Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE
PREMIER MINISTRE

S . G . D . S . N
Agence nationale de la sécurité des systèmes d'information
CERT-FR

Paris, le 18 avril 2014
N° CERTFR-2014-ACT-016

Affaire suivie par :
CERT-FR

BULLETIN D'ACTUALITÉ

Objet : Bulletin d'actualité CERTFR-2014-ACT-016

1 - Services accessibles depuis Internet

Le CERT-FR a récemment été amené à traiter des incidents relatifs à l'utilisation malveillante de services paramétrés par défaut et accessibles sans restriction depuis Internet.

Un service ouvert par mégarde et accessible sur Internet sans filtrage représente un risque élevé de compromission. En effet, les entités exposées sur Internet peuvent être la cible de balayages de ports exécutés de manière plus ou moins furtive. A ce titre, le CERT-FR constate au moins deux types de balayages de ports :

- les balayages massifs effectués généralement depuis une unique adresse IP au cours desquels des dizaines de milliers de paquets sont envoyés en une fraction de seconde sur de larges plages d'adresses. Ces balayages, bien que de très courte durée, peuvent être facilement identifiables et filtrés par les pare-feux installés à l'entrée du site ;
- les balayages silencieux réalisés sur une durée parfois très longue et éventuellement depuis plusieurs IP source. Ces balayages sont plus difficiles à détecter et à filtrer.

Dans les deux cas, l'objectif peut être de cartographier dans un premier temps les services accessibles (IP du serveur, port, bannière, etc.) et d'exploiter ensuite les services vulnérables comme ceux permettant l'administration sans authentification préalable ou encore ceux présentant des vulnérabilités non corrigées ou d'éventuelles portes dérobées à l'instar des récentes révélations portant sur les routeurs D-LINK.

Ces techniques sont également mises en œuvre pour rechercher des serveurs non protégés afin de lancer des dénis de services distribués. Les attaquants ciblent alors de préférence les ports liés aux services DNS, NTP ou encore CHARGEN. En effet, il a été constaté ces derniers mois une recrudescence d'attaques utilisant à leur insu les ressources réseau de systèmes mal configurés.

Le CERT-FR recommande aux administrateurs la vérification périodique de l'exposition de leur système d'information sur Internet et notamment de la configuration des services exposés afin d'éviter d'éventuels accès frauduleux. Des audits périodiques à l'aide d'outils dédiés permettront de mettre en exergue les services, stations de travail et serveurs accessibles depuis Internet et d'effectuer un filtrage pour limiter leur exposition.

Plus généralement, il convient de s'assurer que seuls les services nécessaires au bon fonctionnement d'un système sont activés et qu'ils sont en écoute sur les interfaces réseau adéquates. L'utilisation de parefeux limitant les connexions entrantes aux seuls services légitimement accessibles permet également de contrôler la surface d'exposition du système d'information sur Internet.

Documentation

- Déni de service par amplification CHARGEN :
  http://www.cert.ssi.gouv.fr/site/CERTA-2013-ACT-047/
- Déni de service par amplification NTP :
  http://www.cert.ssi.gouv.fr/site/CERTA-2014-ACT-003/
- Dénis de service par amplification DNS :
  http://www.cert.ssi.gouv.fr/site/CERTA-2013-ACT-014/

– Porte dérobée sur les routeurs D-LINK :
http://www.cert.ssi.gouv.fr/site/CERTA-2013-ACT-042/

2 - Uroburos et Windows 64-bit

Uroburos est un code malveillant entrant dans la catégorie des APT (Advanced Persistent Threat pour menace persistante avancée) qui a récemment défrayé la chronique et a donné lieu à la publication de plusieurs rapports techniques, dont ceux de G Data et BAE (voir liens en référence).

Uroburos est la variante la plus avancée d'une série de codes malveillants connus sous les noms Turla, Pfinet ou Snake, employés dans des attaques (parfois ciblées) depuis au moins 2006.

Il s'agit d'une véritable suite logicielle composée de multiples modules collaborant entre eux pour prendre le contrôle d'une machine infectée. Si cet article n'a pas pour vocation à détailler cette menace de manière exhaustive, il est cependant intéressant de se pencher sur un mécanisme précis employé par Uroburos : son mode de chargement sur les systèmes Windows 64 bits.

Uroburos comprend un pilote (driver) en charge de la furtivité (un rootkit) qui existe en versions 32 bits et 64 bits. Selon le type de système sur lequel il s'installe, Uroburos charge une version 32 bits ou 64 bits adaptée au noyau de Windows.

Or, depuis Windows XP, Microsoft requiert que les pilotes chargés dans le noyau 64 bits de Windows soient tous signés cryptographiquement. Cela permet d'une part de limiter le nombre d'éditeurs logiciels susceptibles de produire des pilotes acceptés par Windows et d'autre part d'identifier les responsables d'éventuelles instabilités du système d'exploitation dues à des pilotes bogués.

Soucieux de leur anonymat, les auteurs d'Uroburos n'ont pas jugé bon de signer leur pilote 64 bits. Afin de parvenir malgré tout à charger leur pilote en mémoire, ils emploient un moyen détourné : un second pilote, légitime et signé, est tout d'abord chargé. Ce pilote présente la particularité d'être vulnérable à une attaque qui permet de désactiver complètement la vérification des signatures. Le pilote malveillant non signé peut alors être chargé à son tour.

Le pilote vulnérable VBoxDrv.sys fait partie d'une ancienne version de VirtualBox, un produit de virtualisation d'Oracle. La vulnérabilité, exploitée par Uroburos, avait été identifiée par Core Security en 2008. Elle permettait une élévation de privilèges par le biais de requêtes mal formées depuis le mode utilisateur vers le pilote (CVE-2008-3431).

Plus précisément, l'exploitation de la faille se déroule en plusieurs étapes.

Tout d'abord le pilote VirtualBox vulnérable est écrit sur disque, sous le faux nom "usbehub.sys", et est chargé en mémoire. Puis un petit exécutable établit une connexion avec le pilote en profitant d'un défaut d'authentification.

Une fois la connexion établie, une série de requêtes vers le pilote exploite une vulnérabilité de conception (absence de validation des paramètres) aboutissant à une primitive permettant des écritures arbitraires.

La primitive d'écriture est utilisée pour mettre à zéro la variable globale `g_CiEnabled` du noyau, désactivant ainsi la vérification des signatures.

Notons au passage que la variable `g_CiEnabled` n'est pas exportée par le noyau, ce qui donne lieu à des manipulations complexes pour retrouver son adresse.

Si les astuces décrites ci-dessus ne sont pas nouvelles (elles datent au moins de 2010), leur emploi met en exergue la capacité des auteurs d'Uroburos à utiliser des techniques d'exploitation avancées.

De manière générale on peut s'attendre à revoir la technique d'élévation de privilège en mode noyau par l'emploi de pilotes signés vulnérables.

Par définition, il est difficile de détecter un code malveillant une fois qu'il a pris pied en mode noyau puisque toutes les fonctionnalités du système et de la machine lui sont ouvertes pour assurer sa furtivité. Néanmoins il est possible que certaines traces demeurent en mémoire ou sur disque. Dans le cas d'Uroburos, un moyen simple de déterminer si la sécurité d'un noyau 64 bits a été compromise est de tenter d'y charger un pilote non signé. (Pour plus d'informations sur les pilotes et le noyau de Windows en général, visiter le site d'OSR cité en référence).

Une autre possibilité est de configurer Windows pour conserver la trace des pilotes chargés au démarrage du système. Pour cela, il suffit d'effectuer les opérations suivantes :

– lancer l'utilitaire msconfig ;
– aller dans le panneau "Démarrer" ;
– cocher l'option "Journaliser le démarrage" ;
– redémarrer la machine.

Un fichier "ntbtlog.txt" sera créé dans le répertoire `"C:\Windows"`. S'il contient une ligne similaire à la suivante :

```
Loaded driver \SystemRoot\$NtUninstallQ923283$\usbehub.sys
```

cela indique que le pilote vulnérable a été chargé et la machine est compromise. (A noter que les chiffres "923283" sont variables, il faut chercher un répertoire de la forme `"$NtUninstallQXXXXXX$"`.)

Avec l'outil Object Viewer d'OSR, on peut également chercher, dans la catégorie BaseNamedObjects, un événement nommé

```
shell.{F21EDC09-85D3-4eb9-915F-1AFA2FF28153}
```

Cet événement est créé par Uroburos à des fins de communication interne et indique que la machine est infectée.

Plus généralement, pour détecter certaines attaques fondées sur des pilotes signés vulnérables, il faut se méfier de la présence de fichiers isolés de leur environnement normal (par exemple, le pilote VirtualBox, alors que VirtualBox n'est pas installé sur la machine), renommés, ou situés dans des répertoires illégitimes ou cachés.

Malheureusement il semble impossible d'empêcher un pilote de se charger par le biais de l'enregistrement de son certificat en liste noire. De plus la date d'expiration du certificat d'un pilote de périphérique n'est pas prise en compte lors de son chargement. Dans le cas du pilote VirtualBox vulnérable, le certificat est périmé depuis 2010 mais le pilote se charge sans problème sous Windows 7 64 bits.

Pour détecter les fichiers et clés de registre spécifiques à Uroburos et cachés par le module de furtivité, il est possible d'utiliser un outil de détection de rootkit.

Le CERT-FR recommande l'emploi régulier de ce type d'outil d'analyse.

Documentation

- Rapport G Data
  https://public.gdatasoftware.com/Web/Content/INT/Blog/2014/02_2014/documents/GData_Uroburos_RedPaper_EN_v1.pdf
- Rapport BAE
  http://www.baesystems.com/what-we-do-rai/the-snake-campaign?r=ai
- Rapport deresz et tecamac
  http://artemonsecurity.com/uroburos.pdf
- Avis de Core Security
  http://www.coresecurity.com/content/virtualbox-privilege-escalation-vulnerability
- CVE de la faille dans VBoxDrv.sys
  http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-3431
- Site d'OSR
  https://www.osronline.com/

3 - Bon usage du pare-feu Windows

L'utilisation d'un pare-feu intégré aux postes de travail est indispensable afin de mettre en application le principe de défense en profondeur, qui consiste notamment à ne pas se limiter à la défense périmétrique. Ceci est d'autant plus important que les postes nomades sont de nos jours largement répandus. L'activation du pare-feu directement sur les postes de travail permet, en outre, de réduire le risque de compromission d'une machine à l'autre au sein de la même zone de confiance.

Ce type d'attaque, dite données d'authentification d'un compte disposant de privilèges d'administration, que ce soit par extraction de données en mémoire, par récupération du mot de passe d'un compte de service, du cache du domaine, etc.

Le pare-feu intégré à Windows a été entièrement réarchitecturé avec Windows Vista. Il offre dorénavant toutes les fonctionnalités attendues telles qu'un filtrage fin des ports réellement nécessaires ou un filtrage sur le trafic entrant et sortant. Il offre toutes les fonctionnalités attendues pour un pare-feu de poste de travail.

Outre son couplage fort avec le pilote IPsec de Windows, lui permettant de réaliser du filtrage selon différents critères liés à l'authentification (appartenance d'un utilisateur à un groupe Active Directory, authentification de la machine, etc.), le pare-feu Windows permet de segmenter suivant différents profils d'utilisation (privé, public, en domaine) et par type d'interface (réseau local, accès distant, sans-fil).

Il est également à noter qu'il est possible de définir des règles par service et, depuis Windows 8, pour les applications « modernes » (par exemple le Windows Store) et dans une certaine mesure des règles selon les *capabilities*. La configuration par défaut de ce pare-feu est différente concernant le trafic entrant et le trafic sortant :

- pour le trafic entrant, le comportement par défaut est de bloquer les connexions et d'appliquer ensuite une liste blanche pour les programmes (sous la forme d'une fenêtre présentée à l'utilisateur lorsqu'aucune règle n'est applicable) ;
- pour le trafic sortant, le comportement par défaut est d'autoriser les connexions et d'appliquer ensuite une liste noire.

Outre le filtrage entrant, rappelons l'importance d'opérer également un filtrage du trafic sortant et d'autoriser par une liste blanche les programmes sur les ports qu'ils utilisent et qui sont nécessaires. Ceci n'est évidemment pas une défense absolue, mais permet de complexifier l'exfiltration de données par d'éventuels programmes malveillants non exécutés avec des privilèges d'administration.

Les règles de bon usage veulent également que la journalisation soit activée. La journalisation peut être effectuée soit par fichier texte (fichier `pfirewall.log`), soit dans le journal de sécurité [1]. Notons également que le système doit être configuré pour sélectionner automatiquement le profil public lorsque le poste est connecté à un réseau inconnu. Ce profil public devra comporter le minimum de règles et idéalement uniquement de quoi monter un VPN vers le réseau interne (pour utilisation du serveur DNS interne, passage par le proxy Web, maintien en condition de sécurité minimal, etc.).

Une des difficultés lors de l'utilisation du pare-feu Windows provient de l'ordre d'application des règles définies. En effet, contrairement à la majorité des solutions de type pare-feu, ces règles ne sont pas ordonnées. Le critère de sélection repose sur la règle la plus adéquate, qui est choisie dans l'ordre préférentiel suivant :

1. Règles relatives aux services (autorisation ou blocage) ;
2. Règles relatives aux machines voulant réaliser une association IPsec ;
3. Règles pour lesquelles l'option « Substituer les règles de blocage » est activée (valable uniquement si IPsec est utilisé, car ce mécanisme est nécessaire pour valider l'authentification des utilisateurs ou des machines) ;
4. Règles correspondant à un blocage du trafic ;
5. Règles correspondant à une autorisation du trafic ;
6. Comportement par défaut pour le profil considéré (autorisation ou blocage).

Dès lors qu'une règle correspond à l'un de ces critères, elle sera appliquée. Ainsi, si deux règles pour un même programme et sur un port donné sont présentes et si l'une spécifie un blocage et l'autre une autorisation, la règle de blocage est prioritaire et sera appliquée, indépendamment de la politique par défaut. Pour illustrer ce fonctionnement et sur les pièges possibles, trois exemples sont détaillés ci-dessous.

Exemple 1

Prenons un service représenté par le processus hôte svchost.exe, où les règles définies sont les suivantes :

Action	Programme	Adresse locale	Adresse distante	Port local	Port distant	Service
Bloquer	svchost.exe	Tout	Tout	Tout	Tout	(non défini)
Autoriser	svchost.exe	Tout	Tout	Tout	Tout	`mon_service`

La règle relative au service `mon_service` est la première qui est appliquée et devrait impliquer une autorisation pour le service. Néanmoins, pour le service `mon_service`, c'est un blocage qui est effectif en pratique. Ceci est dû au fait que lors de la création de la règle d'interdiction pour `svchost.exe`, la règle s'applique par défaut à tous les programmes et services. Il n'est donc pas correct de créer une règle de blocage pour `svchost.exe` (des programmes malveillants l'utilisant dans certains cas pour se « cacher »), en pensant que l'autorisation sera réalisée au cas par cas pour des services.

Exemple 2

Soit les deux règles suivantes applicables au navigateur Internet Explorer en sortie :

Action	Programme	Adresse locale	Adresse distante	Port local	Port distant
Bloquer	Tout	Tout	Tout	Tout	Tout
Autoriser	`iexplore.exe`	Tout	Tout	Tout	Tout

En suivant l'ordre d'application des règles, la règle de blocage est analysée avant la règle d'autorisation, il y a donc un blocage du navigateur. Le piège est ici de croire qu'en raffinant une règle, elle sera prioritaire et donc appliquée.

1. http://msdn.microsoft.com/en-us/library/windows/desktop/bb309058(v=vs.85).aspx

Exemple 3

Dernier cas, les règles suivantes sont configurées en sortie pour le navigateur :

Action	Programme	Adresse locale	Adresse distante	Port local	Port distant
Autoriser	Tout	Tout	Tout	Tout	Tout
Autoriser	`iexplore.exe`	Tout	Tout	Tout	80/TCP

En pratique, le navigateur peut utiliser tous les ports et n'est pas limité au port 80/TCP. Toutes les règles présentes dans la configuration sont effectivement appliquées afin de déterminer le résultat final. La configuration du pare-feu Windows n'est donc pas chose simple dès lors que l'on souhaite réaliser un filtrage le plus fin possible. Comme tout logiciel de filtrage, la bonne mise en œuvre doit être validée grâce à des balayages de ports.

Documentation

– Recommandations pour la définition d'une politique de filtrage réseau d'un pare-feu : http://www.ssi.gouv.fr/fr/guides-et-bonnes-pratiques/recommandations-et-guides/securite-des-reseaux/recommandations-pour-la-definition-d-une-politique-de-filtrage-reseau-d-un-pare.html

4 - Rappel des avis émis

Dans la période du 11 au 17 avril 2014, le CERT-FR a émis les publications suivantes :

– CERTFR-2014-AVI-176 : Multiples vulnérabilités dans VMWare vSphere
– CERTFR-2014-AVI-177 : Vulnérabilité dans plusieurs produits McAfee
– CERTFR-2014-AVI-178 : Vulnérabilité dans plusieurs produits Sophos
– CERTFR-2014-AVI-179 : Vulnérabilité dans IBM AIX
– CERTFR-2014-AVI-180 : Vulnérabilité dans plusieurs produits F5
– CERTFR-2014-AVI-181 : Multiples vulnérabilités dans les produits VMware
– CERTFR-2014-AVI-182 : Multiples vulnérabilités dans F5 ARX
– CERTFR-2014-AVI-183 : Multiples vulnérabilités dans Oracle Database
– CERTFR-2014-AVI-184 : Multiples vulnérabilités dans les produits Oracle PeopleSoft
– CERTFR-2014-AVI-185 : Multiples vulnérabilités dans Oracle Java SE
– CERTFR-2014-AVI-186 : Multiples vulnérabilités dans Oracle Solaris
– CERTFR-2014-AVI-187 : Multiples vulnérabilités dans Oracle Virtualization
– CERTFR-2014-AVI-188 : Multiples vulnérabilités dans Oracle MySQL
– CERTFR-2014-AVI-189 : Vulnérabilité dans strongSwan
– CERTFR-2014-AVI-190 : Vulnérabilité dans Drupal
– CERTFR-2014-AVI-191 : Vulnérabilité dans les produits F5
– CERTFR-2014-AVI-192 : Vulnérabilité dans HP Network Node Manager I
– CERTFR-2014-AVI-193 : Multiples vulnérabilités dans Xerox FreeFlow Print Server
– CERTFR-2014-AVI-194 : Multiples vulnérabilités dans EMC Cloud Tiering Appliance

Durant la même période, les publications suivantes ont été mises à jour :
– CERTFR-2014-ALE-003-001 : Vulnérabilité dans OpenSSL (mise à jour de l'alerte.)
– CERTFR-2014-AVI-161 : Vulnérabilité dans plusieurs produits Cisco (mise à jour des systèmes affectés.)
– CERTFR-2014-AVI-174 : Vulnérabilité dans Juniper Junos (mise à jour des systèmes affectés.)

Gestion détaillée du document

18 avril 2014 version initiale.