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 15 février 2016
N° CERTFR-2016-ACT-007

Affaire suivie par :
CERT-FR

BULLETIN D'ACTUALITÉ

Objet : Bulletin d'actualité CERTFR-2016-ACT-007

1 - Bonnes pratiques pour les certificats - 2ème partie

Signature de code

La signature de fichiers s'appuie également sur l'utilisation de certificats pour assurer l'intégrité des données. Cette technique garantit à un utilisateur que le fichier n'a pas été modifié depuis son édition par son auteur. Cela concerne principalement des fichiers binaires (exécutables, librairies, pilotes), mais des scripts peuvent également être signés. En pratique, l'éditeur calcule un condensat du fichier, le signe avec sa clé privée et l'associe au fichier. La signature peut être stockée dans le fichier (signature embarquée, et dans ce cas la section* utilisée pour stocker la signature est exclue du calcul du condensat) ou être stockée dans un fichier annexe (fichier catalogue avec l'extension .cat). Exemple de commande pour la signature d'un script

```
C:\temp>signtool.exe sign /f Certificat.pfx /p MotDePasse dir.ps1
Done Adding Additional Store
Successfully signed: dir.ps1
```

L'outil signtool.exe fait partie du kit de développement Windows (Windows SDK) disponible sur le site Internet de Microsoft. Et voici le fichier dir.ps1 ainsi signé :

```
Get-ChildItem -Path "c:\"
# SIG # Begin signature block
# MIIFVAYJKoZIhvcNAQcCoIIFRTCCBUECAQExCzAJBgUrDgMCGgUAMGkGCisGAQQB
[...]
# SIG # End signature block
```

Lors de la vérification de la signature, l'ordinateur calcule le condensat selon la même méthode et déchiffre le condensat présent dans le fichier avec la clé publique de l'éditeur. Si les deux valeurs correspondent, la signature est valide. Il vérifie ensuite que le certificat utilisé pour signer le fichier est valide. La validité temporelle de la signature est limitée par l'expiration du certificat utilisé. Pour que celle-ci reste valide au-delà, on utilise une contre-signature. Celle-ci garantit que le certificat utilisé pour la signature était valide au moment de la contre-signature. Un fichier signé et contresigné restera valide au-delà de l'expiration de ces deux certificats. Sous Microsoft Windows, la vérification d'un fichier signé peut se faire via l'explorateur et les propriétés du fichier, onglet "Signatures numériques". On peut également utiliser l'outil signtool.exe pour la vérification. Par exemple :

```
C:\temp>signtool.exe verify /pa /v c:\Windows\System32\kernel32.dll
Verifying: c:\Windows\System32\kernel32.dll
Signature Index: 0 (Primary Signature)
Hash of file (sha256): 43299B19931529685EBE1ADE807047F57FF3AFDB2364A23D115FB7C79732BA60
[...]
Successfully verified: c:\Windows\System32\kernel32.dll
```

Messagerie sécurisée

La messagerie sécurisée met en oeuvre les trois aspects de la cryptographie à clé publique : la signature pour garantir l'intégrité et la non-répudiation de message et le chiffrement pour la sécurité des données. La norme décrivant ces mécanismes est connue sous le nom de S/MIME (Secure / Multipurpose Internet Mail Extensions). Pour la signature de message, l'émetteur utilise sa clé privée pour encoder un condensat du message et envoie le message, le condensat encodé et son certificat. Le destinataire calcule lui aussi le condensat du message et décode le condensat avec la clé publique présente dans le certificat de l'émetteur. Si les deux valeurs correspondent, la signature est validée. Pour le chiffrement, la clé publique du destinataire est nécessaire. L'émetteur a donc besoin de disposer au préalable du certificat de son interlocuteur. Du fait des ressources nécessaires pour un chiffrement asymétrique, il n'est pas possible de chiffrer de grande quantité de données de cette façon. Dans ces situations, c'est une clé symétrique qui est utilisée pour chiffrer les données et la clé publique du destinataire est utilisée pour chiffrer cette clé symétrique (le chiffrement par clé asymétrique nécessite entre 50 et 100 fois plus de ressources qu'avec une clé symétrique). L'émetteur envoie alors le message chiffré avec la clé symétrique, accompagné de la clé symétrique chiffrée avec la clé publique du destinataire. Le destinataire va alors déchiffrer la clé symétrique avec sa clé privée puis déchiffrer le message original.

Chiffrement de fichier

Le chiffrement de fichier Windows connu sous le nom système de fichier chiffré (Encrypted File System) utilise également l'architecture à clé asymétrique pour protéger les fichiers. Cependant, pour les mêmes raisons de performance que la messagerie sécurisée, les fichiers ne sont pas chiffrés directement avec la clé publique de l'utilisateur mais avec une clé symétrique, elle-même protégée par la clé privée de l'utilisateur. Pour partager entre plusieurs utilisateurs un fichier chiffré, la clé symétrique utilisée pour protéger le fichier sera chiffrée avec la clé publique de chacun des utilisateurs et associée au fichier. Les informations de chiffrement peuvent être consultées via l'explorateur (propriétés du fichier, onglet général, Avancé, Détails). On peut également utiliser l'outil cipher.exe.

```
C:\temp>cipher.exe /c C:\temp\chiffre.txt
Liste de C:\temp\
Les nouveaux fichiers ajoutes à ce repertoire ne seront pas chiffres.
E chiffre.txt
Niveau de compatibilite~:
Windows XP/Server 2003
Utilisateurs pouvant dechiffrer~:
MonOrdinateur\Utilisateur [Utilisateur(Utilisateur@ MonOrdinateur)]
Empreinte numerique du certificat~: F8D8 EA73 60E6 34C7 1755 E4F2 551E 518C A7B5 C777

Aucun certificat de recuperation trouve.
Informations sur la cle~:
Algorithme~: AES
Longueur de la cle~: 256
Entropie de la cle~: 256
```

Recommandations générales

Architecture

La mise en place d'une infrastructure à clés privées doit commencer par une phase de réflexion sur l'usage à l'origine de sa mise en place et les usages ultérieurs qui en seront fait. Il est compliqué de modifier une architecture en place, principalement en raison des risques d'interruptions de service. Une des premières questions à se poser concerne la réelle nécessité de monter une infrastructure d'autorités de certification par rapport à l'acquisition de certificats publics. Il convient ensuite de s'assurer que les différents éléments sont les plus adaptés. Les points principaux qu'il faut étudier avant la mise en place sont :

- Le nombre d'autorités. Une seule est la solution la plus facile à mettre en oeuvre et à administrer. Mais pour de grandes entités découpées en différents services avec des besoins multiples, il peut être intéressant d'avoir une autorité racine et plusieurs autorités émettrices.
- Les points de distribution des listes de révocation. Ces informations sont inscrites dans les certificats émis et leurs modifications imposent l'émission de nouveaux certificats pour remplacer ceux existants.

– Les serveurs de vérification en ligne de certificat ("Online Certificate Status Protocol" ou OCSP en anglais). Comme les points de distribution des listes de révocation, leurs adresses sont inscrites dans les certificats.

Administration

Tout comme il faut réfléchir à l'implémentation technique d'une telle solution, il faut également mettre en place un modèle de gestion organisationnel et définir les rôles et responsabilités de chacun. Ainsi, il est nécessaire d'identifier le ou les responsables du système d'exploitation du ou des serveurs autorités de certification. De la même manière, il faut identifier qui est autorisé à modifier la configuration des autorités de certification elle-même. Enfin, il faut aussi déterminer les personnes autorisées à demander les différents types de certificat mis à disposition.

Sauvegarde des autorités de certification

Comme tout élément d'un système d'information, une autorité de certification doit être sauvegardée. Celle-ci contiendra le certificat et la clé privée de l'autorité ainsi que la base de données des certificats et la configuration du service. Ces sauvegardes, sur média amovible de préférence, doivent être sécurisées au maximum, physiquement (dans un lieu sûr) et logiquement (par un mot de passe fort). Il convient également d'identifier les personnes habilitées à effectuer les sauvegardes et les restaurations.

Dimensionnement

Par défaut, tous les certificats émis sont stockés dans la base de données de l'autorité de certification. Cela permet par exemple de gérer leur révocation. La taille de la base de données influe sur le temps de démarrage et d'arrêt du service. Certaines configurations peuvent entrainer une augmentation importante de la taille de la base. Par exemple, des applications demandant à intervalle régulier des certificats, soit en raison d'une mauvaise configuration, soit parce qu'il s'agit de certificat à faible durée de vie. Pour le premier cas, il faudra vérifier rapidement la configuration mise en place, pour le second, il n'est sans doute pas nécessaire de les stocker dans la base de données des certificats. Dans tous les cas, il faut s'assurer de la cohérence de la taille de la base de données des certificats avec l'espace disque disponible.

Récupération des données

La fonctionnalité de chiffrement qu'apporte la cryptographie à clés asymétriques doit être considérée avec attention avant d'être utilisée. En particulier il faut étudier la capacité à récupérer des données chiffrées en cas de perte de la clé privée nécessaire à leur déchiffrement. Pour cela, différentes options sont possibles : sauvegarder les certificats et les clés privées une fois émis, mettre en séquestre les clés privées au moment de l'émission de certificat ou configurer un agent de récupération de données.

Sauvegarde de certificat et de la clé privée associée En sauvegardant le certificat, la clé publique et la clé privée associée sur un média externe, on s'assure de pouvoir la restaurer en cas de perte ou suppression de ces informations et ainsi permettre à nouveau l'accès aux données. Ces informations doivent être protégées physiquement et logiquement pour en interdire l'accès par un tiers. Pour sauvegarder le certificat et la clé privée d'un utilisateur, l'outil certutil.exe ou le cmdlet PowerShell Export-PfxCertificate peuvent être utilisés.

```
C:\temp>certutil.exe -user -exportPFX my 0 MySelf.pfx
my "Personnel"
================ Certificat 0 ================
Numero de serie~: d55abed631f2838044fef734a04ab523
Emetteur: CN=RootCA
NotBefore~: 05/02/2016 15:12
NotAfter~: 01/01/2040 00:59
Objet: CN=MySelf
Il ne s'agit pas d'un certificat racine
Hach. cert. (sha1)~: 70 54 25 5c f8 4e 33 47 32 5d ec c6 a0 dc 78 38 a8 c7 37 30
Conteneur de cle = 66268dc4-a188-41df-baf3-2932a86f1e3e
Nom de conteneur unique: a1e22c0144c0641467f63ae083ead3ea\_3fd28a33-6cc8-4828-8b22-759dc
Fournisseur = Microsoft Strong Cryptographic Provider
Le test de signature a reussi
Entrez un nouveau mot de passe pour le fichier de sortie MySelf.pfx~:
```

```
Nouveau mot de passe~:
Confirmer le nouveau mot de passe~:
CertUtil: -exportPFX La commande s'est terminee correctement.
PS C:\temp> Export-PfxCertificate -Cert Cert:\CurrentUser\My\7054255CF84E3347325DECC6A0D
-AsPlainText -Force)
Repertoire~: C:\temp
Mode
----a----

LastWriteTime
Length Name
------------------ ---05/02/2016 15:27
3382 Export-PfxCertificate.pfx
```

Profil itinérant et Informations d'identification itinérantes Sous Microsoft Windows, les certificats et clé privée des utilisateurs sont stockés dans leur profil. La mise en place de profil itinérant permettra leur récupération en cas de corruption de profil, mais ne protégera pas d'une suppression accidentelle. Un autre mécanisme permet également la sauvegarde de ces informations : Informations d'identification itinérantes ("Credential Roaming" en anglais). Celui-ci ne sauvegarde que les informations d'authentification et les stocke non pas sur un serveur de fichier mais dans des attributs confidentiels du compte utilisateur dans l'Active Directory. Ces deux solutions nécessitent néanmoins un système d'exploitation Windows et que les ordinateurs soient intégrés à un domaine Active Directory.

Agent de récupération de clés

En parallèle de la sauvegarde du certificat et de la clé privée, une fois attribués, il est également possible de mettre sous séquestre la clé privée ces éléments au moment de l'émission du certificat. Cela nécessite que l'autorité de certifications soit préalablement configurée pour le permettre. En pratique cela consiste en la création d'un agent de récupération de clé ("Key Recovery Agent" ou KRA en anglais). Cette entité dispose d'un certificat ayant l'usage "Agent de récupération de clés" (OID = 1.3.6.1.4.1.311.10.3.11) dont la clé publique est utilisée pour chiffrer la clé privée de tout nouveau certificat émis. Cette clé privée est stockée dans la base de données de l'autorité. Il faut ensuite configurer l'autorité de certification pour que la clé privée associée au certificat soit stockée dans le base de certificat, chiffrée avec la clé privée de l'agent de récupération. Ainsi, en cas de perte de clé privée, celle-ci peut être extraite du séquestre.

Agent de récupération de données

Une autre solution pour permettre l'accès aux données chiffrées en cas de perte des informations est la configuration d'un agent de récupération de données ("Data Recovery Agent" ou DRA en anglais). Cela se traduit par un certificat où l'usage est "Récupération de fichiers" (OID = 1.3.6.1.4.1.311.10.3.4.1). Cette fois encore, cette fonctionnalité doit avoir été configurée avant que des données ne soient chiffrées.

Lorsqu'un utilisateur chiffre des données, la clé symétrique utilisée pour effectivement protéger les données sera chiffrée avec la clé publique de l'utilisateur et également avec la clé publique de l'agent de récupération. Ainsi, il pourra, en cas de besoin, déchiffrer la clé symétrique et donc les données. Les informations d'agent de récupération peuvent être vérifiées soit via l'explorateur (propriétés du fichier, onglet général, Avancé, Détail, soit à l'aide de l'outil cipher.exe).

```
C:\temp>cipher.exe /c C:\temp\chiffre.txt
Liste de C:\temp\
Les nouveaux fichiers ajoutes à ce repertoire ne seront pas chiffres.
E chiffre.txt
Niveau de compatibilite~:
Windows XP/Server 2003
Utilisateurs pouvant dechiffrer~:
MyPC\Myself [Myself (Myself@MyPC)]
Empreinte numerique du certificat~: F8D8 EA73 60E6 34C7 1755 E4F2 551E 518C A7B5 C777
Certificats de recuperation~:
DRA(DRA@MyPC)
Empreinte numerique du certificat~: 5E15 9492 6ED1 419E 68D2 1FD4 DCA4 F622 126A 88CA
```

```
Informations sur la cle~:
Algorithme~: AES
Longueur de la cle~: 256
Entropie de la cle~: 256
```

Expiration de certificat

La date d'expiration d'un certificat rend son utilisation impossible. Cela peut avoir des conséquences importantes, comme l'échec d'une connexion HTTPS vers un site marchand ou d'accès à la messagerie (connexions SSL), l'impossibilité de se connecter au réseau (authentification 802.1x) ou de chiffrer des fichiers ou des messages. L'impact peut être encore plus important si c'est le certificat d'une autorité de certification qui a expiré. Dans ce cas, tous les certificats qu'elle a émis ainsi que ceux émis par d'éventuelles autorités enfants sont considérés invalides. Depuis Microsoft Windows 8 pour les clients et Microsoft Windows 2012 pour les serveurs, un nouveau composant permet d'être notifié d'événements relatifs aux certificats et en particulier leur expiration : Notifications du cycle de vie du certificat ("Certificate Services Lifecycle" en anglais) Deux nouveaux journaux d'événement tracent les opérations associées aux certificats, en particulier l'avertissement 1003 informant de l'expiration prochaine d'un certificat. On peut également utiliser PowerShell pour énumérer les certificats dont la date d'expiration est proche. Exemple :

```
PS C:\temp> Get-ChildItem -Path Cert:\LocalMachine\My\ -ExpiringInDays 7
```

Pour surveiller une architecture d'autorités de certification intégrées à l'Active Directory (dont les certificats et liste de révocation y sont publiées), il est également possible d'utiliser la console "PKI d'entreprise" qui vérifie la validité, l'accessibilité et l'expiration de tous ces éléments.

Expiration de liste de révocation

Tout comme un certificat, une liste de révocation possède une date d'expiration. Si celle-ci est dépassée, la liste est considérée invalide, et avec elle, tous les certificats émis par son autorité de certification émettrice. Il est donc impératif de surveiller les dates d'expiration des listes de révocation, en particulier pour celles qui ne sont pas publiées automatiquement ou qui nécessitent une action manuelle, comme une copie depuis l'emplacement de publication vers le point de distribution mentionné dans les certificats.

Annexes

- Introduction to Code Signing
  https://msdn.microsoft.com/en-us/library/ms537361(v=vs.85).aspx
- Windows Authenticode Portable Executable Signature Format
  http://download.microsoft.com/download/9/c/5/9c5b2167-8017-4bae-9fde-d599bac8184a/Authenticode_PE.docx
- SignTool
  https://msdn.microsoft.com/en-us/library/windows/desktop/aa387764(v=vs.85).aspx
- S/MIME Version 3.2 - Certificate Handling
  https://tools.ietf.org/html/5750
- S/MIME Version 3.2 - Message Specification
  https://tools.ietf.org/html/5751
- Enhanced Security Services for S/MIME
  https://tools.ietf.org/html/2634
- Using Encrypting File System
  https://technet.microsoft.com/en-us/library/bb457116.aspx
- Credential Roaming
  https://technet.microsoft.com/en-us/library/cc770797.aspx
- Using credential roaming
  https://technet.microsoft.com/en-us/library/cc773373(v=ws.10).aspx
- Certificate Services Lifecycle Notifications
  http://social.technet.microsoft.com/wiki/contents/articles/14250.certificate-services-lifecycle-notifications.aspx

2 - Mise à jour mensuelle de Microsoft

Le 09 février, lors de sa mise à jour mensuelle, Microsoft a publié treize bulletins de sécurité, dont sept considérés critiques et six importants :

- MS16-009 (critique) concernant Internet Explorer ;
- MS16-011 (critique) concernant le navigateur Edge ;
- MS16-012 (critique) concernant la gestion du format PDF ;
- MS16-013 (critique) concernant le journal Windows ;
- MS16-015 (critique) concernant Microsoft Office ;
- MS16-019 (critique) concernant le cadriciel .NET ;
- MS16-022 (critique) concernant le lecteur Flash Player d'Adobe ;
- MS16-014 (important) concernant Microsoft Windows ;
- MS16-016 (important) concernant WebDAV ;
- MS16-017 (important) concernant le pilote d'affichage de Remote Desktop ;
- MS16-018 (important) concernant le noyau de Windows ;
- MS16-020 (important) concernant Active Directory Federation Services ;
- MS16-021 (important) concernant le serveur NPS Radius.

Navigateurs

Cette mise à jour corrige treize vulnérabilités dans Internet Explorer dont neuf permettent une exécution de code à distance. La vulnérabilité CVE-2016-0041 concerne le chargement des fichiers DLL en mémoire par le navigateur. Un attaquant pourrait ainsi prendre le contrôle d'un système s'il parvient à faire exécuter à un utilisateur un programme dans un répertoire contenant une bibliothèque DLL malveillante. Les huit autres vulnérabilités permettant une exécution de code à distance sont causées par une mauvaise gestion de l'accès des objets en mémoire et peuvent être exploitées si l'utilisateur se rend sur un site piégé. La vulnérabilité CVE-2016-0059 peut potentiellement conduire à une fuite d'informations, car la bibliothèque Hyperlink Object ne cloisonne pas correctement son contenu en mémoire. Les vulnérabilités CVE-2016-0068 et CVE-2016-0069 peuvent être exploitées par un attaquant afin d'élever ses privilèges en contournant la politique inter-domaines. La vulnérabilité CVE-2016-0077 est provoquée par une analyse syntaxique incorrecte des réponses HTTP : un attaquant peut profiter de cette vulnérabilité pour induire un utilisateur en erreur, en lui faisant croire qu'il navigue sur un site légitime alors qu'il se trouve en réalité sur un site malveillant sous son contrôle. Edge reçoit six correctifs, parmi lesquels quatre permettent une exécution de code à distance. À noter que trois des vulnérabilités impactent également Internet Explorer (CVE-2016-0061, CVE-2016-0062 et CVE-2016-0077). La vulnérabilité CVE-2016-0080 peut permettre de contourner la protection de disposition aléatoire de l'espace d'adressage (ASLR). Bien que non-critique en elle-même, cette vulnérabilité peut être utilisée pour fiabiliser un code d'exploitation pour obtenir une exécution de code à distance. Le lecteur Flash Player d'Adobe pour les navigateurs de Microsoft bénéficie également de son lot de correctifs pour empêcher des exécutions de code à distance.

Bureautique

Six vulnérabilités de type corruption de mémoire ont également été corrigées dans Microsoft Office. Celles-ci sont susceptibles de permettre une exécution de code à distance lors de l'ouverture d'un fichier spécialement conçu. La vulnérabilité CVE-2016-0039 peut déboucher sur une injection de code indirecte à distance (XSS) permettant à un attaquant d'exécuter des scripts malveillants dans le contexte de sécurité de l'utilisateur ciblé. Les détails de cette vulnérabilité ont été publiquement révélés.

Windows

Le cadriciel .NET voit deux de ses failles de sécurité comblées. La première prend la forme d'un débordement de tampon qui survient lors de l'analyse syntaxique de fichiers XSLT (Extensible Stylesheet Language Transformations) malveillants, ce qui conduit à un déni de service. La seconde peut déboucher sur une atteinte à la confidentialité des données si un attaquant envoie des données d'icônes malveillantes à un serveur pour tenter de récupérer des informations sensibles sur ce dernier. Microsoft corrige également treize autres vulnérabilités dans Windows permettant soit une exécution arbitraire de code à distance, soit une élévation de privilèges, soit un contournement de la politique de sécurité de Kerberos, soit encore un déni de service.

Recommandations

Le CERT-FR recommande l'application de ces correctifs de sécurité dès que possible.

Documentation

- Avis CERTFR-2016-AVI-055
  http://www.cert.ssi.gouv.fr/site/CERTFR-2016-AVI-055
- Avis CERTFR-2016-AVI-056
  http://www.cert.ssi.gouv.fr/site/CERTFR-2016-AVI-056
- Avis CERTFR-2016-AVI-057
  http://www.cert.ssi.gouv.fr/site/CERTFR-2016-AVI-057
- Avis CERTFR-2016-AVI-058
  http://www.cert.ssi.gouv.fr/site/CERTFR-2016-AVI-058
- Avis CERTFR-2016-AVI-059
  http://www.cert.ssi.gouv.fr/site/CERTFR-2016-AVI-059
- Avis CERTFR-2016-AVI-060
  http://www.cert.ssi.gouv.fr/site/CERTFR-2016-AVI-060

3 - Rappel des avis émis

Dans la période du 08 au 14 février 2016, le CERT-FR a émis les publications suivantes :

- CERTFR-2016-AVI-049 : Vulnérabilité dans Oracle Java
- CERTFR-2016-AVI-050 : Multiples vulnérabilités dans le noyau Linux de Suse
- CERTFR-2016-AVI-051 : Multiples vulnérabilités dans les produits Cisco
- CERTFR-2016-AVI-052 : Multiples vulnérabilités dans le noyau Linux de Suse
- CERTFR-2016-AVI-053 : Multiples vulnérabilités dans Google Chrome
- CERTFR-2016-AVI-054 : Multiples vulnérabilités dans les produits Adobe
- CERTFR-2016-AVI-055 : Multiples vulnérabilités dans Microsoft Internet Explorer
- CERTFR-2016-AVI-056 : Multiples vulnérabilités dans Microsoft Edge
- CERTFR-2016-AVI-057 : Multiples vulnérabilités dans Microsoft Office
- CERTFR-2016-AVI-058 : Multiples vulnérabilités dans Microsoft .NET Framework
- CERTFR-2016-AVI-059 : Vulnérabilité dans Microsoft Active Directory Federation Services
- CERTFR-2016-AVI-060 : Multiples vulnérabilités dans Microsoft Windows
- CERTFR-2016-AVI-061 : Multiples vulnérabilités dans les produits Cisco
- CERTFR-2016-AVI-062 : Multiples vulnérabilités dans SCADA Siemens SIMATIC
- CERTFR-2016-AVI-063 : Multiples vulnérabilités dans Citrix NetScaler Application Delivery Controller et NetScaler Gateway
- CERTFR-2016-AVI-064 : Multiples vulnérabilités dans Mozilla Firefox

Gestion détaillée du document

15 février 2016 version initiale.