Cible de sécurité CSPN

dm-crypt version 4.4.2

Catégorie « Stockage sécurisé »

Référence : CSPN-ST-dm-crypt-1.01
Date : le 24/02/2016
Code interne : ANS014

Siège : 4 bis allée du Bâtiment ● 35000 Rennes ● France ● www.amossys.fr
SIRET : 493 348 890 00036 ● **NAF** : 6202 A ● RCS Rennes B 493 348 890 ● SAS au capital de 38.000 Euros

FICHE D'ÉVOLUTIONS

RÉVISION	DATE	DESCRIPTION	RÉDACTEUR
1.00	19/02/2016	Création du document	Kelly RESCHE Antoine COUTANT
1.01	24/02/2016	Précision sur la version de cryptsetup	Julie LEMETEYER

SOMMAIRE

1. INTRODUCTION

1.1. OBJET DU DOCUMENT

Ce document est réalisé dans le cadre de l'évaluation, selon le schéma CSPN[1] promu par l'ANSSI[2], du produit open source « dm-crypt ».

Ce document est soumis au contrôle technique et qualité d'**AMOSSYS** ainsi qu'à la validation de l'**ANSSI**. Les mises à jour de ce document sont effectuées par l'équipe projet d'**AMOSSYS**.

1.2. IDENTIFICATION DU PRODUIT

Éditeur	Projet Open Source
Lien vers l'organisation	kernel.org
Nom commercial du produit	dm-crypt
Numéro de la version évaluée	Noyau Linux 4.4.2 associé à cryptsetup 1.7.0
Catégorie du produit	Stockage sécurisé

[1] Certification de Sécurité de Premier Niveau
[2] Agence nationale de la sécurité des systèmes d'information

2. DESCRIPTION DU PRODUIT

2.1. DESCRIPTION GÉNÉRALE

`dm_crypt` est un *mapper* permettant le chiffrement de manière transparente d'un périphérique en mode bloc. Il est intégré au noyau Linux depuis la version 2.6 et utilise les fonctions de chiffrement issues de l'API cryptographique du noyau.

Globalement, `dm_crypt` permet de :

- créer un disque virtuel chiffré contenu dans un fichier ou sur une partition du disque dur et de le monter comme un disque physique réel ;
- chiffrer/déchiffrer de façon automatique, à la volée et transparente ;
- chiffrer une partition entière (y compris le système d'exploitation, les fichiers temporaires, swap, etc.) ou un périphérique de stockage (clé USB ou disque dur).

Dans le cas du chiffrement de la partition système, l'authentification de l'utilisateur est réalisée au démarrage du poste client (avant que le système d'exploitation soit démarrée) à l'aide d'un mot de passe ou en insérant un périphérique (clé USB).

On distingue deux modes de chiffrement pour `dm_crypt` :

- mode « *Plain* » :
  - une *seule passphrase* pour ouvrir un volume ;
  - la clé maître est dérivée de la *passphrase* et utilisée pour le chiffrement/déchiffrement des données ;
  - pas de métadonnée sur le disque ;
  - les paramètres sont lus depuis le fichier de configuration (par défaut) ou donnés en ligne de commande ;
- mode LUKS :
  - jusqu'à 8 *passphrase* peuvent être définies pour ouvrir un même volume LUKS ;
  - les *passphrase* sont utilisées pour déchiffrer une unique clé maître ;
  - les paramètres sont stockés dans un bloc de métadonnées (en-tête LUKS) au début du disque ;
  - sauvegarde et restauration des en-têtes LUKS.

La Figure 1 illustre le fonctionnement global de chacun des modes.

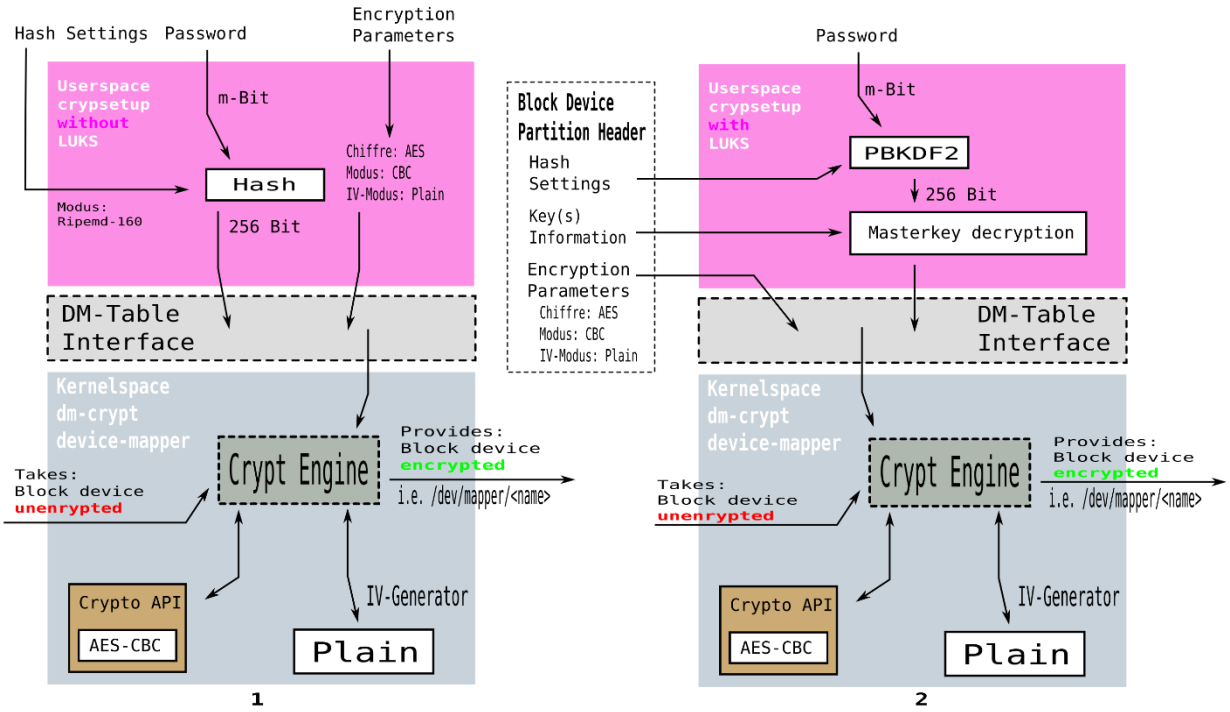

Figure 1 - Comparaison des modes Plain et LUKS du chiffrement dm_crypt

L'évaluation se focalisera en particulier sur le mode `dm_crypt/LUKS`.

2.2. PRINCIPE DE FONCTIONNEMENT

`dm_crypt` peut être utilisé de deux façons :

- au moment de l'installation du système d'exploitation (phase de partitionnement des disques) ;
- sur un système déjà installé.

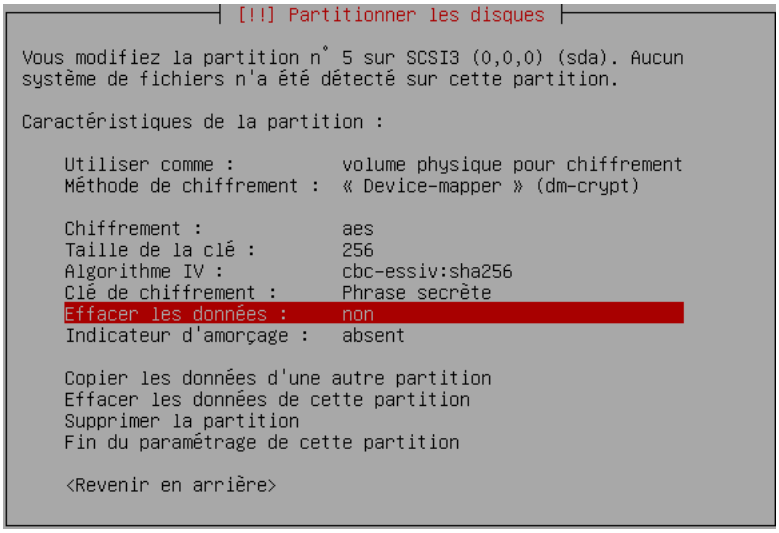

Figure 2 - Exemple de création d'une partition chiffrée avec l'assistant d'installation Ubuntu ou Debian[3]

[3] Source : http://doc.ubuntu-fr.org/tutoriel/chiffrer_son_disque

Dans le second cas, `dm_crypt` repose sur des outils en espace utilisateur pour la gestion des volumes chiffrés et de l'authentification.

L'utilitaire en ligne de commande `cryptsetup`[4] permet de gérer les disques chiffrés avec `dm_crypt`. Il permet de créer des conteneurs chiffrés « *plain dm_crypt* » ou LUKS (usage recommandé[5]).

Les commandes suivantes permettent de créer et ouvrir (mapper) une partition chiffrée au format LUKS :

```
# cryptsetup luksFormat <device>
# cryptsetup luksOpen <device> <name>
```

Le fichier `/dev/mapper/<name>` (indicatif) créé correspond à la vision déchiffrée de la partition `<device>`.

Par la suite, il est possible de formater, monter et démonter la partition :

```
# mkfs.ext4 /dev/mapper/<name>
# mount /dev/mapper/<name> /mnt
# umount /mnt
```

Le volume déchiffré est fermé avec la commande suivante :

```
# cryptsetup luksClose <name>
```

L'utilisateur fournit les informations nécessaires à la création du volume `dm_crypt` :

- le type de conteneurs chiffrés ;
- l'emplacement du volume ;
- la suite cryptographique utilisée pour le chiffrement comprenant :
  - l'algorithme de chiffrement ;
  - le mode opératoire de chiffrement ;
  - le mode de génération des vecteurs d'initialisation ;
- la fonction hachage utilisée pour la dérivation des clés (fonction PBKDF2) ;
- la taille du volume chiffré à créer ;
- le mot de passe utilisateur ;
- la liste de fichiers *keyfiles* (optionnels) ;
- les options de formatage du volume.

Les algorithmes cryptographiques qui seront utilisés par la TOE sont à choisir parmi la liste algorithmes connus du noyau.

2.3. DESCRIPTION DES DÉPENDANCES

Afin de configurer `dm_crypt`, la bibliothèque `device mapper library` et l'outil `cryptsetup` sont nécessaires. Ces paquets sont généralement déjà fournis par la plupart des distributions Linux. Pour l'évaluation, la version 1.7.0 de cryptsetup sera installée.

[4] http://code.google.com/p/cryptsetup/
[5] https://code.google.com/p/cryptsetup/wiki/DMCrypt#Configuration_using_cryptsetup

2.4. DESCRIPTION DE L'ENVIRONNEMENT TECHNIQUE DE FONCTIONNEMENT

`dm_cypt` est un *mapper* de chiffrement de données inclus dans les systèmes d'exploitation Linux (à partir de la version 2.6 du noyau) et Android (depuis la version 3.0).

`dm_crypt` ne nécessite pas de matériel dédié particulier. La TOE est directement installée intégrée dans le noyau Linux. Son utilisation est conditionnée par l'activation (valeur `CONFIG_DM_CRYPT`) dans la configuration de ce dernier.

2.5. PÉRIMÈTRE DE L'ÉVALUATION

L'évaluation porte sur les fonctionnalités du module `dm_crypt` en mode LUKS. Le périmètre de l'évaluation du produit comprend notamment :

- l'authentification de l'utilisateur ;
- le chiffrement/déchiffrement des données de l'utilisateur ;
- la génération de clés cryptographiques associées à un volume `dm_crypt`.

Le système d'exploitation hôte est considéré hors TOE.

Le système d'exploitation retenu pour la réalisation de l'étude est Debian Jessie 64 bits.

3.PROBLÉMATIQUE DE SÉCURITÉ

3.1. DESCRIPTION DES UTILISATEURS TYPIQUES

Par définition, les utilisateurs concernent les personnes et services applicatifs qui interagissent avec le produit évalué.

Les utilisateurs qui ont un accès à la TOE sont les utilisateurs des postes où est installé `dm_crypt` afin de protéger leurs données.

3.2. DESCRIPTION DES BIENS SENSIBLES

Par définition, un bien sensible est une donnée (ou fonction) jugée comme ayant de la valeur par la TOE. Sa valeur est estimée selon trois critères de sécurité (aussi appelés besoins de sécurité) : disponibilité, intégrité et confidentialité.

Les biens sensibles que la TOE doit protéger sont les suivants :

- **B1.DONNÉES-STOCKÉES**

  Les données de l'utilisateur stockées dans un volume chiffré et protégées en intégrité et en confidentialité par `dm_crypt`.

- **B2.CLÉS-CRYPTOGRAPHIQUES**

  L'ensemble des clés symétriques utilisées pour protéger les données dans un volume `dm_crypt`. Il s'agit :

  - des clés maitresses utilisées pour le chiffrement/déchiffrement des données de l'utilisateur ;
  - des clés dérivées utilisées pour protéger les clés maitresses. Elles sont dérivées des données d'authentification.

Les besoins de sécurité de chacun de ces biens sont donnés ci-dessous :

Biens sensibles	Disponibilité	Intégrité	Confidentialité
B1.DONNÉES-STOCKÉES		√	√
B2.CLÉS-CRYPTOGRAPHIQUES		√	√

Tableau 1 - Besoins de sécurité des biens sensibles

3.3. DESCRIPTION DES HYPOTHÈSES SUR L'ENVIRONNEMENT

Par définition, les hypothèses sont des déclarations portant sur le contexte d'emploi de la TOE ou de son environnement.

Les hypothèses sur l'environnement de la TOE suivantes doivent être considérées :

- **H1.OS**

  Le système d'exploitation, support de la TOE, met en œuvre des mécanismes de protection adéquats (confinement, contrôle d'accès, etc.) paramétrés et configurés selon les règles de l'état de l'art. De plus, il est à jour des correctifs en vigueur au moment de l'installation, sain et exempt de virus, chevaux de Troie, etc.

- **H2.INSTALLATION-INTÈGRE**

  Toutes les bibliothèques et outils nécessaires à l'installation et à l'utilisation de la TOE sont intègres.

- **H3.RÉMANENCE**

  La mémoire vive (RAM) utilisée par la machine qui exécute le produit n'est pas rémanente par construction.

- **H4.PROTECTION-PROGRAMME-MALVEILLANT**

  Une fois démarré, le système d'exploitation empêche les programmes malveillants d'exfiltrer les données en clair issues des volumes protégés par la TOE.

- **H5.UTILISATEUR**

  L'utilisateur est de confiance et sait utiliser la TOE. Il doit aussi être sensibilisé à la protection de son poste de travail et aux bonnes pratiques en matière de sécurité (règles sur la qualité des mots de passe ou phrases secrètes, règles permettant d'éviter la non-compromission de ces secrets, utilisation de mécanismes de verrouillage de session en cas d'inactivité, etc.).

3.4. DESCRIPTION DES MENACES

Les agents de menace considérés sont les **entités non autorisées** c'est-à-dire un attaquant humain ou entité qui interagit avec la TOE mais qui ne dispose pas d'accès légitime à la TOE.

Les menaces qui portent sur les biens sensibles de la TOE sont les suivantes :

- **M1.ACCÈS-ILLÉGITIME-DONNÉES**

  Un attaquant parvient à accéder, à l'insu du l'utilisateur légitime, aux données sensibles de l'utilisateur stockées sur la partition `dm_crypt` alors que celle-ci est montée.

- **M2.RÉCUPÉRATION-PARTIELLES-INFORMATIONS**

  Un attaquant à identifier la présence de données sur un conteneur chiffré (si le formatage complet n'a pas été réalisé à la création du volume) en analysant la mémoire de travail de l'application (RAM).

- **M3.ACCÈS-DONNÉES-TEMPORAIRES**

  Un attaquant parvient à récupérer les données sensibles de l'utilisateur ou des clés cryptographiques après analyse de la mémoire de travail de l'application (RAM).

Remarque concernant le contexte d'utilisation : dans le cas du chiffrement de la partition système, les menaces ne s'appliquent que lorsque le système est arrêté ou en cours de démarrage. La TOE n'a pas vocation à contrer les attaques lorsque le système d'exploitation est totalement démarré.

Remarque concernant les données d'authentification : `dm_crypt` ne permet pas de protéger le mot de passe de l'utilisateur. Un attaquant pourrait parvenir à connaître les mots de passe d'accès aux données sensibles contenues dans un volume `dm_crypt` en menaçant l'utilisateur légitime.

3.5. DESCRIPTION DES FONCTIONS DE SÉCURITÉ

Par définition, les fonctions de sécurité sont l'ensemble des mesures techniques et mécanismes mis en œuvre dans la TOE pour protéger de façon proportionnée les biens sensibles de la TOE contre les menaces identifiées.

Les fonctions de sécurité essentielles de la TOE sont les suivantes :

- **F1.(DÉ)CHIFFREMENT-DONNÉES**

  `dm_crypt` protège en confidentialité les données propres à l'utilisateur via du chiffrement.

- **F2.AUTHENTIFICATION-UTILISATEURS**

  `dm_crypt` authentifie l'utilisateur avant de libérer l'accès aux données chiffrées.

- **F3.PROTECTION-CLÉS-CHIFFREMENT-DONNÉES**

  La TOE assure la protection en confidentialité des clés maitresses utilisées pour le chiffrement de données en les chiffrant avec des clés dérivées des données d'authentification.

3.6. MATRICES DE COUVERTURES

3.6.1. Menaces et biens sensibles

La matrice suivante présente la couverture des menaces sur les biens sensibles (les lettres "D", "I" et "C" représentent respectivement les besoins de Disponibilité, Intégrité et Confidentialité) :

	B1.DONNÉES-STOCKÉES	B2.CLÉS-CRYPTOGRAPHIQUES
M1.ACCÈS-ILLÉGITIME-DONNÉES	**IC**	
M2.RÉCUPÉRATION-PARTIELLES-INFORMATIONS	**IC**	**C**
M3.ACCÈS-DONNÉES-TEMPORAIRES	**C**	**IC**

Tableau 2 - Couverture des biens sensibles par les menaces

3.6.2. Menaces et fonctions de sécurité

La matrice suivante présente la couverture des menaces par les fonctions de sécurité :

	F1.(DÉ)CHIFFREMENT-DONNÉES	F2.AUTHENTIFICATION-UTILISATEURS	F3.PROTECTION-CLÉS-CHIFFREMENT-DONNÉES
M1.ACCÈS-ILLÉGITIME-DONNÉES	√	√	
M2.RÉCUPÉRATION-PARTIELLES-INFORMATIONS			√
M3.ACCÈS-DONNÉES-TEMPORAIRES			√

Tableau 3 - Couverture des menaces par les fonctions de sécurité

Fin du document