Audit : que faire ?

----- mail de cgz -----

------trouvé ailleurs -------

Partie1: Les paramètres

Partie 2: les fichiers de données

Partie3: Les redo

Partie 4: La sécurité

etc.

--

Rarement d'audit général, c'est trop vague. Le client veut axer sur un domaine : sécurité, performance, capacité, architecture..

En ce qui concerne la sécurité :

paramétrage : utl\_file\_dir

directories

comptes externes

comptes avec droits sysdba/sysoper

multiplexage des redo+controlfile

+ ce qui est indiqué dans le post de pifor.

Pour les perfs (liste non-exhaustive), en plus de awr/ash/statspack :

fréquence des switchs de log

opérations longues

sélectivité des index

index redondants (FK à ne pas prendre en compte)

nombre de tables ayant plus de 10 index

débordement de workarea (en "live") => écriture des tris dans le temp

attentes : types et fichiers

statistiques du parallélisme

tables accédées en FTS

comparaison buffer keep déclaré et taille des segments prévus pour ce pool

idem avec le multiblock size

éléments audités

...

Exemple de checklist dans le Security Guide 10.2: http://download.oracle.com/docs/cd/B...s.htm#i1010421

ici : http://docs.oracle.com/cd/B19306\_01/network.102/b14266/checklis.htm#i1010098

Extraits ci-dessous :

--------------------------

The following security checklist includes guidelines that help secure your database:

1. **Install only** **what is required.**

**Options and Products**

The Oracle Database CD pack contains a host of options and products in addition to the database server. Install additional products and options only as necessary. Use Custom Installation to avoid installing unnecessary products or, perform a typical installation, and then deinstall unrequired options and products. There is no need to maintain additional products and options if they are not being used. They can always be properly and easily reinstalled as required.

**Sample Schemas**

Oracle Corporation provides Sample Schemas to provide a common platform for examples. If your database will be used in a production environment, then do not install the Sample Schema. If you have installed the Sample Schema on a test database, then before going production, remove or relock the Sample Schema accounts.

1. **Lock and expire default user accounts.**

Oracle Database installs with a number of default (preset) database server user accounts. Upon successful installation of the database server, the Database Configuration Assistant automatically locks and expires most default database user accounts.

If a manual (without using Database Configuration Assistant) installation of Oracle Database is performed, then no default database users are locked upon successful installation of the database server. Left open in their default states, these user accounts can be exploited to gain unauthorized access to data or disrupt database operations.

Therefore, after performing any kind of initial installation that does not use the Database Configuration Assistant, you should *lock* and *expire*all default database user accounts. Oracle Database provides SQL statements to perform such operations.

Installing additional products and components later also results in creating more default database server accounts. Database Configuration Assistant automatically locks and expires all additionally created database server user accounts. Unlock only those accounts that need to be accessed on a regular basis and assign a strong, meaningful password to each of these unlocked accounts. Oracle provides SQL and password management to perform such operations.

[Table 7-2](http://docs.oracle.com/cd/B19306_01/network.102/b14266/policies.htm#g1013758) shows the database users after a typical Oracle Database installation using Database Configuration Assistant.

***Table 7-2 Default Accounts and Their Status (Standard Installation)***

| **Username** | **Account Status** |
| --- | --- |
| ANONYMOUS | EXPIRED & LOCKED |
| CTXSYS | EXPIRED & LOCKED |
| DBSNMP | EXPIRED & LOCKED |
| DIP | EXPIRED & LOCKED |
| DMSYS | EXPIRED & LOCKED |
| EXFSYS | EXPIRED & LOCKED |
| HR | EXPIRED & LOCKED |
| MDDATA | EXPIRED & LOCKED |
| MDSYS | EXPIRED & LOCKED |
| MGMT\_VIEW | EXPIRED & LOCKED |
| ODM | EXPIRED & LOCKED |
| ODM\_MTR | EXPIRED & LOCKED |
| OE | EXPIRED & LOCKED |
| OLAPSYS | EXPIRED & LOCKED |
| ORDPLUGINS | EXPIRED & LOCKED |
| ORDSYS | EXPIRED & LOCKED |
| OUTLN | EXPIRED & LOCKED |
| PM | EXPIRED & LOCKED |
| QS | EXPIRED & LOCKED |
| QS\_ADM | EXPIRED & LOCKED |
| QS\_CB | EXPIRED & LOCKED |
| QS\_CBADM | EXPIRED & LOCKED |
| QS\_CS | EXPIRED & LOCKED |
| QS\_ES | EXPIRED & LOCKED |
| QS\_OS | EXPIRED & LOCKED |
| QS\_WS | EXPIRED & LOCKED |
| RMAN | EXPIRED & LOCKED |
| SCOTT | EXPIRED & LOCKED |
| SH | EXPIRED & LOCKED |
| SI\_INFORMTN\_SCHEMA | EXPIRED & LOCKED |
| SYS | OPEN |
| SYSMAN | EXPIRED & LOCKED |
| SYSTEM | OPEN |
| TSMSYS | EXPIRED & LOCKED |
| WK\_TEST | EXPIRED & LOCKED |
| WKPROXY | EXPIRED & LOCKED |
| WKSYS | EXPIRED & LOCKED |
| WMSYS | EXPIRED & LOCKED |
| XDB | EXPIRED & LOCKED |

If any default database server user account other than the ones left open is required for any reason, then a database administrator (DBA) need simply unlock and activate that account with a new, secure password.

**Enterprise Manager Accounts**

The preceding list of accounts depends on whether you choose to install Enterprise Manager. If so, SYSMAN and DBSNMP are open as well, unless you configure Enterprise Manager for Central Administration. In this case, the SYSMAN account (if present) will be locked as well.

If you do not install Enterprise Manager, then only SYS and SYSTEM are open. Database Configuration Assistant locks and expires all other accounts (including SYSMAN and DBSNMP).

1. **Change default user passwords.**

The most trivial method by which Oracle Database can be compromised is a default database server user account which still has a default password associated with it *even after installation*. The following guidelines are recommended:

* 1. **Change default passwords of administrative users.**

Oracle Database 10g enables you to use the same or different passwords for the SYS, SYSTEM, SYSMAN and DBSNMP administrative accounts. Use different passwords for each: in any Oracle environment (production or test), assign strong, secure, and distinct passwords to these administrative accounts. If Database Configuration Assistant is used, then it requires you to enter passwords for the SYS and SYSTEM accounts, disallowing the use of the defaults CHANGE\_ON\_INSTALL and MANAGER.

Similarly, for production environments, do not use default passwords for any administrative accounts, including SYSMAN and DBSNMP.

At the end of database creation, Database Configuration Assistant displays a page requiring you to enter and confirm new passwords for the SYS and SYSTEM user accounts.

* 1. **Change default passwords of all users.**

In Oracle Database, SCOTT no longer installs with default password TIGER, but instead is locked and expired, as is DBSNMP. Each of the other accounts install with a default password that is exactly the same as that user account (For example, user MDSYSinstalls with the password MDSYS).

If any of the default user accounts that were locked and expired upon installation need to be activated, then assign a new secure password to each such user account.

Even though Oracle does not explicitly mandate changing the default password for the user SCOTT, Oracle recommends that this user account also be locked in a production environment.

* 1. **Enforce password management.**

Oracle recommends that basic password management rules (such as password length, history, complexity, and so forth) as provided by the database be applied to all user passwords and that all users be required to change their passwords periodically.

Oracle also recommends, if possible, using Oracle Advanced Security (an option to the Enterprise Edition of Oracle Database) with network authentication services (such as Kerberos), token cards, smart cards or X.509 certificates. These services enable strong authentication of users to provide better protection against unauthorized access to Oracle Database.

1. **Enable data dictionary protection.**

Oracle recommends that customers implement data dictionary protection to prevent users having the ANY system privileges from using such privileges on the data dictionary.

To enable dictionary protection, set the following configuration parameter to FALSE, in the init<sid>.ora control file:

O7\_DICTIONARY\_ACCESSIBILITY = FALSE

By doing so, only those authorized users making DBA-privileged (for example CONNECT / AS SYSDBA) connections can use the ANY system privilege on the data dictionary. If O7\_DICTIONARY\_ACCESSIBILITY is not set to FALSE, then any user with a DROP ANY TABLE (for example) system privilege will be able to maliciously drop parts of the data dictionary.

However, if a user *needs* view access to the data dictionary, then it is permissible to grant that user the SELECT ANY DICTIONARY system privilege.

**Notes:**

* 1. Regarding O7\_DICTIONARY\_ACCESSIBILITY, note that in Oracle Database, the default is FALSE. However, in Oracle8i, this parameter is set to TRUE by default and must specifically be changed to FALSE to enable this security feature.
  2. Regarding the SELECT ANY DICTIONARY privilege: this privilege is not included in the GRANT ALL PRIVILEGES statement, but it can be granted through a role.

1. **Practice the principle of least privilege.**

The following guidelines are recommended:

* 1. **Grant necessary privileges only.**

Do not provide database users more privileges than are necessary. In other words, the *principle of least privilege* is that users be given only those privileges that are actually required to efficiently perform their jobs.

To implement this principle, restrict the following as much as possible:

1) The number of SYSTEM and OBJECT privileges granted to database users, and

2) The number of people who are allowed to make SYS-privileged connections to the database.

For example, there is generally no need to grant CREATE ANY TABLE to any non-DBA-privileged user.

* 1. **Revoke unnecessary privileges from the** **PUBLIC** **role.**

Revoke all unnecessary privileges and roles from the role PUBLIC. PUBLIC acts as a default role granted to every user in an Oracle database. Any database user can exercise privileges that are granted to PUBLIC. Such privileges include EXECUTE on various PL/SQL packages, potentially enabling a minimally-privileged user to access and execute functions that he would not otherwise be permitted to access directly. The more powerful packages that may potentially be misused are listed in the following table:

| **Package or Subtype** | **Description** |
| --- | --- |
| DBMS\_RANDOM | This package can be used to encrypt stored data. Generally, most users should not have the privilege to encrypt data since encrypted data may be non-recoverable if the keys are not securely generated, stored, and managed. |
| HTTPURITYPE1 | This subprogram is a subtype of the UriType that provides support for the HTTP protocol. It uses the UTL\_HTTP package underneath to access the HTTP URLs. Proxy and secure wallets are not supported in this release. Downgrade grants on HTTPURITYPE to the minimum needed in your environment, or revoke all grants if none of your applications need it. |
| UTL\_HTTP1 | This package allows the database server to request andretrieve data using HTTP. |
| UTL\_INADDR1 | This package allows arbitrary domain name resolution to be performed from the database server. Granting this package to the PUBLIC role may permit unauthorized domain name resolution. |
| UTL\_SMTP[Foot 1](http://docs.oracle.com/cd/B19306_01/network.102/b14266/policies.htm#sthref1027) | This package permits arbitrary mail messages to be sent from one arbitrary user to another arbitrary user. Granting this package to the PUBLIC role may permit unauthorized exchange of mail messages. |
| UTL\_TCP1 | This package permits outgoing network connections to be established by the database server to any receiving (or waiting) network service. Granting this package to PUBLICmay permit arbitrary data may to be sent between the database server and any waiting network service. |

Footnote 1These packages and subtype should be revoked from PUBLIC and made executable for an application only when absolutely necessary.

These packages are extremely useful to the applications that need them. They require proper configuration and usage for safe and secure operation, and may not be suitable for most applications.

For applications that need these packages, create roles with EXECUTE privilege on the particular packages needed and assign those roles only to applications that specifically need to use them. Oracle intends to revoke such privileges from PUBLIC in subsequent releases.

* 1. **Grant a role to users only if they need all privileges of the role.**

Roles (groups of privileges) are useful for quickly and easily granting permissions to users. Although you can use Oracle-defined roles, you have more control and continuity if you create your own roles containing only the privileges pertaining to your requirements. Oracle may change or remove the privileges in an Oracle-defined role, as it has with CONNECT, which now has only the CREATE SESSION privilege. Formerly this role had eight other privileges. Both CONNECT and RESOURCE roles will be deprecated in future Oracle versions.

Ensure that the roles you define contain only the privileges that reflect job responsibility. If your application users do not need all the privileges encompassed by an existing role, then apply a different set of roles that supply just the right privileges. Alternatively, create and assign a more restricted role.

For example, it is imperative to strictly limit the privileges of SCOTT. Drop the CREATE DBLINK privilege for SCOTT. Then drop the entire role for the user, because privileges acquired by means of a role cannot be dropped individually. Recreate your own role with only the privileges needed, and grant that new role to that user. Similarly, for even better security, drop the CREATE DBLINKprivilege from all users who do not require it.

* 1. **Restrict permissions on run-time** **facilities.**

Do not assign all permissions to any database server run-time facility such as the Oracle Java Virtual Machine (OJVM). Grant specific permissions to the explicit document root file paths for such facilities that may execute files and packages outside the database server.

Here is an example of a vulnerable run-time call:

call dbms\_java.grant\_permission('SCOTT', 'SYS:java.io.FilePermission','<<ALL FILES>>','read');

Here is an example of a better (more secure) run-time call:

call dbms\_java.grant\_permission('SCOTT', 'SYS:java.io.FilePermission','<<actual directory path>>','read');

1. **Enforce access controls effectively and authenticate clients stringently.**

**Authenticate clients properly.**

By default, Oracle allows operating-system-authenticated logins only over secure connections, which precludes using Oracle Net and a shared server configuration. This default restriction prevents a remote user from impersonating another operating system user over a network connection.

Setting the initialization parameter REMOTE\_OS\_AUTHENT to TRUE forces the RDBMS to accept the client operating system user name received over a nonsecure connection and use it for account access. Since clients, such as PCs, are not trusted to perform operating system authentication properly, it is very poor security practice to turn on this feature.

The default setting, REMOTE\_OS\_AUTHENT = FALSE, creates a more secure configuration that enforces proper, server-based authentication of clients connecting to an Oracle database.

You should not alter the default setting of the REMOTE\_OS\_AUTHENT initialization parameter, which is FALSE.

Setting this parameter to FALSE does not mean that users cannot connect remotely. It simply means that the database will not trust that the client has already authenticated, and will therefore apply its standard authentication processes.

1. **Restrict operating system access.**

Limit the number of operating system users. Limit the privileges of the operating system accounts (administrative, root-privileged or DBA) on the Oracle Database host (physical machine) to the least privileges needed for the user's tasks.

Oracle also recommends:

* 1. Restricting the ability to modify the default file and directory permissions for the Oracle Database home (installation) directory or its contents. Even privileged operating system users and the Oracle owner should not modify these permissions, unless instructed otherwise by Oracle.
  2. Restricting symbolic links. Ensure that when providing a path or file to the database, neither the file nor any part of the path is modifiable by an untrusted user. The file and all components of the path should be owned by the DBA or some trusted account, such as *root*.

This recommendation applies to all types of files: data files, log files, trace files, external tables, bfiles, and so on.

1. **Restrict** **network access.**

The following guidelines are recommended:

* 1. **Use a firewall.**

Keep the database server behind a firewall. Oracle Database network infrastructure, Oracle Net (formerly known as Net8 and SQL\*Net), offers support for a variety of firewalls from various vendors. Supported proxy-enabled firewalls include' Gauntlet from Network Associates and Raptor from Axent . Supported packet-filtering firewalls include PIX Firewall from Cisco, and supported stateful inspection firewalls (more sophisticated packet-filtered firewalls) include Firewall-1 from CheckPoint .

* 1. **Never poke a hole through a firewall.**

If Oracle Database is behind a firewall, then do not, under any circumstances, poke a hole through the firewall. For example, do not leave open port 1521 for Oracle Listener to make a connection to the Internet or vice versa.

Doing this will introduce a number of significant security vulnerabilities including more port openings through the firewall, multi-threaded operating system server issues, and revelation of crucial information on databases behind the firewall. Furthermore, an Oracle Listener running without an established password may be probed for critical details about the databases on which it is listening such as trace and logging information, banner information and database descriptors and service names.

All this information and the availability of an ill-configured firewall will provide an attacker ample opportunity to launch malicious attacks on the target databases.

* 1. **Protect the Oracle listener.**

Because the listener acts as the database gateway to the network, it is important to limit the consequences of malicious interference:

* + - Restrict the privileges of the listener, so that it cannot read or write files in the database or the Oracle server address space.

This restriction prevents external procedure agents spawned by the listener (or procedures executed by such an agent) from inheriting the ability to do such reads or writes. The owner of this separate listener process should not be the owner that installed Oracle or executes the Oracle instance (such as ORACLE, the default owner).

Sample configuration:

EXTPROC\_LISTENER=

(DESCRIPTION=

(ADDRESS=

(PROTOCOL=ipc)(KEY=extproc)))

SID\_LIST\_EXTPROC\_LISTENER=

(SID\_LIST=

(SID\_DESC=

(SID\_NAME=plsextproc)

(ORACLE\_HOME=/u1/app/oracle/9.0)

(PROGRAM=extproc)))

* + - Secure administration of the database by doing the following:

i. Prevent online administration by requiring the administrator to have write privileges on the LISTENER.ORA file and the listener password:

Add or alter this line in the LISTENER.ORA file

ADMIN\_RESTRICTIONS\_LISTENER=ON

Then RELOAD the configuration.

ii. Use SSL when administering the listener, by making the TCPS protocol the first entry in the address list as follows:

LISTENER=

(DESCRIPTION=

(ADDRESS\_LIST=

(ADDRESS=

(PROTOCOL=tcps)

(HOST = ed-pdsun1.us.oracle.com)

(PORT = 8281)))

To administer the listener remotely, you need to define the listener in the listener.ora file on the client computer. For example, to access listener USER281 remotely, use the following configuration:

user281 =

(DESCRIPTION =

(ADDRESS =

(PROTOCOL = tcps)

(HOST = ed-pdsun1.us.oracle.com)

(PORT = 8281))

)

)

iii. Always establish a secure, well-formed password for the Oracle listener to prevent remote configuration of the Oracle listener. Password protect the listener as follows:

LSNRCTL> CHANGE\_PASSWORD

Old password: lsnrc80

New password: lsnrc90

Reenter new password: lsnrc90

LSNRCTL> SET PASSWORD

Password:

The command completed successfully

LSNRCTL> SAVE\_CONFIG

The command completed successfully

* + - Remove the external procedure configuration from the listener.ora file if you do not intend to use such procedures.
    - Monitor listener activity.
  1. **Monitor who accesses your systems.**

Authenticating client computers over the Internet is problematic. Do user authentication instead, which avoids client system issues that include falsified IP addresses, hacked operating systems or applications, and falsified or stolen client system identities. The following steps improve client computer security:

* + - Configure the connection to use SSL. Using SSL (Secure Sockets Layer) communication makes eavesdropping unfruitful and enables the use of certificates for user and server authentication.
    - Set up certificate authentication for clients and servers such that:

i. The organization is identified by unit and certificate issuer and the user is identified by distinguished name and certificate issuer.

ii. Applications test for expired certificates.

iii. Certificate revocation lists are audited.

* 1. **Check network IP addresses.**

Use the Oracle Net *valid node checking* security feature to allow or deny access to Oracle server processes from network clients with specified IP addresses. To use this feature, set the following protocol.ora (Oracle Net configuration file) parameters:

tcp.validnode\_checking = YES

tcp.excluded\_nodes = {list of IP addresses}

tcp.invited\_nodes = {list of IP addresses}

The first parameter turns on the feature whereas the latter parameters respectively deny and allow specific client IP addresses from making connections to the Oracle listener (This helps in preventing potential Denial of Service attacks).

* 1. **Encrypt network traffic.**

If possible, use Oracle Advanced Security to encrypt network traffic between clients, databases, and application servers.

**Note:**

Oracle Advanced Security is available only with the Enterprise Edition of the Oracle database. It installs in Typical Installation mode and can be configured, after licensing, with the Oracle Net Manager tool or by manually setting six sqlnet.ora parameters to enable network encryption.

* 1. **Harden the operating system.**

Harden the host operating system by disabling all unnecessary operating system services. Both UNIX and Windows platforms provide a variety of operating system services, most of which are not necessary for most deployments. Such services include FTP, TFTP, TELNET, and so forth. Be sure to close both the UDP and TCP ports for each service that is being disabled. Disabling one type of port and not the other does not make the operating system more secure.

1. **Apply all security patches and workarounds.**

Always apply all relevant and current security patches for both the operating system on which Oracle Database resides and Oracle Database itself, and for all installed Oracle Database options and components.

Periodically check the security site on Oracle Technology Network for details on security alerts released by Oracle Corporation at

<http://www.oracle.com/technetwork/topics/security/alerts-086861.html>

Also check Oracle Worldwide Support Service site, My Oracle Support, for details on available and upcoming security-related patches at

[https://support.oracle.com](https://support.oracle.com/)

1. **Contact Oracle Security Products if you come across a vulnerability in Oracle Database.**

If you believe that you have found a security vulnerability in Oracle Database, then submit an iTAR to Oracle Worldwide Support Services using My Oracle Support, or e-mail a complete description of the problem, including product version and platform, together with any exploit scripts and examples to the following address:

secalert\_us@oracle.com