University of Dayton
Center for Cybersecurity
& Data Intelligence

Instructions for installing and using the GoPhish open-source phishing system.

GoPhish is an open-source platform that enables you to Phish test your organization. It is a web-based interface that is very intuitive. It is well supported, and there is a robust community behind this software. It provides features that are expected of professional grade phish tools, such as real time reporting and results. GoPhish is very easy to use and install and is available for all major platforms. A third-party SMTP server will be required to use this service. This can be your own self hosted SMTP server or a paid service (usually not very expensive). Your choice of SMTP sever should be whitelisted with your primary Email provider in order to allow for mail delivery.

GoPhish - https://getgophish.com

GoPhish Installation Instructions

We have set up a GoPhish installation as a proof of concept; organizations that are interested in a low-cost phishing installation could set this up at their organization, or ask for space from OCRI to set up Windows virtual machine on which this can be run. Details about getting space on OCRI can be found by contacting the OCRI (https://www.ohiocyberrangeinstitute.org/).

Prerequsites: A Windows machine(virtual or physical) capable of hosting a webserver and is accessible via the internet.

Setup Windows GoPhish Server (Beginners)

1. On your Windows host, navigate to https://github.com/gophish/gophish/releases and download the latest release compiled for Windows.
2. Once Downloaded, Extract the GoPhish Package to a folder in a location of your choosing on the local drive.
3. To Start the GoPhish Service, double click on “gophish.exe” This will open a command prompt window telling you that a webserver has been started on your local machine at https://localhost:3333 . It will also generate admin credentials for the server which will be displayed in the command prompt window. It will say “Please login with the username admin and the password” with the password being randomly generated.
4. After initial login, GoPhish will force you to create your own password for the admin user.
5. Initial Installation is Finished!

Pictures and selected content from the GoPhish documentation:
https://docs.getgophish.com/user-guide/documentation

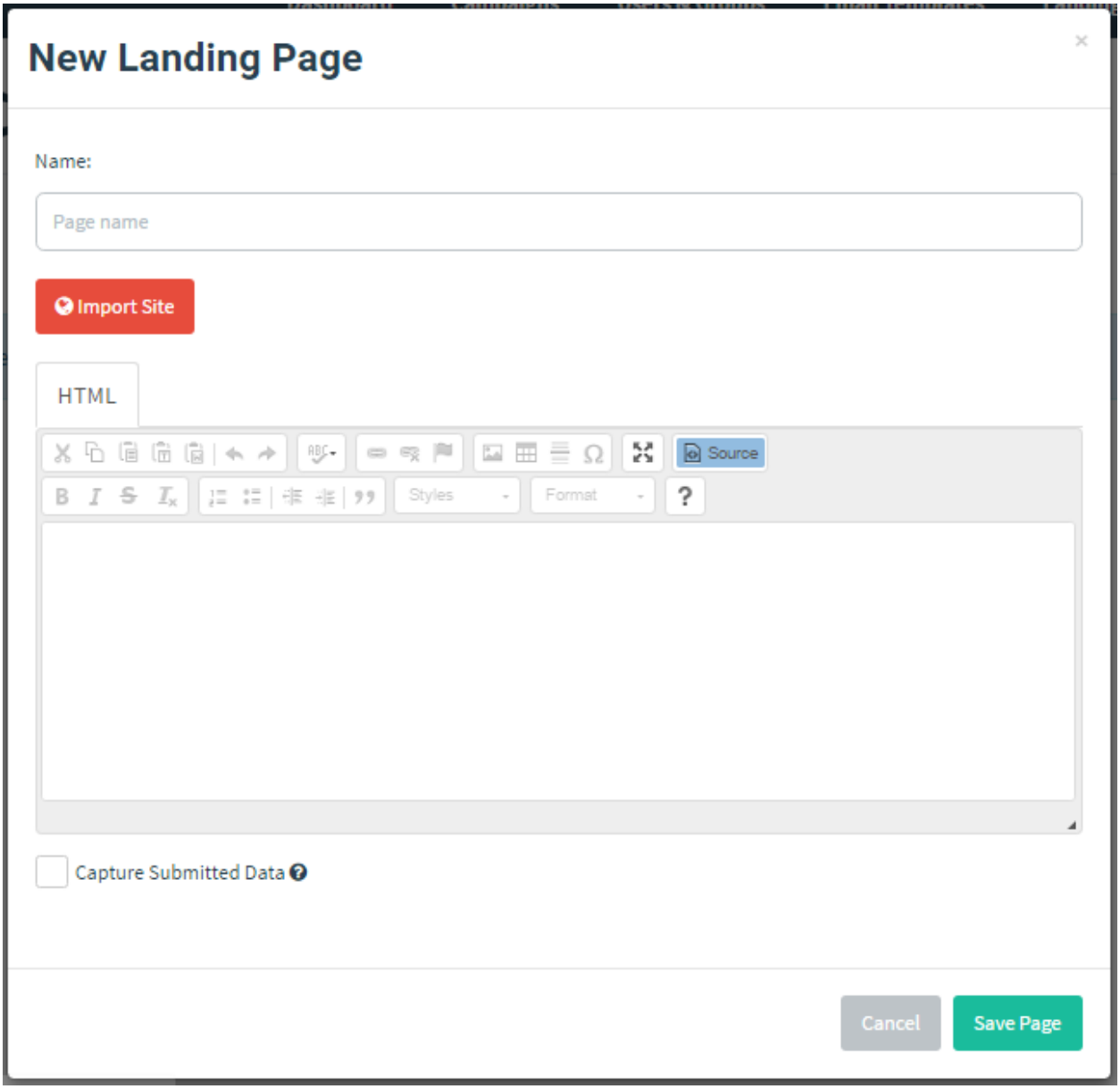
Setup a Landing Page

Landing pages are the actual HTML pages that are returned to the users when they click the phishing links they receive.

Landing pages support templating, capturing credentials, and redirecting users to another website after they submit their credentials.

To preview what a landing page will look like, you will need to either use the HTML editor seen below, or launch a test campaign. Simply browsing directly to the GoPhish listener.

To create a landing page, click on the "Landing Pages" entry in the sidebar and click the "New Page" button.



Landing Page Dialog

Importing a Site From URL

A powerful feature of GoPhish is the ability to import a site from a URL. To import a site, click the "Import Site" button. This is one of the coolest features of this software, it allows for you to input any login page and GoPhish will recreate it for you automatically.

Import Site Dialog

After entering the URL and clicking "Import", you should see the HTML of the URL populated into the editor.

Capturing Credentials

GoPhish makes it easy to capture credentials from the landing page. To capture credentials, simply select the checkbox that says "Capture Submitted Data".

Note: Credentials are stored **in plaintext**. If you don't want to capture passwords, don't select the "Capture Passwords" checkbox. GoPhish will still capture other text fields, such as usernames.

Capturing passwords would not be recommended, as the plaintext way they are captured could lead to data loss or compromise.

Redirecting Users

Red team assessments are all about preventing suspicion. To prevent users from becoming suspicious after entering credentials, you may want to redirect them to the original URL.

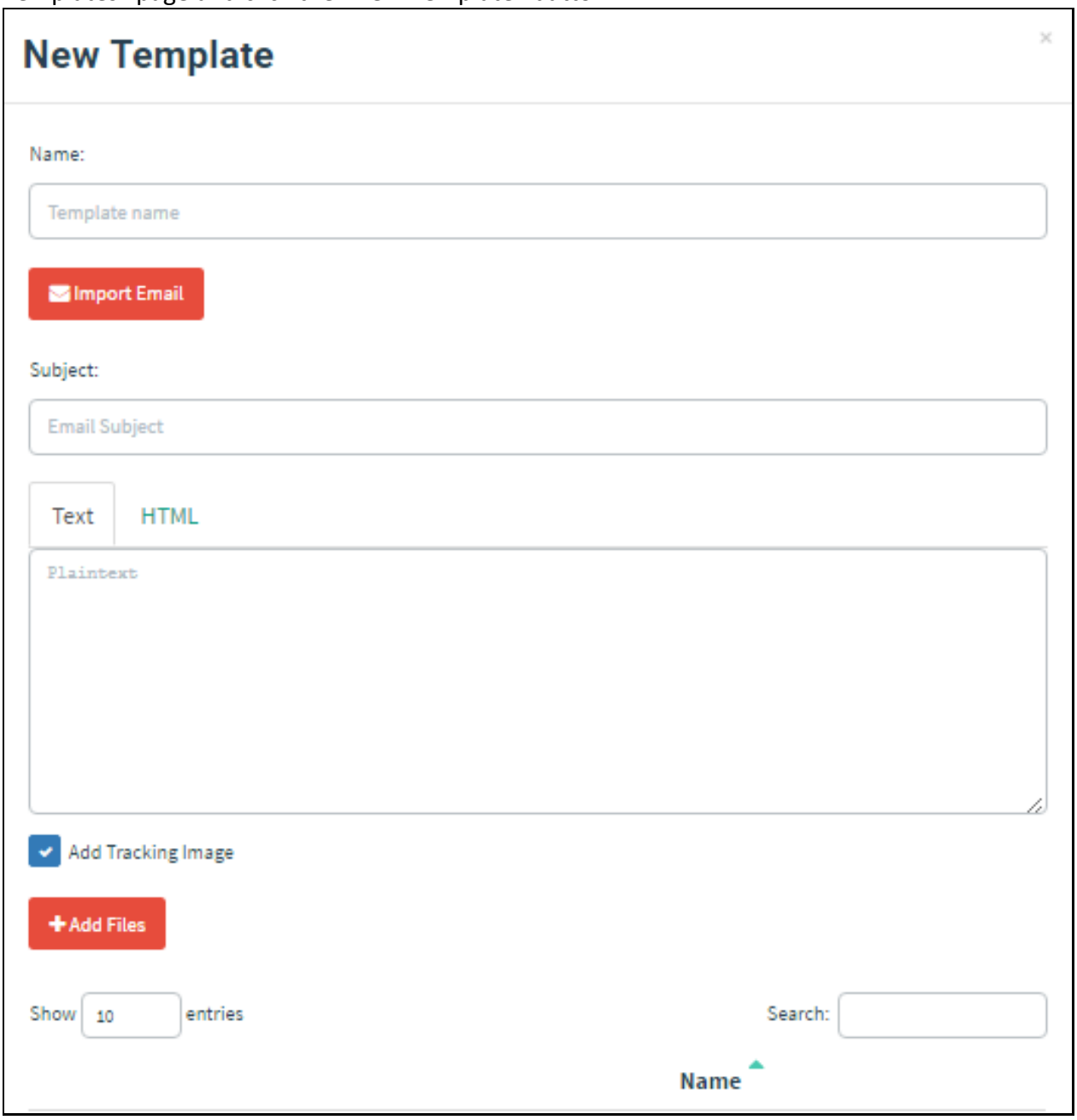
GoPhish makes it easy to redirect users after they submit credentials. To redirect users, enter a URL in the "Redirect To:" text field that appears after the "Capture Submitted Data" checkbox is selected.

This redirected URL can be a “You have been Phished Page” or it can be a login success page if you do not want to let the user know they have been phished quite yet.

Note: Make sure to include the full URL (including the scheme such as http:// or https://). Otherwise, browsers may interpret the URL as being relative to the GoPhish URL.

Setup an Email Template

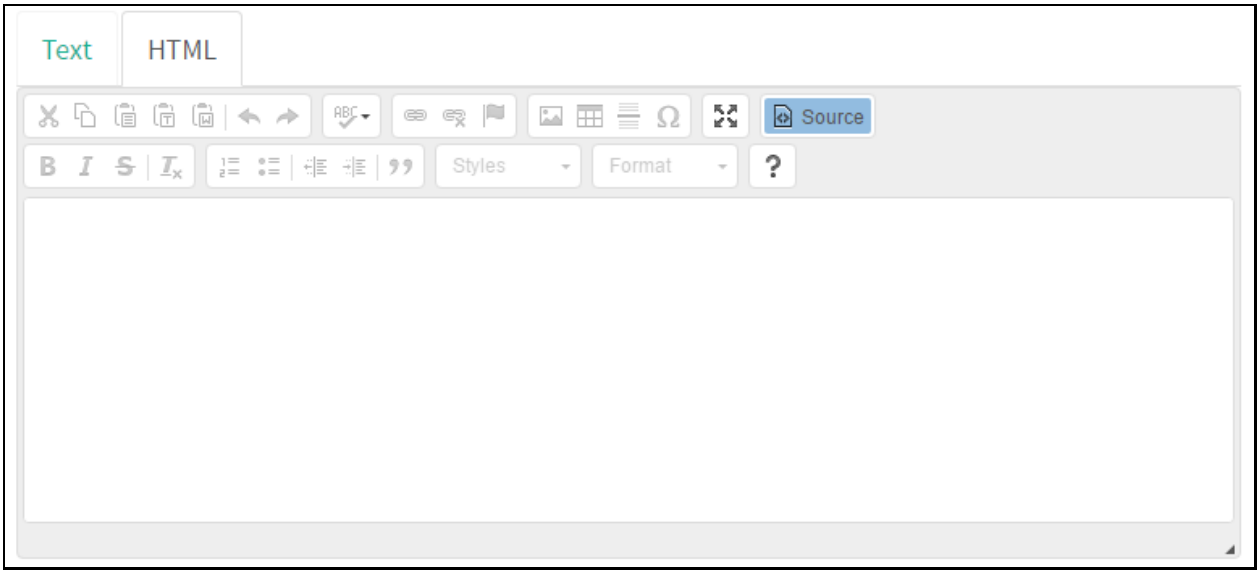
To create the template we will use for our Morning Catch campaign, first navigate to the "Email Templates" page and click the "New Template" button.



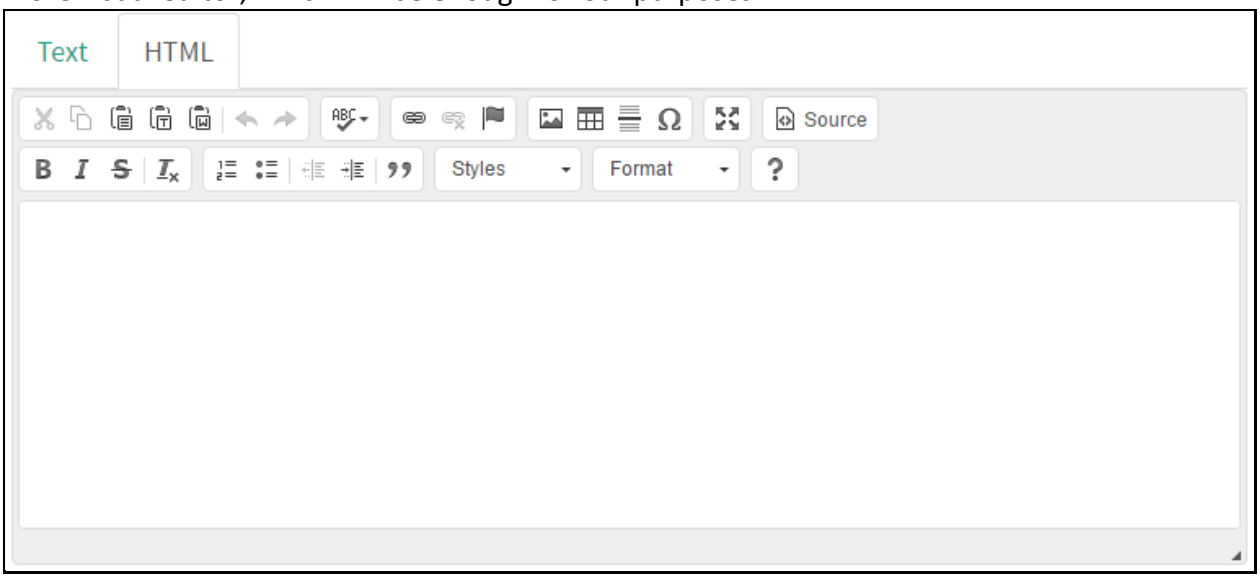
New Template Dialog

We notice that Morning Catch comes with a webmail portal. Let's craft a simple template that suggests the user needs to go reset their password. Obviously, this is a simple scenario, and by

using the "Import Email" feature, you can import existing emails directly into GoPhish for a greater effect.
We'll use the following subject line:
1
Password Reset for {{.Email}}
Copied!
You'll notice we used the {{.Email}} template value. This will populate with the target's email address when the emails are sent. This is GoPhish's way to tailor emails to individuals to increase the chance of success.
By clicking the "HTML" tab, we will see the editor we can use to create our HTML content:

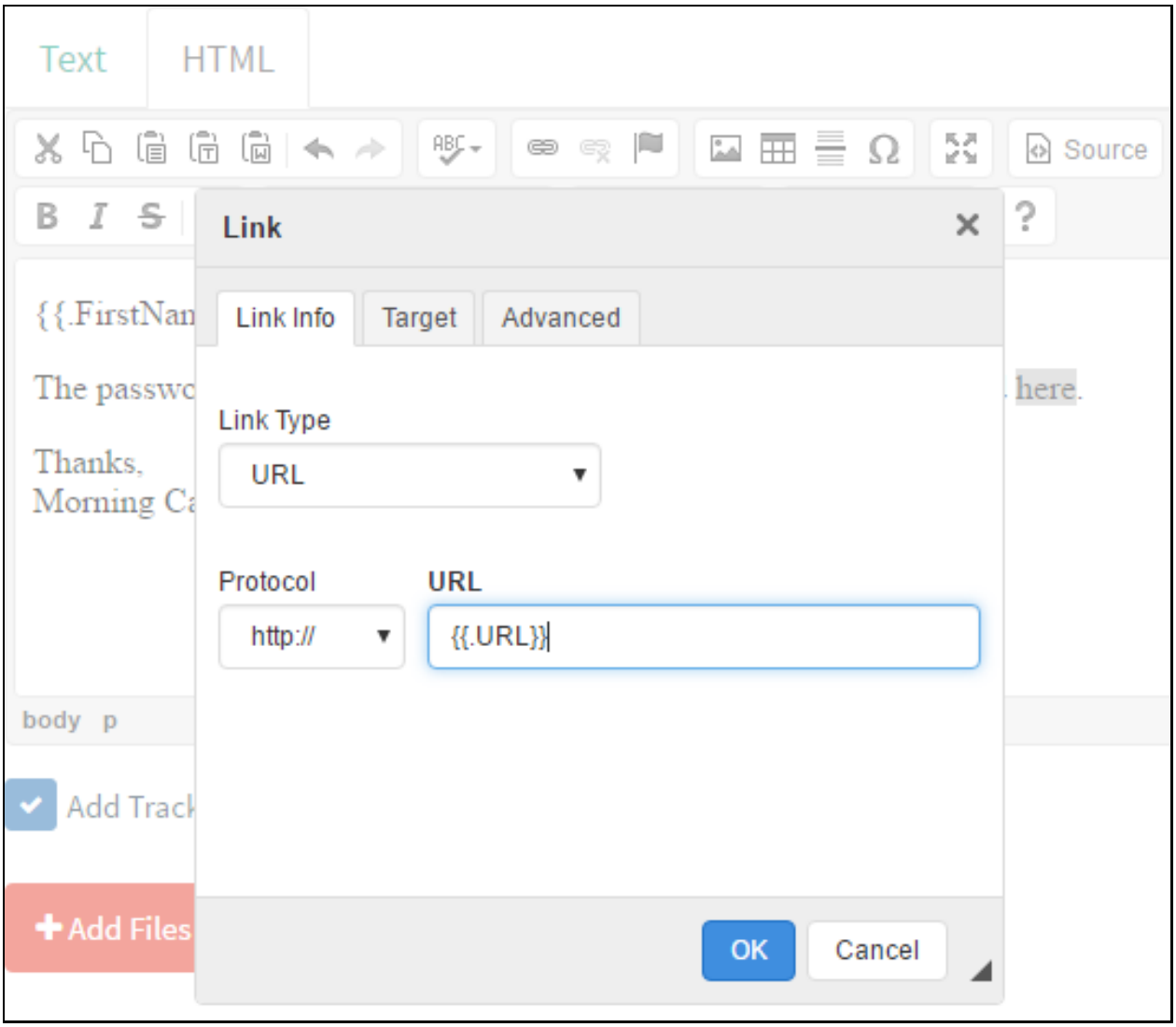


HTML Editor
Since our content is pretty simple, we can just click the "Source" button and be taken to the more visual editor, which will be enough for our purposes:



Visual Editor

Our template will be simple for the sake of demonstration. I'll start by adding the message:
1
{{.FirstName}},
2
3
The password for {{.Email}} has expired. Please reset your password here.
4
5
Thanks,
6
Morning Catch IT Team
Copied!
Speaking of links, now we need to add our phishing link. Highlight the word "here" and click the chain icon in the menu, exposing the "Link" dialog. In this dialog, we'll set the link to {{.URL}}, another template value, so that our link is automatically created and inserted into the email.



Link Menu

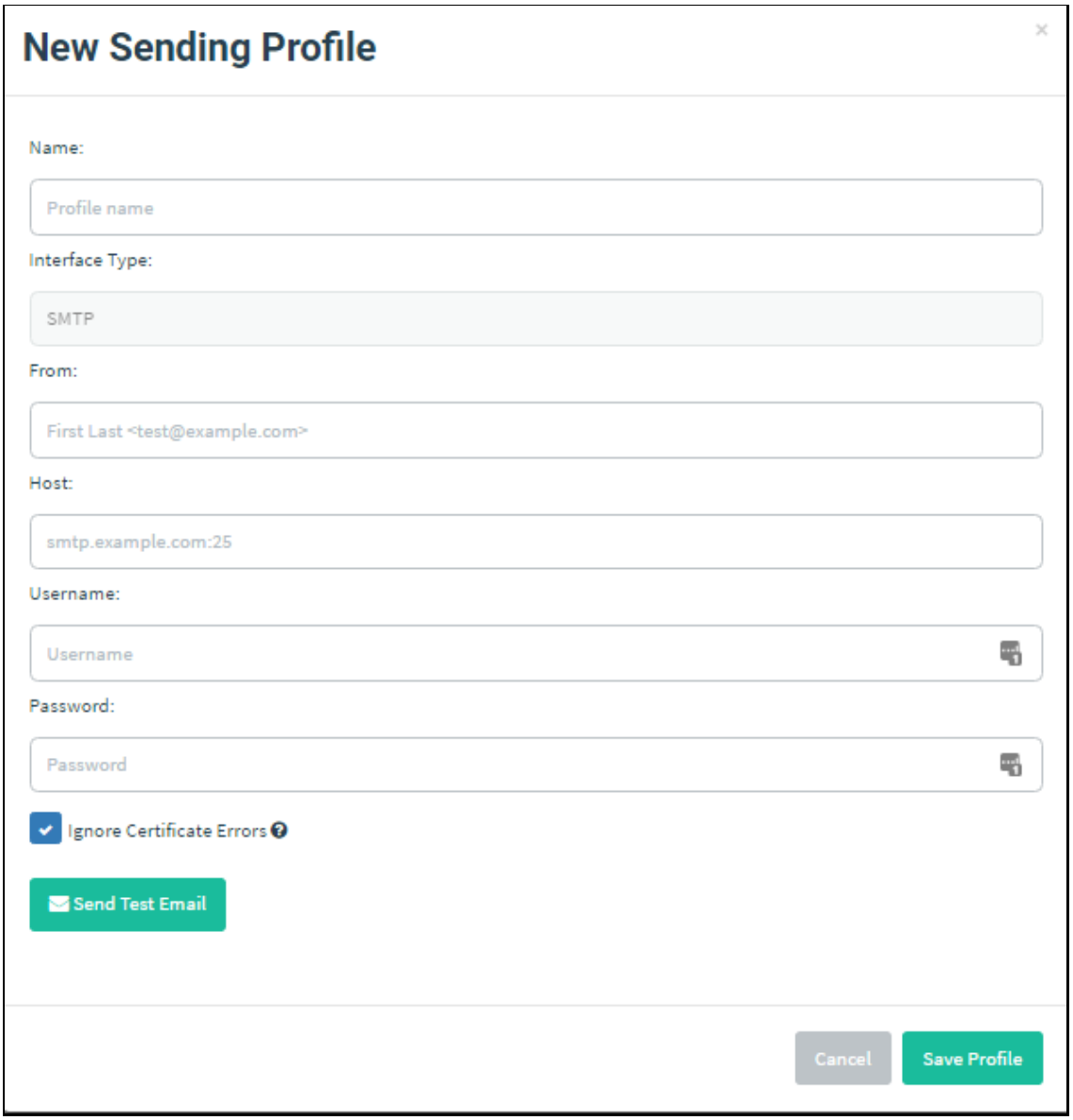
Finally, make sure the "Add Tracking Image" checkbox is checked, and click "Save Template".

Setup a Sending Profile

This portion of the setup may be the most difficult part to get right, as many mail relays now check for masquerading traffic. You may need to contact your mail provider or whitelist this application on your SMTP server.

To send emails, GoPhish requires you to configure SMTP relay details called "Sending Profiles". To setup a sending profile, click the "Sending Profiles" navigation entry in the sidebar and click the "New Profile" button.

Note: If you're looking for a good testing SMTP server, GoPhish recommends Mailhog, but you can try using your exisiting SMTP server or Microsoft or Google's however most times they will block sending an email that is masquerading itself as a person that is a different sender.



Sending Profile Dialog

Sending Profiles

It's important to make sure that your "From" address is a valid email address format. Additionally, make sure you setup your "Host" in the full host:port format.

To test your SMTP configuration, you can click the "Send Test Email" button:

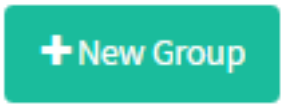
Send Test Email Dialog

After entering the recipient details and clicking "Send", you should see a message indicating if the email was sent successfully.

Adding Sending Groups

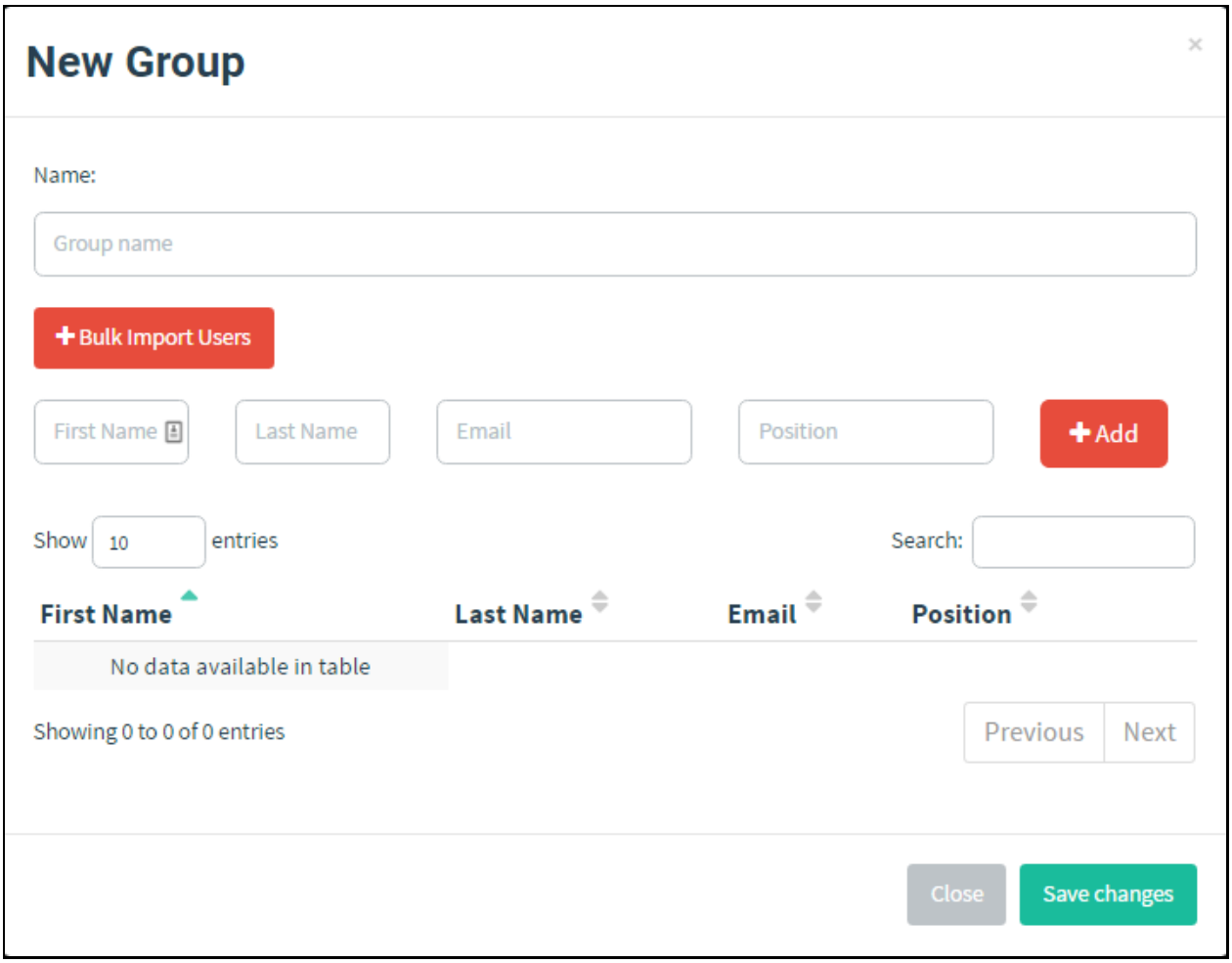
This portion is very easy to do, these will be the groups of people **receiving** the phish from you.

Creating Groups

To create a group, first navigate to the "Users & Groups" page in the navigation menu and click the button 

.

You will see the following dialog appear:



New Group Dialog

To add a group, you need to specify a *unique* group name, as well as at least one recipient.

Adding Users to the Group

You can add the users to the group in two ways:

Manually Adding Users

To add users manually, fill in the text boxes for "First Name", "Last Name", "Email", and "Position" and click the "Add" button.

Bulk Uploading Users

Adding users manually can be a pain. To fix this, GoPhish lets you upload users in bulk from a CSV file.

The CSV format GoPhish expects has the following header values:

- First Name
- Last Name
- Email
- Position

To upload a CSV with user information, click the "Bulk Import Users" button and select the CSV you want to upload. Users are then uploaded and displayed in the dialog.

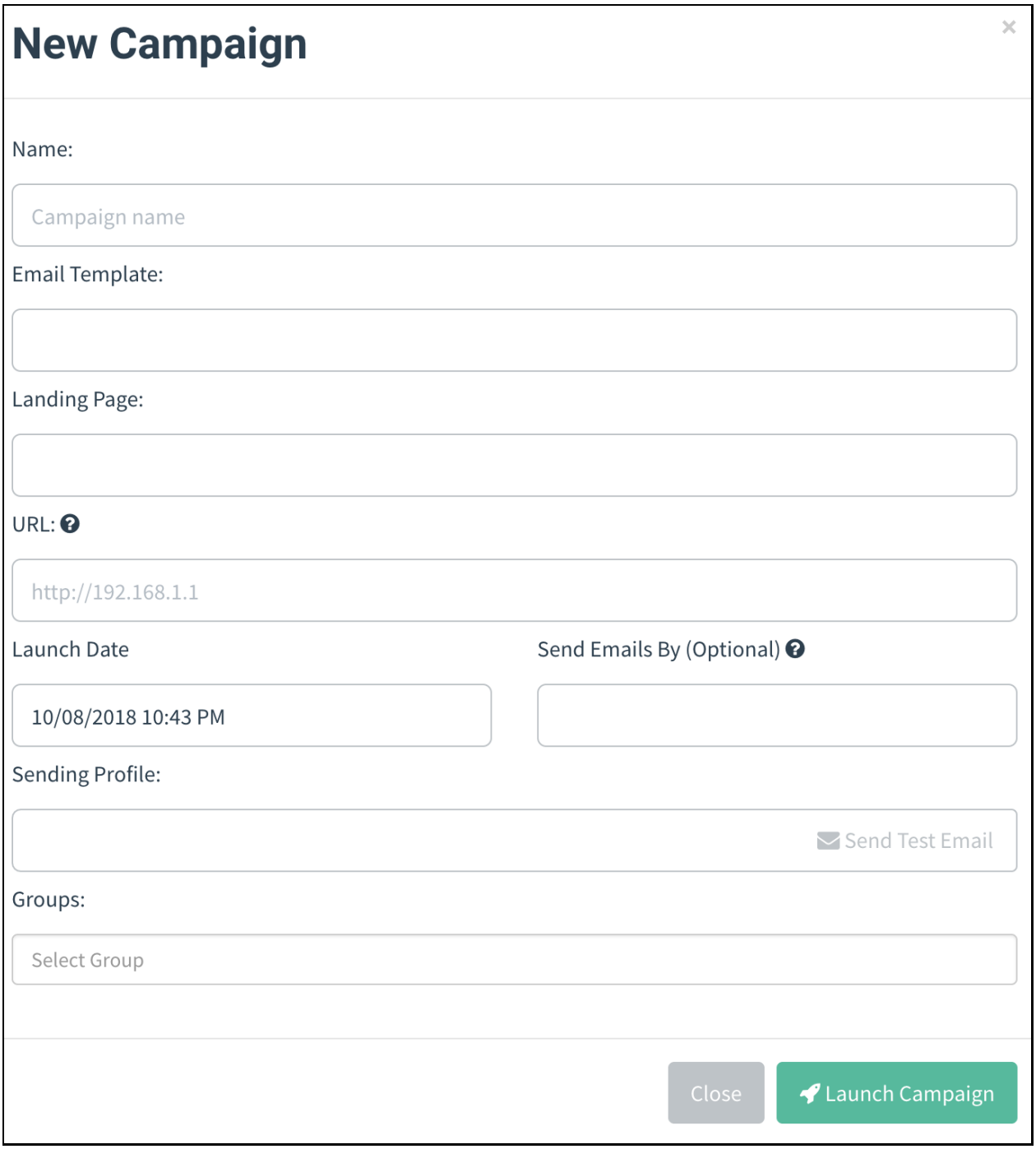
To save the group, click "Save changes".

Setting up a Campaign

A campaign is the “wrapper” for the rest of the elements that you have setup in previous steps. It is here that you choose which groups emails should be sent out to, as well as which email templates, landing pages, and sending profiles are used.

Launching a Campaign

To configure and launch a campaign, click the "Campaigns" entry in the navigation sidebar.



New campaign dialog

Setting up a campaign requires the following fields to be provided:

- **Name** - The name of the campaign
- **Email Template** - The email that is sent to campaign recipients. This is created in the Email Templates section of the documentation.

- **Landing Page** - The HTML that is returned when a recipient clicks the link in the email template. This is created in the Landing Pages section of the documentation.
- **URL** - This is the URL that populates the {{.URL}} template value, commonly used in email templates. This should be a URL or IP address that points to the GoPhish phishing server and is reachable by the recipient.
- **Launch Date** - This is the date that the campaign will begin. See Scheduling Campaigns for more information.
- **Send Emails By** - This is the date all emails will be sent by. See Scheduling Campaigns for more information.
- **Sending Profile** - This is the SMTP configuration to use when sending emails. This is created in the Sending Profiles section of the documentation.
- **Groups** - This defines which groups of recipients should be included in the campaign.

Scheduling Campaigns

GoPhish supports scheduling campaigns, making it easy to plan campaigns in advance. There are two fields to consider when scheduling campaigns: the **Launch Date** and the **Send Emails By** date.

The **Launch Date** is when GoPhish should start sending emails. By default, GoPhish assumes you want the campaign to be launched immediately.

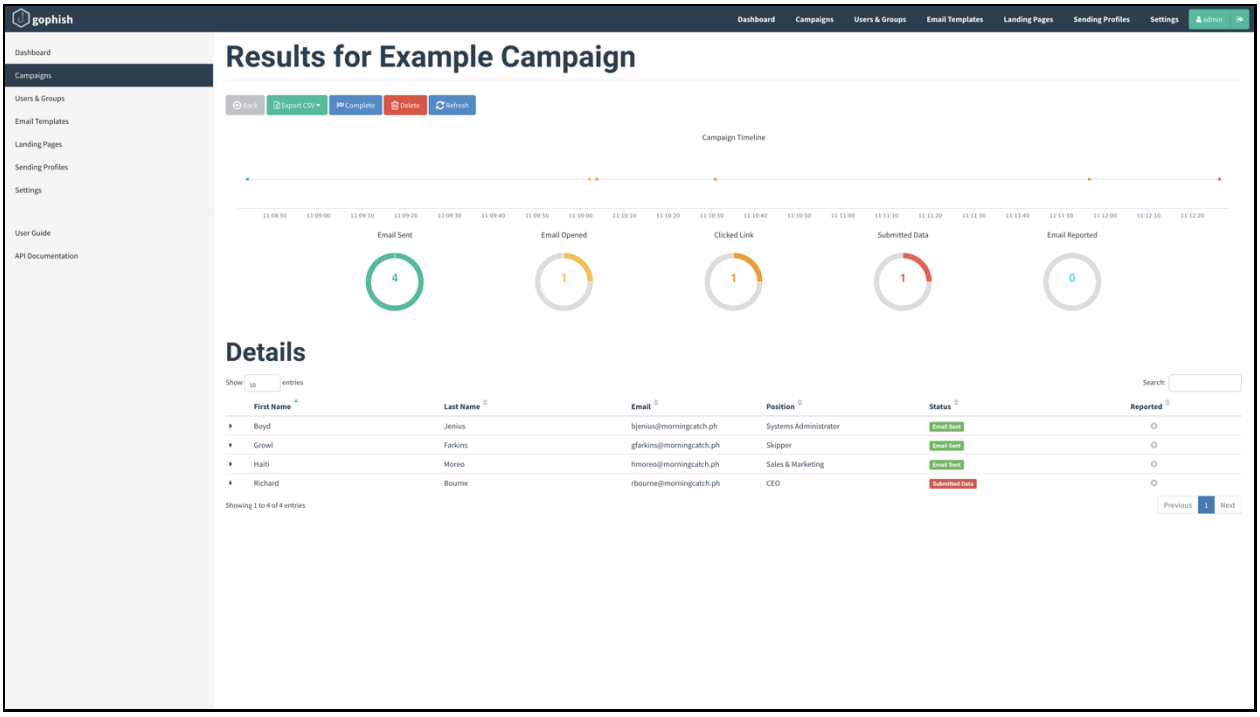
GoPhish also assumes that you want all emails to be sent immediately after the campaign is launched, and to be sent as quickly as possible. However, there are times where you may wish to spread the emails over a period of time. Setting the **Send Emails By** date tells GoPhish to spread emails evenly between the launch date and this date.

Launching the Campaign

After you have the campaign configuration ready to go, click the "Launch Campaign" button, click through the confirmation message, and you're good to go! Depending on your scheduling settings, GoPhish will either launch the campaign immediately or will schedule the campaign to be launched at a later date.

Viewing Campaign Results

When a campaign is launched, you are automatically redirected to the campaign results screen:



On the results page, you will see overview information on the campaign status as well as detailed results for each target.

Exporting Campaign Results

To export campaign results in CSV format, click the "Export CSV" format and select the type of results you want to export:

- **Results** - The current status for each target in the campaign.

  Contains the following fields:

  id, email, first_name, last_name, position, status, ip, latitude, longitude

- **Raw Events** - Contains a stream of events as they occurred during the campaign.

Completing a Campaign

To complete a campaign, click the "Complete" button and confirm that you want to mark the campaign as completed.

Deleting a Campaign

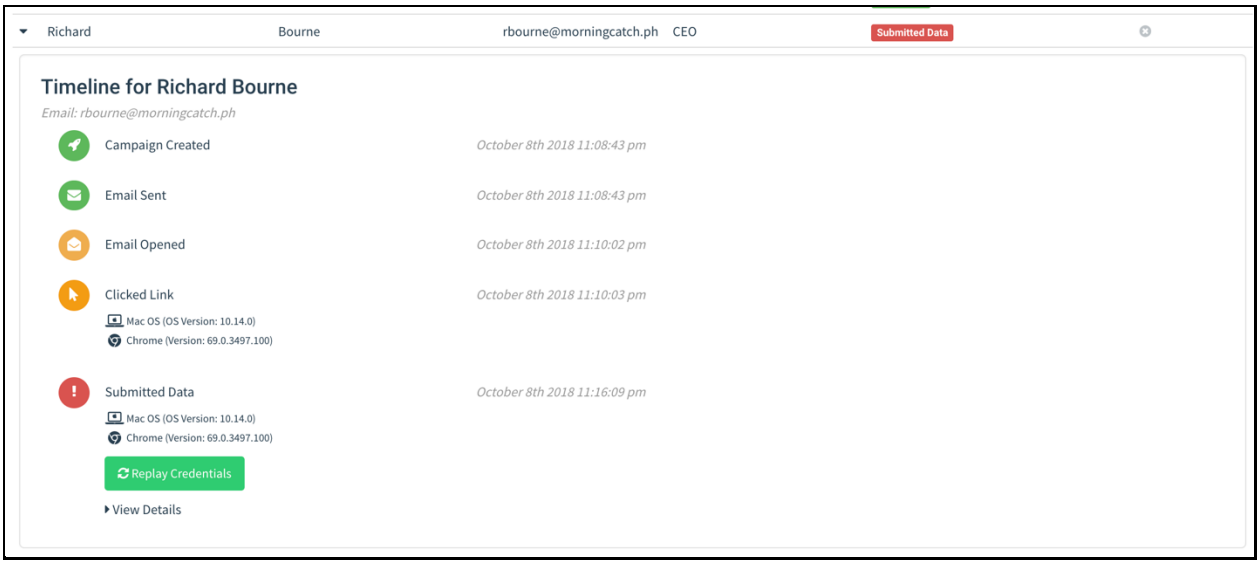
To delete a campaign, click the "Delete" button and confirm that you want to delete the campaign.

Note: This **cannot** be undone, so be careful when deleting a campaign!

Viewing Result Details

GoPhish makes it easy to view the campaign results in a timeline format.

To view the timeline for each recipient, expand the row with the recipient's name.



The results pane shows what a campaign recipient did, such as opening the email, clicking the link, or attempting to submit data from the landing page.

GoPhish also records information about the device that clicked the link or submitted data. This data is parsed from the browser's user-agent string. The operating system and browser version is displayed below the event details.

Viewing Captured Credentials

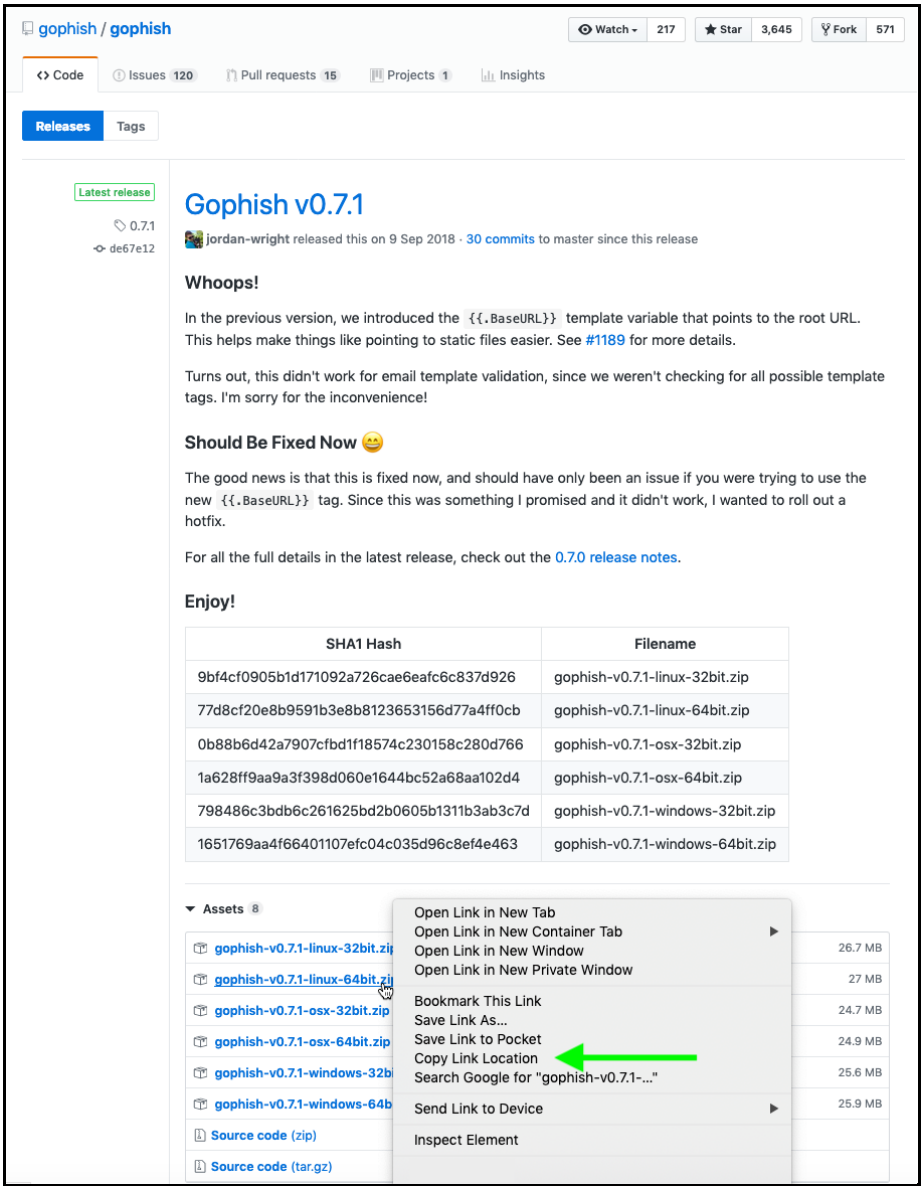
If you selected the "Capture Credentials" option when building a landing page, GoPhish displays the credentials in the results pane. To view them, click the "View Details" dropdown which renders the captured credentials in a table.

Linux Instructions (Advanced Users)

The instructions here can be used on On-Premise or Cloud-Based servers.

Instructions based on https://medium.com/airwalk/practical-phishing-with-GoPhish-7dd384ad1840

Software installation
Once logged on to the instance via ssh, I needed to download a binary package for GoPhish. In the browser on my laptop, I went to the GoPhish releases page on GitHub and found the URL for the 64-bit Linux distribution by right-clicking (your browser experience may differ slightly).



Back on the instance, create a user called GoPhish and downloaded the binary:
sudo adduser -c “GoPhish user” GoPhish
sudo su — GoPhish
curl -L https://github.com/gophish/gophish/releases/download/0.7.1/gophish-v0.7.1-linux-64bit.zip -o gophish-v0.7.1-linux-64bit.zip
Please note:

- Do not simply copy paste URLs above as there may be a newer version available since writing
- Remember to use -L as the URL on the webpage doesn’t link directly to the binary, your curl will have to follow redirects

Create a directory for this version of Gophish and unzipped:
mkdir gophish-v0.7.1
cd gophish-v0.7.1/
unzip ../gophish-v0.7.1-linux-64bit.zip
rm ../gophish-v0.7.1-linux-64bit.zip

To make life easier with future upgrades and startup scripts, Make a symbolic link to the current version:

ln -s /home/gophish/gophish-v0.7.1 /home/gophish/gophish
Gophish Configuration
Next, I inspected the config.json using the nano editor:

```
{
   “admin_server”: {
      “listen_url”: “0.0.0.0:3333”,
      “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”: “”
```

admin_server -> listen_url

In order for the admin server to listen on our public IP address on TCP port 3333 (access to which was restricted to our office IP address in the security group rules above), I needed to update the admin_server -> listen_url to be 0.0.0.0:3333
I chose to leave everything else at the defaults. While this configured HTTPS for our interaction with the admin server (GoPhish creates this automatically), it still used unencrypted HTTP for any communication between email client and GoPhish server. This could be switched to use HTTPS but would require an SSL certificate signed by a trusted certificate authority, otherwise recipients' email clients would display a warning and give the game away. I therefore chose not to enable HTTPS now, as I would not be collecting data through forms, I would only be using the landing pages to display a message that this was a phishing test..

Startup Scripts

While I could have just started GoPhish in an interactive shell, my preference was to start it as a service, meaning it would start at system boot. For this, I referred to this issue, linked from the installation page.
I adapted the script slightly to match my installation location so it became:

```
#!/bin/bash
# /etc/init.d/gophish
# initialization file for stop/start of gophish application server
#
# chkconfig: — 64 36
# description: stops/starts gophish application server
# processname:gophish
# config:/opt/goapps/src/github.com/gophish/gophish/config.json# define script
variablesprocessName=Gophish
process=gophish
appDirectory=/home/gophish/gophish
logfile=/var/log/gophish/gophish.log
errfile=/var/log/gophish/gophish.errostart() {
 echo 'Starting '${processName}'…'
 cd ${appDirectory}
 nohup ./$process >>$logfile 2>>$errfile &
 sleep 1
}stop() {
 echo 'Stopping '${processName}'…'
 pid=$(/usr/sbin/pidof ${process})
 kill ${pid}
 sleep 1
}status() {
 pid=$(/usr/sbin/pidof ${process})
 if [[ "$pid" != "" ]]; then
 echo ${processName}' is running…'
```

```
 else
 echo ${processName}' is not running…'
 fi
}case $1 in
 start|stop|status) "$1" ;;
esac
```

Yes, it runs as root, but this is necessary to bind to port 80. You could configure your favorite webserver as a proxy in front of GoPhish, but this is left as an exercise to the reader.
As I now needed to take some steps with sudo privilege, I exited my shell which was running as the gophish user:
exit

I placed the above init script at /etc/init.d/gophish:
sudo vi /etc/init.d/gophish
(paste and save)
I changed the permission of this file to be a proper init script:
sudo chmod 744 /etc/init.d/gophish

I made GoPhish start on boot:
sudo chkconfig gophish on

I created the log directory for GoPhish that was referenced in the startup script:
sudo mkdir /var/log/gophish

I started Gophish:
sudo service gophish start

I checked the logs in /var/log/gophishand it seemed happy. I also checked the process was still up:
ps -ef |grep [g]ophish

From here you can follow the GoPhish instructions, as they are the same as the Windows version from the Web.

This document was created as part of a research project funded by:

OHIO CYBER RANGE INSTITUTE

UNLOCKING POTENTIAL,
SECURING THE FUTURE