Evaluation pour le module Sécurité des Infrastructures Virtualisées

Pierre-Louis Palant / Landry Serin

Prelude

Les fichiers correspondants aux question 1 et 2 sont dans le dossier systemd

Les fichiers correspondants à la question 3 sont dans le dossier selinux

Les fichiers correspondants aux question 4 et 5 sont dans le dossier docker

Question 1:

Préparons l'environnement pour l'execution du service:

```
#Création de l'utilisateur qui executera le service
useradd httptest
#Le script appartient au nouvel utilisateur créé
chown httptest httpserver.py
#Le script est maintenant executable
chmod u+x httpserver.py
#Copie du script dans /usr/bin
sudo cp httpserver.py /usr/bin
```

Il faut placer httpserver.service dans /etc/systemd/system/

Pour lancer le service on execute les commandes suivante:

```
sudo systemctl daemon-reload
sudo systemctl start httpserver.service
```

Question 2:

Le service est executé avec l'utilisateur httptest qui n'as pas de bash ni de privilèges

```
~ » ps aux | grep httpserver                                    lserin@cheldon
httptest 11503  0.0  0.1 232212 16796 ?        Ss   17:51   0:00 python3 /usr/bin/httpserver.py
```

Le service à bien son propre dossier privé dans /tmp:

```
~ » sudo ls /tmp | grep httpserver
systemd-private-d5e5c1e97eb94f7d86f1104b4816821a-httpserver.service-tYhOka
```

Question 3:

Policy module généré avec sepolgen, complété à l'aide de audit2allow

On peut acceder en lecture/ecriture à /var/www/html

```
[ppalant@localhost selinux]$ curl  -d "test test test" localhost:7070/var/www/html/ftetet
[ppalant@localhost selinux]$ curl  -d "test test test" localhost:7070/etc/ssl/test
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
        "http://www.w3.org/TR/html4/strict.dtd">
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html;charset=utf-8">
        <title>Error response</title>
    </head>
    <body>
        <h1>Error response</h1>
        <p>Error code: 404</p>
        <p>Message: Could not create dir.</p>
        <p>Error code explanation: HTTPStatus.NOT_FOUND - Nothing matches the given URI.</p>
    </body>
</html>
```

On voit aussi qu'on ne peut pas accéder à /etc/ssl

Autre exemple, en lecture:

```
[ppalant@localhost selinux]$ curl   localhost:7070/var/www/html/ftetet
test test test[ppalant@localhost selinux]$ curl   localhost:7070/etc/passwd
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
        "http://www.w3.org/TR/html4/strict.dtd">
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html;charset=utf-8">
        <title>Error response</title>
    </head>
    <body>
        <h1>Error response</h1>
        <p>Error code: 404</p>
        <p>Message: File not found.</p>
        <p>Error code explanation: HTTPStatus.NOT_FOUND - Nothing matches the given URI.</p>
    </body>
</html>
```

Question 4:

Pour créer ce conteneur j'ai créer un dockerfile que vous pouvez trouver en pièce jointe Ce dockerfile permet de respecter ces contraintes: - Image de fédora à jour - Variable d'environnement - Utilisateur non privilégié

Pour que le système soit en lecture seule et que l'on crée un point de montage il faudra le préciser dans la commande.

Question 5:

On construit ensuite l'image avec la commande:

```
docker build -t imagehttp .
```

Puis on l'execute avec la commande suivante:

```
docker run --read-only -v volhttp:/tmp -d --name httpserver -p 8080:8080
imagehttp
```

Il faudra d'abord créer un volume

Afin de simplifier les choses j'ai utilisé docker-compose, il suffit ainsi d'executer:

```
sudo docker-compose up -d
```

Et docker-compose créer les volumes les montes et lance le conteneur