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 21 mars 2014
N° CERTFR-2014-ACT-012

Affaire suivie par :
CERT-FR

BULLETIN D'ACTUALITÉ

Objet : Bulletin d'actualité CERTFR-2014-ACT-012

1 - Risques liés à l'entête HTTP « X-Forwarded-For »

L'adresse IP est utilisée, dans les protocoles réseau, pour identifier un hôte distant. Cependant, dans certaines circonstances il est possible d'usurper cette adresse. Sa seule utilisation à des fins d'authentification est donc à proscrire.

Dans leur fonctionnement interne, les serveurs Web renseignent une variable de session nommée REMOTE_ADDR qui contient l'adresse IP distante du client se connectant au serveur Web. C'est sur cette variable qu'une application web est susceptible de s'appuyer pour des actions telles que la journalisation, le contrôle d'accès, etc.

Lorsqu'un serveur mandataire inverse (« reverse proxy ») ou un répartiteur de charge (« load balancer ») sont présents, la valeur renseignée par le serveur web dans REMOTE_ADDR correspond à l'adresse IP du serveur intermédiaire, empêchant ainsi les applications hébergées sur un serveur terminal d'avoir l'adresse IP du client ayant effectué la requête.

Une des solutions qui s'est imposée de fait est l'ajout d'entêtes HTTP non standards par ces serveurs intermédiaires contenant l'adresse IP du client. Les noms les plus courants pour ces entêtes sont « X-Forwarded-For » et « X-Client-IP » (le préfixe « X » signifiant que ces champs ne sont pas standards). Lorsqu'un équipement intermédiaire reçoit une requête contenant déjà un entête « X-Forwarded-For », il ajoute l'IP du client à la suite des données déjà présentes dans le champ.

On retrouve ainsi dans cet entête toute la chaîne d'équipements traversée par un client avant d'arriver au serveur final.

Cependant un client peut forger une requête HTTP de son navigateur pour ajouter les entêtes cités précédemment. Or, certains modules Apache tels que `mod_rpaf` et `mod_remoteip` remplacent la valeur de REMOTE_ADDR avec la dernière valeur présente dans le champ X-Forwarded-For. Dans ce cas, un attaquant forgeant un entête X-Forwarded-For pourra tromper les applications effectuant de l'authentification en s'appuyant sur l'IP source renvoyée par Apache, voire exploiter une vulnérabilité sur une application ne traitant pas correctement cette donnée (vulnérabilité de type injection de code: SQL, HTML, etc.).

Étant donné la confiance limitée que l'on peut avoir en cet entête, le CERT-FR recommande de ne pas utiliser ce champ comme seul mécanisme d'authentification du client. Par ailleurs, étant donné qu'il s'agit d'un champ fourni par l'utilisateur, il doit être préalablement filtré (vérification syntaxique) avant utilisation.

2 - Dissimulation et protections de The Mask - Careto

Careto est la porte dérobée fonctionnant en mode utilisateur de la campagne The Mask. Elle permet de recueillir des informations système sur les postes compromis et d'exécuter du code arbitraire fourni par un serveur de C&C. Cette porte dérobée est utilisée pour déployer le code malveillant SGH, plus sophistiqué et fonctionnant principalement en mode noyau. SGH dispose de nombreux modules permettant entre autres d'intercepter le trafic réseau, les conversations Skype, les clefs de chiffrement (PGP, SSH, etc), et les fichiers de configuration (VPN, RDP, etc).

Careto est intéressant d'un point de vue technique car il utilise plusieurs mécanismes pour masquer sa présence.

Persistance de Careto

Pour ne pas éveiller les soupçons de compromission et garder un moyen de persistance sur le système, Careto détourne un objet COM utilisé par le programme `explorer.exe`. Cet objet COM est associé au CLSID ECD4FC4D-521C-11D0-B792-00A0C90312E1 et correspond à :

- `browseui.dll` sous Windows XP ;
- `explorerframe.dll` pour les versions ultérieures.

Pour effectuer le détournement, il modifie le chemin original de l'objet COM vers la bibliothèque malveillante SHMGR :

- code s'exécutant avec des droits Administrateur :
  - HKLM\Software\Classes\CLSID\{ECD4FC4D-521C-11D0-B792-00A0C90312E1}\InprocServer32
    - Default: %SystemRoot%\system32\shmgr.dll
- code s'exécutant avec des droits utilisateur ou UAC activée :
  - HKCU\Software\Classes\CLSID\{ECD4FC4D-521C-11D0-B792-00A0C90312E1}\InprocServer32
    - Default: %APPDATA%\Microsoft\shmgr.dll

Un détournement d'objet COM implique le chargement de l'objet original après le déploiement de la charge malveillante afin que cela n'affecte pas la stabilité du processus ciblé. Le chemin de l'objet COM original est alors sauvegardé en utilisant le CLSID E6BB64BE-0618-4353-9193-0AFE606D6F0C :

- {HKCU,HKLM}\Software\Classes\CLSID\{E6BB64BE-0618-4353-9193-0AFE606D6F0C}\InprocServer32
  - Default: %SystemRoot%\system32\{explorerframe.dll,browseui.dll}

Minimisation des indices de compromission sur disque

Le nom et la description de la bibliothèque malveillante sont renseignés dans un bloc de configuration (chiffré et compressé) du code responsable du téléchargement de la bibliothèque. Cela permet à l'attaquant de définir un nom de bibliothèque sur le disque différent pour chaque campagne d'espionnage. De plus, les informations fichiers (horodatage) sont recopiées depuis la bibliothèque `kernel32.dll` pour nettoyer ses traces.

La bibliothèque malveillante `shmgr.dll` embarque plusieurs modules compressés et chiffrés :

- `waiter{32,64}.jpg` : module principal, chef d'orchestre, récupère et distribue les commandes ;
- `chef{32,64}.jpg` : module de communication réseau ;
- `dinner{32,64}.jpg` : module spécifique au navigateur web.

Ces modules sont déployés en mémoire et ne sont jamais présents directement sur le disque de la victime.

Dissimulation en mémoire des modules malveillants

Pour dissimuler la présence du module principal dans le processus explorer.exe, le module SHMGR se supprime de la liste des modules chargés dans le processus. Pour cela, il retire son entrée (LDR_DATA_TABLE_ENTRY) de la liste doublement chaînée de LDR_DATA du PEB (Process Environment Block). Il minimise aussi ses traces en mémoire en remplaçant sa précédente entrée LDR_DATA_TABLE_ENTRY par celle du module original `explorerframe.dll` ou `browseui.dll` (qui est aussi chargé en mémoire).

Careto déploie son code malveillant et ses modules dans le processus d'une façon assez furtive. En effet, il charge en mémoire une bibliothèque système signée qui n'est pas déjà chargée dans le processus. Le but de l'opération est de remplacer les données de la bibliothèque signée (c'est-à-dire les sections `.text` et `.data`) par ses données malveillantes. Après avoir déployé son code dans des bibliothèques systèmes, SHMGR nettoie ses traces en mémoire à l'aide de la fonction UnmapViewOfFile. Cette technique permet de dissimuler et d'exécuter du code illégitime dans des zones mémoires considérées comme légitimes (bibliothèques systèmes signées). Cette méthode de dissimulation permet de tromper les outils qui analysent la mémoire des processus pour détecter des codes malveillants, tel que Volatility.

Camouflage des routines de détournement

Pour cacher sa présence, Careto est capable d'intercepter des fonctions de l'API de Windows. La méthode utilisée pour les routines de détournement de fonction (hooks IAT) est assez originale. En effet, en plus de modifier l'adresse de la fonction ciblée dans la table d'import (Import Address Table ou IAT) du module ciblé, Careto va ajouter le code de la fonction illégitime à la fin de la section .text (en mémoire) de la bibliothèque légitime. Cela permet de positionner le code malveillant dans la bibliothèque légitime qui contient la fonction originale interceptée.

Par exemple, une interception de la fonction GetSidSubAuthority (exportée par la bibliothèque `advapi32.dll`) dans le module `iertutil.dll` est présentée ci-dessous :

- ajout du code de la fonction illégitime hook_GetSidSubAuthority à la fin de la section `.text` de `advapi32.dll` ;
- remplacement de l'adresse de GetSidSubAuthority dans l'IAT du module `iertutil.dll` vers la fonction hook_GetSidSubAuthority.

Ce mécanisme trompe les logiciels qui détectent les routines de détournement (hooks IAT) car la fonction illégitime se trouve dans l'espace mémoire du module légitime (`advapi32.dll`, qui exporte la fonction GetSidSubAuthority originale).

Protections Anti-debug

La première protection, peut-être involontaire, est l'utilisation de la fonction CreateFile en omettant l'option FILE_SHARE_READ pour la lecture sur disque de son propre code malveillant. Ce code malveillant effectue cette opération pour se déployer en tant que bibliothèque système signée :

```
GetModuleFileNameW(NULL, pathCodeMalveillant, 0x104);
CreateFile(pathCodeMalveillant, GENERIC_READ | GENERIC_EXECUTE,
        NULL, NULL, OPEN_EXISTING, NULL, NULL);
```

L'omission de l'option FILE_SHARE_READ empêche le déploiement du code malveillant lorsque le fichier sur disque est déjà ouvert par une autre application (par exemple un débogueur).

Une deuxième protection est l'utilisation de l'API IsBadWritePtr sur une zone mémoire qui n'est pas réinscriptible. En effet, la documentation MSDN de Microsoft indique que l'utilisation de cette fonction avec un débogueur attaché provoque une exception STATUS_ACCESS_VIOLATION lorsque la zone vérifiée n'est pas réinscriptible. Cette protection empêche l'exécution du code malveillant lorsque celui-ci est débogué.

Le CERT-FR recommande de vérifier que les indicateurs de compromission identifiés par la société Kaspersky (Annexe 1 du rapport [1]) ainsi que les marqueurs de persistance décrits au paragraphe "Persistance de Careto" ne sont pas présents sur le parc informatique et de se référer à la note d'information du CERT-FR "Les bons réflexes en cas d'intrusion sur un système d'information" [3] en cas de suspicion de compromission.

Documentation

1. The Careto/Mask APT :
   http://www.securelist.com/en/blog/208216078/The_Careto_Mask_APT_Frequently_Asked_Questions
2. Careto Attack - The Mask :
   https://kc.mcafee.com/resources/sites/MCAFEE/content/live/PRODUCT_DOCUMENTATION/25000/PD25037/en_US/McA
3. Les bons réflexes en cas d'intrusion sur un système d'information :
   http://www.cert.ssi.gouv.fr/site/CERTA-2002-INF-002

3 - Risques liés aux formats Microsoft Office et recommandations

Office est à l'origine une collection d'applications de bureautique indépendantes publiée par Microsoft pour la première fois en 1989, contenant Word (.doc), Excel (.xls, développé à l'origine pour Macintosh) et PowerPoint (.ppt). À cette époque, les 3 formats de fichiers sont donc complètement indépendants.

Au fur et à mesure des développements, des fonctionnalités communes vont émerger, notamment l'utilisation de scripts VBA (« Visual Basic for Applications ») ou de composants OLE (« Object Linking and Embedding »), jusqu'à ce que ces différents formats soient regroupés au sein d'un même méta-format nommé CFB (« Compound File Binary »). Ce format est en réalité une partition FAT (« File Allocation Table ») contenant des sous fichiers correspondants aux formats originels. L'avantage de cette structure réside dans son caractère évolutif, ainsi que

dans la possibilité d'intégrer différentes briques logicielles au sein du même fichier (par exemple, un conteneur ActiveX dans un fichier Word). Il est important de noter que le format CFB ne se restreint d'ailleurs pas qu'à Office, mais qu'il est également utilisé par exemple par les installateurs Windows (.msi), ou encore des miniatures d'images (Thumbs.db). De plus, il peut être embarqué dans d'autres types de formats de fichiers très populaires, comme le format RTF (« Rich Text Format ») ou le format PDF (« Portable Document Format », voir Bulletin d'actualité CERTFR-2014-ACT-004). Par ailleurs, il peut à l'inverse embarquer lui-même d'autres formats, comme par exemple des fichiers Flash (SWF, voir Bulletin d'actualité CERTFR-2014-ACT-009) ou encore des images (JPG, PNG, BMP, etc.).

Il en résulte une surface d'attaque considérable, et ces formats constituent donc un vecteur d'attaque privilégié pour les attaquants, d'autant plus que malgré l'apparition depuis Office 2007 d'un nouveau format s'appuyant sur OpenXML (.docx, .xlsx, .pptx), le format CFB reste encore aujourd'hui très largement répandu, notamment pour des raisons de compatibilité entre les différentes versions de la suite logicielle. Par ailleurs, Microsoft annonce que plus d'un milliard de personnes utilisent aujourd'hui la suite Office.

L'analyse automatisée de ces formats par des outils de sécurité (par exemple des antivirus) n'est pas forcément aussi aisée que pour certains autres formats. En particulier, il s'agit d'un format dont la représentation est binaire, contrairement aux fichiers PDF par exemple, dont le contenu brut est du texte éditable de façon plus simple. De plus, les spécifications du format Office sont restées confidentielles jusqu'en 2008, date où Microsoft les a finalement publiées, sous certaines réserves (de nombreux cas rencontrés en pratique ne sont en réalité pas documentés). Par ailleurs, la popularité du format a amené d'autres outils bureautiques (OpenOffice, LibreOffice, ou Pages par exemple) à prendre en compte ces formats, introduisant par ce biais d'autres subtilités, comme des malformations de fichiers par rapport aux spécifications strictes de Microsoft, mais sans pour autant présenter de caractère malveillant. Enfin, certains fichiers Microsoft comme les installateurs MSI ne respectent délibérement pas les spécifications du format. Toutes ces contraintes complexifient l'analyse des fichiers Office par les solutions de sécurité afin de détecter une attaque potentielle.

Exemple de vulnérabilité

La CVE-2012-0158 est une vulnérabilité qui a touché Microsoft Office 2003 SP3, 2007 SP2, 2007 SP3, 2010 Gold, 2010 SP1 et Office 2003 Web Components SP3. Elle permet à un attaquant d'exécuter du code à distance lors de l'ouverture d'un document piégé. La vulnérabilité se situe dans l'utilisation d'un objet ActiveX ListView ou TreeView présent dans la bibliothèque MSCOMCTL.OCX par l'intermédiaire de macro VBA par exemple.

Durcissement de la configuration de Microsoft Office et bonnes pratiques

Pour prévenir autant que possible les attaques exploitant les formats Office, le CERT-FR préconise d'utiliser les options suivantes afin d'améliorer la sécurité de Microsoft Office :

- désactivation des macros ;
- désactivation des contrôles ActiveX ;
- activation du « Mode Protégé » pour les documents provenant de l'Internet et de courriels ;
- activation du mode sans échec ;
- activation du mode de prévention d'exécution des données.

Comme pour tout autre type de fichiers, il est recommandé de n'accepter et de n'ouvrir que les documents provenant de personnes ou d'entités de confiance. Ainsi, la possible ouverture de fichiers embarqués dans ces documents ne sera réalisée que de manière ponctuelle et uniquement sur les fichiers pour lesquels cela semble être légitime et nécessaire. Il est donc essentiel de sensibiliser l'utilisateur quant aux risques que présentent les formats Office.

De plus, la mise à jour régulière des lecteurs de documents, des systèmes d'exploitation et des antivirus, ainsi que l'installation de mécanismes de sécurité tels qu'EMET permettront de réduire l'exposition aux failles applicatives des stations de travail. Enfin, la pré-visualisation de document dans l'explorateur de fichiers et dans Outlook, ainsi que l'utilisation d'ActiveX pourra être désactivée afin de réduire le risque de compromission.

Rappel des avis et alertes émis sur Microsoft Office

Depuis septembre 2013, la suite Microsoft Office a fait l'objet des avis suivants :

- http://www.cert.ssi.gouv.fr/site/CERTA-2014-AVI-014/CERTA-2014-AVI-014.html
- http://www.cert.ssi.gouv.fr/site/CERTA-2013-AVI-672/CERTA-2013-AVI-672.html

- http://www.cert.ssi.gouv.fr/site/CERTA-2013-AVI-670/CERTA-2013-AVI-670.html
- http://www.cert.ssi.gouv.fr/site/CERTA-2013-AVI-564/CERTA-2013-AVI-564.html
- http://www.cert.ssi.gouv.fr/site/CERTA-2013-AVI-519/CERTA-2013-AVI-519.html
- http://www.cert.ssi.gouv.fr/site/CERTA-2013-AVI-518/CERTA-2013-AVI-518.html

4 - Rappel des avis émis

Dans la période du 14 au 20 mars 2014, le CERT-FR a émis les publications suivantes :

- CERTFR-2014-AVI-126 : Multiples vulnérabilités dans le système SCADA Siemens SIMATIC
- CERTFR-2014-AVI-127 : Multiples vulnérabilités dans Spip
- CERTFR-2014-AVI-128 : Multiples vulnérabilités dans Moodle
- CERTFR-2014-AVI-129 : Multiples vulnérabilités dans Google Chrome
- CERTFR-2014-AVI-130 : Vulnérabilité dans les produits Huawei
- CERTFR-2014-AVI-131 : Multiples vulnérabilités dans Apache httpd
- CERTFR-2014-AVI-132 : Multiples vulnérabilités dans PHP
- CERTFR-2014-AVI-133 : Multiples vulnérabilités dans les produits Mozilla
- CERTFR-2014-AVI-134 : Vulnérabilité dans Cisco AsyncOS
- CERTFR-2014-AVI-135 : Vulnérabilité dans EMC Connectrix Manager
- CERTFR-2014-AVI-136 : Vulnérabilité dans nginx
- CERTFR-2014-AVI-137 : Multiples vulnérabilités dans le système SCADA Siemens SIMATIC
- CERTFR-2014-AVI-138 : Vulnérabilité dans Bluecoat Content Analysis System

Gestion détaillée du document

21 mars 2014 version initiale.