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 28 août 2017
N° CERTFR-2017-ACT-034

Affaire suivie par :
CERT-FR

BULLETIN D'ACTUALITÉ

Objet : Bulletin d'actualité CERTFR-2017-ACT-034

1 - Protéger les fichiers d'historique sur Linux pour garantir la protection apportée par un conteneur chiffré

Contexte

Pour protéger les informations d'un projet sur un poste bureautique, il est possible d'utiliser des conteneurs chiffrés. Lorsqu'un conteneur est ouvert, les fichiers qu'il contient sont utilisables comme des fichiers non-chiffrés : ils peuvent être ouverts dans un éditeur de texte, un visualiseur d'image, un navigateur web, etc. Un des avantages des conteneurs chiffrés est la possibilité de le fermer, rendant ainsi inaccessibles les informations qu'il contient. Ceci est par exemple utile afin de ne pas rendre accessibles les informations d'un projet quand l'ordinateur est utilisé pour une tâche sans lien avec celles-ci.

Toutefois, l'environnement bureautique de l'utilisateur n'étant pas cloisonné, certains programmes conservent des traces d'utilisation des fichiers avec lesquels ils interagissent. Par exemple lorsque l'éditeur de texte Vim est utilisé, le fichier `.viminfo` du répertoire personnel contient les dernières commandes effectuées (dont les mot-clés utilisés dans les recherches) et les noms des derniers fichiers édités. Un autre exemple est donné par le fichier `.recently-used`, qui contient les chemins d'accès des derniers fichiers ouverts de certains logiciels. Quand ces chemins contiennent des éléments sensibles (par exemple le nom d'une entreprise cliente), cela contrevient à la protection mise en place avec les conteneurs chiffrés.

Mesures de protection

Sur Linux, plusieurs méthodes permettent de prévenir l'enregistrement d'informations concernant des éléments sensibles dans de tels fichiers. Parmi celles-ci, on peut citer :

- introduire un cloisonnement au niveau de l'environnement utilisateur du système, par exemple en mettant en place des comptes utilisateurs spécifiques ;
- nettoyer *a posteriori* les fichiers contenant les informations à l'aide d'un outil d'effacement sécurisé ;
- prévenir l'inscription d'informations dans ces fichiers (en retirant le droit d'écriture par exemple).

La première option répond au problème dont il est question mais est peu pratique d'utilisation. La seconde option n'est réalisable que s'il existe des moments où il est possible de nettoyer les traces, ce qui n'est pas toujours possible selon les contraintes. Il reste donc la troisième, qui peut être mise en œuvre soit en configurant les logiciels quand ceux-ci le permettent (par exemple dans Vim avec `set viminfo="NONE"`), soit en utilisant les commandes shell suivantes pour créer un fichier vide immuable :

1. Supprime le fichier des fichiers récemment utilisés

```
shred -u ~/.recently-used
```

2. Crée un fichier vide

```
touch ~/.recently-used
```

3. Rend le fichier immuable, ce qui nécessite d'être administrateur

```
sudo chattr +i ~/.recently-used
```

Voici une liste non-exhaustive de fichiers dans le répertoire personnel d'un utilisateur sur un système Linux qui peuvent être traités de cette manière :

- `.bash_history`, `.sh_history`, `.zsh_history` : historique des commandes shell
- `.mysql_history` : historique des commandes MySQL et MariaDB
- `.lesshst`, `.lesshsQ` : historique des commandes du programme de présentation Less
- `.viminfo` : historique des commandes de l'éditeur de texte Vim
- `.xsession-errors` : journal des erreurs des programmes lancés dans une session graphique X (dont des éditeurs de texte qui peuvent écrire des informations au sujet du contenu des conteneurs chiffrés)
- `.recently-used`, `.local/share/recently-used.xbel` : liste des fichiers récemment utilisés par certaines applications (Keepass 2 et gedit par exemple)
- `.config/libreoffice/4/user/registrymodifications.xcu` : liste des fichiers récemment utilisés par LibreOffice

En remplaçant tous ces fichiers par des fichiers vides immuables, les programmes concernés ne pourront plus écrire d'information pouvant contrevenir à la protection mise en œuvre par les conteneurs chiffrés. Cela peut toutefois entraîner des effets de bord (par exemple rendre immuable `.bash_history` empêche d'utiliser les fonctionnalités de recherche dans l'historique de Bash), c'est pourquoi il est nécessaire d'analyser le cas d'utilisation de chaque fichier avant de le rendre immuable.

2 - Rappel des avis émis

Dans la période du 21 au 27 août 2017, le CERT-FR a émis les publications suivantes :

- CERTFR-2017-AVI-271 : Multiples vulnérabilités dans Mozilla Thunderbird
- CERTFR-2017-AVI-272 : Vulnérabilité dans Xen
- CERTFR-2017-AVI-273 : Vulnérabilité dans Hewlett Packard Enterprise Integrated Lights-out

Gestion détaillée du document

28 août 2017 version initiale.