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 23 février 2015
N° CERTFR-2015-ACT-008

Affaire suivie par :
CERT-FR

BULLETIN D'ACTUALITÉ

Objet : Bulletin d'actualité CERTFR-2015-ACT-008

1 - Substitution de bibliothèque

Cet article décrit le fonctionnement de la technique de substitution de bibliothèque sous Windows et présente ensuite des recommandations pour s'en protéger.

Substitution de bibliothèque

La technique de substitution de bibliothèque a été rendue publique notamment à cause de son utilisation par l'outil malveillant PlugX. La charge est délivrée par les vecteurs classiques de distribution comme, par exemple une pièce jointe à un message électronique. Une fois que la victime a ouvert le fichier, la technique consiste à déposer sur l'ordinateur un exécutable et une bibliothèque.

L'exécutable déposé est un programme légitime. Il est spécialement choisi car il est vulnérable à la technique de substitution de bibliothèque. Une des bibliothèques qu'il essaye de charger est substituée par la bibliothèque précédemment déposée sur l'ordinateur, malveillante.

Cette attaque est rendue possible en raison de la façon dont Windows gère le chargement des bibliothèques. Si le nom de la bibliothèque n'est pas spécifié dans la clé de registre `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\KnownDLLs`, alors Windows essaye de la trouver en parcourant les répertoires suivants :

1. le répertoire où l'application réside ;
2. le répertoire où l'application est exécutée ;
3. le répertoire système ;
4. le répertoire système 16-bit ;
5. le répertoire de Windows ;
6. les répertoires spécifiés dans la variable d'environnement `PATH`.

L'ordre dans lequel Windows cherche une bibliothèque dans ses répertoires est bien défini. Dès qu'une bibliothèque est trouvée, la recherche se termine. Ainsi, une bibliothèque malveillante peut se substituer à une bibliothèque légitime. Pour ce faire, il suffit que la bibliothèque malveillante porte le même nom et qu'elle soit placée dans un dossier examiné avant le dossier qui contient la bibliothèque légitime. Cette substitution permet alors l'exécution de code arbitraire dans le contexte de l'exécutable.

Détection d'une substitution de bibliothèque

Le but de ce type d'attaque est de se dissimuler un maximum, aussi bien au niveau du système que de l'utilisateur. N'étant pas malveillant, l'exécutable déposé n'attire pas l'attention. Il est même parfois choisi parmi des listes bien spécifiques, comme par exemple la liste NIST [2], qui référence des logiciels considérés comme légitimes. L'attaquant peut également choisir une application dont le certificat apparaît de confiance pour l'utilisateur.

Le fichier malveillant peut donc s'exécuter dans le contexte du processus légitime. Il acquiert les niveaux de privilèges qui étaient octroyés à l'application légitime ainsi que son niveau de confiance. Le chargement du code malveillant présent dans la bibliothèque permet ensuite la compromission.

Il est donc plus aisé de détecter ce genre d'attaque en identifiant une bibliothèque malveillante. Si un doute subsiste sur l'authenticité d'une bibliothèque, il est conseillé de la comparer avec la bibliothèque originale. Il est important d'en maîtriser sa provenance : celle-ci doit se trouver sur le support authentique de l'application, qui a permis son installation.

Utilisation de mécanismes Windows

Développeurs

Windows permet aux administrateurs et aux développeurs de redéfinir l'algorithme de recherche de bibliothèques. Si la clé de registre `HKLM\System\CurrentControlSet\Control\Session Manager\SafeDllSearchMode` n'est pas activée, alors l'ordre est celui mentionné au chapitre précédent. Sinon, le répertoire où l'application est exécutée prend la 5ème position.

Les développeurs peuvent aussi redéfinir cet ordre de parcours durant l'exécution du programme grâce à la fonction `SetDllDirectory`. Appelée avec une chaîne vide, elle réduit le nombre de dossiers susceptibles de pouvoir héberger une bibliothèque malveillante, en enlevant le répertoire où l'application est exécutée.

Afin de mieux contrôler les bibliothèques qui sont chargées lors de l'utilisation d'un programme, les développeurs peuvent utiliser des manifestes. Ils permettent entres autres d'exprimer toutes les dépendances d'un exécutable. Les développeurs peuvent donner plus de détails sur les bibliothèques qu'ils désirent charger. Les manifestes donnent la possibilité d'utiliser des moyens de contrôle d'intégrité au chargement de la bibliothèque. Pour cela, il suffit d'indiquer une fonction de hachage ainsi que le condensat qui sera vérifié pour garantir que la bibliothèque est bien légitime.

Administrateurs

Les administrateurs du système peuvent mettre à jour l'entrée du registre `CWDIllegalInDLLSearch` afin d'influencer l'algorithme de recherche des fonctions `LoadLibrary` et `LoadLibraryEx`. Deux options sont envisageables :

- la clé située dans `HKLM\SYSTEM\CurrentControlSet\Control\Session Manager` permet d'appliquer le changement à toutes les applications sur un ordinateur ;
- la clé `HKLM\Software\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\<Nom de l'application>` permet de lister le nom des applications impactées.

Cette mesure ne peut être mise en oeuvre qu'à condition d'installer le HotFix proposé par Microsoft dans son bulletin [1].

Les administrateurs peuvent définir des règles sur l'utilisation des exécutables d'un système avec les stratégies de restrictions logicielles [3][4]. Cette fonctionnalité permet notamment de contrôler les applications qui s'exécutent sur l'ordinateur. Le programme Windows AppLocker [5] permet également une configuration de sécurité sur les applications.

Recommandations

L'ANSSI recommande aux développeurs :

- de préférer les chemins absolus aux noms des bibliothèques lors de l'utilisation de fonctions telles que `LoadLibrary` ou de spécification de bibliothèque dans les manifestes ;
- d'utiliser des manifestes pour les bibliothèques qui ne sont pas présentes dans la clé de registre des bibliothèques connues. Il est alors préférable d'indiquer une fonction de hachage ainsi qu'un condensat dans leur utilisation ;
- d'ajuster l'algorithme de recherche de bibliothèques, notamment avec la fonction `SetDllDirectory`.

L'ANSSI recommande aux administrateurs :

- de déployer des stratégies de restrictions logicielles ou de configurer l'application Windows AppLocker ;
- d'ajuster l'algorithme de recherche de bibliothèques, notamment grâce aux clés de registre évoquées dans l'article.

Références

1 http://support.microsoft.com/kb/2264107

2 http://www.nsrl.nist.gov/

3 https://technet.microsoft.com/fr-fr/library/hh831534.aspx

4 http://www.ssi.gouv.fr/entreprise/guide/recommandations-pour-la-mise-en-oeuvre-dune-politique-de-restrictions-logicielles-sous-windows/

5 https://technet.microsoft.com/fr-fr/library/hh831409.aspx

6 https://msdn.microsoft.com/en-us/library/windows/desktop/ms682586%28v=vs.85%29.aspx

7 http://binaryplanting.com/guidelinesAdministrators.htm

2 - Vulnérabilité UXSS dans Microsoft Internet Explorer

Le 1 février dernier, le chercheur en sécurité David Leo a annoncé sur la liste de distribution « *Full Disclosure* » l'existence d'une grave faille dans le navigateur Internet Explorer. Celle-ci s'est vu assignée l'identifiant CVE-2015-0072, et a fait l'objet de l'alerte CERTFR-2015-ALE-004. Elle n'est pas encore corrigée à ce jour. Il s'agit d'une exécution de script trans-origine universelle (« *Universal Cross-Site Scripting - UXSS* ») permettant de contourner un mécanisme de sécurité fondamental du Web : la politique appelée « *Same-Origin-Policy* » (SOP).

La SOP assigne une origine à chaque élément téléchargé depuis un site Web, et stipule que seuls les éléments possédant une même origine peuvent partager un contexte commun ou interagir entre eux. L'origine, définie précisément dans la RFC 6454, correspond en général au triplet (« protocole », « domaine », « port ») d'un URI.

Par exemple, la page Web `https://www.labanquebnt.fr/site/login.html` a pour origine le triplet ("https", "www.labanquebnt.fr", 443).

Il en est de même de la page `https://www.labanquebnt.fr/site/recapitulatif_compte.html` qui peut donc recevoir un cookie de session établi par la page précédente. Les scripts contenus ou importés dans une page Web héritent de l'autorité de cette page , et peuvent interagir avec les contenus partageant la même origine que la page.

En revanche, la page Web `http://www.mechantsite.net/index.html` ne peut interagir avec les pages précédentes, puisqu'elle ne possède pas la même origine.

Les vulnérabilités de type XSS (« *Cross-Site Scripting* ») permettent à un attaquant d'injecter du code JavaScript dans les pages d'un site Web tiers, se jouant ainsi de la séparation établie par la SOP. Ces vulnérabilités sont nombreuses, cependant chacune relève d'une erreur de programmation particulière dans le code source du site affecté. L'impact individuel d'une faille XSS classique est donc réduit à l'horizon d'un seul site.

La particularité de la faille CVE-2015-0072 est de permettre une injection de code JavaScript dans un site quelconque, d'où le qualificatif de XSS « universelle ». L'erreur ne réside pas dans le site ciblé, mais dans le navigateur Internet Explorer.

Le mode d'exploitation de la faille consiste à attirer un utilisateur sur une page Web maligne, par exemple par hameçonnage. La page contient tout d'abord deux `iframe`, l'un pointant vers le site de l'attaquant, et l'autre vers le site ciblé dont l'attaquant souhaite usurper l'origine. Le site de l'attaquant est configuré pour répondre à la requête provenant du premier `iframe` par une redirection vers le site ciblé, avec un léger délai. Puis vient un script qui s'exécute dans le contexte du premier `iframe` et effectue les actions suivantes :

- le script sauvegarde dans une variable temporaire le contexte du second `iframe` ;
- il effectue une action bloquante, par exemple en invoquant la fonction `alert()`, ou plus discrètement en faisant une requête `XMLHttpRequest` synchrone ;
- il redirige, à travers la variable temporaire, le second `iframe` vers une URL JavaScript qui injecte du code dans le site ciblé.

Le déroulement temporel de l'attaque est le suivant :

- Internet Explorer effectue la requête vers le site de l'attaquant correspondant au premier `iframe` ;
- Internet Explorer effectue la requête vers le site ciblé correspondant au second `iframe` ;
- Internet Explorer commence à exécuter le code JavaScript, jusqu'à la phase bloquante ;
- le site de l'attaquant répond à la première requête par une redirection ;
- Internet Explorer suit la redirection vers le site ciblé ;
- la phase bloquante expire et la fin du script s'exécute, désormais avec l'origine du site ciblé.

Le script s'exécute donc finalement dans le contexte du site ciblé, et peut effectuer toute action nuisible sur le site, comme de modifier son contenu. Il peut éventuellement accéder au cookie de session du site par la variable `document.cookie` et l'exfiltrer vers l'attaquant.

Sans présumer d'une connaissance fine des arcanes d'Internet Explorer, la faille semble venir du fait que le script, bien qu'il débute avec l'origine de la page de l'attaquant, est capable de lancer un script embarqué, qui lui s'exécute avec l'origine du site ciblé. Il s'agit d'une erreur logique dans l'implantation d'Internet Explorer, qui n'affecte d'ailleurs pas les autres navigateurs.

Contrairement aux nombreuses failles dues à des corruptions de la mémoire, vouées à une disparition progressive, voire totale, avec l'amélioration des méthodes de développement et des langages de programmation, il est probable que ce genre de bogue logique existera toujours dans les programmes complexes, et occupera donc une place importante dans le panorama futur de la sécurité informatique.

Des mesures de mitigations de la faille CVE-2015-0072, du côté client et du côté serveur, sont proposées dans la page de l'alerte CERTFR-2015-ALE-004.

Documentation

- http://cert.ssi.gouv.fr/site/CERTFR-2015-ALE-004/index.html
- http://tools.ietf.org/html/rfc6454

3 - Rappel des avis émis

Dans la période du 16 au 22 février 2015, le CERT-FR a émis les publications suivantes :

- CERTFR-2015-AVI-072 : Multiples vulnérabilités dans les produits IBM
- CERTFR-2015-AVI-073 : Multiples vulnérabilités dans les produits Cisco
- CERTFR-2015-AVI-074 : Multiples vulnérabilités dans PHP
- CERTFR-2015-AVI-075 : Vulnérabilité dans TYPO3
- CERTFR-2015-AVI-076 : Multiples vulnérabilités dans les produits Cisco

Gestion détaillée du document

23 février 2015 version initiale.