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 05 décembre 2016
N° CERTFR-2016-ACT-049

Affaire suivie par :
CERT-FR

BULLETIN D'ACTUALITÉ

Objet : Bulletin d'actualité CERTFR-2016-ACT-049

1 - Artéfacts inforensiques liés aux services Windows

Cet article présente plusieurs artéfacts inforensiques liés à l'installation ou à l'exécution d'un service sous Windows.

Les services Windows sont parfois exécutés avec des privilèges élevés et souvent lancés automatiquement. Ils sont donc particulièrement utilisés par des codes malveillants comme mécanisme de persistance. Par conséquent, une attention particulière doit leur être accordée lors d'une investigation afin de relever des traces de leur exploitation.

Ces artéfacts se retrouvent classiquement dans le système de fichiers, la base de registre, les fichiers d'événements Windows, mais aussi en mémoire.

Certains apparaissent lors de l'installation, d'autres à son lancement ou encore pendant son exécution.

Installation d'un service

Lors de l'installation, l'exécutable sur le système de fichiers doit être créé. Ensuite, le service doit être configuré auprès du *Service Control Manager (SCM)*. Cette opération requiert les privilèges d'administration. Le SCM pourra alors automatiquement ou sur demande procéder au démarrage de celui-ci.

Le premier artéfact identifié lors de l'installation d'un service est donc la création du fichier associé sur le système de fichiers.

Le chemin habituel pour les services de type pilote noyau est `%SYSTEMROOT%\System32\Drivers\` alors que celui des services système est `%SYSTEMROOT%\System32\`. Les services applicatifs ou tierce partie peuvent choisir d'autres dossiers.

Ensuite, un installeur peut appeler les API Windows pour la configuration du service (*OpenSCManager*, *CreateService*). Des outils tiers peuvent aussi se charger de cette étape. Dans ce dernier cas de figure, les techniques suivantes sont les plus répandues :

- l'outil `sc.exe` avec le paramètre *create* ;
- l'outil `installutil.exe` du framework .NET ;
- le cmdlet Powershell `New-Service` ;
- la méthode *create* de la classe WMI `Win32_Service`.

Les artéfacts classiques montrant l'exécution de programmes (*Prefetch*, *ShimCache*, ...) pourront révéler l'utilisation d'un installeur ou d'un des outils mentionnés.

Toutes ces méthodes modifient la base de données des services utilisée par le SCM pour y ajouter le service créé. Celle-ci est stockée dans la base de registre sous la clé :
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services`.
Chaque sous-clé est un service installé. Microsoft recommande d'utiliser les fonctions de l'API Windows pour gérer cette base de données plutôt que de manipuler directement la base de registre.

Chaque sous-clé présente les champs suivants :

- type du service : pilote noyau, pilote noyau d'un système de fichiers, service ayant son propre processus ou service partageant un processus ;
- type de démarrage : au démarrage de la machine, automatique, manuel ou désactivé ; dans le cas d'un pilote, on précise s'il est lancé au démarrage de la machine ou lors de l'initialisation du noyau ;
- niveau de contrôle d'erreur : en cas d'échec lors du lancement d'un service, le système d'exploitation réagit différemment en fonction du niveau indiqué ici. Les valeurs possibles sont : critique, important, normal ou ignorer ;
- chemin complet vers l'exécutable à lancer au démarrage du service. Il est possible que celui-ci soit vide. Dans ce cas le système charge le fichier ayant le même nom que la clé de registre depuis l'un des répertoires connus pour les services ;
- liste de dépendances (optionnel) : les services ou groupes de services qui doivent être démarrés avant ce service, pour que ce dernier puisse se lancer correctement ;
- compte utilisateur (optionnel) : compte utilisateur qui lance le service ; si aucun compte n'est spécifié, le compte *LocalSystem* est utilisé ;
- nom du pilote (optionnel) : le nom indiqué est utilisé par le gestionnaire d'entrées/sorties pour gérer le pilote ; en cas d'absence, le nom du fichier est utilisé.

Ces artéfacts sont retrouvés dans la base de registre. Les techniques et outils habituels (*regripper*, ...) seront utiles pour les relever. `CurrentControlSet` fait référence au *ControlSet* courant mais il est toujours pertinent de tous les analyser.

A partir de Windows Vista, un artéfact d'installation dans les journaux d'événements système peut être recherché. En effet, lorsqu'un service est installé par les fonctions de l'API Windows, un événement (7045) est créé dans le journal des événements "Système". Celui-ci contient des informations utiles pour l'investigation, comme le compte à l'origine de l'installation et celui qui exécute le service, l'exécutable lancé par le service, son type de démarrage et enfin la date d'installation.

Lancement d'un service

Une fois installé, il est possible de gérer un service selon plusieurs méthodes, dont les plus courantes sont citées ci-dessous :

- utilisation de la mmc de Windows pour le lancer ou l'arrêter via l'interface graphique ;
- utilisation du binaire `net.exe` avec les paramètres *start* ou *stop* ;
- utilisation du binaire `sc.exe` avec les paramètres *start* ou *stop* ;
- utilisation du cmdlet Powershell `Start-Service` ;
- utilisation de la méthode *StartService* de la classe WMI `Win32_Service`.

Ces méthodes laissent des traces dans les artéfacts d'exécution usuels (*Prefetch*, *ShimCache*, ...) parmi lesquelles le lancement du binaire associé au service.

Les journaux d'évènements permettent également de retrouver des traces d'exécution de services. Ainsi, à partir de Windows Vista, lorsqu'un service change d'état, les événements 7035 et 7036 du fournisseur *Service Control Manager* sont générés dans le journal des événements "Système". D'autres événements relatifs à des problèmes lors du lancement ou de l'arrêt d'un service attestent indirectement de son exécution. On peut citer par exemple les événements 101 et 203 du fournisseur *Microsoft-Windows-Diagnostics-Performance*.

La base de registre permet aussi de retrouver des traces d'exécution de services. Lors du premier démarrage d'un service de type pilote, une clé de registre est créée dans :
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_<nom du service>`.
Elle est n'est pas automatiquement supprimée lors de la désinstallation.

Enfin, on retrouve des traces du lancement d'un service en mémoire. Le SCM maintient en mémoire une liste des services, sous la forme d'une liste chaînée de structures `SERVICE_RECORD`. Le plugin Volatility `svcscan` parcourt cette liste et effectue également la recherche de motifs en mémoire pour en extraire les structures `SERVICE_RECORD` "orphelines".

Contournement du Service Control Manager

Afin d'être plus furtif, un attaquant peut effectuer manuellement certaines des actions effectuées automatiquement lorsqu'on passe par les API Windows standard, et ainsi éviter de créer une partie des artéfacts vus précédemment.

Par exemple, au lieu d'utiliser les fonctions *CreateService* et *StartService*, un attaquant peut créer les clés de registre dans `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services` manuellement et appeler l'API native `NdrClientCall2` (ou `NtLoadDriver` pour un pilote) pour le lancer. De ce fait, le code du SCM ne sera pas appelé, et les entrées du journal des événements ainsi que les structures `SERVICE_RECORD` en mémoire ne seront pas créées, mais le code du service malveillant s'exécutera quand même.

Dans ce cas, il reste toujours les entrées dans la base de registre et le système de fichiers pour retrouver le service malveillant.

Cependant, un attaquant qui n'aurait pas besoin de mécanisme de persistance pourrait supprimer ces clés de registre ainsi que le fichier associé une fois le code du service chargé en mémoire. Dans ce cas, il faut s'appuyer sur les artéfacts d'exécution mentionnés dans les précédents paragraphes, sachant qu'on peut également espérer retrouver des données dans le système de fichiers et dans la base de registre tant que les blocs de données correspondants n'auront pas été réécrits.

2 - Rappel des avis émis

Dans la période du 28 novembre au 04 décembre 2016, le CERT-FR a émis les publications suivantes :

- CERTFR-2016-ALE-009 : Campagne d'attaque contre des routeurs DSL
- CERTFR-2016-AVI-391 : Vulnérabilité dans Mozilla Firefox
- CERTFR-2016-AVI-392 : Vulnérabilité dans les produits Mozilla
- CERTFR-2016-AVI-393 : Multiples vulnérabilités dans le noyau Linux d'Ubuntu
- CERTFR-2016-AVI-394 : Multiples vulnérabilités dans Google Chrome

Gestion détaillée du document

05 décembre 2016 version initiale.