[NEWS] Multiple Vulnerabilities in Oracle Database (Trigger, Extproc, Wrapped Procedures, PL/SQL Injection)

Source: http://www.derkeiler.com/Mailing−Lists/Securiteam/2004−12/0101.html

---

From: SecuriTeam (*support_at_securiteam.com*)
Date: 12/27/04

```
To: list@securiteam.com
Date: 27 Dec 2004 12:03:35 +0200
```

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: http://www.securiteam.com
− − promotion

The SecuriTeam alerts list − Free, Accurate, Independent.

Get your security news from a reliable source.
http://www.securiteam.com/mailinglist.html

− − − − − − − − −

Multiple Vulnerabilities in Oracle Database (Trigger, Extproc, Wrapped Procedures, PL/SQL Injection)
−−−

the SELECT, INSERT, UPDATE and DELETE object privileges on this table. Consequently, anyone can cause the SDO_CMT_CBK_TRIG trigger to fire by deleting a row from the table. If we examine the text of the trigger we can see that, before the DELETE actually occurs, a list of functions are selected from the SDO_CMT_DBK_FN_TABLE and SDO_CMT_CBK_DML_TABLE tables and then these functions are executed. PUBLIC has no object privileges set for either of these tables so they can not insert their own function name.

However, the PRVT_CMT_CBK package owned by MDSYS has two procedures, CCBKAPPLROWTRIG and EXEC_CBK_FN_DML, that take as their parameters a schema and function name which are then inserted into the SDO_CMT_DBK_FN_TABLE and SDO_CMT_CBK_DML_TABLE tables. PUBLIC has the EXECUTE permission on the PRVT_CMT_CBK package and, as it has not been defined with the 'AUTHID CURRENT_USER' keyword, the package executes using the rights of MDSYS, the definer, and not the invoker. As a result of this anyone can indirectly insert function names into the SDO_CMT_DBK_FN_TABLE and SDO_CMT_CBK_DML_TABLE tables. Thus when a DELETE occurs on SDO_TXN_IDX_INSERTS anyone can influence what actions the SDO_CMT_CBK_TRIG trigger takes − in other words, anyone can get the trigger to execute an arbitrary function. What is more, this function, as it is being executed from the trigger will run with the privileges of MDSYS and an attacker can exploit this to gain elevated privileges.

The MDSYS.SDO_GEOM_TRIG_INS1 is vulnerable to SQL injection on both 9i and 10g.

The trigger executes the following:

```
.
.
EXECUTE IMMEDIATE
'SELECT user FROM dual' into tname;
stmt := 'SELECT count(*) FROM SDO_GEOM_METADATA_TABLE ' ||
'WHERE sdo_owner = '' || tname || '' ' ||
' AND sdo_table_name = '' || :n.table_name || '' '||
' AND sdo_column_name = '' || :n.column_name || '' ';
.
.
```

When an INSERT is performed on MDSYS.USER_SDO_GEOM_METADATA. The :new.table_name and :new.column_name can be influenced by the user and SQL injected. PUBLIC has the permissions to INSERT into this table. As such the trigger can be abused to gain MDSYS privileges − a DBA. The MDSYS.SDO_LRS_TRIG_INS trigger fires when an INSERT occurs on the MDSYS.USER_SDO_LRS_METADATA view. PUBLIC can insert into this view and so cause the trigger to fire. This trigger is vulnerable to SQL injection. Both Oracle 9i and 10g are affected.

It executes:

```
.
.
stmt := 'SELECT count(*) FROM SDO_LRS_METADATA_TABLE ' ||
```

```
' WHERE sdo_owner = ''' || UPPER(user_name) || ''' ' ||
' AND sdo_table_name = ''' || UPPER(:n.table_name) || ''' ' ||
' AND sdo_column_name = ''' || UPPER(:n.column_name) || ''' ';
EXECUTE IMMEDIATE stmt INTO vcount;
.
.
```

And :new.table_name and :new.column_name are user supplied in the insert statement. This is where an attacker can insert malicious SQL code.

2. Extproc Directory Traversal

The Oracle database server supports PL/SQL, a programming language. PL/SQL can execute external procedures via extproc. Over the past few years there has been a number of vulnerabilities in this area (<http://www.nextgenss.com/advisories/oraplsextproc.txt> http://www.nextgenss.com/advisories/oraplsextproc.txt, <http://www.nextgenss.com/advisories/ora−extproc.txt> http://www.nextgenss.com/advisories/ora−extproc.txt). Extproc has been found to suffer from a directory traversal problem that allows attackers access to arbitray libraries.

Extproc verifies that the library to be loaded is in the $ORACLE_HOME\bin directory. This is to ensure that libraries outside of this directory cannot be loaded. However, there exists a directory traversal issue whereby an attacker can break outside of this constraint. This can allow attackers to access libraries such as libc and msvcrt.dll. By calling the system() function attackers can run arbitrary OS commands.

3. Oracle Extproc Local Command Execution

No authentication takes place when extproc is asked to load a library and execute a function. This allows local users to run commands as the Oracle user (Oracle on UNIX and system on Windows). If configured properly, under 10g, extproc runs as nobody on *nix systems so the risk posed here is minimal but still present.

4. Multiple PL/SQL Injection Vulnerabilities

When a PL/SQL procedure executes, it does so with the permissions of the definer unless the AUTHID CURRENT USER keyword has been specified. In this case the procedure executes with invoker privileges. Any procedure that uses definer rights can be abused to gain elevated privileges if they are vulnerable to PL/SQL injection.

Known to be vulnerable are:

Owner	Procedure
SYS	DBMS_EXPORT_EXTENSION
WKSYS	WK_ACL.GET_ACL
WKSYS	WK_ACL.STORE_ACL
WKSYS	WK_ADM.COMPLETE_ACL_SNAPSHOT
WKSYS	WK_ACL.DELETE_ACLS_WITH_STATEMENT
CTXSYS	DRILOAD.VALIDATE_STMT

Each of these can be exploited to gain DBA privileges. Further, attacks can be affected via an Oracle Application Server without the attacker

having a user ID and password.
* Note − CTXSYS is not a DBA in 10g but is on 9i.

5. Wrapped Procedure Overflow
The code for PL/SQL procedures can be encrypted or "wrapped" to use the Oracle term. When a wrapped procedure is created a buffer overflow vulnerability can be triggered. By placing an overly token in the text of a procedure that has been wrapped with version 9 and stack based buffer is overflowed in the Oracle server when the procedure is created. Exploitation of this allows an attacker to run code as the Oracle user.

Vendor Status:
A patch (#68) was released for all the problems described above by Oracle. See <http://metalink.oracle.com/> http://metalink.oracle.com/ for more details.

Original Advisories can be found at:
<http://www.ngssoftware.com/advisories/oracle23122004I.txt>
http://www.ngssoftware.com/advisories/oracle23122004I.txt
<http://www.ngssoftware.com/advisories/oracle23122004B.txt>
http://www.ngssoftware.com/advisories/oracle23122004B.txt
<http://www.ngssoftware.com/advisories/oracle23122004C.txt>
http://www.ngssoftware.com/advisories/oracle23122004C.txt
<http://www.ngssoftware.com/advisories/oracle23122004H.txt>
http://www.ngssoftware.com/advisories/oracle23122004H.txt
<http://www.ngssoftware.com/advisories/oracle23122004J.txt>
http://www.ngssoftware.com/advisories/oracle23122004J.txt

ADDITIONAL INFORMATION

The information has been provided by <mailto:nisr@nextgenss.com>
NGSSoftware Insight Security Research.

==========

This bulletin is sent to members of the SecuriTeam mailing list.
To unsubscribe from the list, send mail with an empty subject line and body to:
list−unsubscribe@securiteam.com
In order to subscribe to the mailing list, simply forward this email to: list−subscribe@securiteam.com

==========
==========

DISCLAIMER:
The information in this bulletin is provided "AS IS" without warranty of any kind.
In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.