Oracle Security Alert #28

Dated: 06 Feburary 2002
Updated: 05 July 2002

1. Oracle mod_plsql v3.0.9.8.2 in Oracle9i Application Server v1.0.2.x (Oracle9iAS v1.0.2.x)

Description

a) Potential buffer overflow-related security vulnerabilities exist in the Oracle mod_plsql v3.0.9.8.2 of the Oracle9iAS, v1.0.2.x. By exploiting excessive string lengths in mod_plsql administration pages, a knowledgeable and malicious user can use Oracle9iAS v1.0.2.x to gain access to Windows OS accounts.

b) By attacking the Oracle mod_plsql directory path traversal mechanism using the double-URL encoding exploit, a knowledgeable and malicious user may be able to access readable OS files that may provide OS account information, and thereby gain access to the OS and Oracle9iAS.

c) By directly accessing the Oracle mod_plsql gateway configuration web pages, a knowledgeable and malicious user may remotely administer PL/SQL DADs without requiring authentication if default passwords for privileged database accounts are not changed in an Oracle9iAS production environment.

d) By sending a malformed authorization HTTP client header to the Oracle mod_plsql gateway, a knowledgeable and malicious user may be able to force a Denial of Service (DoS) attack on Oracle mod_plsql if no authorization type such as “Basic Apache” is established on mod_plsql.

e) A knowledgeable and malicious user may directly access privileged Oracle database server information or write cross-site script attacks to gain unauthorized access to an Oracle9iAS v1.0.2.x installation by utilizing the Oracle PL/SQL OWA and HTP packages that are installed by default during a “Typical” Oracle 9iAS and Oracle9i Database installation.

f) A knowledgeable and malicious user may be able to bypass PL/SQL authentication by substituting an application specific DAD in a given URI and therefore gain unauthorized access to Oracle9iAS.

g) By sending invalid requests to the mod_plsql administration pages, a knowledgeable and malicious user may be able to force a Denial of Service (DoS) attack on Oracle9iAS. By exploiting excessive the mod_plsql administration pages, a knowledgeable and malicious user can use Oracle9iAS v1.0.2.x to gain access to Windows OS accounts.

h) A knowledgeable and malicious user may be able to gain unauthorized access to some known PL/SQL procedures in the database

Oracle9iAS Releases affected
mod_plsql 3.0.9.8.2 in Oracle9iAS v1.0.2.x

Database Releases affected
mod_plsql 3.0.9.0.7 in Oracle8i Database Release 8.1.7.x
mod_plsql 3.0.9.0.7 in Oracle9i Database Release 9.0.1.x

Platforms affected
All (Unix, Linux and Windows)

Workarounds
To remove the potential vulnerabilities identified in c) and g)

- Change the adminPath entry located in $ORACLE_HOME$\Apache\modplsql\cfg**wdbsvr.app** to a path name that does not reveal the exact location of the true administration pages.

- Secure the mod_plsql administration pages by either setting the parameter “administrators” or by setting the “adminDAD” parameter in the DAD configuration file $ORACLE_HOME/Apache/modplsql/cfg/wdbsvr.app. Please refer to Chapter 2 of the guide “Using the PL/SQL Gateway” for more details on the configuration of this parameter.

To remove the security vulnerability identified in f) and h), add the following rule to the following file: $ORACLE_HOME$\Apache\modplsql\cfg**wdbsvr.app**

exclusion_list= *(*, account*, sys.*, dbms_*, owa.*, htp.*,htf.*

where account* is the pattern of URL’s which you wish to protect from direct browser access via the application specific DAD.

The remainder of the vulnerabilities identified above have patches (see below).

Patch Information
Oracle has fixed all of the potential vulnerabilities identified above (except those identified in part c), f) g) and h) that have workarounds) in Oracle mod_plsql v3.0.9.8.3b. In addition to the mod_plsql vulnerabilities reported above, Oracle iAS development has identified and fixed additional mod_plsql issues in Oracle mode_plsql v3.0.9.8.3b as well. Oracle mod_plsql v3.0.9.8.3b can be used with Oracle9iAS v1.0.2.x and Oracle8i Database only.

The patch numbers for the patched release of Oracle mod_plsql for Sun Solaris and Windows platforms are listed below.

Oracle9iAS, v1.0.2.x

Platform	**Patch Number**
Sun Solaris	2209455
Windows	2209455

Oracle9i Database, Release 9.0.1

Platform	**Patch Number**
Sun Solaris	2209455
Windows	2209455

Oracle8i Database, Release 8.1.7.x

Platform	**Patch Number**
Sun Solaris	2209455
Windows	2209455

Download these patches for your platform from Oracle’s Worldwide Support web site, Metalink, http://metalink.oracle.com. Activate the "Patches" button to get to the patches web page. Enter patch number indicated above and activate the "Submit" button.

Please check Metalink and/or Oracle Worldwide Support periodically for patch availability if the patch for your platform is not yet available.

Credits
Oracle Corporation thanks David Litchfield of Next Generation Security Software Limited for discovering and promptly bringing these potential security vulnerabilities to Oracle’s attention.

2. Oracle JSP, SQLJSP and WEB-INF Directories in Oracle9iAS v1.0.2.x

Description

a) A potential vulnerability exists in Oracle JSP and SQLJSP for Oracle9iAS v1.0.2.x. When a user requests a JSP page from a server running OracleJSP, the JSP page is translated, compiled and executed. The results are returned to the requesting client. During this process, three intermediate files are created that contain information such as database user IDs, passwords and business logic. A knowledgeable and malicious user may be able to deduce the location and names of these files from the URI of the JSP page request and thereby unauthorized access to information stored in these files.

b) Just as the Oracle JSP translation files can be accessed directly if not protected, so too can the JSP application's globals.jsa file. This file also contains information such as database user IDs, passwords and business logic. Thus, a knowledgeable and malicious user may gain unauthorized access to information stored in these files.

c) Several different components of OHS and OC4J use the J2EE-standards-defined WEB-INF directory to store information of types determined by the customer. According to the Servlet standard, the contents of these directories should not be available to browsers, so application developers and deployers may reasonably expect that these directories are protected and place files containing user IDs, passwords, business logic, or other confidential information in them. The default configuration of OHS (Oracle HTTP Server) does not prevent access to these directories. A knowledgeable and malicious user may be able to deduce the locations of these directories since they are defined in the J2EE standards, and thereby gain unauthorized access to information stored in these files.

Oracle9iAS Releases affected

v1.0.2.x

Platforms affected

All (Unix, Linux, Windows)

Workaround

To remove the potential vulnerability identified in a), delete the three intermediate files upon creation of the final executable or put the files into another protected directory.

To prevent access to the .java pages, edit **httpd.conf for this sub-application** (not the main httpd.conf) as follows:

For Solaris:

```
<DirectoryMatch /_pages/>
  Order deny,allow
  Deny from all
</DirectoryMatch>
```

For Windows:

```
<DirectoryMatch \_ pages\>
  Order deny,allow
  Deny from all
</DirectoryMatch>
```

For the potential vulnerability identified in b), prevent access to the globals.jsa file by adding the following entry to the **sub-application httpd.conf** file as follows:

```
<Files ~ "^\globals.jsa">
```

```
  Order allow,deny
  Deny from all
</Files>
```

For the potential vulnerability identified in c), prevent access to all WEB-INF directories by adding the following entry to the **main httpd.conf** file as follows:

For Windows AND Solaris:

```
<DirectoryMatch WEB-INF>
  Order deny,allow
  Deny from all
</DirectoryMatch>
```

Credits
Oracle Corporation thanks David Litchfield of Next Generation Security Software Limited for discovering and promptly bringing these potential security vulnerabilities to Oracle's attention.

Oracle Corporation thanks Matt Moore of Westpoint Limited for discovering and promptly bringing potential security vulnerability (c) to Oracle's attention.

3. XSQL 1.0.x in Oracle9iAS, v1.0.2.x

Description

a) A knowledgeable and malicious user can potentially gain unauthorized access to an existing Oracle9iAS installation configuration information by breaking out of virtual Web root to access $ORACLE_HOME/lib/XSQLConfig.xml via XSQL Servlets.

b) Depending upon how an XSQL file is written to the machine hosting Oracle9iAS or Oracle9i Database, a knowledgeable and malicious user can write an exploit by inserting a single quote at the end of the file name to remotely execute arbitrary SQL queries against Oracle9iAS and Oracle9i Database.

c) A knowledgeable and malicious user can use a demo XSQL style sheet to remotely run arbitrary SQL queries against Oracle9iAS and Oracle9i Database.

Oracle9iAS Releases affected
v1.0.2.x

Database Releases affected
Oracle9i Database

Platforms affected
All (Unix, Linux, Windows)

Workarounds
The details on all of the workarounds are available in the Security Release Notes on OTN at http://otn.oracle.com/docs/tech/xml/xdk_java/doc_library/Production9i/java/xsql/readme.html#ID9914.

To remove the potential security vulnerability identified in a), remove the XSQLConfig.xml configuration file from the virtual web root and put it in another protected directory in the operating system.

To remove the potential security vulnerability identified in b), remove all DEMO pages from XSQL production environments. Note that implementing the workaround identified for a) will assist in eliminating the potential security vulnerability identified in part b).

To remove the potential security vulnerability identified in c), remove all XSQL DEMO style sheets from XSQL production environments.

4. Default services, pages, environment variables and files in Oracle9iAS, v1.0.2.x

Description
A knowledgeable and malicious user can gain unauthorized access to Oracle9iAS by exploiting monitoring services, sample JSP pages, environment variables and shell script utilities that are available by default in a “Typical Installation” of Oracle9iAS, v1.0.2.x.

Oracle9iAS Releases affected
v1.0.2.x

Platforms affected
All (Unix, Linux, Windows)

Workarounds
Edit **httpd.conf** to prevent access to the following pages.

Dynamic Monitoring Services
http://oracleserver/dms0
http://oracleserver/dms/DMSDump
http://oracleserver/servlet/DMSDump
http://oracleserver/servlet/Spy
http://oracleserver/soap/servlet/Spy
http://oracleserver/dms/AggreSpy

Oracle Java Process Manager
http://oracleserver/oprocmgr-status
http://oracleserver/oprocmgr-service

Edit **httpd.conf** to prevent access to the "/cgi-bin" directory via the “/perl” directory. Alternatively, if Perl is not being used, then disable or remove this tool from the host machine.

Remove references to the following demos and environment variables from production servers.

http://oracleserver/demo/email/sendmail.jsp
http://oracleserver/demo/basic/info/info.jsp
http://oracleserver/cgi-bin/printenv
http://oracleserver/fcgi-bin/echo
http://oracleserver/fcgi-bin/echo2

Credits
Oracle Corporation thanks David Litchfield of Next Generation Security Software Limited for discovering and promptly bringing these potential security vulnerabilities to Oracle’s attention.