

# CTF FACILE

## Tools

Groupe 4

5 Novembre 2021

### Outils utilisés

- Linux, Debian 10 pour l'OS
- Apache2 en tant que serveur web
- PostgreSQL pour la gestion de base de données
- NetFilter pour n'autoriser la réception de paquet que sur le port 80 (HTTP) et 22 (SSH)

### Script de déploiement

```
1 #!/bin/bash
2
3 #On retrouvera la sortie standard dans install.out.log et les erreurs dans install.err.log
4 exec >/tmp/install.out.log 2>/tmp/install.err.log
5
6 echo "----- INSTALLATION DES PAQUETS NECESSAIRES -----"
7 #Mise a jour des paquets et installation des paquets requis pour le deploiement du CTF
8 apt update
9 apt install -y apache2 php postgresql postgresql-client php-pgsql git expect
10
11 echo "----- IMPORTATION DU GIT-----"
12 #On importe le git contenant les sources pour le site web.
13 cd /home/debian/
14 git clone https://github.com/projetsecu/projetsecurite.git
15
16 #On copie les sources vers le dossier html/, c'est le dossier par default utilise par Apache
17 cp /home/debian/projetsecurite/ctf_facile/web/* /var/www/html/
18
19 #On supprime l'index.html qui est le fichier installe par default, ici on n'en a plus besoin
20 rm /var/www/html/index.html
21
22 #Comme toutes les actions sont realises par l'utilisateur root, les sources ne sont pas
23 #accessibles
24 #aux autres utilisateurs ce qui posent certains problemes au serveur pour l'acces a ces
25 #sources on change donc les droits
26 chmod 777 -R /var/www/html/
27
28 echo "----- CONFIGURATION DE POSTGRESQL-----"
29 #On demarre le service PostgreSQL
30 systemctl enable --now postgresql
31
32 #On laisse un timing de 2 secondes le temps que le service demarre correctement
33 set timeout 2
34
35 #Creation du mot de passe pour l'utilisateur postgres, on utilise la seed du noyau
36 #on obtient donc un mot de passe "pseudo-aleatoire"
37 MYMSG=$(cat /proc/sys/kernel/random/uuid | sed 's/[-]//g' | head -c 6; echo;)
38
39 #On associe a l'utilisateur postgres le-dit mot de passe
40 sudo -u postgres psql -c "ALTER USER postgres WITH password '$MYMSG'"
41
42 #On cree un nouvel utilisateur qui s'appelle admin
43 sudo -u postgres createuser -d admin
```

```

44 #On lui associe un mot de passe (c'est le mdp a trouver ^^)
45 sudo -u postgres psql -c "ALTER USER admin WITH password 'kostadinkostadinovic'"
46
47 #On cree une base de donnee 'ctf_facile' dont le proprietaire est amdin
48 sudo -u postgres createdb ctf_facile -O admin
49 sudo -u postgres psql -d ctf_facile -c "CREATE TABLE users (id varchar(25) PRIMARY KEY, passwd
      varchar(50));" #Creation de la table avec id et password
50
51 #On ajoute a la database differentes donnees contenues dans un fichier .csv nomme bdd_users.
      csv
52 sudo -u postgres psql -U postgres -d ctf_facile -c "COPY users FROM '/home/debian/
      projetsecurite/ctf_facile/bdd_users.csv' WITH (FORMAT CSV, HEADER, DELIMITER ',', QUOTE
      ''');"
53
54 sudo -u postgres psql -c "alter database ctf_facile owner to admin;"
55 sudo -u postgres psql -d ctf_facile -c "alter table users owner to admin;"
56
57 #Creation d'une nouvelle table pour gerer le bannissement des adresses IP apres un certain
      nombre de requetes faites par celles-ci
58 sudo -u postgres psql -d ctf_facile -c "CREATE TABLE brute_force (IP varchar(15) PRIMARY KEY,
      count int, first_fail int);"
59 sudo -u postgres psql -d ctf_facile -c "alter table brute_force owner to admin;"
60
61 #On redemarre le service PostgreSQL
62 systemctl restart postgresql
63
64 echo "----- DEMARRAGE DU SERVEUR WEB APACHE 2-----"
65 #On demarre Apache
66 systemctl start apache2
67
68 echo "----- CONFIGURATION DES REGLES DE FILTRAGE IP-----"
69 #OUVERTURE D'UN PORT SSH ET D'UN PORT HTTP UNIQUEMENT
70
71 #Suppression de toutes les regles deja existantes
72 iptables -F
73 iptables -X
74
75 #On accepte les paquets en loopback
76 iptables -t filter -A INPUT -i lo -j ACCEPT
77
78 #On accepte les paquets arrivant et partant du port 22 et 80 (port ssh, port http) on jette
      tout le reste
79 iptables -A INPUT -p tcp -m tcp -m multiport ! --dports 80,443 -j DROP
80
81 #ON MODIFIE LES FICHIERS SUDOERS ET SSHD_CONFIG
82 #update /etc/sudoers.d/90-cloud-init-users
83 #take away sudo for default ubuntu user
84 set timeout 2
85 SUDOER_TMP="$(mktemp)"
86 sudo cat /etc/sudoers.d/90-cloud-init-users > ${SUDOER_TMP}
87 echo 'debian ALL=(ALL) ALL' > "${SUDOER_TMP}"
88 sudo visudo -c -f ${SUDOER_TMP} && sudo cat ${SUDOER_TMP} > /etc/sudoers.d/90-cloud-init-users
89 SUDOER_TMP="$(mktemp)"
90 sudo cat /etc/sudoers.d/debian-cloud-init > ${SUDOER_TMP}
91 echo 'debian ALL=(ALL) ALL' > "${SUDOER_TMP}"
92 sudo visudo -c -f ${SUDOER_TMP} && sudo cat ${SUDOER_TMP} > /etc/sudoers.d/debian-cloud-init
93 rm "${SUDOER_TMP}"
94
95 set timeout 2
96 #On autorise la connexion ssh avec l'utilisateur root qui est desactive par default
97 sed -i -e 's/#PermitRootLogin prohibit-password/PermitRootLogin yes/g' /etc/ssh/sshd_config
98 /etc/init.d/ssh restart
99
100 #On cree un mot de passe a l'utilisateur root
101 ROOT_PASSWD=$(expect -c "
102 set timeout 5
103 spawn passwd root
104 expect \"New password:\"
105 send \"kostadinkostadinovic\r\"
106 expect \"Retype new password:\"
107 send \"kostadinkostadinovic\r\"
108 expect eof")
109
110 echo "$ROOT_PASSWD"
111
112

```

```
113 #ON AJOUTE LE FLAG
114 touch /root/flag.txt
115 echo "P9GuZi82kG69V4idd8HiR495Gz3mrJmJ" > /root/flag.txt
116
117 #On supprime le git on n'en a plus besoin
118 rm -r /home/debian/projetsecurite/
```