Blackford, M. Bret

From: Tidwell, Bill
Sent: Thursday, May 15, 2008 3:08 PM
To: Blackford, M. Bret; Abbene, Michael
Cc: Greenway, Roy; Telle, Paul; Berberich, Bob; Kelley, Allen; Staten, Don; Paglino, Julie
Subject: RE: 12 May 2008 Mtg Notes: IA & IS discussion of Oracle Db risks/controls

The problems Bret noted are quite serious and we thank him for bringing them to our attention. The level to which he was able to penetrate the database is alarming.

However I would like to point out that we were looking at many of these problems and a number of counter-measures are in progress.. The problem of being able to see passwords passed to sqlplus has already been corrected. Bret that changed was done last night. Developers are also being instructed in the correct method of coding calls to sqlplus to prevent the passwords from being displayed.

A procedure has also been developed that allows developers to remove passwords from scripts that call sqlplus. We have requested the developers to test it The script does require some minor coding changes on existing scripts. However the number of scripts requiring coding changes is fairly small. The script will even block the developer from seeing the password, even if they have sudoed to a user that does have access to the passwords. The information on both these changes has already been forwarded to Bret. We will let you know Bret when we implement the scripts and change the existinging passwords.

It should be fairly easy to create a similar script to use with ftp sessions. For some time now, where possible we have been converting ftp scripts to use sftp, that log on using public/private key pairs. This is the preferred method but won’t work in every area. I’m not aware of any current usage of telnet with embedded passwords.

Use of the orabf tool did take, at least me by surprise and is a concern. Note the actual cracking of the passwords takes place offline, but the database must have been previously accessed to get the password hashes (encrypted passwords). As Bret noted the longer and more complex the password the longer it takes to crack. A short password starting with a may take 30 minutes to crack. A complex password with special characters and numbers could takes weeks or might never be cracked. All the passwords will be changed once the scripts to hide the current passwords have been implemented. It is also necessary to document all the locations where the passwords currently might be stored. For instance WebSphere internally stores the ellipse database ID’s password for use with MMLM. WebView must be rebuilt should the password for the user elluser be changed. We are also looking at the possibility of scripting the orabf tool to allow us to run our own tests and find weak passwords that exist.

The problem with restricting access to dbausers table is there currently is no easy method to do that. Oracle allows you to give access to selected tables or all tables. It does not have a all tables except option. However Bob is still looking into that area. Our thinking at the moment is to make the passwords complex enough to seriously limit the possibility of them being cracked.

Thanks

From: Blackford, M. Bret
Sent: Thursday, May 15, 2008 12:40 PM
To: Abbene, Michael
Cc: Greenway, Roy; Telle, Paul; Berberich, Bob; Kelley, Allen; Tidwell, Bill; Blackford, M. Bret
Subject: 12 May 2008 Mtg Notes: IA & IS discussion of Oracle Db risks/controls

Michael,

During some of my recent work I was able to fully compromise the Company’s various Oracle databases, including production. On Monday of this week, I met with Roy Greenway, Paul Telle, Bob Berberich, and Bill Tidwell to discuss the vulnerabilities that allowed the compromise and possible controls to mitigate. There was good discussion about the significance of the vulnerabilities and possible controls or mitigating steps.

The intent of the meeting was to inform the IT Compliance Manager about system weaknesses that were recently uncovered and provide any information that would assist in the risk assessment for possible mitigation. Although Internal Audit does not intend to open an engagement in this area we would like to be made aware of any security enhancements that are implemented related to the issues noted.

Description of Compromise:

- While on the UNIX test system (testapp1.aci.corp.net) it was noted that someone was using a script calling SQLPLUS using a hard-coded database ID and password. This allowed me to see the database logon information for the Ellipse test/UAT system.

- I was able to use this ID and password to access the Ellipse test database and noted that the user account allowed access to significant tables, such as **dba_users**, which allowed me to find a listing of all accounts for that database and their encrypted Oracle password hashes.

- Using a popular cracking tool – orabf – I was able to find the actual passwords of many significant accounts (sys, system, quest, tripwire_monitor). It should be noted that the breaking of the Oracle passwords was done fully off-line on a home pc, meaning that a physical connection to the database is not necessary to break the passwords.

- Using some of these privileged accounts – tripwire_monitor, sys – I was able to gain access to all remaining Oracle databases; dev, test, production databases for Ellipse, RMS, Allegro, LMS, Oracle Apps, etc.

- Access noted above can occur with any pc that has a physical connection to the Company network. When plugging in the pc will receive an IP address from the DHCP server and can then make a connection to the various databases. Although a tnsnames.ora file is not needed the hostname and port is required (this information is readily obtained by performing a search on 'tnsnames' in \\stlsrv2\stlpub).

- The test Oracle Application Server (http://testapp1.aci.corp.net:7779/) was also compromised using the sys/system password found on the database. No serious attempt was made to compromise the production Application Server.

Issues Noted:

- Scripts have been written with hard coded IDs and Passwords. This is not just for database access but also for FTP, Telnet, etc.
    - This practice allows IDs and Passwords to be found by watching as scripts are run or by performing directory searches (recursive grep).
    - Because IDs and Passwords have been hard-coded it has hobbled the ability of the system administrators and DBAs to change passwords, as doing so requires a change to the passwords hard-coded in a number of scripts.

- The same ID and Password has been used across multiple database instances. This is a pragmatic approach but allows the compromise of an ID and Password on a low risk system (development) to jeopardize production databases.

- Privileged database accounts (those with DBA roles or 'select any table' privileges) have passwords that use few characters, making them easier to break with tools such as orabf.
    - Orabf first attempts to break a password by looking in a dictionary and if that does not work it begins a brute force attack. This brute force attack takes an amount of time that is exponential to the number and type of characters, making a 9 character password several factors harder to break than an 8 character password (where an 8 character password starting with the letter 'a' may take 30 minutes a 9 character password may take more than a month)

- Several individuals are given access to significant database tables. All developers and ID contractors (as well as a few other individuals) have been given access to the **dba_users** table in all environments (dev, test, and production). These users can then crack the system password as noted above.

- No logging of privileged account database access.

- Passwords for privileged accounts such as SYS are not changed frequently

- Passwords are not periodically reviewed for complexity or policy compliance

- Database access privileges are not currently reviewed for appropriateness or conformance with the principle of least privilege. Reviews have recently begun looking at whether access is for legitimate users, but no independent review to determine if access might be excessive.

Caveat:

Monday's discussion focused on the specific weakness that was recently uncovered by Internal Audit and was not intended to discuss all risks and vulnerabilities. There may be several other Oracle or UNIX risks which may need to be addressed by the security arm of the Information Services Department (such as the use of OPS$ accounts or database links).

Company Specific Considerations:
Internal Audit recognizes that the size of the Company's IS department requires certain pragmatic steps to be taken. As an example; developers need to address production problems and are therefore given read access to production data. However, when such access is considered steps should also be taken to mitigate any risk, such as restricting the ability to access or view the **dba_users.password** data, monitoring access to significant tables or the access of significant accounts, etc. It is also understood that Arch specific modifications are often required to be made to existing application, but these modifications should not introduce risk by the use of practices such as hard-coding passwords.

Although Arch 's environment requires certain divestitures from best-practice standard in order to operate efficiently and effectively, there were options discussed during the meeting that would help mitigate much of the risk noted without an excessive amount of resources (time or money). A procedure for eliminating the need to hard-code passwords into program code was discussed, which would go a long way to increasing the control environment. Additional discussions about increasing the complexity of significant database passwords and the frequency in which these are changed was also discussed which would reduce a significant amount of risk with limited expenditures. Another option to be reviewed is the security features of Oracle 10g to hide or further encrypt the password information that could be obtained by developers or contractors.

Let me know if you have any questions or need any detail on the specific steps taken to obtain access. Internal Audit would also appreciate updates on the status of mitigation steps taken to remediate.

--- Bret (x2928)