

tests. This can be achieved in number of steps as:

- One can find the Oracle version through scanning.
- The HTTP verb request XML DB also provides an ample amount of information.
- Packet dissection at the network level. For this the auditor should know the packet design.
- The auditor can use one of many tools available to discover the Oracle version.

For Example see Listing 6. Another way to do this is by using a tool (see Listing 7)  
It's easy to detect the Oracle version.

### Oracle Running Service SID

The SID of running Oracle server is required for in depth analysis while auditing. If an auditor is not able to find the target SID, he will not be able to launch further attacks. Because the SID is such a critical point to the success of the audit, the auditor must discover the SID prior to attack the target. Let's see:

- It has been noticed several times that Host name is same as the SID of service running on the target. So it's good to give a try for Hostname as SID for Oracle.
- One can brute force or perform dictionary attacks to find the Oracle SID. The default list of SID can be discovered.

Let's see:

```
root@knock /cygdrive/d/knock/audit/oak
$ ./ora-getsid.exe 172.16.25.5 1521
sidlist.txt
Found SID: IRIS
```

### Oracle Default Username Enumeration and Password Control

The Oracle default accounts play a small role in hacking most of the Oracle servers. There are a number of default accounts listed. The administration of these accounts requires a basic core knowledge. The system administrators usually lack this skill, which serves as a positive point for hackers while breaking into the servers. The complexity is really high. Let's analyze the sys account

case. It's a default account. Default permissions are set on it. But it exists in a different context in sys as DBA account. Most of the administrators forget or don't know the consequence of this type of configuration. Even default account used for SNMP (i.e. dbsnmp) is least protected. It's good to have a

deeper knowledge of account settings in Oracle which proves beneficial from a testing point of view. The auditor follows certain steps for example:

- The auditor must perform dictionary attack or brute force attack on the default users. Example:

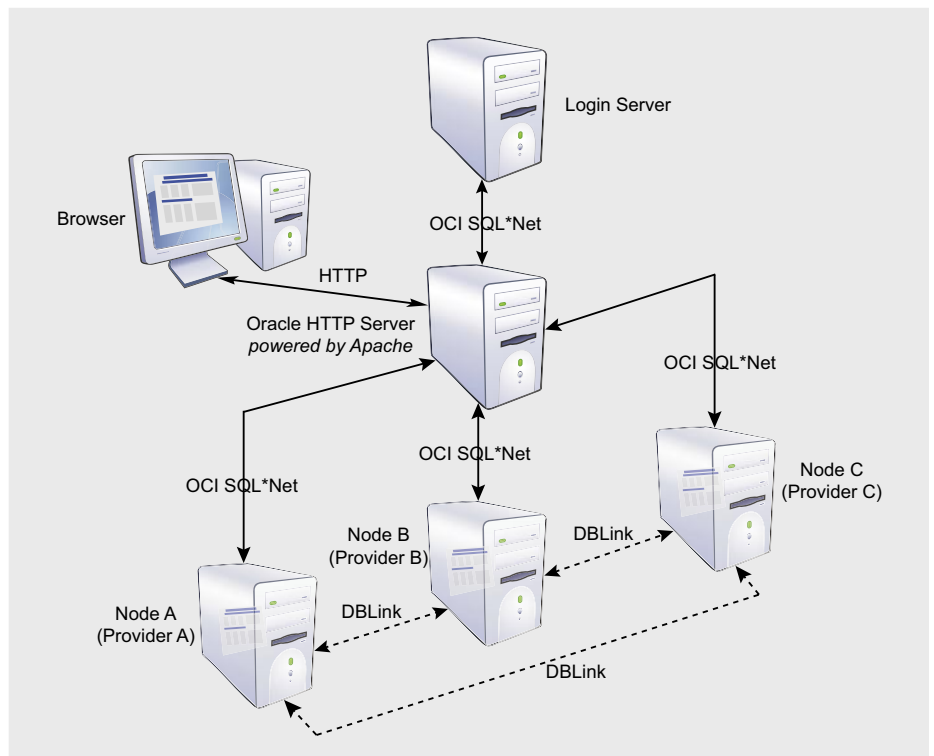


Figure 1. Oracle Database Layout

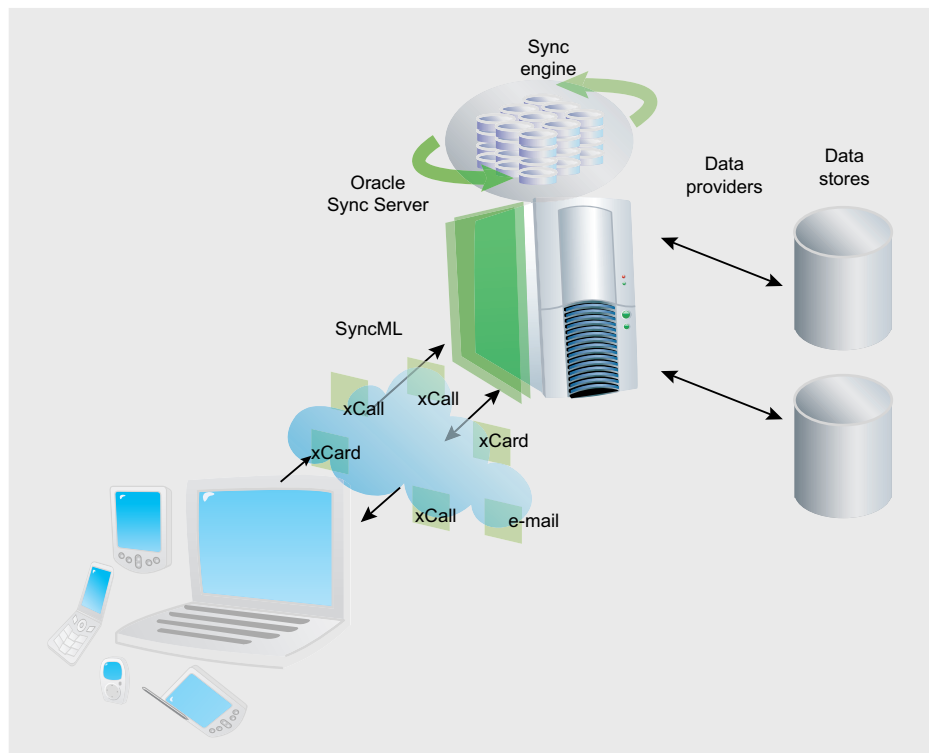


Figure 2. Oracle Syncing

```
$ ./ora-pwdb brute 172.16.25.5 1521 IRIS Version: Oracle9i Enterprise Edition
SYS passwords.txt Release 9.2.0.1.0 - 64bit Production
```

### Listing 7. Oracle Version check through TNS Querying

```
root@knock /cygdrive/d/knock/audit/oak
$ ./ora-ver.exe -l 172.16.25.5 1521
Packet: 1 Size: 69 Type: TNS_ACCEPT

0000 00 45 00 00 02 00 00 01 34 00 01 08 00 7F FF .E.....4....
0010 00 01 00 2D 00 18 0D 01 28 44 45 53 43 52 49 50 ...-....(DESCRIP
0020 54 49 4F 4E 3D 28 54 4D 50 3D 29 28 56 53 4E 4E TION=(TMP=)(VSNN
0030 55 4D 3D 31 35 33 30 39 32 33 35 32 29 28 45 52 UM=153092352)(ER
0040 52 3D 30 29 29 R=0))

Packet: 1
Size: 338
Type: TNS_DATAData Flags: 00
Type: Unknown

0000 01 52 00 00 06 00 00 00 00 54 4E 53 4C 53 4E .R.....TNSLSN
0010 52 20 66 6F 72 20 48 50 55 58 3A 20 56 65 72 73 R for HPUX: Vers
0020 69 6F 6E 20 39 2E 32 2E 30 2E 31 2E 30 2D 20 ion 9.2.0.1.0 -
0030 50 72 6F 64 75 63 74 69 6F 6E 0A 09 54 4E 53 20 Production..TNS
0040 66 6F 72 20 48 50 55 58 3A 20 56 65 72 73 69 6F for HPUX: Versio
0050 6E 20 39 2E 32 2E 30 2E 31 2E 30 2D 20 50 72 n 9.2.0.1.0 - Pr
0060 6F 64 75 63 74 69 6F 6E 0A 09 55 6E 69 78 20 44 oduction..Unix D
0070 6F 6D 61 69 6E 20 53 6F 63 6B 65 74 20 49 50 43 omain Socket IPC
0080 20 4E 54 20 50 72 6F 74 6F 63 6F 6C 20 41 64 61 NT Protocol Ada
0090 70 74 6F 72 20 66 6F 72 20 48 50 55 58 3A 20 56 ptor for HPUX: V
00A0 65 72 73 69 6F 6E 20 39 2E 32 2E 30 2E 31 2E 30 ersion 9.2.0.1.0
00B0 20 2D 20 50 72 6F 64 75 63 74 69 6F 6E 0A 09 4F - Production..O
00C0 72 61 63 6C 65 20 42 65 71 75 65 61 74 68 20 4E racle Bequeath N
00D0 54 20 50 72 6F 74 6F 63 6F 6C 20 41 64 61 70 74 T Protocol Adapt
00E0 65 72 20 66 6F 72 20 48 50 55 58 3A 20 56 65 72 er for HPUX: Ver
00F0 73 69 6F 6E 20 39 2E 32 2E 30 2E 31 2E 30 2D 20 sion 9.2.0.1.0 -
0100 20 50 72 6F 64 75 63 74 69 6F 6E 0A 09 54 43 50 Production..TCP
0110 2F 49 50 20 4E 54 20 50 72 6F 74 6F 63 6F 6C 20 /IP NT Protocol
0120 41 64 61 70 74 65 72 20 66 6F 72 20 48 50 55 58 Adapter for HPUX
0130 3A 20 56 65 72 73 69 6F 6E 20 39 2E 32 2E 30 2E : Version 9.2.0.
```

### Listing 8. Oracle Standard Errors

```
TNS-12518: TNS:listener could not hand off client connection
TNS-12560: TNS:protocol adapter error
TNS-00530: Protocol adapter error
TNS-12545: Connect failed because target host or object does not exist.
ORA-12154 TNS:could not resolve service name
```

- With the Partitioning, OLAP and Oracle Data Mining options
- JServer Release 9.2.0.1.0 – Production
- SYS must log in as SYSDBA!!! Password is MANAGER
- connection to sys should be as sysdba or sysoper

One can see the password which is gathered after the brute force attack.

- Try to find out the active user account and locked account on the target system.

```
root@knock /cygdrive/d/knock/audit/oak$ ./ora-userenum.exe
172.16.25.5 1521 IRIS userlist.txt
> 172.16.25.5_Oracle_user.txt
```

ME, OSM, SYS, SYSTEM, CTXSYS are the default users in the file.

- Try to use the same username and password for logging into the Oracle servers.

The steps provide a bundle of knowledge while auditing.

### Attacking Critical Oracle Service

After performing number of steps, the process should be implemented. It aims at finding the most critical service listen on the target by overall vulnerability analysis. It has been noticed that Oracle MTS and Oracle XML DB can only be used to discover information but cannot be exploited as such. The reason is that remote connections can not be set and queries can not be executed.

Moreover these are functions at a lower layer for providing efficiency and reliability but not connection oriented service.

The Oracle Listener service is always at high risk and needs to be dissected. The major problem found in this service is not configured by administrators and is presented as such. This flaw is quite common in Oracle versions 10. Oracle 8 and 9 versions are most vulnerable. This is because:

- no password is set for Listener,
- no Administrator restrictions are there,

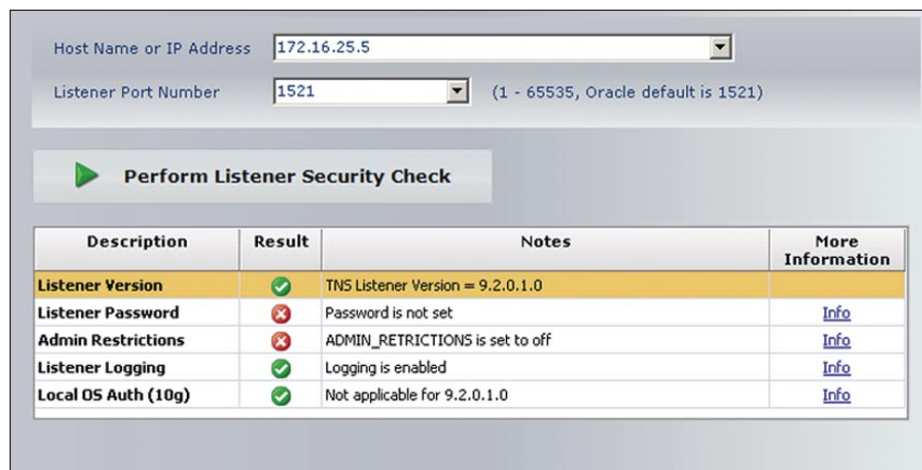


Figure 3. Oracle TNS Listener Checks

- attacker can execute commands remotely,
- attacker can set password for Listener and control the database connection,
- attacker can execute rogue queries for finding password hashes, etc.

All this is possible. Two different tests are carried to check the enumeration of configured TNS listener. The tool by Integrity Company is used for this. Let us see the output in Oracle version 9. By default Oracle version 10 is safe.

Let's see Figure 3.

The TNS Listener is vulnerable on this Oracle version. So one can see the attack surface it generates. Remember the attack driven Oracle Listener is one of the high risk vulnerability in Oracle servers.

## Setting Remote Interface in a Blatant Way

The setting of remote connection by a Oracle Client is one of the rooted problem for executing queries. There are number of ways which can set remote connections. The real network mapping needs to be understood. First we look at the ways the remote connection can be set to Oracle TNS Listener.

- The most general way is to trace the SID and set the Oracle Client from Oracle suite.
- Designing Scripts and programs to run command remotely.
- Remote connection through SQL\*Plus with defining individual database connection string.
- Oracle thin Clients, the most efficient way while auditing.

It has been noticed that setting remote connection by Oracle client software is bit hard task and most of the time result in errors. I think for better control the interface is required so that number of queries can be run. Some of the listener errors are listed. It depends a lot on the Host which Oracle server is set on. Let's see Listing 8.

The error in red is the most common error recognized by auditors while performing tests. The scripts are useful for running limited queries. But for core testing a proper connection is required.

By looking at the time window the auditor has to use Oracle thin client software. This connection software is run through properly selected database drivers for a specific database. After setting a proper database driver, the Oracle account credentials are required to set a remote connection in the context of the user. This is the most targeted way to complete the task.

## Oracle Post Attack Surface

This step depends on the depth the auditor wants to follow. It deals with digging deep into the database after successfully logging into system. The testing can be conducted under following situations as:

- Detecting hidden users in the database as

```
Select name from sys.user$ where
type#=1 minus select username from
SYS.dba_users;
```

- Auditing rootkits in the Oracle server.
- Testing databases for high end vulnerabilities like Cursor Snarfing , Lateral SQL Injections, etc.
- To intercept crypto keys through database crypto mechanism i.e. by dbms\_crypto.

These types of vulnerabilities are hard to trigger and detect. But still it's a part of Oracle Auditing Model.

## Oracle Vulnerability Scanning

Once the above process is completed, the last phase is to conduct in core vulnerability scanning for sustained vulnerabilities. The scan can be run through NISSUS vulnerability scanner. Always define the policy file and desired plug-in according to the target specification. This step

is almost everywhere when a test is conducted. The Oracle has been under the threat sign from last year because of plethora of vulnerabilities in Oracle database server and other components. The combination of NISSUS and METASPLOIT is a good entity setup for exploiting listed vulnerabilities in scanning. Let's look at some of Oracle vulnerabilities:

- Oracle time zone buffer overflow vulnerability.
- Oracle DBS\_Scheduler vulnerability.
- Oracle link overflow vulnerability.
- Oracle XML-SOAP remote Dos vulnerability.

It comprises of both Denial of Service and Buffer Overflow vulnerabilities. Once it is exploited, a system shell is generated based on it. While auditing, this should be the last step. The tests should be conducted in off hours when the production server is in state of reduced load when no user plugged into server.

So at this point of time we have seen the exact way to audit Oracle environment by following hacker psychology.

## Conclusion

The auditing at organization level requires a procedural implementation and testing model to find insecurities that are persisting in network. A responsible behavior is required but at the same time one needs to have hacker psychology to penetrate deep into systems. The Oracle auditing model discussed above suits in every possible environment where Oracle applications and server is to be tested. It has been structured against all type of stringencies and the required ways to perform testing relentlessly. At the end we should not forget our business rely on these technologies. A simple bug in implementation results in loss of business which I think no organization wants to face. So stay protected.

### Aditya K Sood, a.k.a. 0kn0ck

Aditya K Sood, a.k.a. 0kn0ck, is an independent security researcher and founder of SecNiche Security, a security research arena. He works for KPMG as a Security Auditor. His research articles have been featured in Usenix Login. He has given advisories to forefront companies. He is an active speaker at conferences such as EuSecWest, XCON, OWASP, and CERT-IN. His other projects include Mlabs, CERA, and TrioSec.

## On the 'Net

- <http://www.red-database-security.com/>
- <http://www.ngssoftware.com/>
- <http://www.oracle.com>
- <http://www.databasesecurity.com/>
- <http://www.secniche.org>