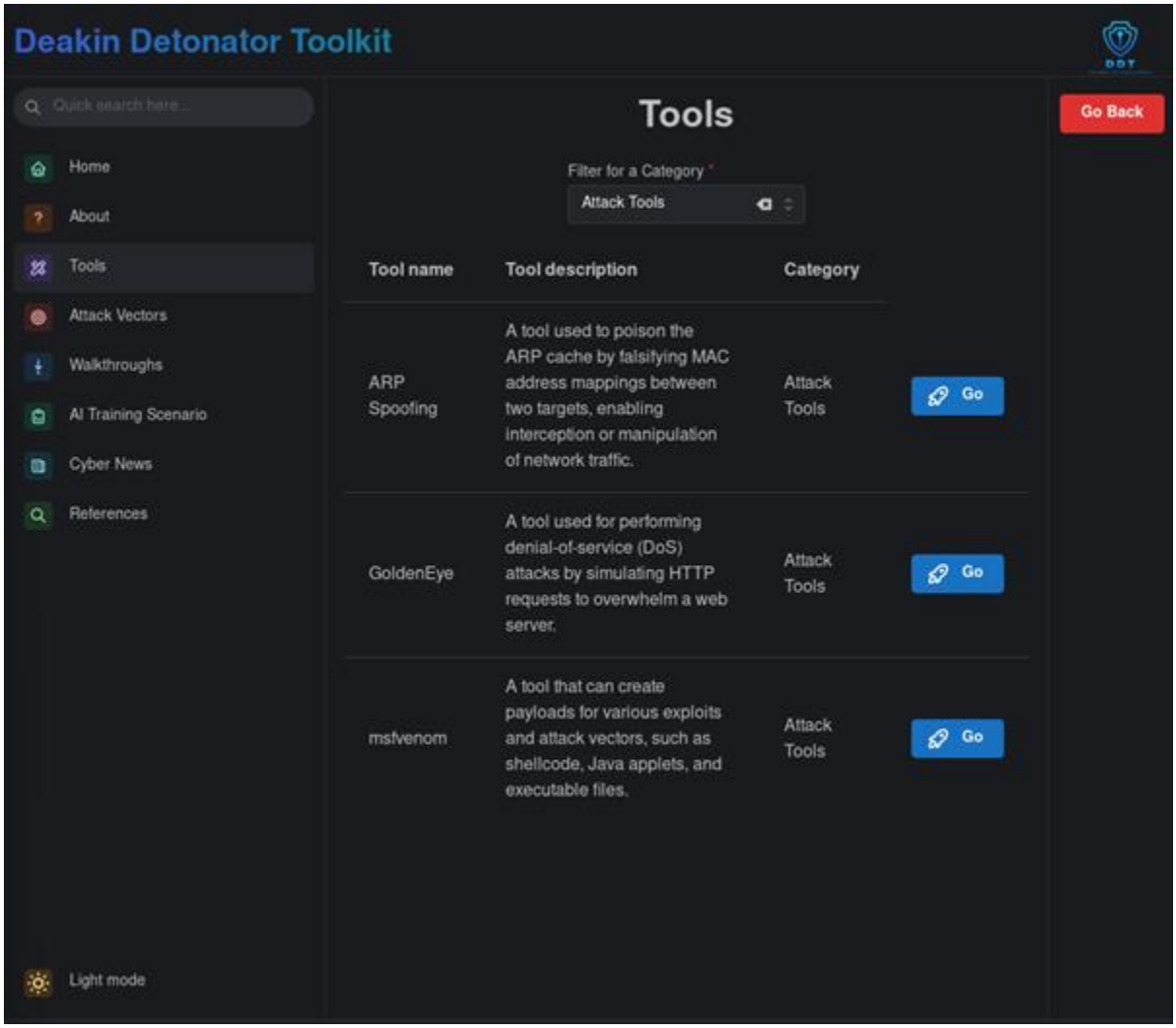
Options shown:

- 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

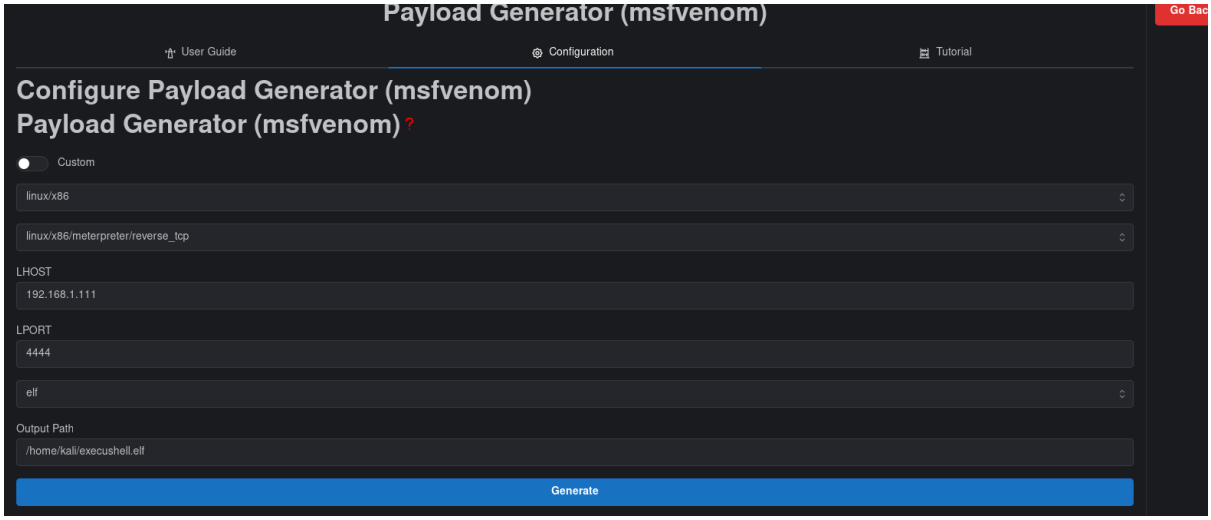
This controls how the payload is packaged.

The How:

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



Step 2: Enter your respective details:



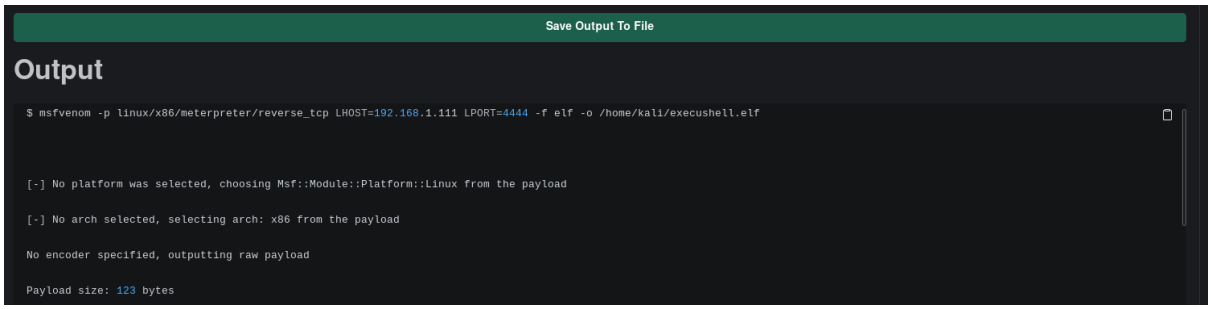
Linux/x86 was selected as I am targeting my Ubuntu server.

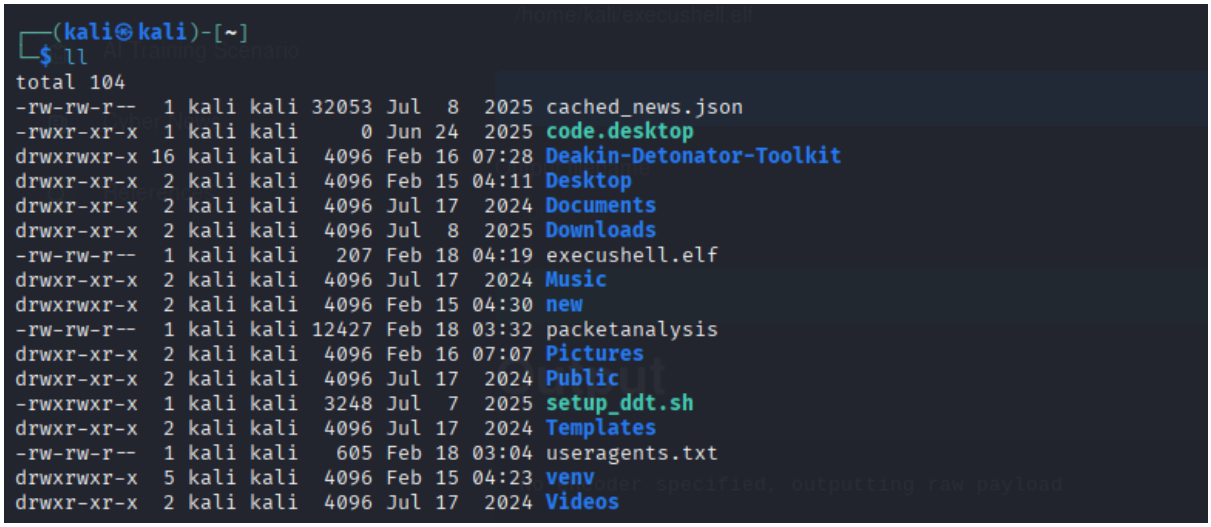
Linux/x86/meterpreter/reverse_tcp was selected to create a Linux-compatible reverse TCP payload, allowing the target Ubuntu server to initiate an outbound callback connection to the specified LHOST and LPORT, thereby establishing a remote session for analysis within the controlled virtual lab.

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!





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.