

January 19, 2005

Security Analysis

Oracle Critical Patch Update – January 2005 *Oracle E-Business Suite Impact*

Overview

Oracle Corporation released the first Critical Patch Update (CPU) on January 18, 2005. The CPU is a collection of security related patches for the Oracle Database, Oracle Application Server, Oracle Collaboration Suite, and Oracle E-Business Suite. There are 23 vulnerabilities addressed in the CPU ranging from buffer overflows to SQL injection to denial of service (DoS) issues. Most of the vulnerabilities are high risk and should be addressed quickly by organizations.

This analysis provides additional information on the vulnerabilities and patches released in the CPU as it relates to the Oracle E-Business Suite (Oracle Applications 11i). The objective of this analysis is to assist IT managers and database administrators in assessing the impact on their Oracle Applications 11i implementations and the risks associated with the vulnerabilities, especially since the CPU addresses a large number of vulnerabilities and impacts all layers of the Oracle Applications technology stack.

On the surface, the Critical Patch Update appears to be similar to other security patches in terms of effort and scope. This is true for the database server and E-Business Suite patches, however, the Oracle Application Server and Oracle Developer 6i security patches may require significant and time consuming upgrades to the technology stack for many non-11.5.10 implementations.

Assessment of Vulnerabilities

You should assume these vulnerabilities can be easily exploited by someone with only limited Oracle knowledge. With the detailed information provided by Oracle and the ability to determine the exact vulnerability through reverse engineering of the Oracle patches, we believe usable exploit code will be readily available on the Internet for some of these vulnerabilities in next 30 to 60 days.

Oracle Database Vulnerabilities (DB01 – DB17)

All the database vulnerabilities can easily be exploited using a direct SQL*Net connection to the database or indirect connection (e.g., reporting tools) that allows a user to execute arbitrary SQL statements. This is especially a problem with Oracle Applications, unless specifically blocked, given that anyone can access the database and logon using the APPLSYS PUB account; the APPLSYS PUB database account has a well known

password and can not be disabled. A number of these vulnerabilities can be exploited using the APPLSYSPUB database account, since APPLSYSPUB has access to some of the vulnerable database packages by default.

In addition, almost all these vulnerabilities can be exploited via the SQL injection vulnerabilities in Oracle Applications also fixed in the CPU (see the Oracle Applications vulnerabilities). By exploiting these vulnerabilities simultaneously, termed a blended threat, attacks on the database server can be made using a web browser, even though the attacker does not have direct access to the database server. This possibility will have the greatest impact on those implementations where Oracle Applications web servers are directly connected to the Internet.

Oracle Application Server Vulnerabilities (AS01 – AS03)

The Report Server issue (**AS01**) can be exploited to retrieve the APPS database account password by accessing a standard administrative URL in the Report Server. This issue has been known for a long time and Integrigy released a security alert in November 2002 to address this issue [\[link\]](#). It was corrected in the AutoConfig templates as of June 2003.

Oracle Applications is vulnerable to the Forms vulnerability (**AS02**), although, this issue is a denial of service bug in the Forms Server and should be considered low to medium risk.

The mod_plsql vulnerability (**AS03/DB17**) can not be exploited in the default configuration of Oracle Applications since web access to the OWA_OPT_LOCK database package is not allowed. To verify, check that OWA_OPT_LOCK does not exist in the table APPLSYS.FND_ENABLED_PLSQL. One exception is that if Oracle Portal is installed in the Oracle Applications database, then potentially the OWA_OPT_LOCK vulnerability can be exploited through the portal DAD.

Oracle E-Business Suite Vulnerabilities (APPS01 – APPS02)

Both of these vulnerabilities are SQL injection issues in web pages accessible by anyone with a web browser, thus anyone can execute arbitrary SQL statements in the database. By exploiting a SQL injection bug, an attacker is able to select, insert, update, or delete data in the database. We believe an attacker will choose to exploit known buffer overflows in standard Oracle functions rather than attempt traditional SQL injection, since exploiting function buffer overflows is much easier and may allow an attacker to gain access to the database server operating system (for more information on SQL injection [\[link\]](#)).

We disagree with Oracle's risk assessment for **APPS01** and believe it is much easier to exploit this vulnerability than stated by Oracle. It is very easy to obtain a valid session for Oracle Applications without a valid application user account or password. Several Oracle Application modules automatically grant valid application sessions (e.g., iStore) without any authentication and the module may not have to be installed or configured to obtain a session.

These two vulnerabilities should be considered high risk for any Oracle Applications implementations with web servers directly connected to the Internet.

Patch Analysis

For the Oracle E-Business Suite, install the patches as specified in "Note 293741.1 Oracle Critical Patch Update January 2005 Pre-Installation Note for Oracle E-Business Suite" and you should also review the Pre-Installation Notes for the Oracle Database and Oracle Application Server prior to installing those patches.

Oracle Database Patches

The database portion of the patch fixes over 40 security and non-security related bugs in most components of the database. The database portion of the patch is straight forward. The Oracle HTTP Server (OHS) may be installed if you have upgraded to Oracle 9iR2, thus, the mod_plsql component patch may have to be installed.

Oracle Database security patches are cumulative, therefore, the patches for Oracle Security Alert #68 and other patches are included. However, the patches for Oracle Security Alert #62 (3169446) and #48 (2611482) are not included and may need to be applied before this patch.

Testing

An abbreviated testing cycle should be performed similar to testing for a minor database updated (e.g., 9.2.0.4 to 9.2.05). We can not provide specific recommendations as to where to focus testing efforts since the database patch touches all aspects of the database.

Oracle Application Server Patches

1.0.2.1.x

The patches for the Oracle Application Server require 1.0.2.1.x environments to upgrade to 1.0.2.2.x is significant as this upgrade requires a full installation of 1.0.2.2.x and implementation of AutoConfig as well as the installation of the latest Developer 6i PatchSet. All implementations installed using Rapid Install from 11.5.1 to 11.5.5 must upgrade or have previously upgraded in order to install the security patch.

1.0.2.2.x

Patch 3169446 is from Oracle Security Alert #62, thus you may have already applied it. Patch 3072811 is a roll-up patch and may already have been applied, especially if you have upgraded to FND.H or 11.5.10.

The next step in the process in the process is to upgrade the \$IAS_HOME to 8.1.7.4 with patch 2376472. This is not a trivial task, but requires only a few steps.

The last step is to install patch 4005880, which patches the 8.1.7.4 Oracle client in the \$IAS_HOME. The patch includes a number of additional steps including an upgrade of JInitiator to 1.1.8.24. It is unclear based on the information provided by Oracle as to what the exact risk is by not applying this patch and prerequisite patches. The only stated

vulnerability is with the mod_plsql OWA_OPT_LOCK database package, which may not be vulnerable in your environment. Most likely the security fixes associated with 4005880 are related to vulnerabilities in the Oracle client software that were addressed in Oracle Security Alert #68.

Testing

Since these patches impact both the Apache and JInitiator, a brief walk-through and execution of critical web pages and Forms should be performed to test the patches.

To verify that the mod_plsql portion of patch 4005880 was successfully applied, the following SQL should return one row –

```
select text from dba_source
where owner = 'SYS'
and name = 'OWA_OPT_LOCK'
and upper(text) like '%PROCEDURE_VALIDATE_OBJECT_NAME%';
```

No additional testing should be required for the mod_plsql component of the patch.

Oracle Developer 6i Patches

The patches for Forms and Reports are dependent on either Developer 6i PatchSet 15 or 16 and the installation any Developer 6i PatchSet is a significant undertaking. If you are not running Patchset 15 or 16, you should assess the risk of a potential denial of service attack against your Forms Server and consider only implementing the workaround for the Reports Server at this time.

The Oracle Forms Server patch (4107597 for PatchSet 15 and 3201067 for PatchSet 16) is simple to install and should require no testing other than making the Forms Server still works.

The Oracle Reports Server patch 3281229 is also simple to install and should require minimal testing. Although, the workaround is more effective as it blocks access to all the Report Server administrator functions and should be implemented even if the patch is applied. For environments running AutoConfig since June 2003, the workaround should already be implemented.

The last step of the Developer 6i security patches is 4005880, which patches the 8.0.6 Oracle Home client. The patch includes a number of additional steps including an upgrade of JInitiator to 1.1.8.24. It is unclear based on the information provided by Oracle as to what the exact risk is by not applying this patch and prerequisite patches. The only stated vulnerability is with the mod_plsql OWA_OPT_LOCK database package, which may not be vulnerable in your environment. Most likely the security fixes associated with 4005880 are related to vulnerabilities in the Oracle client software that were addressed in Oracle Security Alert #68.

Alternate Workaround for Reports Server (AS01)

The preferred work-around for **AS01** (Reports Server APPS Password Disclosure) is to include the following lines near the end of the \$IAS_HOME/Apache/Apache/conf/apps.conf file –

```
SetEnv REPORTS60_CGINODIAG YES
SetEnv REPORTS60_OWSNODIAG YES
```

If you are running AutoConfig, check the current settings in apps.conf to see if the above lines are included as the AutoConfig templates have included these settings since June 2003 (apps.conf = 115.25 or higher). Both values need to be set to "YES", however, the values may have been changed at some point in order to troubleshoot the Report Server.

To test the workaround, access the following URLs –

Windows -

```
http://<host>:<port>/dev60cgi/rwcgi60.exe/showjobs?server=REP60_<sid>
```

UNIX/Linux -

```
http://<host>:<port>/dev60cgi/rwcgi60/showjobs?server=REP60_<sid>
```

If the work-around is correctly applied, when accessing the above URLs you should receive one of the following error messages –

```
"Oracle Reports Server CGI Error: The requested URL was not found,  
or cannot be served at this time."
```

```
"Incorrect usage."
```

Oracle E-Business Suite Patches

The two required Oracle Applications patches were released in February 2001 (1632947) and June 2002 (2343999). We believe these two patches are fairly benign and should not cause any issues.

Testing

Patch 2343999 changes the Oracle Applications authentication mechanism for mod_plsql (direct access to database packages from a web browser). Critical web pages should be accessed to ensure the authorization mechanism is working correctly for all sites running versions of Oracle Applications 11.5.6 or earlier that have not applied mini-pack FND.F or later.

We do not believe any specific testing is warranted for patch 1632947.

Patching Strategy

With the number of patches required and testing effort, the patches need to be prioritized. A number of factors will affect the order and timing of the patches –

- Are the Oracle Applications servers directly connected to the Internet?
- Does the Oracle Applications database contain sensitive data (employee information, credit card numbers, etc.)?
- Is the internal network secure?
- Can anyone directly connect to the database and execute SQL statements?
- Is there a large technical or Oracle skilled user population?

Every organization and Oracle Applications environment is unique and will have individual requirements, testing procedures, and criteria for applying security patches. The following guidelines are meant to be a reference and guide to assist you in determining how you will apply the patches.

High Risk and Secure Environment Strategy

- Apply the Oracle E-Business Suite patches as soon as possible
 - 1632947
 - 2343999
- Implement work-around for Oracle Report Server (AS01) as soon as possible (if needed)
 - Requires Apache to be restarted
- Apply the Oracle Database patches as soon as possible or during next scheduled downtime
 - 4002909 (8.1.7.4), 400994 (9.2.0.4), or 4003006 (9.2.0.5)
- Evaluate the requirements and risks associated with applying the Oracle Application Server and Developer 6i patches

Non-High Risk Environment Strategy

- Apply the Oracle E-Business Suite patches as soon as possible or during next scheduled downtime
 - 1632947
 - 2343999
- Implement work-around for Oracle Report Server (AS01) as soon as possible (if needed)
 - Requires Apache to be restarted
- Apply the Oracle Database patches during next scheduled maintenance period
 - 4002909 (8.1.7.4), 400994 (9.2.0.4), or 4003006 (9.2.0.5)
- Evaluate the requirements and risks associated with applying the Oracle Application Server and Developer 6i patches

About Integrigy Corporation

Integrigy Corporation is a leader in application security for large enterprise, mission critical applications. Our application vulnerability assessment tool, AppSentry, assists companies in securing their largest and most important applications. AppDefend is an intrusion prevention system for Oracle Applications and blocks common types of attacks against application servers. Integrigy Consulting offers security assessment services for leading ERP and CRM applications.

AppSentry and AppDefend have been updated to detect and/or block the vulnerabilities addressed in the Oracle Critical Patch Update – January 2005.

For more information, visit www.integrigy.com

Integrigy Corporation
2052 Lincoln Park West, Suite 1301
Chicago, Illinois 60614 USA
888/542-4802
www.integrigy.com

Copyright © 2005 Integrigy Corporation.

Written by: Stephen Kost, Jack Kanter, and Bob Campbell

If you have any questions, comments or suggestions regarding this document, please send them via e-mail to alerts@integrigy.com.

Integrigy, AppSentry, and AppDefend are trademarks of Integrigy Corporation. Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

Integrigy's Vulnerability Disclosure Policy - Integrigy adheres to a strict disclosure policy for security vulnerabilities in order to protect our clients. We will never release detailed information regarding individual vulnerabilities and will only provide information regarding vulnerabilities that is publicly available or readily discernable. We do not develop or distribute any type of exploit code. We provide verification or testing instructions for specific vulnerabilities only if the instructions do not disclose the exact vulnerability or if the information is publicly available.

The Information contained in this document includes information derived from various third parties. While the Information contained in this document has been presented with all due care, Integrigy Corporation does not warrant or represent that the Information is free from errors or omission. The Information is made available on the understanding that Integrigy Corporation and its employees and agents shall have no liability (including liability by reason of negligence) to the users for any loss, damage, cost or expense incurred or arising by reason of any person using or relying on the information and whether caused by reason of any error, negligent act, omission or misrepresentation in the Information or otherwise.

Furthermore, while the Information is considered to be true and correct at the date of publication, changes in circumstances after the time of publication may impact on the accuracy of the Information. The Information may change without notice.