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 30 mars 2015
N° CERTFR-2015-ACT-013

Affaire suivie par :
CERT-FR

BULLETIN D'ACTUALITÉ

Objet : Bulletin d'actualité CERTFR-2015-ACT-013

1 - Vulnérabilité CVE-2015-0096 dans Microsoft Windows

Le 11 mars 2015, le CERT-FR a publié l'avis CERTFR-2015-AVI-096 concernant deux vulnérabilités dans Microsoft Windows (bulletin MS15-020).

L'une d'elles, référencée par l'identifiant CVE-2015-0096, a été signalée à ZDI par le chercheur en sécurité Michael Heerklotz en janvier 2015.

Elle a été corrigée par Microsoft lors de la distribution mensuelle des mises àjour, le 10 mars 2015.

Cette vulnérabilité est un reliquat de la vulnérabilité CVE-2010-2568, qui avait été exploitée par Stuxnet en 2009 et dont le correctif fourni par Microsoft dans le bulletin MS10-046 en août 2010 s'avère être partiel.

Détails de la vulnérabilité originale

L'exploitation de la vulnérabilité originale était déclenchée lors de l'ouverture, avec l'explorateur Windows, d'un dossier contenant un fichier .LNK malveillant (par exemple lors du branchement d'une clé usb).

Windows permet aux fichiers .LNK d'utiliser des icônes personnalisées au format .CPL.

Ce type de format est une bibliothèque de liens dynamiques (DLL), qui est donc capable d'exécuter du code.

Lors du chargement des icônes par l'explorateur Windows, un appel à la fonction "LoadLibrary" est réalisé sur le fichier .CPL et le point d'entrée de celui-ci est exécuté.

S'il contient du code, il est alors exécuté.

Premier correctif de Microsoft, août 2010

Le correctif décrit par le bulletin MS10-046 a modifié la fonction "CControlPanelFolder::GetUiObjectOf()" de la bibliothèque "shell32.dll" afin d'introduire une liste blanche des bibliothèques autorisées à charger des icônes personnalisées.

Problème du premier correctif

Lorsque la bibliothèque ne fait pas partie de la liste blanche, une chaîne de caractères de 554 octets au maximum est créée. Elle contient le chemin de la DLL, un identifiant icône (valeur '-1') ainsi que le nom de la DLL.

Lorsque la valeur de l'identifiant icône est différente de 0, la bibliothèque n'est pas chargée. Chacun de ces champs est séparé par une virgule.

Une vérification est ensuite effectuée sur le champ "wszModuleFullPath" afin de s'assurer qu'il ne contient pas de virgule, afin d'éviter une attaque par formatage de chaîne sur l'identifiant icône. Cette structure est ensuite utilisée lors de la création d'un objet de type "CCtrlExtIconBase". Le constructeur de la classe "CCtrlExtIconBase" est ainsi appelé :

```
01 int __stdcall CCtrlExtIconBase::CCtrlExtIconBase(unsigned __int16 *, char)
```

```
02 arg_0= dword ptr 8
03 arg_4= byte ptr 0Ch
04 mov edi, edi
05 push ebp
06 mov ebp, esp
07 push esi
08 mov esi, ecx
09 call CExtractIconBase::CExtractIconBase(void)
10 push [ebp+arg_0]
11 mov al, [ebp+arg_4]
12 and dword ptr [esi+214h], 0
13 or dword ptr [esi+218h], 0FFFFFFFFh
14 mov [esi+21Ch], al
15 push 104h~; taille de la nouvelle chaine
16 lea eax, [esi+0Ch]
17 push eax
18 mov dword ptr [esi], offset CCtrlExtIconBase@BIExtractIconA
19 mov dword ptr [esi+4], offset CDrvExtIconBase@BIExtractIconW
20 call StringCchCopyW(ushort *,uint,ushort const *)
21 mov eax, esi
22 pop esi
23 pop ebp
24 retn 8
```

Comme indiqué à la ligne 17 de ce code, la chaîne de caractères précédente est copiée dans un espace de 260 octets à l'aide la fonction "StringCchCopyW".

Si la taille de la première chaîne est supérieure à 259, alors elle sera tronquée (le dernier octet est réservé pour un '\x00'). Le programme cherche ensuite à retrouver la valeur de l'identifiant icône en localisant la première virgule puis en convertissant sa valeur en entier avec un appel à la fonction "StrToIntW". Si le chemin de la DLL possède une taille de 257 octets, la virgule sera présente à l'octet 258, elle sera suivie par le '-' provenant du '-1' et l'octet nul à l'adresse 260.

Ainsi pour déterminer l'identifiant icône, le programme fera à un appel à StrToIntW("-"), qui retourne 0. Le code de la bibliothèque sera donc chargé grâce à un appel à la fonction "LoadLibrary" et le code sera exécuté.

Le principe de liste blanche est ainsi contourné.

Second correctif de Microsoft, mars 2015

Dans le correctif du 10 mars 2015, plusieurs fonctions ont été modifiées :

```
; public: __thiscall CCtrlExtIconBase::CCtrlExtIconBase(void)
mov edi, edi
push esi
mov esi, ecx
call CExtractIconBase::CExtractIconBase(void)
and dword ptr [esi+460h], 0
or dword ptr [esi+464h], 0FFFFFFFFh
or dword ptr [esi+468h], 0FFFFFFFFh
xor eax, eax
mov [esi+0Ch], ax
mov dword ptr [esi], offset CCtrlExtIconBase@BIExtractIconA
mov dword ptr [esi+4], offset CDrvExtIconBase@BIExtractIconW
mov byte ptr [esi+46Ch], 0
mov eax, esi
pop esi
retn
??0CCtrlExtIconBase@QAE@XZ endp
```

Le constructeur de l'objet "CCtrlExtIconBase" ne contient plus l'appel à la fonction "StringCchCopyW". L'appel a été déplacé dans une nouvelle fonction :

```
01 __int32 __thiscall CCtrlExtIconBase::Initialize(CCtrlExtIconBase *this,
```

```
 const unsigned __int16 *, int, bool)
02 arg_0= dword ptr 8
03 arg_4= dword ptr 0Ch
04 arg_8= byte ptr 10h
05 mov edi, edi
06 push ebp
07 mov ebp, esp
08 mov al, [ebp+arg_8]
09 push [ebp+arg_0]
10 mov [ecx+46Ch], al
11 mov eax, [ebp+arg_4]
12 mov [ecx+464h], eax
13 push 22Ah~; taille 554
14 add ecx, 0Ch
15 push ecx
16 call StringCchCopyW
17 pop ebp
18 retn 0Ch
```

La taille de la nouvelle chaîne a été corrigée (ligne 13), elle est maintenant de 554 octets, taille identique à celle de la structure initiale.

Recommandations

Le CERT-FR recommande d'installer les mises à jour du bulletin MS15-020 dès que possible.

Documentation

- Bulletin de Microsoft à propos de la vulnérabilité CVE-2010-2568 : https://technet.microsoft.com/fr-fr/library/security/ms10-046
- Bulletin de Microsoft à propos de la vulnérabilité CVE-2015-0096 : https://technet.microsoft.com/en-us/library/security/MS15-020
- Référence CVE CVE-2015-0096 : http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-0096
- Avis du CERT-FR AVI-096 à propos de la vulnérabilité CVE-2015-0096 : http://www.cert.ssi.gouv.fr/site/CERTFR-2015-AVI-096/CERTFR-2015-AVI-096.html
- Analyse détaillée du premier correctif de Microsoft par HP : http://h30499.www3.hp.com/t5/HP-Security-Research-Blog/Full-details-on-CVE-2015-0096-and-the-failed-MS10-046-Stuxnet/ba-p/6718459

2 - Dernières publications sur le site de l'ANSSI

L'ANSSI a récemment publié deux nouveaux guides sur son site Web.

Le premier guide, fruit d'une collaboration entre l'ANSSI et la Confédération Générale des Petites et Moyennes Entreprises (CGPME), présente douze règles essentielles pour la sécurité des systèmes d'information des petites et moyennes entreprises.

Ces règles simples d'hygiène informatique sont accompagnées pour chacune d'entre elles d'un exemple inspiré de faits réels auxquels l'ANSSI a été confrontée.

Parmi ces règles on peut citer à titre d'exemples :

- choisir avec soin son mot de passe ;
- mettre à jour régulièrement ses logiciels ;
- sécuriser le WiFi de votre entreprise ;
- protéger ses données lors de ses déplacements.

En complément de ces douze règles, on trouve en fin de ce guide une liste de mesures afin de renforcer efficacement la sécurité et, en cas d'incident, les bons réflexes à adopter.

Le deuxième guide, rédigé en collaboration avec des sociétés et opérateurs de télécommunications, couvre les attaques par déni de service distribué ("Distributed Denial of Service" ou "DDoS") ainsi que les éléments à prendre en compte pour s'en protéger.

Il est destiné aux responsables de la sécurité des systèmes d'information des sociétés et organismes clients d'opérateurs de transit ou de fournisseurs d'accès à Internet, et cherchant à se protéger contre les attaques DDoS.

Ces attaques visant à rendre un ou plusieurs service(s) informatique(s) indisponible(s) sont aujourd'hui fréquentes car elles sont relativement simples à mettre en œuvre et efficaces contre une cible non préparée.

Parmi les vecteurs d'attaques couverts dans ce guide, on peut citer :

- les botnets ;
- les attaques basées sur la réflexion ;
- les attaques par amplification (DNS et NTP notamment) ;
- les attaques ciblant une application.

Les solutions de protection suivantes sont détaillées :

- filtrage en bordure de réseau de l'entité (équipements de type pare-feu ou spécifiques) ;
- protection externalisée (par l'hébergeur, par l'opérateur de transit, en ayant recours à un "Content Delivery Network" et enfin par des services de protection dédiés).

Documentation

- http://www.ssi.gouv.fr/actualite/petites-et-moyennes-entreprises-decouvrez-le-guide-des-bonnes-pratiques-de-linformatique-adapte-a-vos-besoins/
- http://www.ssi.gouv.fr/actualite/publication-du-guide-comprendre-et-anticiper-les-attaques-ddos/

3 - Rappel des avis émis

Dans la période du 23 au 29 mars 2015, le CERT-FR a émis les publications suivantes :

- CERTFR-2015-AVI-119 : Multiples vulnérabilités dans Mozilla Firefox
- CERTFR-2015-AVI-120 : Vulnérabilité dans IBM Rational ClearCase
- CERTFR-2015-AVI-121 : Multiples vulnérabilités dans le noyau Linux d'Ubuntu
- CERTFR-2015-AVI-122 : Vulnérabilité dans Cisco IOS
- CERTFR-2015-AVI-123 : Vulnérabilité dans Cisco WebEx Meetings Server
- CERTFR-2015-AVI-124 : Vulnérabilité dans Cisco IOS XR
- CERTFR-2015-AVI-125 : Multiples vulnérabilités dans les produits Cisco
- CERTFR-2015-AVI-126 : Vulnérabilité dans EMC Isilon OneFS
- CERTFR-2015-AVI-127 : Multiples vulnérabilités dans Citrix Command Center
- CERTFR-2015-AVI-128 : Multiples vulnérabilités dans le noyau Linux de RedHat

Gestion détaillée du document

30 mars 2015 version initiale.