TADJER Badr | TRAN Leo | ARRADI Naoufal
M1-APP-LS1

TP4

INSTALLATION ET CONFIGURATION DE FREERADIUS

1. **Installer FreeRadius sur une machine Linux.**

apt install freeradius

2. **Quel est le répertoire d'installation de FreeRadius ? Sur quels ports le serveur écoute les requêtes ? sous quel compte le serveur est lancé et pourquoi ?**

répertoire d'installation : /usr/sbin/freeradius
port d'écoute : 1812 (commande : freeradius -X ou dans /etc/freeradius/3.0/sites-enables/default.conf)
compte server : freerad (commande : ps faux)

3. **Lancer Freeradius en mode debug et vérifier son bon fonctionnement.**

lancer en mode débug : /sbin/freeradius -X

4. **Ouvrez le fichier de configuration clients.conf. Que déclare-t-on dans ce fichier ? assurez-vous que le client localhost est bien présent. Le paramètre secret est testing123 par défaut. A quoi sert ce paramètre ? Comment s'authentifie le point d'accès auprès du serveur ?**

path : /etc/freeradius/3.0/clients.conf
Ce fichier déclare les clients RADIUS.
Le secret testing123 sert à chiffrer et signer les paquets entre le NAS et FREERADIUS.
La sécurité du protocole RADIUS dépend entièrement de ce secret.
Le protocole radius permet de se connecter via un échange de paquets UDP, généralement sur le port 1812.

5. **Ouvrez le fichier users se trouvant dans le même répertoire et ajoutez un utilisateur.**

fichier : /etc/freeradius/clients.conf →

```
client   192.168.0.1/24 {
         secret          = passer
         shortname       = localhost
}
```

fichier : /etc/freeradius/users → testing Cleartext-Password := "passer"
redémarrage du serveur : /etc/init.d/freeradius restart

6. **Relancer votre serveur freeradius en mode debug puis tester l'authentification avec la commande radtest (n'oubliez pas de lire le manuel de cette commande). Capturer les paquets et faire une analyse des AVP échangés. Analyser les messages affichés sur la console.**

```
root@debian:/# /etc/init.d/freeradius restart
Restarting freeradius (via systemctl): freeradius.service.
root@debian:/# radtest testing passer localhost 0 testing123
Sent Access-Request Id 135 from 0.0.0.0:47947 to 127.0.0.1:1812 length 77
        User-Name = "testing"
        User-Password = "passer"
        NAS-IP-Address = 127.0.1.1
        NAS-Port = 0
        Message-Authenticator = 0x00
        Cleartext-Password = "passer"
Received Access-Accept Id 135 from 127.0.0.1:1812 to 127.0.0.1:47947 length 20
root@debian:/#
```

commande pour capturer → tcpdump -i lo

```
root@debian:/home/leo# tcpdump -i lo
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on lo, link-type EN10MB (Ethernet), snapshot length 262144 bytes
10:59:57.172910 IP localhost.58902 > localhost.radius: RADIUS, Access-Request (1), id: 0xa2 length:
77
10:59:57.173030 IP localhost.radius > localhost.58902: RADIUS, Access-Accept (2), id: 0xa2 length: 2
0
```

7. **Freeradius est un serveur AAA. Que signifie ce terme ? Interdire l'accès de votre utilisateur pour une plage horaire de votre choix. Quelles sont les autres restrictions possibles ? Que fait Freeradius en premier : authentification ou contrôle d'accès ? Pourquoi ?**

serveur AAA → c'est un serveur qui utilise un protocol qui réalise trois fonction : l'authentification, l'autorisation, et la traçabilité

fichier : /etc/freeradius/users → testing Cleartext-Password := "passer", Login-Time := "Al1000-1800"

TEST ET ANALYSE DES MÉTHODES EAP

1. **Si ce n'est pas encore fait, installer le package eapol_test Lire son manuel.**

apt install eapoltest
utiliser : eapol_test

2. **Pour chacune des méthodes suivantes, analyser l'échange entre le client et le serveur (nombre de requêtes/réponses, nature des messages dans le tunnel, génération finale d'une clé MK ...etc) et indiquer la robustesse de la méthode :**

   a. PEAP-MSCHAPv2
      - nombre de requêtes/réponses : 20
      - nature des messages dans le tunnel : EAP
      - génération finale d'une clé MK : Oui

   b. TTLS-EAPMD5
      - nombre de requêtes/réponses : 14
      - nature des messages dans le tunnel : RADIUS
      - génération finale d'une clé PMK : Oui

   c. TTLS-PAP
      - nombre de requêtes/réponses : 12
      - nature des messages dans le tunnel : RADIUS
      - génération finale d'une clé PMK : Oui

   d. TTLS-MSCHAPv2
      - nombre de requêtes/réponses : 14
      - nature des messages dans le tunnel : RADIUS
      - génération finale d'une clé PMK : Oui

   e. TTLS-EAP-MSCHAPv2
      - nombre de requêtes/réponses : 18
      - nature des messages dans le tunnel : RADIUS
      - génération finale d'une clé PMK : Oui

3. **Faites un tableau comparatif entre les deux méthodes EAP-PEAP et EAP-TTLS.**

	EAP-PEAP	EAP-TTLS
nature du message tunnel SSL	EAP	Radius
génération finale d'une clé PMK	Oui	Oui
Robustesse	Très bonne	Très bonne

4. **Que faut-il faire en plus pour utiliser EAP-TLS ? Que faire pour tester cette méthode ?**

Afin d'utiliser EAP-TLS, il faut disposer du certificat et de la clé du client.

Afin de tester cette méthode :
/home/user/cert.crt → certificat
/home/user/key.pem → clé
/home/user/eap-tls.conf → fichier conf
exécuter la commande → **eapol_test -c eap-tls.conf -s testing123**