TP 3 – Cassage de mots de passe

Première partie

Authentification Linux

Le mécanisme d'authentification reposant sur DES utilisé par Linux (plus généralement les systémes fondés sur Unix), consiste à générer l'empreinte d'un mot de passe en chiffrent 25 fois, avec une variante de l'algorithme DES, une chaine de caractéres nulles (64 bits valant 0). Le fait de chiffrer 25 fois a pour objectif de ralentir le processus d'authentification et donc de ralentir de potentiels attaques. Les empreintes sur Linux, sont stockées dans le fichier `/etc/shadow` dont l'accés en lecture et re-ecriture nécessite de droits privilégié.
Le fichier `/etc/shadow` contient des lignes de la forme :

```
username:passwd:last:may:must:warn:expire:disable:reserved
```

où :
- `username` : nom de l'utilisateur du compte
- `passwd` : sel et empreinte du mot de passe
- `last` : date de la derniére modification du mot de passe
- `may` : nombre de jours avant que le mot de passe puisse étre modifié
- `must` : nombre de jours avant que le mot de passe doive étre modifié
- `warn` : nombre de jours durant lesquels l'utilisateur est prévenu de l'expiration
- `expire` : nombre de jours entre l'expiration du mot de passe et le blocage du compte
- `disable` : date du blocage du compte (en nombre de jours depuis le 01/01/70)
- `reserved` : champ réservé pour un usage futur.

Nous allons travailler à l'intérieur de la machine virtuelle installé lors du TP1 à fin de pouvoir utiliser les droits d'administrateur.

Exercice 1. *Empreintes et Authentification*

1. Quelle est l'utilité de stocker les empreintes des mots de passe plûtot que les mots de passe eux-même ?
2. Pourquoi doit-on protéger l'accès aux empreintes des mots de passe ?
3. Sous quelle condition cette precaution ne serait-elle pas nécessaire ?
4. Lancez la VM et afficher le contenu du fichier `/etc/shadow`, comment lisez vous cette information (on regardera surtout la dernière ligne du fichier) ?
5. La structure de `passwd` est la suivante `$chiffrement$sel$empreinte`. Recréez l'empreinte du mot de passe en utilisant une commande de la forme suivante :

   ```
   echo motdepasse | mkpasswd -m sha-512 -S sel -s
   ```

   où `motdepasse` et `sel` doivent être remplacés par les informations correspondantes. On utilise l'algorithme de hash-512 qui correspond à l'indice 6.

Deuxième partie

Le but de ce premier tp est de vous faire tester John the Ripper, un logiciel libre de cassage de mots de passe par dictionnaire. Ce logiciel peut notamment être utilisé pour tester la sécurité d'un mot de passeàposteriori dans le cas d'un audit de sécurité ou encore pour évaluer son niveau de sécuritéàpriori lorsque l'utilisateur choisit son mot de passe. En pratique, l'utilisation de ce type de logiciel doit se faire d'une maniére responsable et éthique, dans le but principal de contrôler la sécurité de vos mots de passe. Vous devrez commencer par télécharger le logiciel John the Ripper Í'adresse suivante : `http://www.openwall.com/john/` Plus spécifiquement, il vous est conseillé de télécharger la version 1.8.0-jumbo-1 community-enhanced de John the Ripper. Installez ensuite John the Ripper sur votre machine et localisez la documentation ainsi que le fichier de configuration joint.conf. Prenez ensuite le temps de lire la documentation afin de vous familiariser avec le fonctionnement du logiciel. Téléchargez ensuite le fichier de mots de passage á casser, contenu dans le fichier `password_tp.txt`, á cette adresse : `https://www.giuseppe-dimolfetta.com/securite-internet-reseaux`. Ce fichier de mots de passe contient 8 entrées utilisateur contenant chacune un login et le mot de passe correspondant obtenu en le passant dans la fonction de hachage MD5 sans application de salage au préalable. Votre objectif est de récupérer chacun des mots de passe correspondants à l'aide de John the Ripper ainsi que grâce aux indices qui vous sont fournis.

Important

Vous devrez rendre un rapport résumant le travail effectué dans le TP. Dans ce rapport, vous devrez préciser pour chacun des mots de passe récupérés, la démarche que vous avez utilisé pour récupérer ce mot de passe en montrant par exemple le code que vous avez utilisé ainsi que des statistiques d'utilisation comme le temps que le logiciel a mis pour trouver le mot de passe ou le nombre associé de tentatives.

Exercice 2. *John the Ripper, un premier rendez-vous*

JTR dispose de trois mode de fonctionnement : (i) **single** le mode *single* est le plus basique mais pas forcement le plus efficace ; (ii) **wordlist**, permet d'utiliser des listes de mots pour générer des candidats ; et (iii) **incremental**. Nous on se concentrera sur le premiére de ces modes.

User1 et *User2* n'ont pas pris la peine de sécuriser leur mot de passe d'une maniére appropriée. Essayez de casser ces mots de passe par le mode "single" de John the Ripper. [1]

1. Comment fonctionne ce mode à votre avis ?
2. Quel bon conseil vous donneriez aux utilisateurs *User1* et *User2*

Exercice 3. *Attaque par dictionnaire*

User3, *User4* et *User5* sont des fans de Pokémon et ont dérivé leurs mots de passe de cet univers. En particulier, *User3* avoue avoir directement pris comme mot de passe le nom de son pokémon préféré. *User4* quant à lui a utilisé une recette un peu plus complexe. Plus précisément, il a choisi un nom de pokémon au hasard, remplacé toutes les voyelles par un chiffre fixé de 0 à 9, puis mis le tout en majuscules. Enfin, *User5* a choisi un pokémon au hasard avant d'ensuite inverser les lettres de ce pokémon puis enfin de dédoubler le résultat généré afin d'obtenir son mot de passe. [2]

1. Pour lancer John en mode single, il suffit de taper : `john --single fichier_passwd` John va prendre le fichier passwd et tester différents algorithmes. Ces algorithmes se trouvent dans le fichier john.conf dans la partie List.Rules : single. Aide possible, le site de john the ripper `http://www.openwall.com/john/doc/RULES.shtml`. Pour voir les mots de passe trouvés : `john --show fichier_passwd`
2. Pour lancer en mode dictionnaire, il suffit de taper : `john --wordlist=nom_dico fichier_passwd`

1. Retrouvez les mots de passe de ces 3 utilisateurs. Vous pourrez trouver un dictionnaire `pokemon.txt` contenant le nom des pokémons à l'adresse suivante : `https://www.giuseppe-dimolfetta.com/securite-internet-reseaux`. Indice : un bon point de départ pourrait être de lire la documentation RULES de john.
2. Suite aux exercises 2 et 3, proposez une liste d'au moins 5 règles évidentes pour choisir un bon mot de passe.

`--rules`. Les algorithmes testés seront dans la sections : `List.Rules:wordlist`. Pour voir les mots de passe trouvés : `john --show fichier_passwd`