****

**SSI - Laboratoire 1**

**Craquage de mots de passe**

12.10.2011

IL-2012

Brahim Lahlou

Numa Trezzini

1. Introduction 3

2. But 3

3. Détermination de la complexité 3

3.1 Fonctionnement des algorithmes 3

3.1.1 Généralités sur les algorithmes de tri 3

# 1. Résumé

# 2. Introduction

Dans le monde informatique, l’une des principales protections contre les intrusions et autres tentatives d’accès frauduleuses à des machines tierses sont les mots de passe. Ils jouent un role premier dans la sécurité des systèmes d’information. Pour pouvoir faire usage de ces mots de passe, il faut bien que l’ordinateur puisse les stocker, sous une forme ou une autre, afin de les vérifier. L’idée d’exploiter cette contraite à des fins malveillantes ne date pas d’hier, et les techniques y relatives sont, si ce n’est plus nombreuses, au moins plus complexes. Les techniques de protection se sont elles aussi développées, mais le « bon sens » de l’utilisateur est toujours un facteur proimordial dans la sécurité de sa machine : les mots de passe doivent etre « forts », c’est-à-dire résister, le plus longtemps possible, aux attaques et au décryptage.

Afin de créer des mots de passe forts, il convient de connaître les méchanismes de craquage sous-jacents. Ceci nous permettra de comprendre les risques et ainsi d’anticiper les menaces.

Nous allons en premier lieu présenter le but de ce laboratoire, avant de fournir une explication sur la façon de réaliser ce laboratoire. Nous répondrons ensuite aux questions posées, analyserons les réponses et nous terminerons par une conclusion sur les méthodes de craquage.

# 3. But

Le but de ce laboratoire est de tester nous-mêmes quelques unes des techniques de craquage de mots de passe. Par l’intermédiaire du programme **John The Ripper** (**JtR** plus loin dans le rapport), nous allons tester une attaque par « dictionnaire » et une attaque par « force brute ». Le programme **rainbowcrack** nous donnera un aperçu de la façon d’utiliser des tables rainbow (tables arc-en-ciel) pour trouver des mots de passe.

# 4. Mise en œuvre

Les mots de passe sont stockés sur l’ordinateur sous forme de chaine de caractères hashée (cryptée), afin d’assurer une première protection : les mots de passe ne sont pas stockés « en clair » et ne sont donc pas lisibles par l’humain. De plus, ils sont situés dans des emplacements protégés du système.

## 4.1 Obtention des empreintes des mots de passe

La première étape consiste donc à récupérer ces chaines de caractères. Selon l’OS utilisé, la méthode sera différente, car les protections ne sont pas pareilles. Pour réaliser l’ensemble du laboratoire, deux comptes fictifs seront crées sur nos machines : Paul et Roger, avec comme mots de passe *sawrin* et *amazing*, respectivement.

### 4.1.1 Question 1

* Quel est l’OS utilisé :   
  L’OS utilisé pour effectuer ce laboratoire est Windows 7.
* Votre fichier contient-il des empreintes de type LM, NTLM, MD5 ou autre :  
  Les mots de passe sont donc codés en…

## 4.2 John The Ripper

JtR fonctionne selon plusieurs méthodes, détaillées plus loin. JtR est normalement capable, avec assez de temps à disposition, de trouver n’importe quel mot de passe. Pour ce faire,

### 4.2.1 Question 2

* Comment JtR fonctionne-t-il ? Par dictionnaire ? Par force brute ? Autre ? Dans quel ordre ? (Donner des explications sur chaque méthode)

JtR effectue, dans l’ordre, les actions suivantes : test de mots de passe simple (« single crack » [JTR]), par attaque dictionnaire et finalement par « force brute ».

* Attaque par mots de passe simple: JtR commence par tester des mots de passe dérivés des informations disponibles sur l’utilisateur (nom, prénom, username, etc.). Dans la même lancée, JtR teste les mots de passe les plus couramment rencontrés (plus de détails sur ce sujet à la question suivante). Cette étape du craquage contient aussi plusieurs modifications des mots de passe de base, avec notamment l’ajout de prefixes et suffixes, le remplacement de lettres par des chiffres ou la duplication des mots.
* Attaque dictionnaire : L’attaque dictionnaire consiste à tester

# X. Conclusion

# X. Références

# X. Liste des symboles de référence

# X. Liste des figures

# X. Annexes