Database Health Check

Table of Contents

[1. WARNING 2](#_Toc476912218)

[2. Security 3](#_Toc476912219)

[3. General 6](#_Toc476912220)

[4. Database Design 7](#_Toc476912221)

# WARNING

**This document contains plain text passwords for various accounts and must be treated as strictly confidential.**

# Security

* All the supplied Oracle schemas have their default password still. However, all accounts in this state are currently "expired & locked" and cannot be logged into. This is acceptable, however, should the accounts be unlocked - which is not usually required - then the password must be changed.
* Passwords - appear mostly insecure, are hard coded into the application and are, therefore, unable to be changed without a deployment. Examples:
  + SVC\_AURA\_SERV/password1
  + UVSCHEDULER/abc123
  + ONLOAD/onload123
  + BBAPUSER/bb4pus3r
* Additionally, password1 is known to be rife throughout the database.
* Only accounts with APP\_USER or DBA profiles have any form of password complexity requirements. Other accounts can have any password they desire, however weak or simple. This does cover the majority of accounts however.
* The password verification function does perform certain validity checks, however, some of the checks that are performed will only ever be carried out under certain specific circumstances - when the user changes their own password and supplies the old password.
* Reference: <<http://qdosmsq.dunbar-it.co.uk/blog/2013/11/so-how-do-you-change-a-users-password/>>
* Profiles are set up for varying uses. None have password expiry enabled. All passwords are perpetual and can be reused an unlimited number of times.
* A number of profiles allows unlimited login attempts with an incorrect password. Brute force attacks are easy on users with these profiles.
  + Anyone logging onto the database server as "uadmin" has free access to the database *without* a password. The server's "uadmin" account connects to the database's "OPS$UADMIN" account and that user has:
  + CREATE SESSION (allows logon)
  + ALTER USER (can alter ANY database user)
  + SELECT on FCS.MUSER table.
* This user appears to be for *application user* maintenance. But can actually change *any* *database user*, including SYS, SYSTEM etc.
* The FCS account has DBA privileges granted. Anyone logging into this account has full access to the database and can carry out any task.
* The following "ANY" privileges are granted to the listed users. FCS obtains its list from the DBA role mentioned above. 'ANY' privileges have wide ranging effects, when misused. The first, GRANT ANY ROLE is surprising as it allows the AURA\_ADMIN user to potentially grant DBA to an otherwise unprivileged user, and revoke it afterwards.
  + AURA\_ADMIN GRANT ANY ROLE
  + AURA\_ADMIN SELECT ANY DICTIONARY
  + DATABASE\_READER\_UV EXECUTE ANY PROCEDURE
  + DATABASE\_READER\_UV SELECT ANY SEQUENCE
  + DATABASE\_READER\_UV SELECT ANY TABLE
  + FCS ALTER ANY INDEX
  + FCS ALTER ANY TABLE
  + FCS CREATE ANY INDEX
  + FCS CREATE ANY SYNONYM
  + FCS CREATE ANY TABLE
  + FCS DROP ANY SYNONYM
  + FCS DROP ANY TABLE
  + FCS GRANT ANY ROLE
  + FCS SELECT ANY DICTIONARY
  + SCHEDULER\_ADMIN CREATE ANY JOB
  + SCHEDULER\_ADMIN EXECUTE ANY CLASS
  + SCHEDULER\_ADMIN EXECUTE ANY PROGRAM
  + SYSMAN SELECT ANY DICTIONARY
  + UVSCHEDULER AUDIT ANY
  + WEBSERVICE\_USER SELECT ANY DICTIONARY
  + WEBSERVICE\_USER SELECT ANY SEQUENCE
  + WEBSERVICE\_USER SELECT ANY TABLE
* In the above, AURA\_ADMIN has alo received the ADMIN option on its grants. This allows AURA\_ADMIN to grant its received roles to other users.
* Roles and Privileges are a mess. An exercise in amalgamating *exactly* which roles and privileges are *required* is advised, with a view to reducing the amount of these which have been granted, are overlapping, etc.
* CONNECT and RESOURCE roles, granted in 9i, should never have been granted. These allow many more privileges than their name suggests, especially CONNECT. Under 11g, CONNECT has been reduced to a simple CREATE SESSION (allows logon), but RESOURCE is still over endowed.
* A number of profiles have been created but have not been used by any account:
  + BANKMANAGER\_PROFILE
  + CAPSIL\_STP\_USER
  + COMMS\_USER
  + FISHER
* A number of user accounts have been given EXECUTE access to certain packages, procedures etc. They have also been given the GRANT OPTION, which allows them to make similar grants to other users.
* There doesn't appear to be a process where by users leaving the company are notified to the IT Department so that all access, emails etc can be closed down and secured immediately. At present some accounts are checked and locked, but at least 30 days can pass before the account is totally shut down.
* Penetration Testing? - Has an exercise of this type ever been run against the application/database/servers?
* SQL Injection attacks - Has the application ever been tested for any of the various SQL Injection attacks that are currently known?

# General

* Are there any Production Support Documents? - None appear to be available.
* What monitoring is actually done? Are alerts raised? Are they visible? Are they acted upon? - OEM is apparently in use, but recent events have shown that metrics well outside the warning/critical settings - by many magnitudes, were classed as "ok". Example - Recent "logon storm" when a scheduler server (or email server?) went down. Critical was set to around 300 but the actual was into the 20,000 and also 30,0000s.
* Under normal circumstances, the following should be monitored at the absolute minimum:
  + Backups. Have they succeeded? Have any failed? When was the last successful backup?
  + Tablespace capacity - 80% full means they need looking at.
  + Failed login attempts - could indicate brute force attacks.
  + Failed scheduler jobs.
  + Accounts with passwords due to expire in the next 14 days, or sooner.
  + Are the standby databases running? Are they suffering from lags in transport or application of logs?
  + Are the OEM agents actually running and communication with OEM?
  + What errors are reported in the alert log?
* Have the backup tapes/files ever been used to restore a database to a different server? If not actually restoring, have they ever been used in a "restore validate" to at least prove that they can be read? Has the evidence been documented? Are these tests carried out regularly?
* During the migration to 11g, three accounts had their default tablespace set to SYSTEM. This is unacceptable in any Oracle Database, and was rectified as part of the migration itself.
* The database contains numerous "TEMP" named tables. If these are temporary, they should have been cleaned out after use/warranty period.
* Some private and/or public synonyms do not point at any existing objects.
* A number of database constraints exist on application tables, but are disabled. Mainly CHECK constraints, but there are also a couple of Foreign Key constraints.

# Database Design

* A number of tables appear to be over indexed. FCS.INVESTOR and FCS.ORDTRAN specifically, have 46 different indexes each. Some indexes have numerous columns - many over 10 columns wide.
* A number of tables have overlapping indexes. These are indexes with the same leading columns.
* A number of child tables have FK columns which differ in data type from the parent table's referenced columns. This causes implicit data conversion to take place on joins, and will disable any indexes present on the columns, parent or child.
* Some child tables have no index on the FK columns. While this is not necessarily a problem, it will be a major one if any of the following are permitted:
  + The parent table's referenced columns are allowed to be updated;
  + The parent table's rows can be deleted;
  + The parent and child tables are joined in a query, using the FK and referenced columns.
* In any one or more of these three cases, a missing index will affect performance as exclusive table locks are acquired, full scans carried out and the locks released.
* There are a number of tables, in the FCS schema, which have mixed case names. This usually indicates a code generator or poorly set up database design tools. The tables must always be accessed using double quotes, and the exact letter case that have been stored with. While not a database problem as such, it does make life difficult when exporting by table name, and for developers needing to access the tables.
* 21 check constraints in the FCS schema *do not work*. For example:

... CHECK(Mortality in ('Y','N',NULL)) ...

These constraints are totally disabled by the presence of NULL, and any value is allowed in the column, not just the desired Ys and Ns. A more correct specification would be:

... CHECK(NVL(Mortality, 'N') in ('Y','N')) ...

or:

... CHECK(Mortality in ('Y','N') or Mortality IS NULL) ...

A number of the various TAKEON\_XXXXX schemas are also afflicted with this problem.

Reference: <<http://qdosmsq.dunbar-it.co.uk/blog/2016/08/dropping-temporary-tables-with-bonus-broken-check-constraints/>>

* A number of tables have record lengths that are bigger than the database block size. The design should have considered this, and catered for it with either a bigger block size, or, special tablespaces with larger block sizes to facilitate these tables. (This may not have been possible in 9i, but I think it was.)