Gophish

7/7/25 12:36 AM

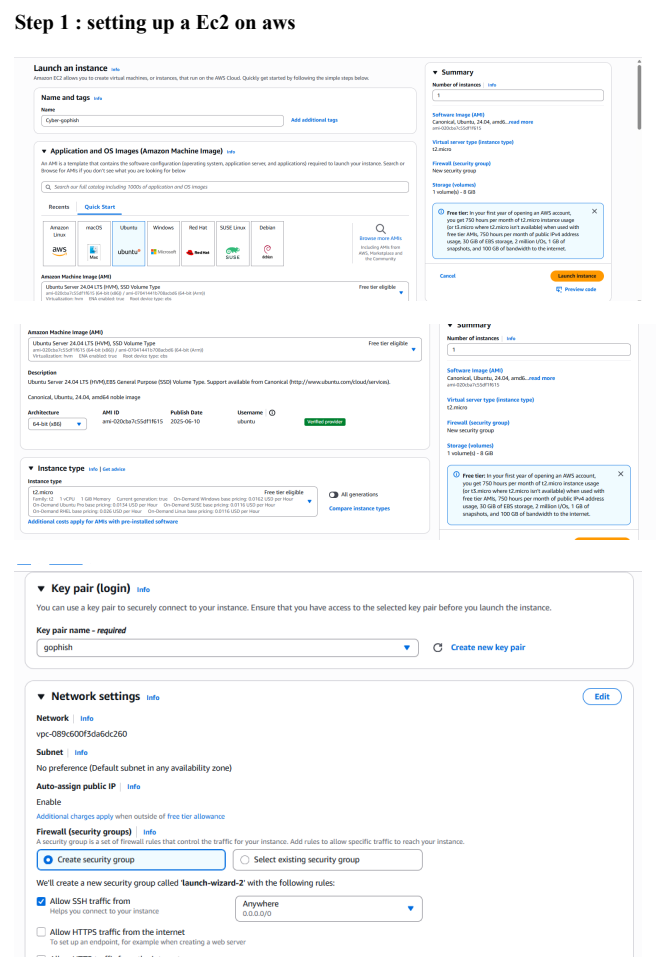
GoPhish is an open-source **phishing simulation tool** that lets you create, manage, and track phishing campaigns — mainly used for **security awareness training** and **red teaming**.

Key Features of GoPhish

- Create realistic phishing emails
- Build fake login pages (landing pages)
- Track who opened, clicked, or submitted data
- Helps organizations test user vulnerability
- Web-based dashboard for full control

Setting up gophish on aws :

Step 1 : setting up a Ec2 on aws

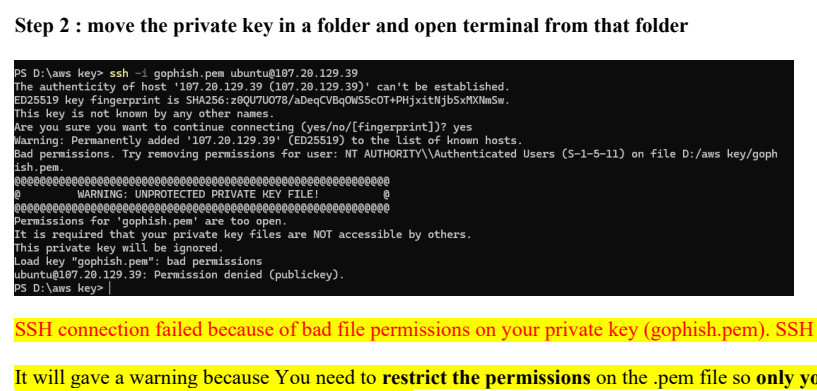


Key pair name - required: gophish

Then launch the instance

Don't click on connect we are connecting it using ssh

Step 2 : move the private key in a folder and open terminal from that folder



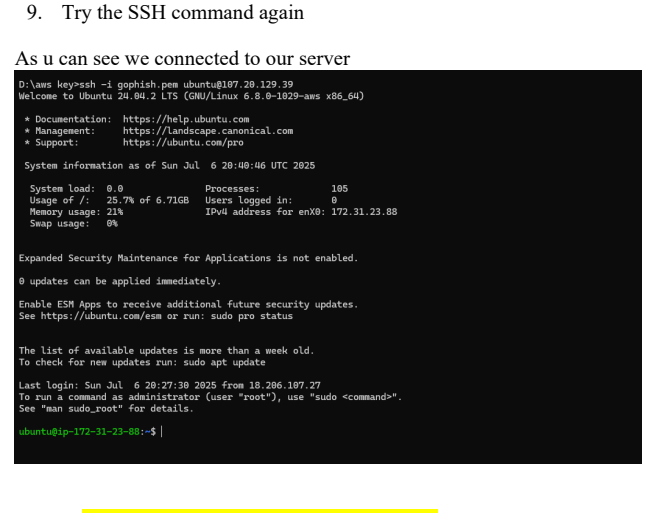
```
PS D:\aws key> ssh -i gophish.pem ubuntu@107.20.129.39
The authenticity of host '107.20.129.39 (107.20.129.39)' can't be established.
ED25519 key fingerprint is SHA256:z0QU7UO78/aDeqCVBqOWS5cOT+PHjxitNjbSxMXNmSw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '107.20.129.39' (ED25519) to the list of known hosts.
Bad permissions. Try removing permissions for user: NT AUTHORITY\\Authenticated Users (S-1-5-11) on file D:/aws key/goph
ish.pem.
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@         WARNING: UNPROTECTED PRIVATE KEY FILE!          @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Permissions for 'gophish.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "gophish.pem": bad permissions
ubuntu@107.20.129.39: Permission denied (publickey).
PS D:\aws key>
```

SSH connection failed because of bad file permissions on your private key (gophish.pem). SSH is rejecting the key for security reasons.

It will gave a warning because You need to **restrict the permissions** on the .pem file so **only your user can read it**.
Since you're using **Windows**, do the following:

1. Right-click gophish.pem → **Properties**.
2. Go to the **Security** tab.
3. Click **Advanced**.
4. Click **Disable inheritance**.
5. Remove all users except your own account.
6. Click on add and select principles and enter the logged in user name
7. Ensure **only you** have Read access.
8. Apply and close everything.
9. Try the SSH command again

As u can see we connected to our server



```
D:\aws key>ssh -i gophish.pem ubuntu@107.20.129.39
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Sun Jul  6 20:40:46 UTC 2025

  System load:  0.0               Processes:             105
  Usage of /:   25.7% of 6.71GB   Users logged in:       0
  Memory usage: 21%               IPv4 address for enX0: 172.31.23.88
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Sun Jul  6 20:27:30 2025 from 18.206.107.27
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-23-88:~$
```

Step 3 : sudo apt update & sudo apt upgrade -y

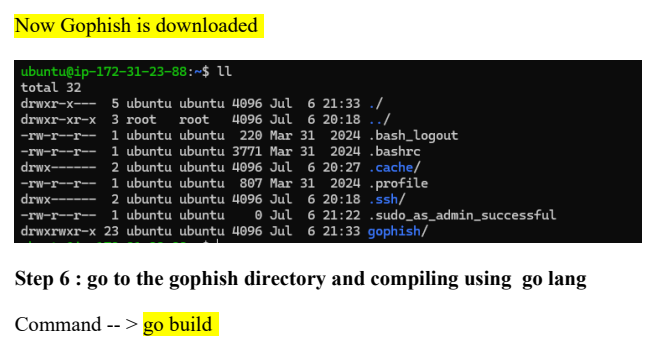
Step 4 : install golang.. Golang is a programming language.. Gophish is developed in golang

Command -- > sudo apt install golang-go -y

Step 5 : cloning GitHub repository

Command -- > git clone https://github.com/gophish/gophish.git

Now Gophish is downloaded

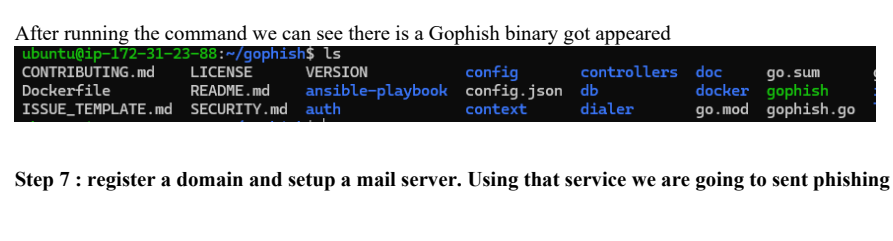


```
ubuntu@ip-172-31-23-88:~$ ll
total 32
drwxr-x---  5 ubuntu ubuntu 4096 Jul  6 21:33 ./
drwxr-xr-x  3 root   root   4096 Jul  6 20:18 ../
-rw-r--r--  1 ubuntu ubuntu  220 Mar 31  2024 .bash_logout
-rw-r--r--  1 ubuntu ubuntu 3771 Mar 31  2024 .bashrc
drwx------  2 ubuntu ubuntu 4096 Jul  6 20:27 .cache/
-rw-r--r--  1 ubuntu ubuntu  807 Mar 31  2024 .profile
drwx------  2 ubuntu ubuntu 4096 Jul  6 20:18 .ssh/
-rw-r--r--  1 ubuntu ubuntu    0 Jul  6 21:22 .sudo_as_admin_successful
drwxrwxr-x 23 ubuntu ubuntu 4096 Jul  6 21:33 gophish/
```

Step 6 : go to the gophish directory and compiling using go lang

Command -- > go build

After running the command we can see there is a Gophish binary got appeared

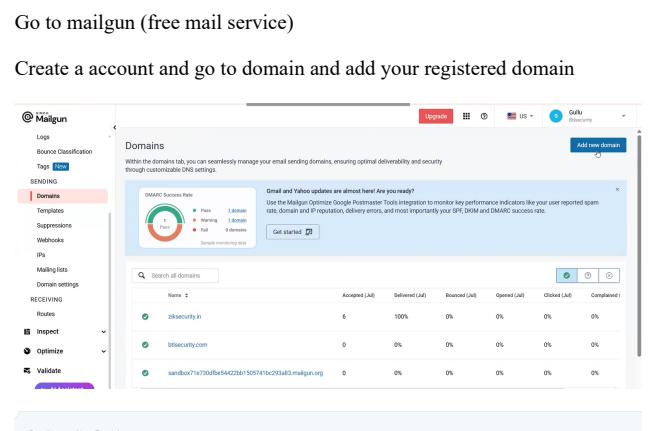


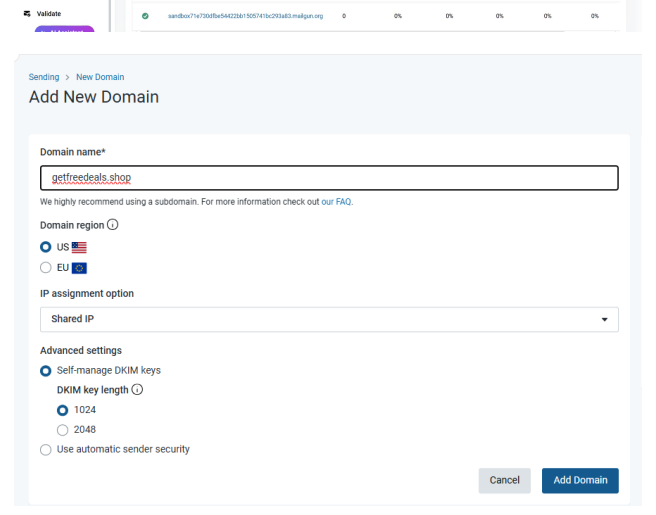
```
ubuntu@ip-172-31-23-88:~/gophish$ ls
CONTRIBUTING.md    LICENSE      VERSION           config       controllers  doc     go.sum
Dockerfile         README.md    ansible-playbook  config.json  db           docker  gophish
ISSUE_TEMPLATE.md  SECURITY.md  auth              context      dialer       go.mod  gophish.go
```

Step 7 : register a domain and setup a mail server. Using that service we are going to sent phishing email

Go to mailgun (free mail service)

Create a account and go to domain and add your registered domain





Add New Domain

Domain name*: getfreedeals.shop

Domain region: ● US ○ EU

IP assignment option: Shared IP

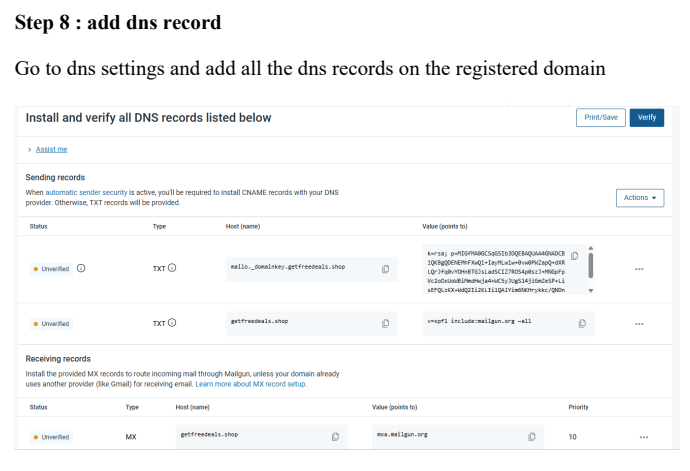
Advanced settings: ● Self-manage DKIM keys

DKIM key length: ● 1024 ○ 2048

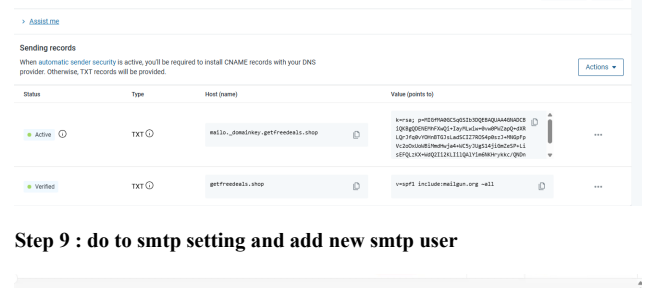
☐ Use automatic sender security

Step 8 : add dns record

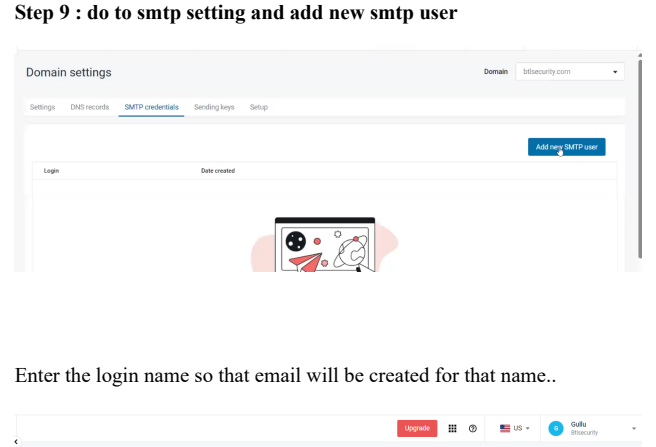
Go to dns settings and add all the dns records on the registered domain



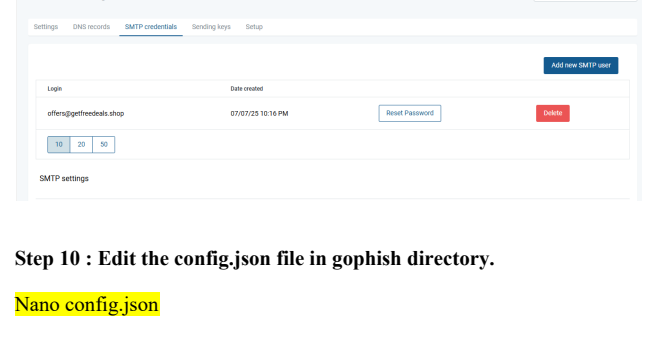
And verify the status.. As u can see that the records are verified



Step 9 : do to smtp setting and add new smtp user

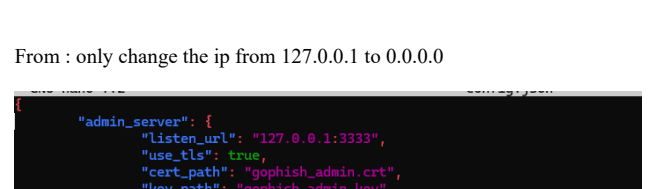


Enter the login name so that email will be created for that name..



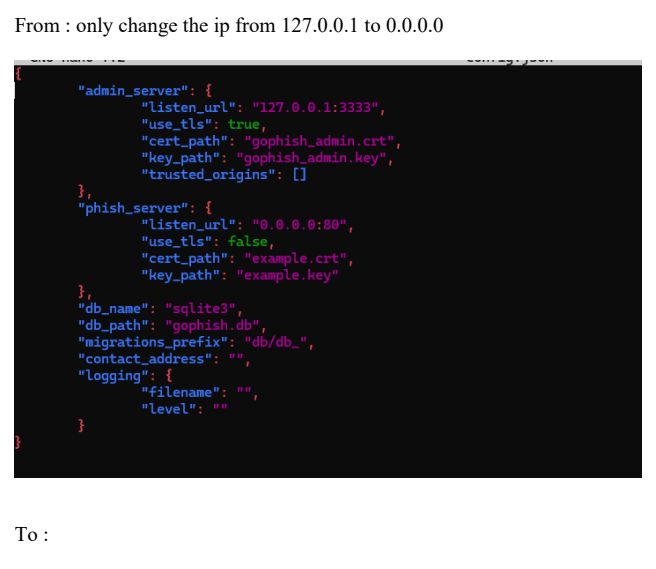
Step 10 : Edit the config.json file in gophish directory.

Nano config.json



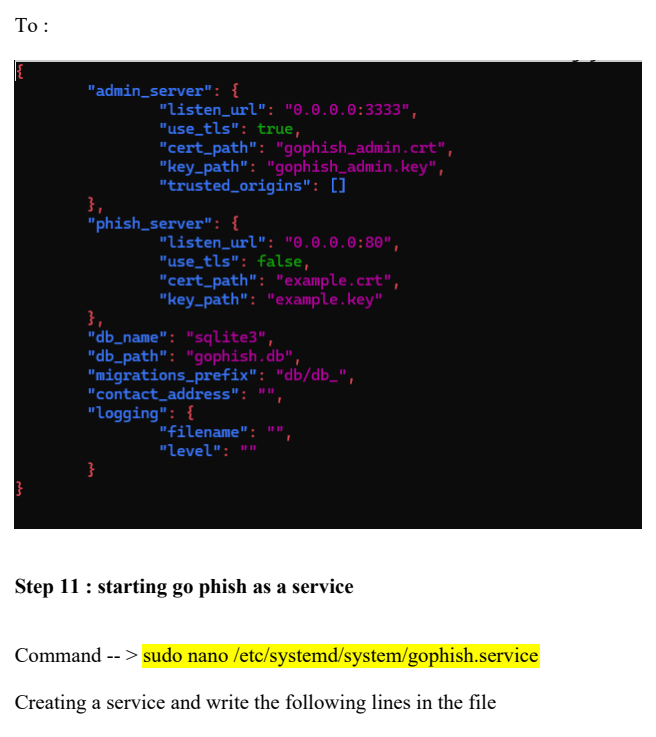
```
ubuntu@ip-172-31-23-88:~$ cd gophish/
ubuntu@ip-172-31-23-88:~/gophish$ ls
CONTRIBUTING.md    SECURITY.md       config.json  doc      gophish.go   middleware    util
Dockerfile         VERSION           context      docker   gulpfile.js  models        webhook
ISSUE_TEMPLATE.md  ansible-playbook  controllers  go.mod   imap         package.json  webpack.config.js
LICENSE            auth              db           go.sum   logger       static        worker
README.md          config            dialer       gophish  mailer       templates     yarn.lock
ubuntu@ip-172-31-23-88:~/gophish$
```

From : only change the ip from 127.0.0.1 to 0.0.0.0



```
{
        "admin_server": {
                "listen_url": "127.0.0.1:3333",
                "use_tls": true,
                "cert_path": "gophish_admin.crt",
                "key_path": "gophish_admin.key",
                "trusted_origins": []
        },
        "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": "",
        "logging": {
                "filename": "",
                "level": ""
        }
}
```

To :

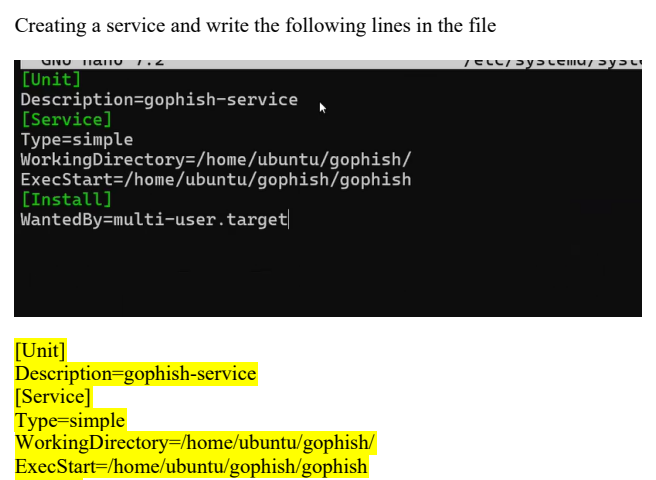


```
{
        "admin_server": {
                "listen_url": "0.0.0.0:3333",
                "use_tls": true,
                "cert_path": "gophish_admin.crt",
                "key_path": "gophish_admin.key",
                "trusted_origins": []
        },
        "phish_server": {
                "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": "",
        "logging": {
                "filename": "",
                "level": ""
        }
}
```

Step 11 : starting go phish as a service

Command -- > sudo nano /etc/systemd/system/gophish.service

Creating a service and write the following lines in the file

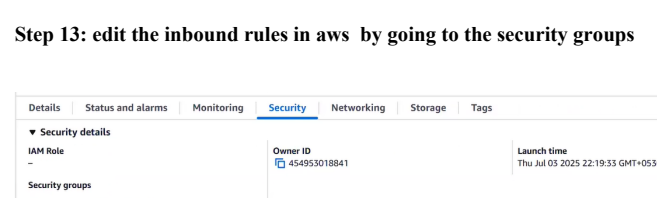


```
[Unit]
Description=gophish-service
[Service]
Type=simple
WorkingDirectory=/home/ubuntu/gophish/
ExecStart=/home/ubuntu/gophish/gophish
[Install]
WantedBy=multi-user.target
```

[Unit]
Description=gophish-service
[Service]
Type=simple
WorkingDirectory=/home/ubuntu/gophish/
ExecStart=/home/ubuntu/gophish/gophish
[Install]
WantedBy=multi-user.target

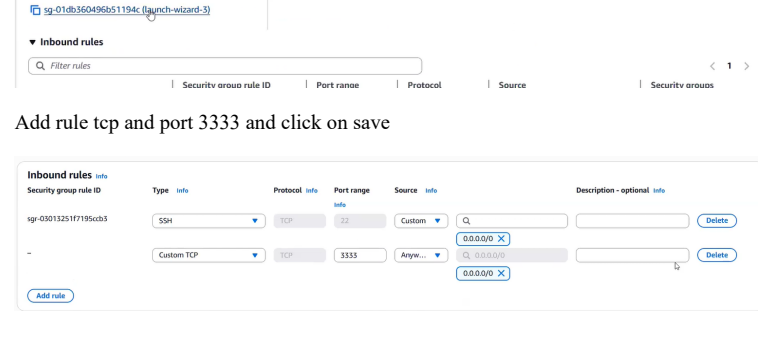
Step 12 : enable the service

Sudo systemctl enable gophish.service then start the service sudo systemctl start gophish.service and check the status sudo systemctl status gophish.service

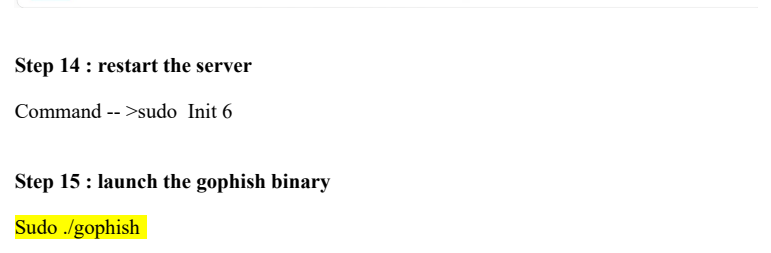


```
ubuntu@ip-172-31-23-88:~/gophish$ sudo systemctl enable gophish.service
Created symlink /etc/systemd/system/multi-user.target.wants/gophish.service → /etc/systemd/system/gophish.service.
ubuntu@ip-172-31-23-88:~/gophish$ sudo systemctl start gophish.service
ubuntu@ip-172-31-23-88:~/gophish$ sudo systemctl status gophish.service
● gophish.service - gophish-service
     Loaded: loaded (/etc/systemd/system/gophish.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-07-07 16:55:03 UTC; 12s ago
   Main PID: 2423 (gophish)
      Tasks: 6 (limit: 1124)
     Memory: 18.4M (peak: 18.5M)
        CPU: 138ms
     CGroup: /system.slice/gophish.service
             └─2423 /home/ubuntu/gophish/gophish
```

Step 13: edit the inbound rules in aws by going to the security groups



Add rule tcp and port 3333 and click on save



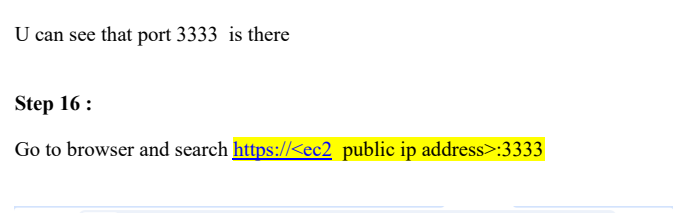
Step 14 : restart the server

Command -- >sudo Init 6

Step 15 : launch the gophish binary

Sudo ./gophish

It will give u a username and password..

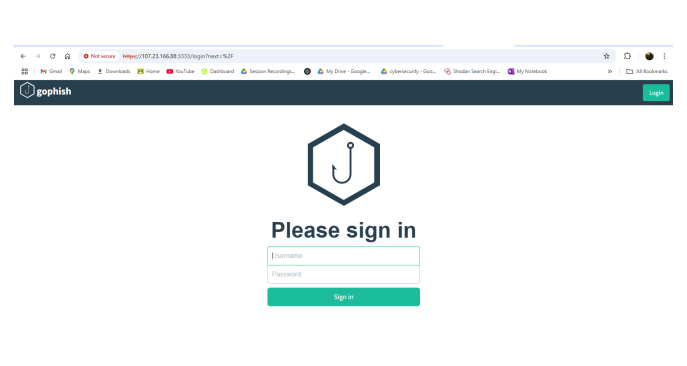


```
ubuntu@ip-172-31-23-88:~$ cd gophish/
ubuntu@ip-172-31-23-88:~/gophish$ sudo ./gophish
time="2025-07-07T16:59:05Z" level=warning msg="No contact address has been configured."
time="2025-07-07T16:59:05Z" level=warning msg="Please consider adding a contact_address entry in your config.json"
goose: no migrations to run. current version: 20220321133237
time="2025-07-07T16:59:05Z" level=info msg="Please login with the username admin and the password 7a656eba0eebf64e"
time="2025-07-07T16:59:05Z" level=info msg="Starting admin server at https://0.0.0.0:3333"
time="2025-07-07T16:59:05Z" level=fatal msg="listen tcp 0.0.0.0:3333: bind: address already in use"
ubuntu@ip-172-31-23-88:~/gophish$
```

U can see that port 3333 is there

Step 16 :

Go to browser and search https://<ec2 public ip address>:3333

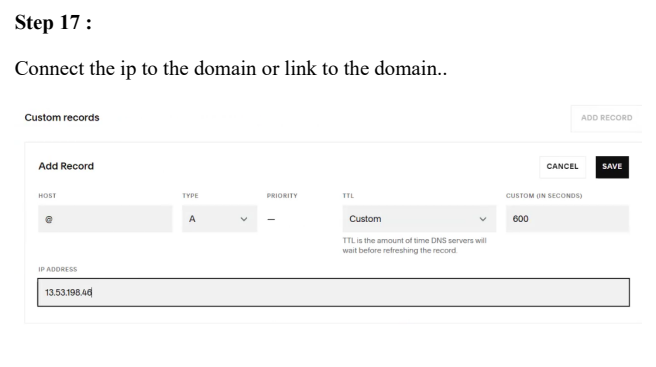


U can get the login page of go phish

Reset pw-- > khaja@123

Step 17 :

Connect the ip to the domain or link to the domain..

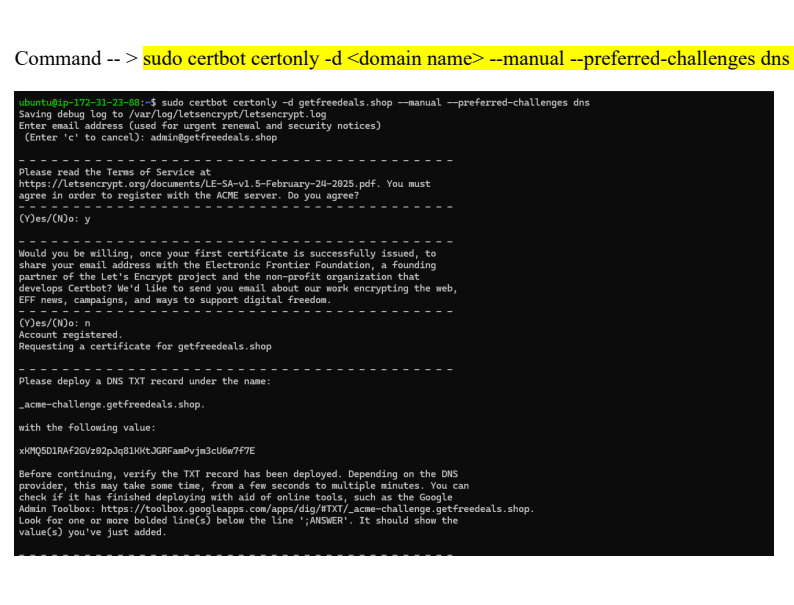


Step 18 : creating a TLS certificate for https

Command -- > sudo apt install certbot

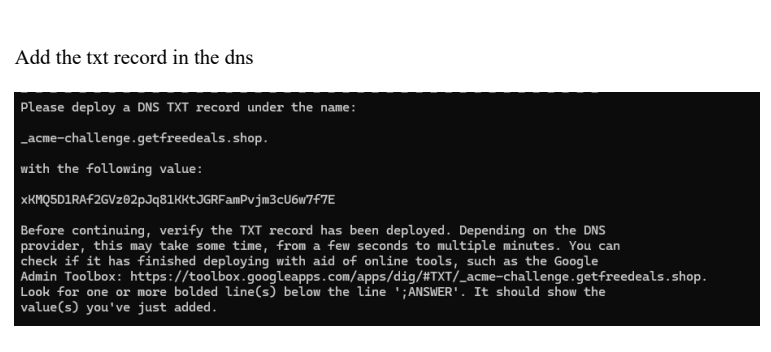
Certbot is a package using that we can create a tls certificate

Command -- > sudo certbot certonly -d <domain name> --manual --preferred-challenges dns



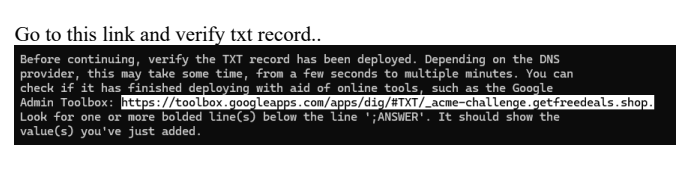
```
ubuntu@ip-172-31-23-88:~$ sudo certbot certonly -d getfreedeals.shop --manual --preferred-challenges dns
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Enter email address (used for urgent renewal and security notices)
 (Enter 'c' to cancel): admin@getfreedeals.shop

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.5-February-24-2025.pdf. You must
agree in order to register with the ACME server. Do you agree?
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
(Y)es/(N)o: y

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Would you be willing, once your first certificate is successfully issued, to
share your email address with the Electronic Frontier Foundation, a founding
partner of the Let's Encrypt project and the non-profit organization that
develops Certbot? We'd like to send you email about our work encrypting the web,
EFF news, campaigns, and ways to support digital freedom.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
(Y)es/(N)o: n
Account registered.
Requesting a certificate for getfreedeals.shop

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Please deploy a DNS TXT record under the name:

_acme-challenge.getfreedeals.shop.

with the following value:

xHMQSD1RAF2GVz82pJq81KHtJGRFamPvjm3cU6w7F7E

Before continuing, verify the TXT record has been deployed. Depending on the DNS
provider, this may take some time, from a few seconds to multiple minutes. You can
check if it has finished deploying with aid of online tools, such as the Google
Admin Toolbox: https://toolbox.googleapps.com/apps/dig/#TXT/_acme-challenge.getfreedeals.shop.
Look for one or more bolded line(s) below the line ';ANSWER'. It should show the
value(s) you've just added.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

Add the txt record in the dns

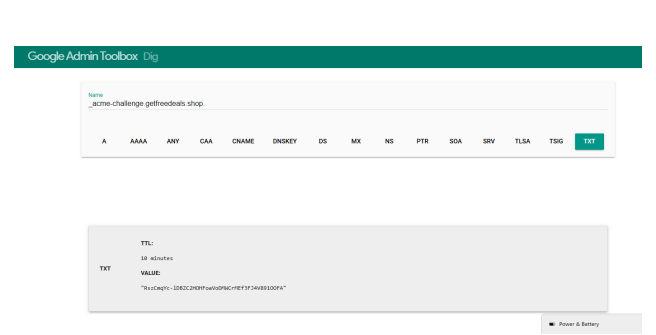


```
Please deploy a DNS TXT record under the name:

_acme-challenge.getfreedeals.shop.

with the following value:

xHMQSD1RAF2GVz82pJq81KHtJGRFamPvjm3cU6w7F7E

Before continuing, verify the TXT record has been deployed. Depending on the DNS
provider, this may take some time, from a few seconds to multiple minutes. You can
check if it has finished deploying with aid of online tools, such as the Google
Admin Toolbox: https://toolbox.googleapps.com/apps/dig/#TXT/_acme-challenge.getfreedeals.shop.
Look for one or more bolded line(s) below the line ';ANSWER'. It should show the
value(s) you've just added.
```

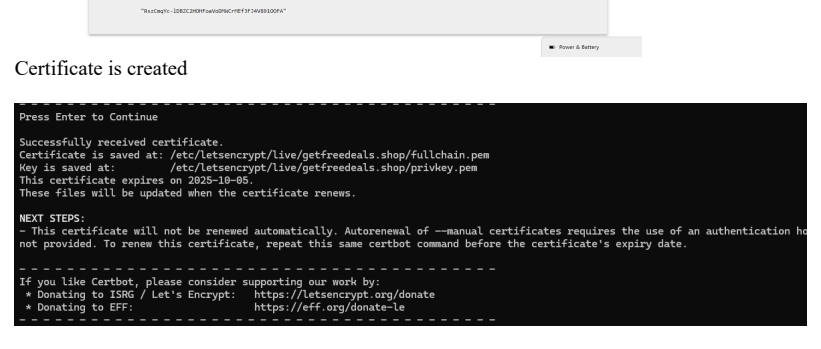
Go to this link and verify txt record..



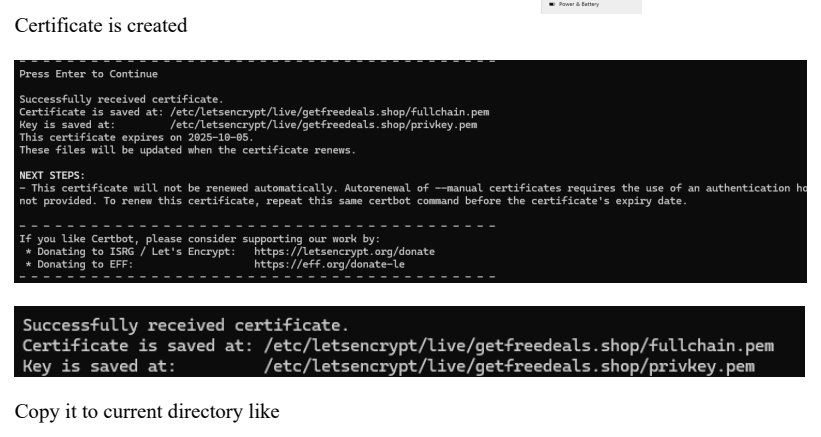
```
Before continuing, verify the TXT record has been deployed. Depending on the DNS
provider, this may take some time, from a few seconds to multiple minutes. You can
check if it has finished deploying with aid of online tools, such as the Google
Admin Toolbox: https://toolbox.googleapps.com/apps/dig/#TXT/_acme-challenge.getfreedeals.shop.
Look for one or more bolded line(s) below the line ';ANSWER'. It should show the
value(s) you've just added.
```



Certificate is created



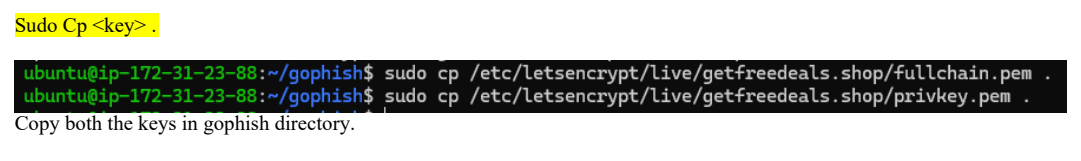
```
Press Enter to Continue

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/getfreedeals.shop/fullchain.pem
Key is saved at:         /etc/letsencrypt/live/getfreedeals.shop/privkey.pem
This certificate expires on 2025-10-05.
These files will be updated when the certificate renews.

NEXT STEPS:
- This certificate will not be renewed automatically. Autorenewal of --manual certificates requires the use of an authentication ho
not provided. To renew this certificate, repeat this same certbot command before the certificate's expiry date.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
If you like Certbot, please consider supporting our work by:
 * Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
 * Donating to EFF:                    https://eff.org/donate-le
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```



```
Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/getfreedeals.shop/fullchain.pem
Key is saved at:         /etc/letsencrypt/live/getfreedeals.shop/privkey.pem
```

Copy it to current directory like

Sudo Cp <key> .

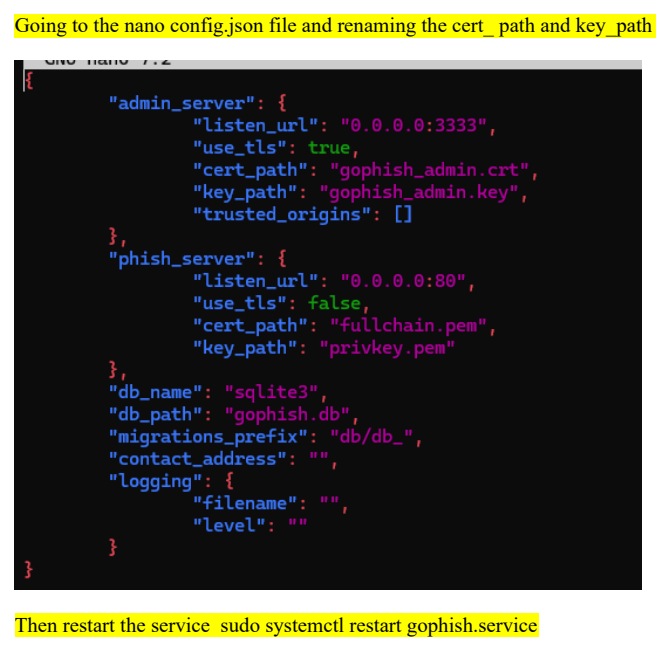


```
ubuntu@ip-172-31-23-88:~/gophish$ sudo cp /etc/letsencrypt/live/getfreedeals.shop/fullchain.pem .
ubuntu@ip-172-31-23-88:~/gophish$ sudo cp /etc/letsencrypt/live/getfreedeals.shop/privkey.pem .
```

Copy both the keys in gophish directory.

Step 19: attach the certificate

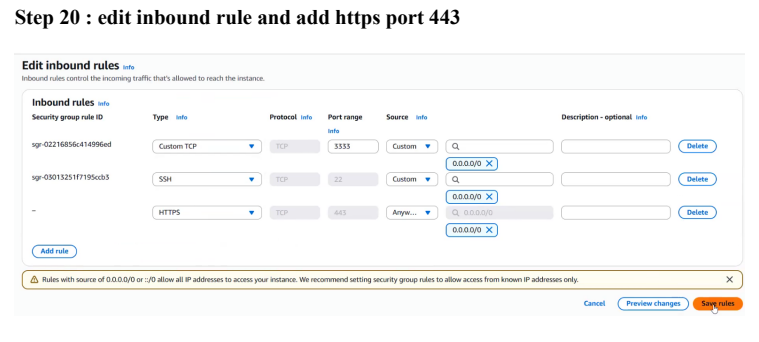
Going to the nano config.json file and renaming the cert_ path and key_path with public and private key under phish_server



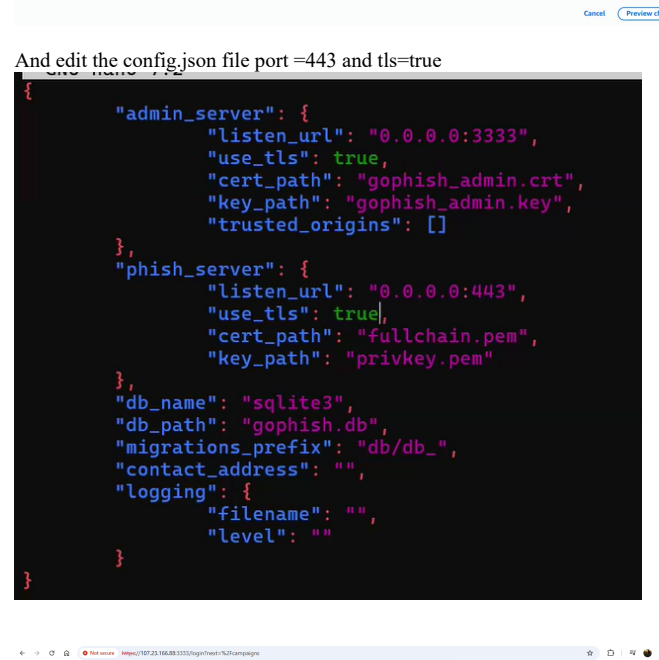
```
{
        "admin_server": {
                "listen_url": "0.0.0.0:3333",
                "use_tls": true,
                "cert_path": "gophish_admin.crt",
                "key_path": "gophish_admin.key",
                "trusted_origins": []
        },
        "phish_server": {
                "listen_url": "0.0.0.0:80",
                "use_tls": false,
                "cert_path": "fullchain.pem",
                "key_path": "privkey.pem"
        },
        "db_name": "sqlite3",
        "db_path": "gophish.db",
        "migrations_prefix": "db/db_",
        "contact_address": "",
        "logging": {
                "filename": "",
                "level": ""
        }
}
```

Then restart the service sudo systemctl restart gophish.service

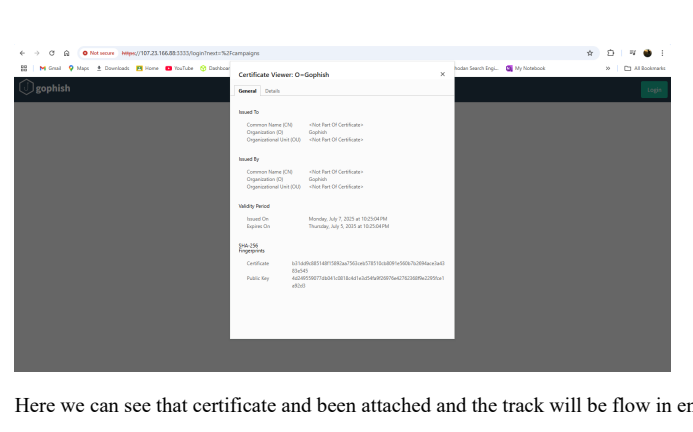
Step 20 : edit inbound rule and add https port 443



And edit the config.json file port =443 and tls=true

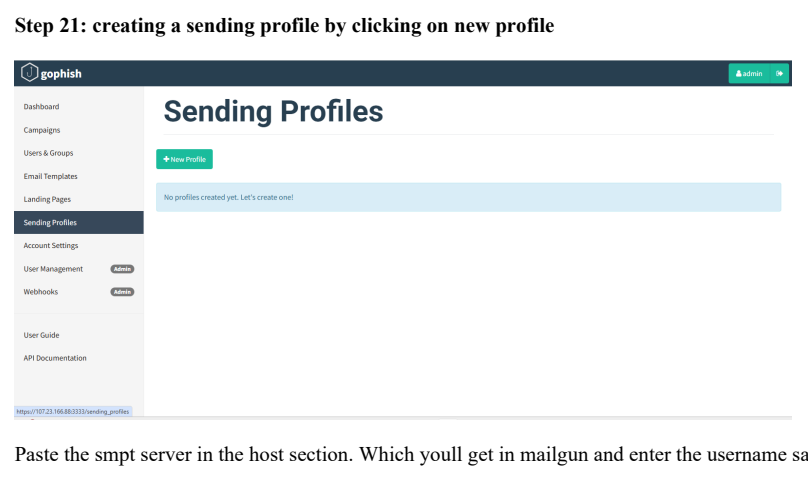


```
{
        "admin_server": {
                "listen_url": "0.0.0.0:3333",
                "use_tls": true,
                "cert_path": "gophish_admin.crt",
                "key_path": "gophish_admin.key",
                "trusted_origins": []
        },
        "phish_server": {
                "listen_url": "0.0.0.0:443",
                "use_tls": true,
                "cert_path": "fullchain.pem",
                "key_path": "privkey.pem"
        },
        "db_name": "sqlite3",
        "db_path": "gophish.db",
        "migrations_prefix": "db/db_",
        "contact_address": "",
        "logging": {
                "filename": "",
                "level": ""
        }
}
```

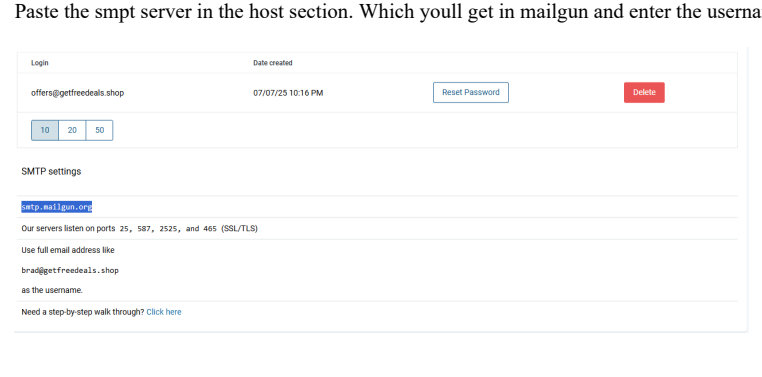


Here we can see that certificate and been attached and the track will be flow in encrypted text

Step 21: creating a sending profile by clicking on new profile



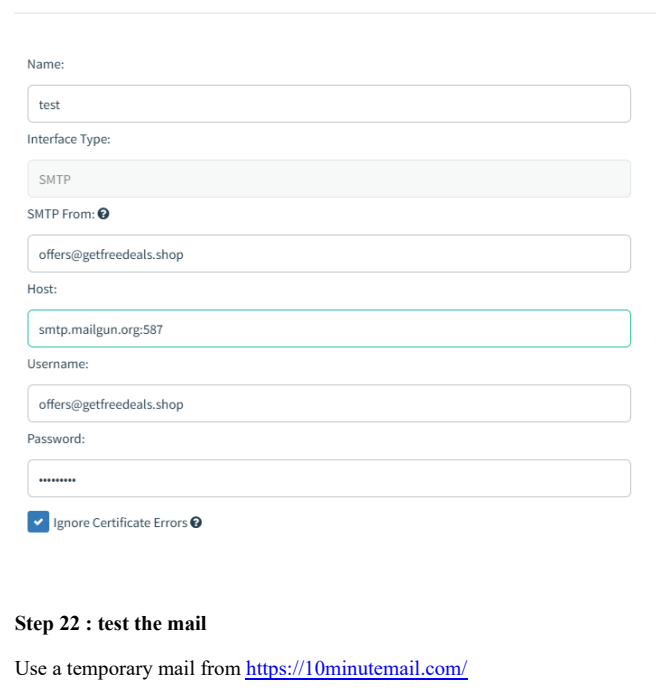
Paste the smpt server in the host section. Which youll get in mailgun and enter the username same and password from mailgun



SMTP settings

smtp.mailgun.org

Our servers listen on ports 25, 587, 2525, and 465 (SSL/TLS)



Name: test

Interface Type: SMTP

SMTP From: offers@getfreedeals.shop

Host: smtp.mailgun.org:587

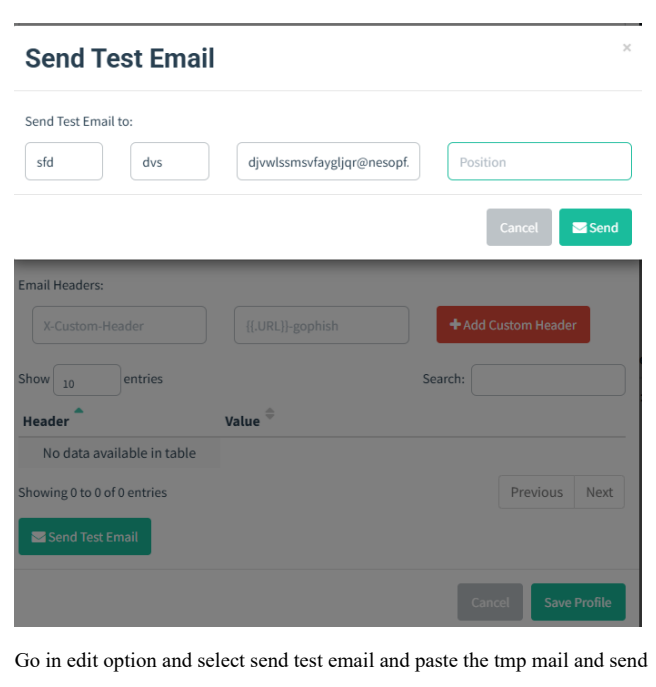
Username: offers@getfreedeals.shop

Password: ••••••••••

☑ Ignore Certificate Errors

Step 22 : test the mail

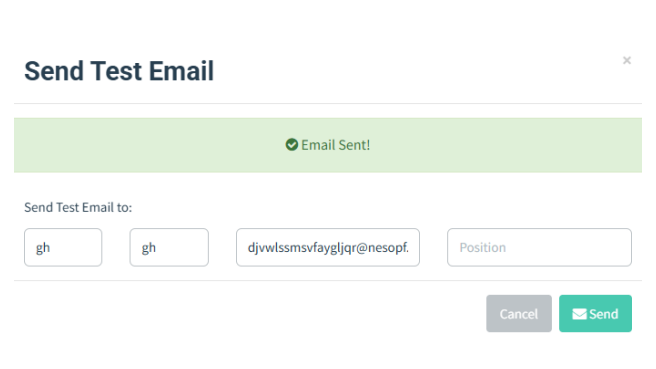
Use a temporary mail from https://10minutemail.com/



Send Test Email

Send Test Email to: sfd | dvs | djvwlssmsvfaygljqr@nesopf. | Position

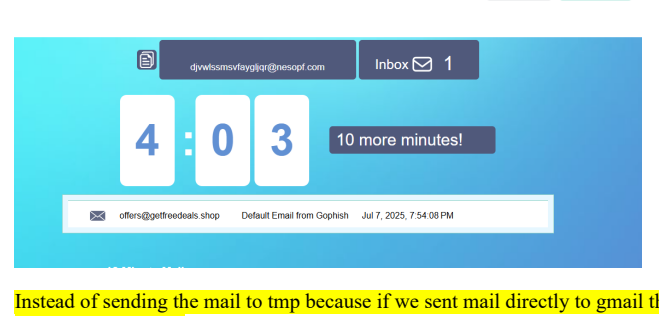
Go in edit option and select send test email and paste the tmp mail and send



Send Test Email

Email Sent!

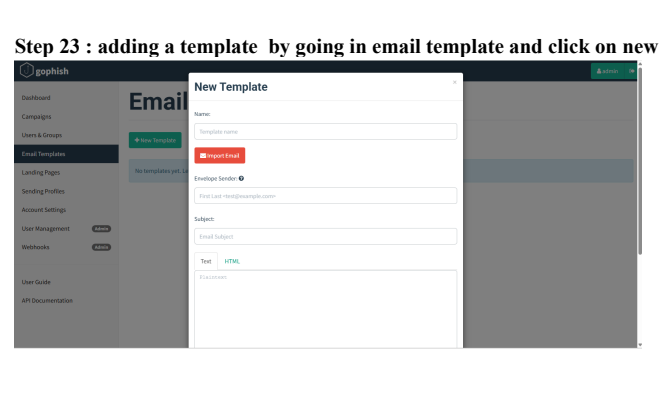
Send Test Email to: gh | gh | djvwlssmsvfaygljqr@nesopf. | Position

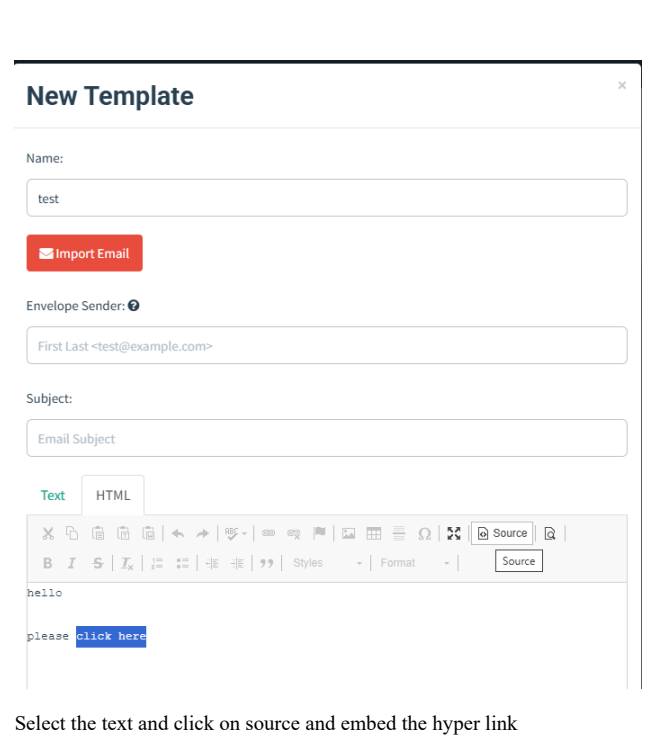


offers@getfreedeals.shop   Default Email from Gophish   Jul 7, 2025, 7:54:08 PM

Instead of sending the mail to tmp because if we sent mail directly to gmail the email security block the mail and blacklist the domain by seeing defualt email from gophish

Step 23 : adding a template by going in email template and click on new template





New Template

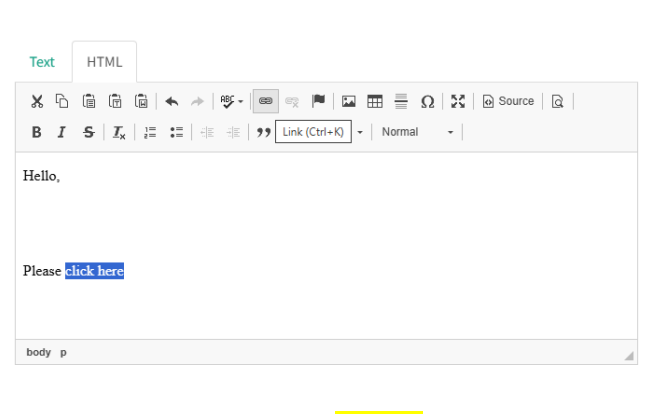
Name: test

Envelope Sender: First Last <test@example.com>

Subject: Email Subject

```
hello

please click here
```

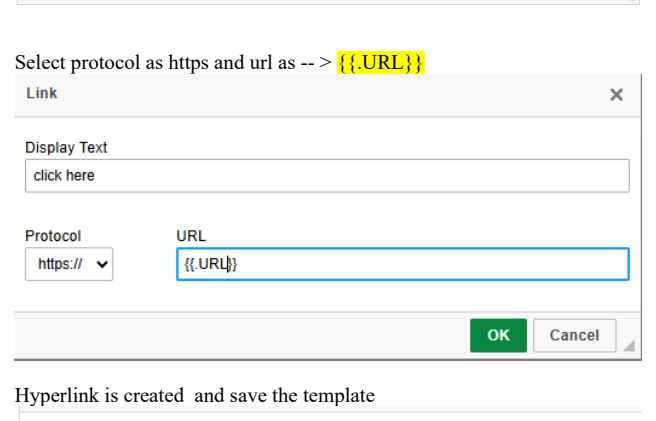
Select the text and click on source and embed the hyper link



Hello,

Please click here

Select protocol as https and url as -- > {{.URL}}



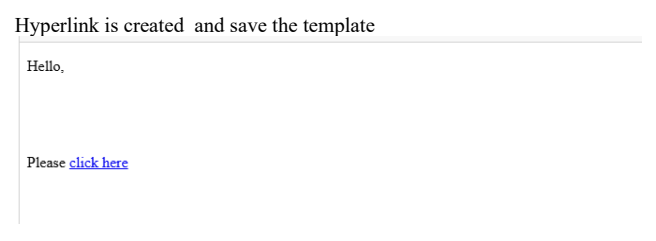
Link

Display Text: click here

Protocol: https://

URL: {{.URL}}

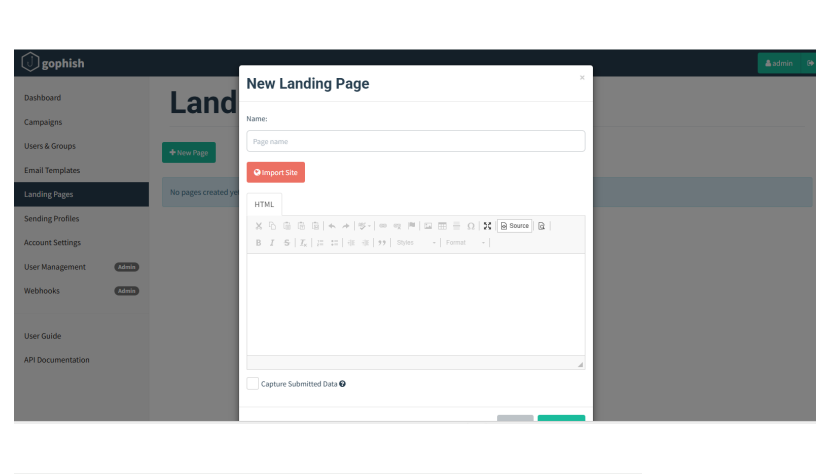
Hyperlink is created and save the template

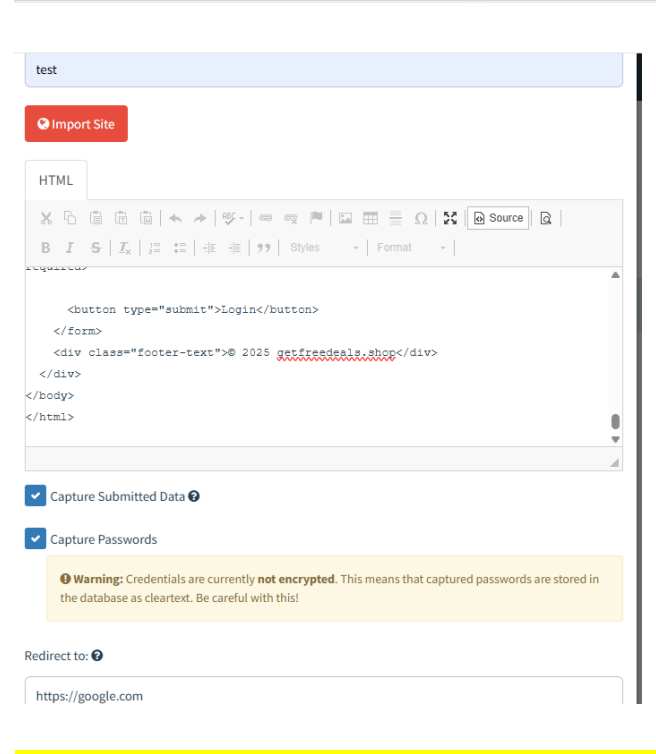


Hello,

Please click here

Step 24 : creating a landing page.. A page will open after victim click on the link.. Creating it by going in landing page and then click on new page





test

```
        <button type="submit">Login</button>
    </form>
    <div class="footer-text">© 2025 getfreedeals.shop</div>
  </div>
</body>
</html>
```

☑ Capture Submitted Data

☑ Capture Passwords

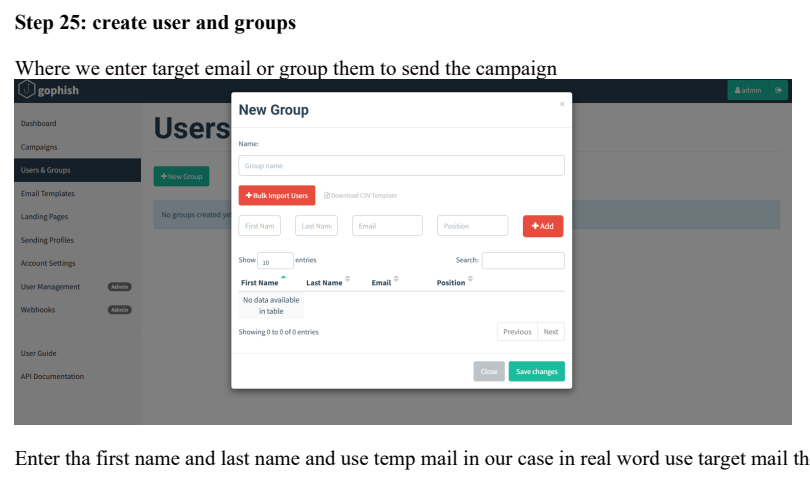
Warning: Credentials are currently **not encrypted**. This means that captured passwords are stored in the database as cleartext. Be careful with this!

Redirect to: https://google.com

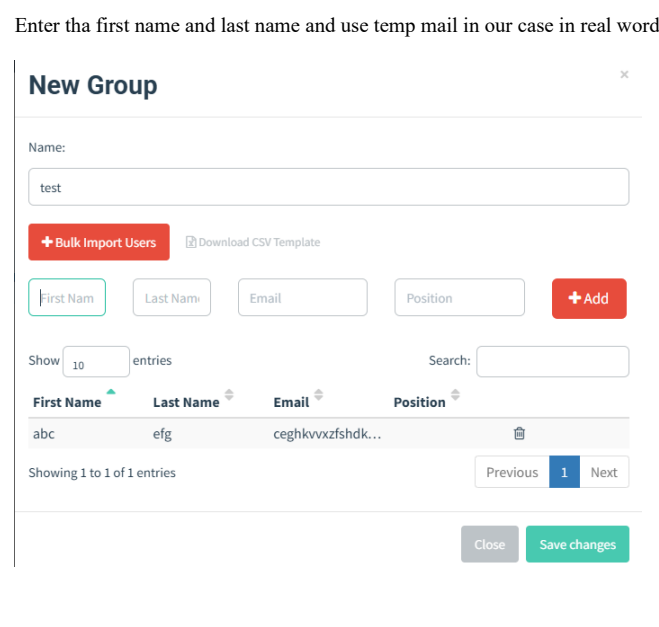
Import the html code which you can generate it form ai.. And enter the redirect link so after entering the credential victim will redirected to google.com in our case and click on save..

Step 25: create user and groups

Where we enter target email or group them to send the campaign



Enter tha first name and last name and use temp mail in our case in real word use target mail then click on add and save changes



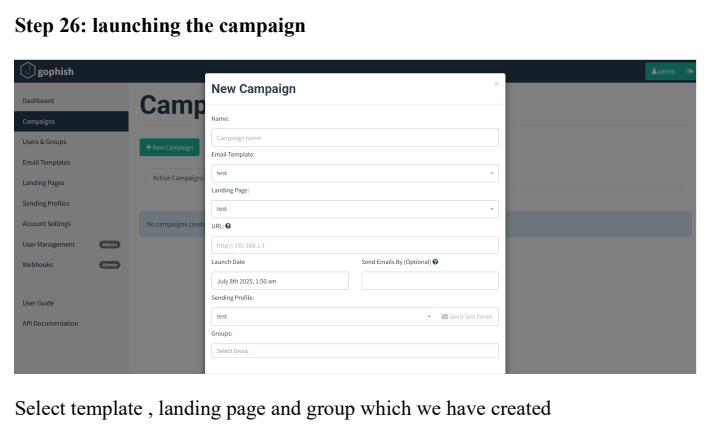
New Group

Name: test

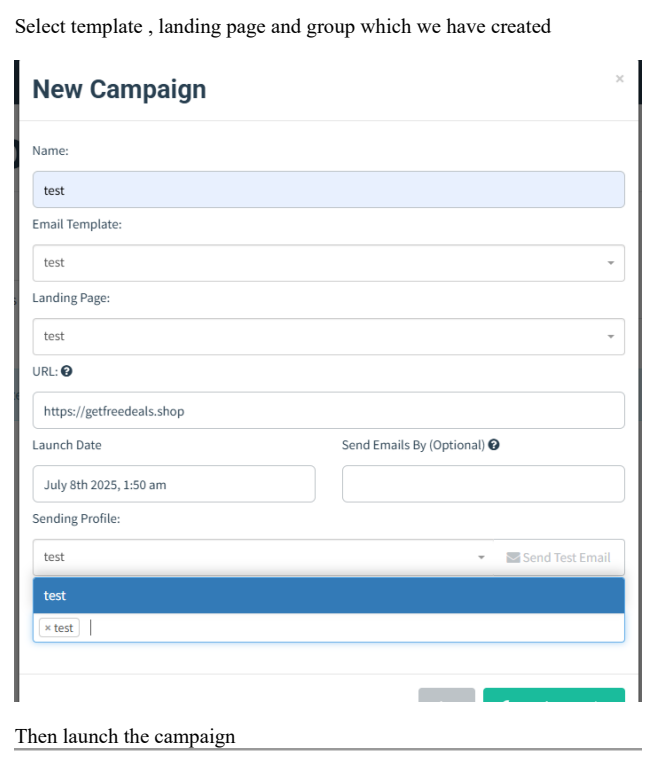
First Name	Last Name	Email	Position
abc	efg	ceghkvvxzfshdk...	

Showing 1 to 1 of 1 entries

Step 26: launching the campaign



Select template , landing page and group which we have created



New Campaign

Name: test

Email Template: test

Landing Page: test

URL: https://getfreedeals.shop

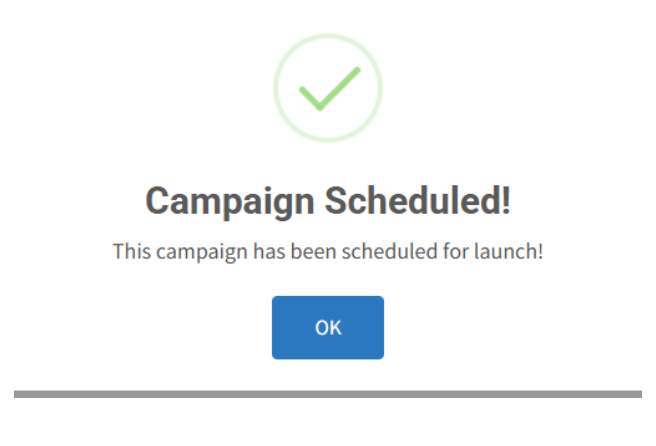
Launch Date: July 8th 2025, 1:50 am

Send Emails By (Optional):

Sending Profile: test

Groups: test

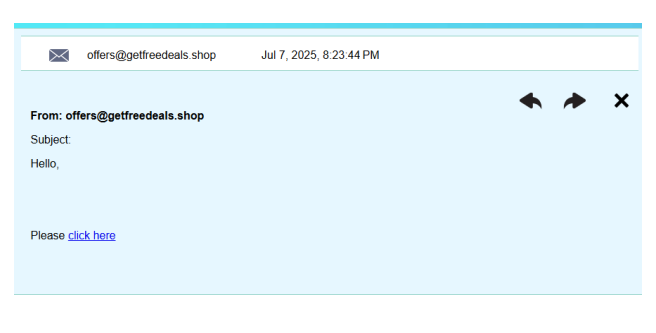
Then launch the campaign



Campaign Scheduled!

This campaign has been scheduled for launch!

Email will be sending to the target mail



offers@getfreedeals.shop   Jul 7, 2025, 8:23:44 PM

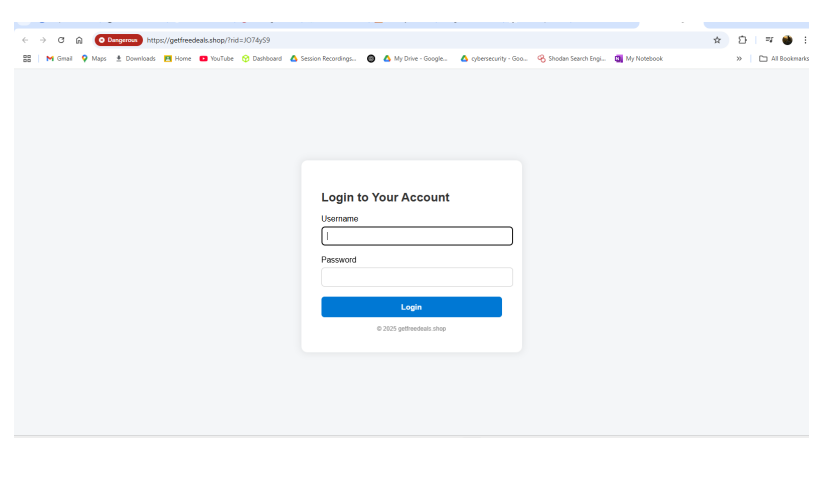
From: offers@getfreedeals.shop

Subject

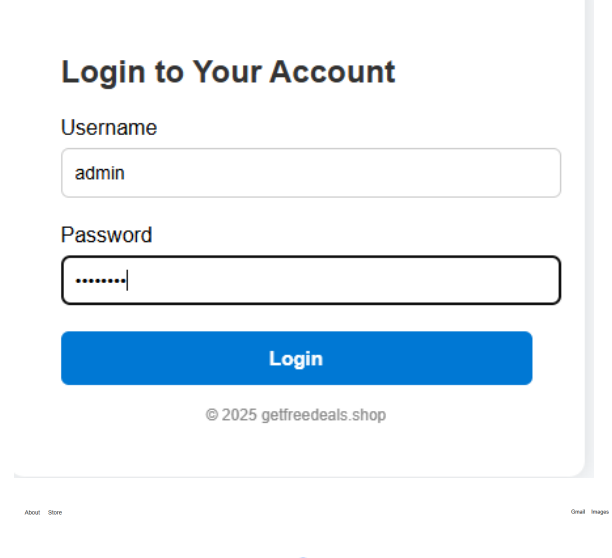
Hello,

Please click here

After the user clicks on the link it will redirected to our domain.. Or landing page



After victim enter the credentials and enter login.. It will redirect to the google.com

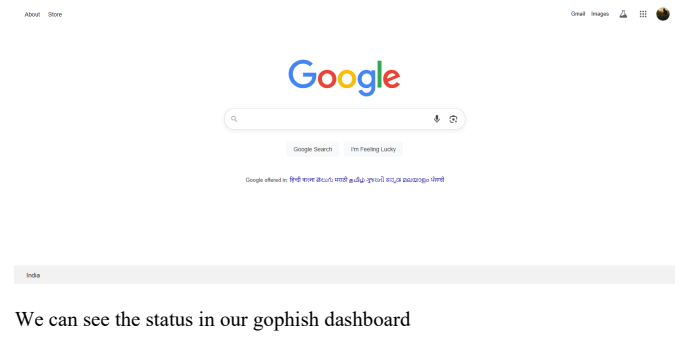


Login to Your Account

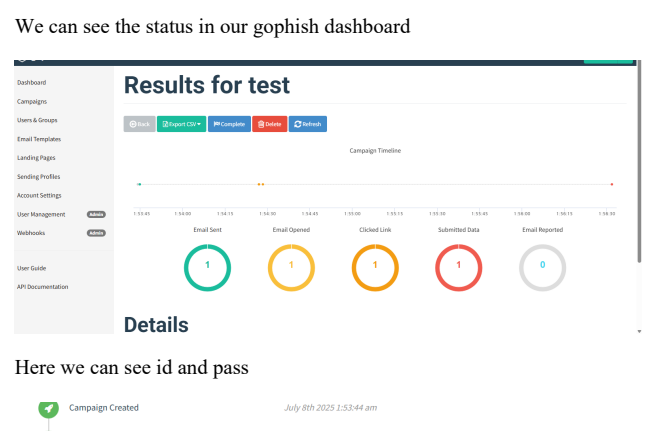
Username: admin

Password: ••••••••

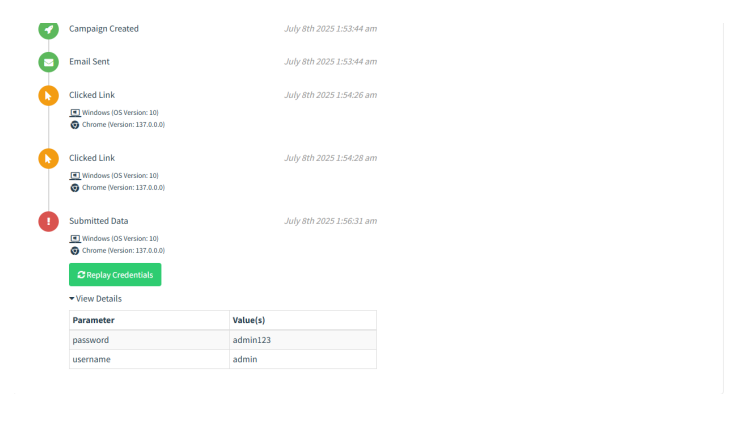
© 2025 getfreedeals.shop



We can see the status in our gophish dashboard



Here we can see id and pass



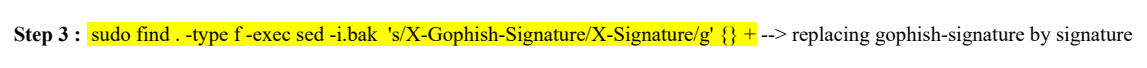
Parameter	Value(s)
password	admin123
username	admin

Hardening phishing server:

Hardening the server so it won't get detected by the email security or defensive solution. There are some changes u must need to do

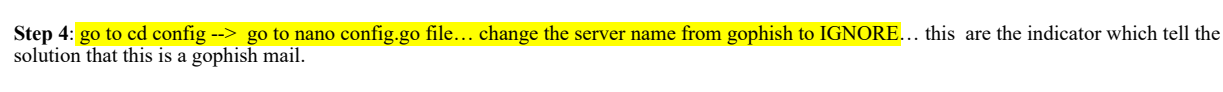
Step 1 : go to gophish directory..

Step 2 : sudo find . -type f -exec sed -i.bak 's/X-Gophish-Contact/X-Contact/g' {} + -- > replacing the gophish header by contact .. Like in email header it will display gophish so we are replacing that header so it wont get detect by the security solution



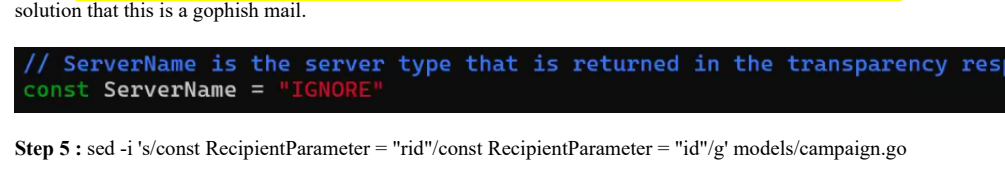
```
ubuntu@ip-172-31-23-88:~/gophish$  sudo find . -type f -exec sed -i.bak  's/X-Gophish-Contact/X-Contact/g' {} +
ubuntu@ip-172-31-23-88:~/gophish$
```

Step 3 : sudo find . -type f -exec sed -i.bak 's/X-Gophish-Signature/X-Signature/g' {} + --> replacing gophish-signature by signature



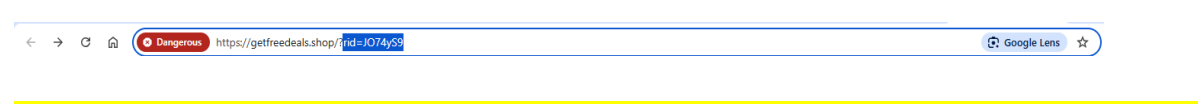
```
ubuntu@ip-172-31-23-88:~/gophish$  sudo find . -type f -exec sed -i.bak  's/X-Gophish-Signature/X-Signature/g' {} +
ubuntu@ip-172-31-23-88:~/gophish$
```

Step 4: go to cd config --> go to nano config.go file… change the server name from gophish to IGNORE… this are the indicator which tell the security solution that this is a gophish mail.

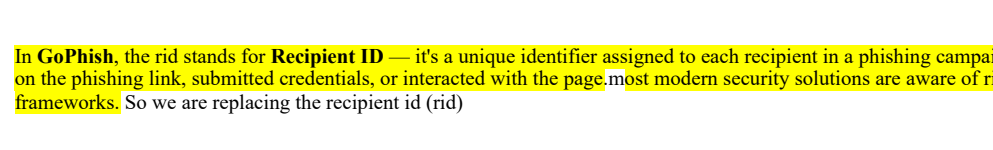


```
// ServerName is the server type that is returned in the transparency response.
const ServerName = "IGNORE"
```

Step 5 : sed -i 's/const RecipientParameter = "rid"/const RecipientParameter = "id"/g' models/campaign.go



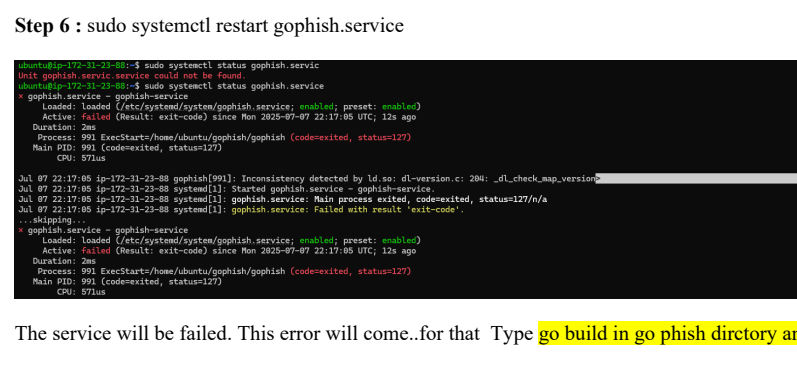
```
ubuntu@ip-172-31-23-88:~/gophish$ sed -i 's/const RecipientParameter = "rid"/const RecipientParameter = "id"/g' models/campaign.go
```



https://getfreedeals.shop/?rid=JO74yS9

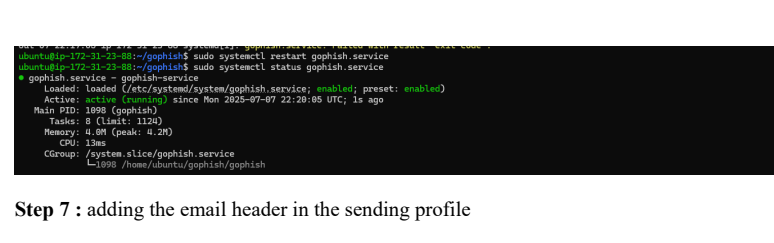
In **GoPhish**, the rid stands for **Recipient ID** — it's a unique identifier assigned to each recipient in a phishing campaign. It is used in tracking who clicked on the phishing link, submitted credentials, or interacted with the page.most modern security solutions are aware of rid used by GoPhish and similar frameworks. So we are replacing the recipient id (rid)

Step 6 : sudo systemctl restart gophish.service



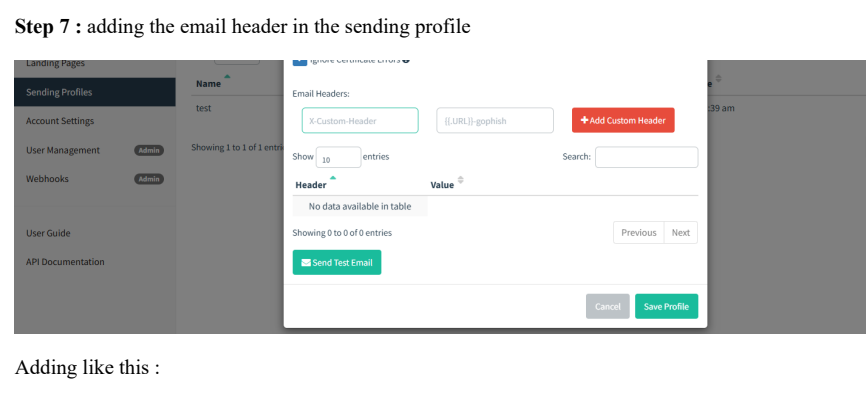
```
ubuntu@ip-172-31-23-88:~$ sudo systemctl status gophish.servic
Unit gophish.servic.service could not be found.
ubuntu@ip-172-31-23-88:~$ sudo systemctl status gophish.service
× gophish.service - gophish-service
     Loaded: loaded (/etc/systemd/system/gophish.service; enabled; preset: enabled)
     Active: failed (Result: exit-code) since Mon 2025-07-07 22:17:05 UTC; 12s ago
   Duration: 2ms
    Process: 991 ExecStart=/home/ubuntu/gophish/gophish (code=exited, status=127)
   Main PID: 991 (code=exited, status=127)
        CPU: 971us

Jul 07 22:17:05 ip-172-31-23-88 gophish[991]: Inconsistency detected by ld.so: dl-version.c: 204: _dl_check_map_version
Jul 07 22:17:05 ip-172-31-23-88 systemd[1]: Started gophish.service - gophish-service.
Jul 07 22:17:05 ip-172-31-23-88 systemd[1]: gophish.service: Main process exited, code=exited, status=127/n/a
Jul 07 22:17:05 ip-172-31-23-88 systemd[1]: gophish.service: Failed with result 'exit-code'.
...skipping...
× gophish.service - gophish-service
     Loaded: loaded (/etc/systemd/system/gophish.service; enabled; preset: enabled)
     Active: failed (Result: exit-code) since Mon 2025-07-07 22:17:05 UTC; 12s ago
   Duration: 2ms
    Process: 991 ExecStart=/home/ubuntu/gophish/gophish (code=exited, status=127)
   Main PID: 991 (code=exited, status=127)
        CPU: 971us
```

The service will be failed. This error will come..for that Type go build in go phish dirctory and restart the service. As u can see that the service is active

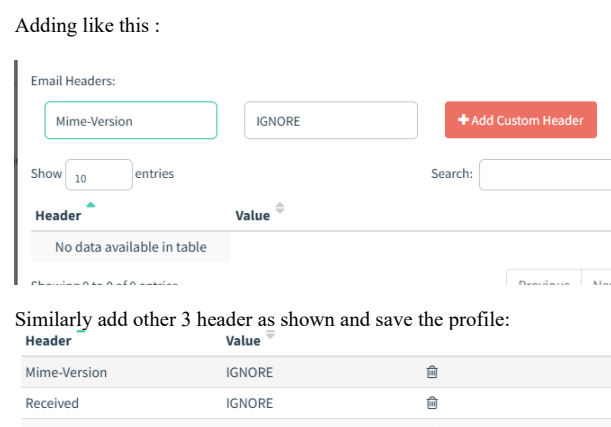


```
ubuntu@ip-172-31-23-88:~/gophish$ sudo systemctl restart gophish.service
ubuntu@ip-172-31-23-88:~/gophish$ sudo systemctl status gophish.service
● gophish.service - gophish-service
     Loaded: loaded (/etc/systemd/system/gophish.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-07-07 22:28:05 UTC; 1s ago
   Main PID: 1898 (gophish)
      Tasks: 8 (limit: 1124)
     Memory: 4.0M (peak: 4.2M)
        CPU: 13ms
     CGroup: /system.slice/gophish.service
             └─1898 /home/ubuntu/gophish/gophish
```

Step 7 : adding the email header in the sending profile

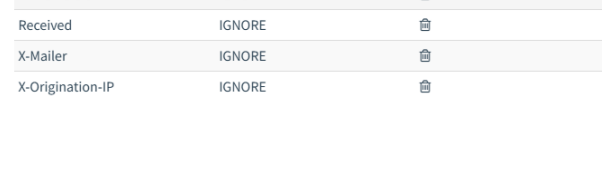


Adding like this :



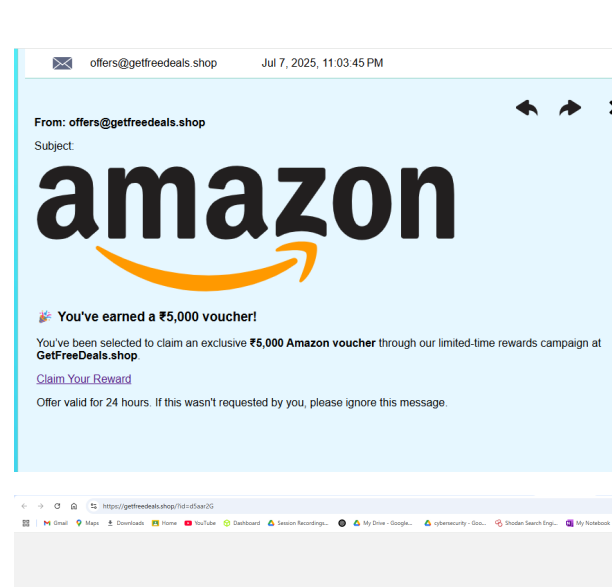
Email Headers: Mime-Version | IGNORE

Similarly add other 3 header as shown and save the profile:



Header	Value
Mime-Version	IGNORE
Received	IGNORE
X-Mailer	IGNORE
X-Origination-IP	IGNORE

My project :



offers@getfreedeals.shop   Jul 7, 2025, 11:03:45 PM

From: offers@getfreedeals.shop

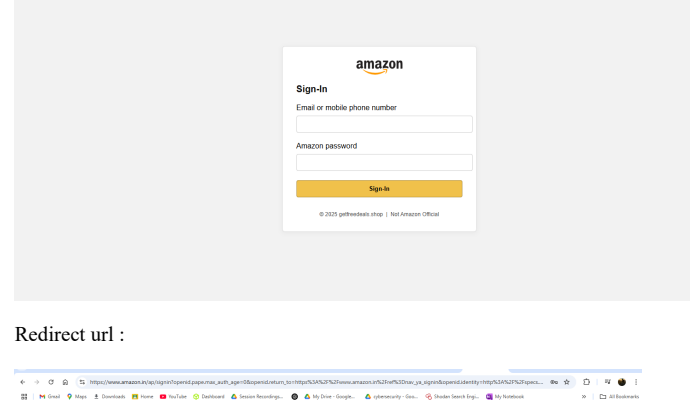
Subject

🎉 You've earned a ₹5,000 voucher!

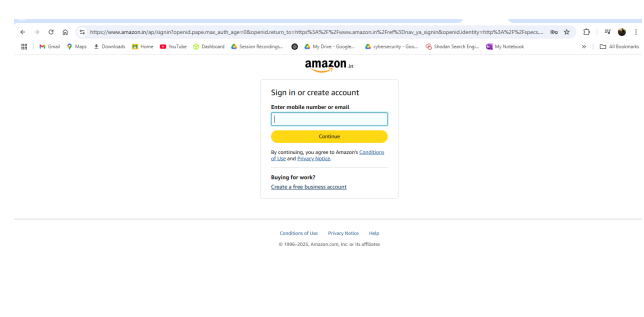
You've been selected to claim an exclusive **₹5,000 Amazon voucher** through our limited-time rewards campaign at **GetFreeDeals.shop**.

Claim Your Reward

Offer valid for 24 hours. If this wasn't requested by you, please ignore this message.



Redirect url :



Captured username and password :

Parameter	Value(s)
email	admin
id	d5aar2G
password	admin123