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 04 juillet 2014
N° CERTFR-2014-ACT-027

Affaire suivie par :
CERT-FR

BULLETIN D'ACTUALITÉ

Objet : Bulletin d'actualité CERTFR-2014-ACT-027

1 - Introduction au contrôle d'accès Windows reposant sur les revendications (« claims »)

Le mécanisme de contrôle d'accès historique de Windows repose sur la confrontation entre des jetons de sécurité (*tokens*) associés à des processus et des descripteurs de sécurité associés à des objets. Lors d'une demande d'accès, le jeton de l'utilisateur ou du processus est comparé au descripteur de sécurité de l'objet accédé. À partir de Windows 8, Microsoft a introduit un nouveau mécanisme d'authentification appelé revendications (ou *claims* en anglais), se détachant du modèle utilisé jusqu'à présent. Cet article vise à introduire les principes généraux de ce mécanisme.

Définition

Une revendication (ou claim) est un élément d'information unique relatif à un utilisateur, un périphérique ou une ressource. Cet élément d'information peut être par exemple un nom d'utilisateur, une adresse de messagerie, l'appartenance à un groupe du domaine ou une version de système d'exploitation. Ces *claims* sont émises par un contrôleur de domaine (DC) sous Windows Server 2012 qui joue le rôle de fournisseur d'identité. Il existe trois types de *claims*:

- *claim* utilisateur : attribut Active Directory associé à un compte utilisateur spécifique ;
- *claim* périphérique : attribut Active Directory associé à un compte machine spécifique ;
- attribut de ressource : propriété utilisée pour *étiqueter* un objet. Ces objets sont marqués pour une utilisation dans les décisions d'autorisation.

Les *claims* utilisateur et périphérique sont contenues dans les jetons de sécurité, tandis que les attributs de ressource sont stockés dans le descripteur de sécurité. Les *claims* permettent une approche complémentaire au contrôle d'accès historique en faisant intervenir des propriétés plus fines que les groupes.

Contrôle d'accès conditionnel

Il est rappelé que le descripteur de sécurité d'un objet est, entre autres, constitué d'une liste de contrôle d'accès composée d'ACE. Le système de *claims* introduit un mécanisme d'ACE permettant de formuler une politique de contrôle d'accès conditionnelle. Les expressions contenues dans ces ACE peuvent être liées par un ou plusieurs opérateurs logiques (AND, OR et NOT). Les ACE conditionnelles utilisent les 3 types de *claims* dans leurs expressions :

- les *claims* utilisateur sont représentées par la chaîne de caractères `@USER`;
- les *claims* périphériques sont représentées par la chaîne de caractères `@DEVICE`;
- les attributs de ressource sont représentées par la chaîne de caractères `@RESOURCE`.

Par exemple, une ACE comportant pour condition la *claim* country de l'utilisateur est égale à la chaîne de caractères FR s'écrit `@USER.country == FR`.

Intérêts et exemple concret

Parmi les nouvelles possibilités offertes, notons surtout la possibilité d'utiliser des propriétés de la machine. Ainsi, il est possible de former des règles conditionnelles pour gérer l'accès aux ressources. Par exemple : l'accès à cette ressource est autorisée si l'utilisateur est français et que son adresse mail est une adresse en `@ssi.gouv.fr`. Les *claims* permettent également de réduire la complexité inhérente à la gestion des groupes. Prenons par exemple un projet d'une multinationale, pour lequel nous souhaitons que seules les RH du projet aient accès au fichier contenant l'emploi du temps des salariés. Nous posons également la condition que seules les RH à plein temps sur la filiale européenne puissent y avoir accès. En utilisant le mécanisme des groupes de sécurité de l'AD et en créant un groupe pour chaque type de salarié, ce scénario implique la création de 12 groupes :

- 4 groupes pour diviser les employés répartis dans les 4 zones du globe ;
- 4 groupes pour sélectionner uniquement les RH des 4 zones du globe ;
- 4 groupes pour sélectionner les employés à plein temps des 4 zones du globe.

L'utilisation des *claims* permet de simplifier ce problème. Dans notre exemple, il est possible de remplacer certains groupes par des *claims* :

- tous les groupes `RH` peuvent être remplacés par la *claim* `Department`,
- tous les groupes `Employés_plein_temps` peuvent être remplacés par la *claim* `Employee-Type`.

Avec une telle configuration, nous avons divisé le nombre de groupes par 3 et évité de nombreuses redondances.

Conclusion

Sans le remplacer, les *claims* apportent au contrôle d'accès une souplesse supplémentaire facilitant certaines démarches d'administration. Les *claims* sont d'ailleurs utilisées par Microsoft pour un usage interne à certains de leurs logiciels :

- AppLocker utilise des *claims* appelées Local *claims* ;
- Sharepoint (à partir de la version 2010) possède un fournisseur de *claims* interne à l'application pour la gestion des utilisateurs ;
- l'Infrastructure de Classification de Fichiers (FCI) introduite avec Windows Server 2008 R2 automatise les processus de classification de documents pour optimiser la gestion des données à l'aide d'attributs de ressources.

La gestion de façon native des *claims* dans Windows 8 / Serveur 2012 marque la généralisation de ce nouveau mécanisme à l'ensemble des produits Microsoft.

2 - Compromission de comptes FTP

Le CERT-FR a récemment eu connaissance de la diffusion publique d'une liste de nombreux identifiants d'accès à des comptes FTP. Les incidents de ce type sont fréquents, et peuvent être en grande partie évités avec une gestion appropriée des mots de passe et des accès à un serveur FTP. Les identifiants de connexion ont très certainement pu être collectés par des attaquants suite à la compromission de postes où ces identifiants étaient déjà enregistrés.

Le CERT-FR recommande ainsi de ne pas enregistrer les mots de passe dans un client FTP mais de préférer un outil de stockage sécurisé de mots de passe et, dans la mesure du possible, d'utiliser des postes dédiés et contrôlés pour accéder au serveur FTP. Les identifiants récupérés par les attaquants sont parfois des comptes qui ne sont plus utilisés mais qui sont encore valides et dont les identifiants sont moins bien protégés. Il est également important de désactiver tous les comptes qui ne sont plus utilisés.

Ces recommandations peuvent être généralisées pour tout service accessible par Internet. De plus, lorsque cela est possible, le CERT-FR recommande d'autoriser l'accès à des services seulement depuis des adresses IP bien identifiées afin de limiter l'exposition du service.

De manière générale, il reste cependant préférable d'utiliser des protocoles de transfert permettant l'authentification forte, comme SFTP pour le transfert de fichiers, et des canaux de communication sécurisés tels qu'un VPN.

Documentation

- KeePass, certifié au titre de la CSPN :
  http://www.ssi.gouv.fr/fr/produits-et-prestataires/produits-certifies-cspn/certificat_cspn_2010_07.html

– Note technique Recommandations de sécurité relatives aux mots de passe :
http://www.ssi.gouv.fr/IMG/pdf/NP_MDP_NoteTech.pdf
– Note technique Recommandations pour un usage sécurisé d'(Open)SSH :
http://www.ssi.gouv.fr/IMG/pdf/NP_OpenSSH_NoteTech.pdf

3 - Vulnérabilité dans de nombreuses implémentations des algorithmes LZO et LZ4

Contexte

Une vulnérabilité affectant un certain nombre d'implémentations de l'algorithme de compression LZO (et de son successeur LZ4) a été publiée cette semaine par le blog « *Lab Mouse Security* ». Les algorithmes LZO et LZ4 sont des algorithmes de compression très efficaces qui sont utilisés dans une grande variété d'applications. Cette vulnérabilité a provoqué une réaction forte, principalement en raison de l'utilisation de cet algorithme dans le robot *Curiosity* déployé sur Mars par la NASA.

Analyse de l'impact

Cette faille est une vulnérabilité de type dépassement d'entier qui se produit dans la fonction de décompression dans le cas où des données dites « *Literal Run* » (données non compressées intégrées au sein d'un flux compressé) sont traitées. Cela peut ensuite provoquer une corruption de mémoire lors de la recopie des données dans le tampon de sortie.

Les attaquants éventuels devraient cependant rencontrer une difficulté majeure dans l'exploitation de cette vulnérabilité. En effet, pour provoquer ce dépassement d'entier, il faut que la fonction de décompression LZO ou LZ4 prenne en entrée des blocs de 16M de données au minimum. Or, les applications utilisant cet algorithme fonctionnent toutes avec des blocs de données de 8M au maximum. Une très grande majorité d'applications fonctionne avec des blocs encore plus petits, par exemple 128Ko pour *ZFS* et 16Ko pour *Lucene*.

Malgré l'impact médiatique, cette vulnérabilité ne semble pas facilement exploitable. Cependant, du fait de la grande variété des applications utilisant LZO ou LZ4 et de la duplication du code vulnérable, cette faille pourrait n'être corrigée que tardivement sur certains systèmes. Le noyau Linux, bien qu'utilisant l'algorithme LZ4, n'est pas vulnérable. Le tampon d'entrée lors de la décompression est en effet limité à 8M. Parmi les applications les plus utilisées intégrant ces algorithmes figurent MySQL, OpenVPN ou Dovecot.

Un correctif pour la bibliothèque `liblz4` a déjà été publié.

Recommandations

Le CERT-FR recommande de mettre à jour les systèmes affectés. La liste des applications affectées est disponible ci-dessous (section documentation).

Documentation

– Article de Lab Mouse Security :
http://blog.securitymouse.com/2014/06/raising-lazarus-20-year-old-bug-that.html
– Résumé de la portée de la vulnérabilité :
http://fastcompression.blogspot.fr/2014/06/lets-move-on.html
– Applications affectées (bas de page) :
https://code.google.com/p/lz4

4 - Rappel des avis émis

Dans la période du 27 juin au 03 juillet 2014, le CERT-FR a émis les publications suivantes :

– CERTFR-2014-AVI-284 : Vulnérabilité dans Red Hat JBoss Web Framework Kit
– CERTFR-2014-AVI-285 : Vulnérabilité dans Wireshark
– CERTFR-2014-AVI-286 : Multiples vulnérabilités dans Solaris
– CERTFR-2014-AVI-287 : Multiples vulnérabilités dans Samba

- CERTFR-2014-AVI-288 : Multiples vulnérabilités dans le noyau Linux d'Ubuntu
- CERTFR-2014-AVI-289 : Multiples vulnérabilités dans Asterisk
- CERTFR-2014-AVI-290 : Multiples vulnérabilités dans les produits Huawei
- CERTFR-2014-AVI-291 : Vulnérabilité dans les produits F5
- CERTFR-2014-AVI-292 : Vulnérabilité dans RealNetworks RealPlayer
- CERTFR-2014-AVI-293 : Multiples vulnérabilités dans les produits Apple
- CERTFR-2014-AVI-294 : Multiples vulnérabilités dans les produits EMC
- CERTFR-2014-AVI-295 : Multiples vulnérabilités dans Cisco Unified Communications Domain Manager

Gestion détaillée du document

04 juillet 2014 version initiale.