Social engineering and psychological tools to launch a simple phishing attack

Abstract

Using the SET to create a clone of facebook login page and setup a login credential harvester. Further, we use this fake site to learn how mass mailing phishing attacks work. Further, we use psychology and social engineering concepts learned in class and apply them to create a convincing phishing email.

Aim

The aim of this lab was to demonstrate the tools available to attackers to launch social engineering attacks. Further, to also demonstrate the ease of use of this toolkit.

Introduction and Background

The focus of the lab is to use psychology insights learned in class to create a convincing phishing email that can be used to harvest login credentials(in this case Facebook) from unfortunate victims. To complete this task I try to aggravate victims emotions and give a simple logical solution which leads the said victim to click on the link to the fake website. To instill a sense of urgency, I add a deadline to imply the loss of their account. We have also learned in class that people are more likely to comply if an authority figure directs them to do so.

Methods

To begin with, we use the SET to create a phishing site by cloning a URL as a fake website. For this lab we use Facebook.com. After we have created the phishing website, we create a text based phishing email to lure victims to said website. First we set up Postfix as our own SMTP server. As an example we send this email to our own address. The SET allows us to set up Postfix as an open SMTP relay, which allows us to spoof our email and name.

Finally, we use an online html to create an actual email. As a template I used the security email google sends out whenever a new device logs in, as it is not a stretch to say majority of people have received this email and are familiar with it. This was to make the email seem more genuine and convincing,

Results and Observation

Opening the website, we observe that the cloning was successful and the fake website was virtually similar to the original. After entering login credentials are entered and login button is clicked, the harvester running in the terminal gets a hit. On reviewing the data on the terminal we see the username which was entered and the indication that a password was entered.

```
[*] WE GOT A HIT! Printing the output:
PARAM: jazoest=2938
PARAM: lsd=AVoCXNHvhe0
PARAM: display=
PARAM: enable_profile_selector=
PARAM: isprivate=
PARAM: legacy_return=0
PARAM: profile_selector_ids=
PARAM: return_session=
POSSIBLE USERNAME FIELD FOUND: skip_api_login=
PARAM: signed_next=
PARAM: trynum=1
PARAM: timezone=240
PARAM: lgndim=eyJ3IjoxMjgwLCJoIjo4MDAsImF3IjoxMjA4LCJhaCI6NzczLCJjIjoyNH0=
PARAM: lgnrnd=125401_P-M4
PARAM: lgnjs=1632340455
POSSIBLE USERNAME FIELD FOUND: email=kenilshahseng360lab1
PARAM: prefill_contact_point=
PARAM: prefill_source=
PARAM: prefill_type=
PARAM: first_prefill_source=
PARAM: first_prefill_type=
PARAM: had_cp_prefilled=false
POSSIBLE PASSWORD FIELD FOUND: had_password_prefilled=false
PARAM: ab_test_data=AAAAy/yy/ylAMAAAAMAAAAAAAAAAAAMAMAAAAAAAASc/JJAADARCAC
POSSIBLE PASSWORD FIELD FOUND: encpass=#PWD_BROWSER:5:1632340482:AeRQAL+OKFGC5WmLL6HL/Anc/V
```

After exiting the SET and looking at the generated report, we see the username and password that was entered. The username is shown in plain text, while the password is encrypted.

```
[09/22/21]seed@VM:~/setoolkit$ sudo more '/root/.set/reports/2021-09-22 15:56:00.046256.xml
'
<?xml version="1.0" encoding='UTF-8'?>
<harvester>
   login.facebook.com/login.php
   <url>      <param>----------------------------625289257120166847939498669 38</param>
   </url>
   <url>      <param>----------------------------111899487334661101454722396 9</param>
   </url>
   <url>      <param>jazoest=2938</param>
      <param>lsd=AVoCXNHvhe0</param>
      <param>display=</param>
      <param>enable_profile_selector=</param>
      <param>isprivate=</param>
      <param>legacy_return=0</param>
      <param>profile_selector_ids=</param>
      <param>return_session=</param>
      <param>skip_api_login=</param>
      <param>signed_next=</param>
      <param>trynum=1</param>
      <param>timezone=240</param>
      <param>lgndim=eyJ3IjoxMjgwLCJoIjo4MDAsImF3IjoxMjA4LCJhaCI6NzczLCJjIjoyNH0=</param>
      <param>lgnrnd=125401_P-M4</param>
      <param>lgnjs=1632340455</param>
      <param>email=kenilshahseng360lab1</param>
      <param>prefill_contact_point=</param>
      <param>prefill_source=</param>
```

Question Q1: What is an open SMTP relay? Why is it problematic?

Open SMTP relay is a type of SMTP server configured in a way which allows any user to send email through it, without any need for the sender to authenticate themself. The mail sent through such a server would be bounced around multiple machines on the internet before delivering it to the intended destination. This meant if a malicious user infected the mail with a computer worm, it would possibly infect all the machines on relay list. Additionally, the lack of a need to authenticate themselves, a user can easily spoof their identity and address, thus making spam emails or illegal activity virtually untraceable by the receiver.

While creating a convincing phishing email we choose the template google uses to report new logins to a person's google account. We start with mentioning that there was suspicious activity recorded on their account. To make the email look official we add fake information such as exact time and date of the activity recorded, operating system of the machine used in said suspicious activity, web browser, a random location in the world and the IP address of the machine used. Having the location be on the other side of a world, would aggravate the victim's emotion and right after we provide a logical solution and easy to follow solution to the problem they are currently facing. The solution being the link we attached, which redirects to the fake phishing website. To further nudge the victim to click the link, we try to convey a sense of urgency by mentioning they would lose access to the account in 24 hours if they don't click on the link.