Security Alert #20
Reference Date: 10/18/01

Oracle File Overwrite Security Vulnerability

Overview

There is a potential security vulnerability associated with the Oracle binary `oracle` on UNIX platforms. A non-privileged user (such as “nobody”) invokes the `oracle` executable: as a result of the presence of the `SETUID` bit, the executable can be forced to write to a trace file in `ORACLE_HOME/rdbms/log` directory and thereby overwrite existing log files or create new (unauthorized) files. The non-privileged user can also point the environment variable, `ORACLE_HOME`, to an arbitrary directory in the operating system and thereby corrupt other files as well.

Products

All Oracle database server releases (8.0.x, 8.1.x and 9.0.1)

Platforms

All Unix platforms

Workaround

Change the file permissions on the `oracle` executable as follows:

```
% chmod o-x oracle
```

Notes

The workaround suggested above will permit only the owner of the `oracle` executable and users defined in the OS DBA group to run the `oracle` executable directly. With the execute permissions for “others” removed, other users cannot connect to an Oracle database server using the BEQ driver. If the BEQ driver is being used to connect to an Oracle database, a client program (such as SQLPLUS) will fork its processes and try to execute the `oracle` executable directly. This operation will fail because such a client program will run with the OS user’s privileges who no longer has execute permission on the oracle executable. To avoid this problem, local users must connect to an Oracle database using the IPC driver which makes it possible to connect to a TNS listener listening on an Oracle database. The TNS listener will need to be started by a user that has `execute` permissions on the `oracle` executable.

Patches

The potential security vulnerability will be code-fixed in the next release of the Oracle database server which is Oracle9*i*, Release 2, only. All other releases of the Oracle database (8.0.x, 8.1.x and 9.0.1) must use follow the workarounds specified above to circumvent the potential security vulnerability.

Credits

Oracle wishes to thank Juan Manuel Pascual EscribÃ for discovering these vulnerabilities and promptly bringing them to Oracle's attention.