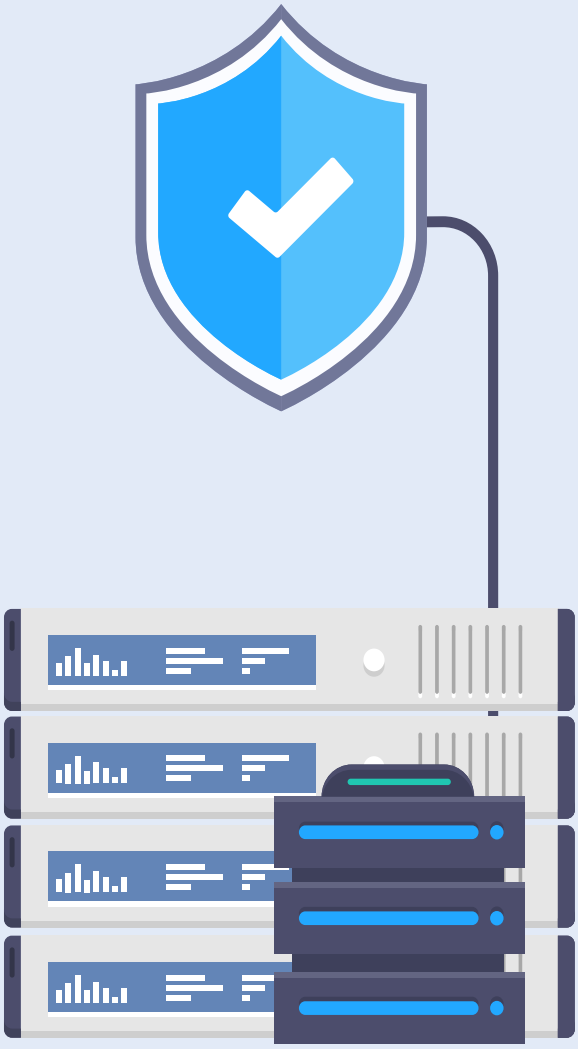
CCN-CERT

BP/22



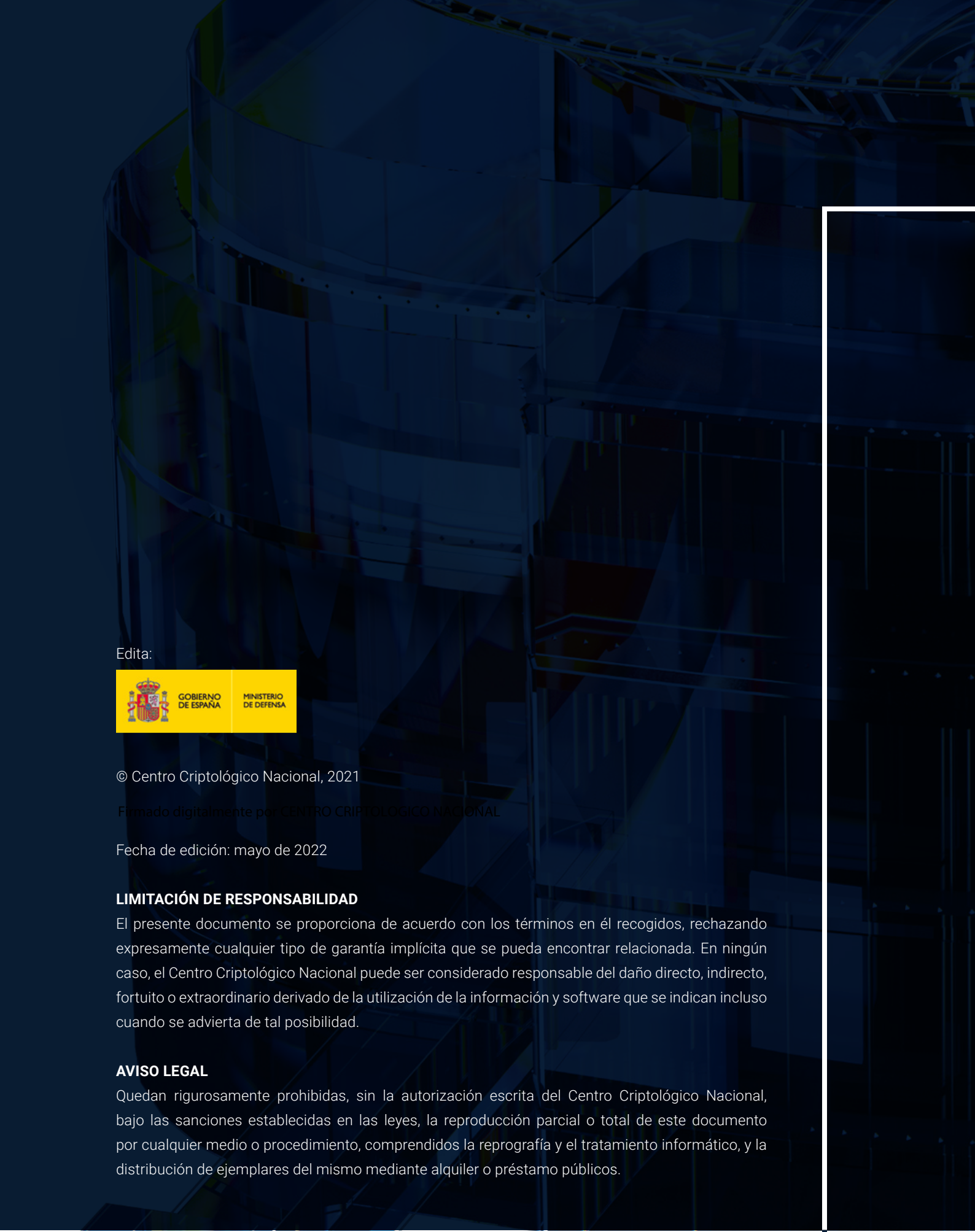
Recomendaciones de seguridad para Oracle Database 19C

INFORME DE BUENAS PRÁCTICAS

MAYO 2022

ccn-cert centro criptológico nacional

CCN centro criptológico nacional



Edita:

GOBIERNO DE ESPAÑA MINISTERIO DE DEFENSA

Firmado digitalmente por CENTRO CRIPTOLOGICO NACIONAL

Fecha de edición: mayo de 2022

Índice

1. Sobre CCN-CERT, CERT Gubernamental Nacional

El CCN-CERT es la Capacidad de Respuesta a incidentes de Seguridad de la Información del Centro Criptológico Nacional, CCN.

El **CCN-CERT** es la Capacidad de Respuesta a incidentes de Seguridad de la Información del Centro Criptológico Nacional, CCN, adscrito al Centro Nacional de Inteligencia, CNI. Este servicio se creó en el año 2006 como **CERT Gubernamental Nacional español** y sus funciones quedan recogidas en la Ley 11/2002 reguladora del CNI, el RD 421/2004 de regulación del CCN y en el RD 3/2010, de 8 de enero, regulador del Esquema Nacional de Seguridad (ENS), modificado por el RD 951/2015 de 23 de octubre.

Su misión, por tanto, es **contribuir a la mejora de la ciberseguridad española**, siendo el centro de alerta y respuesta nacional que coopere y ayude a responder de **forma rápida** y eficiente a los ciberataques y a afrontar de forma activa las ciberamenazas, incluyendo la coordinación a nivel público estatal de las distintas Capacidades de Respuesta a Incidentes o Centros de Operaciones de Ciberseguridad existentes.

Todo ello, con el fin último de **conseguir un ciberespacio más seguro y confiable**, preservando la información clasificada (tal y como recoge el art. 4. F de la Ley 11/2002) y la información sensible, defendiendo el Patrimonio Tecnológico español, formando al personal experto, aplicando políticas y procedimientos de seguridad y empleando y desarrollando las tecnologías más adecuadas a este fin.

De acuerdo a esta normativa y la Ley 40/2015 de Régimen Jurídico del Sector Público es competencia del CCN-CERT la gestión de ciberincidentes que afecten a cualquier organismo o empresa pública. En el caso de operadores críticos del sector público la gestión de ciberincidentes se realizará por el CCN-CERT en coordinación con el CNPIC.

2. Fundamentos de la seguridad de las bases de datos

Los sistemas gestores de bases de datos se ejecutan sobre plataformas específicas y sistemas operativos que les proporcionan los elementos fundamentales de comunicación y de acceso.

El modelo de seguridad de un sistema gestor de bases de datos, por lo tanto, desde un punto de vista simplificado, se puede indicar que estará dividido en estos dos **ámbitos de actuación**:

El ámbito de la plataforma donde se ejecuta el servicio

El entorno y capacidades que proporciona el propio gestor de bases de datos

El producto Oracle 19c es un gestor de bases de datos relacionales de tipo generalista, lo cual quiere decir que puede ser utilizado en múltiples entornos y aplicaciones, que se puede implementar tanto en sistemas Unix, Linux como servidores Microsoft Windows.

En todos los casos, será importante no perder de vista los aspectos de seguridad que se configuran en el ámbito del sistema operativo, como usuarios, servicios, comunicaciones y protocolos, así como los que se configuran en el entorno de Oracle 19c, como los procesos de autorización y control de acceso a los datos que residen en las distintas bases de datos.

La autenticación es el proceso por el cual **un sistema verifica la identidad de un usuario**. En Oracle 19c, este proceso se realiza fuera del entorno de la aplicación, a través de un módulo de autenticación. Mediante distintos módulos que incorpora Oracle 19c, se puede hacer uso de protocolos de autenticación como LDAP, OS, TNS, Kerberos, por SID o nombre de servicio.

La autentificación es el proceso por el cual un sistema verifica la identidad de un usuario.

Para activar y configurar los **distintos métodos de autentificación** puede consultarse la guía:

Enlace: https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/configuring-authentication.html

Puede implementarse **Kerberos** según se indica a través del siguiente enlace:

Enlace: https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/configuring-kerberos-authentication.html

La aplicación de seguridad sobre el motor de base de datos Oracle 19c forma parte una serie de tareas que se deben realizar de manera continua. Oracle es un motor de base de datos muy conocido por lo que sus cuentas de usuario, puertos, rutas de archivos y configuraciones por defecto en una instalación pueden originar una grave amenaza para la seguridad de una organización.

Esta nueva versión de Oracle ofrece una serie de mejoras y características referentes a la indexación automática mediante algoritmos de machine learning y añade un punto de mejora sobre la estabilidad mediante el redireccionamiento DML de Active Data Guard en cuanto a copias de seguridad.

Active Data Guard es un modelo de arquitectura de alta disponibilidad que permite una arquitectura de alta disponibilidad como modelo preventivo de pérdida de datos en modo síncrono o asíncrono.

Las capacidades de Oracle (Active) Data Guard en Oracle Database 19c mejoran aún más su objetivo estratégico de prevenir la pérdida de datos, proporcionar alta disponibilidad, eliminar riesgos y aumentar el retorno de la inversión al habilitar sistemas activos de recuperación ante desastres altamente funcionales que son fáciles de implementar y administrar. Lo logra proporcionando la infraestructura de software de administración, monitoreo y automatización. para crear y mantener una o más bases de datos en espera sincronizadas que protejan los datos de Oracle de fallas, corrupción de datos, errores humanos y desastres.

Active Data Guard utiliza la simplicidad de la replicación física, con su integración con Oracle se proporciona un aislamiento único entre las bases de datos primarias y en espera para ofrecer el mayor nivel de protección contra la pérdida de datos. Active Data Guard admite tanto síncrono (cero garantizado pérdida de datos) y asincrónica (pérdida de datos casi nula).

Active Data Guard utiliza la simplicidad de la replicación física, con su integración con Oracle se proporciona un aislamiento único entre las bases de datos primarias.

Mantener una alta disponibilidad para la misión. aplicaciones críticas, los administradores de bases de datos pueden elegir la conmutación por error manual o automática a la espera de que, el sistema principal no esté disponible por cualquier motivo. Active Data Guard es una opción con licencia para Oracle Database Enterprise Edition. Todas las capacidades que se denominan explícitamente "Active Data Guard" requieren una licencia de Active Data Guard. Todas las capacidades a las que se hace referencia explícitamente como "Data Guard" son incluido con Oracle Enterprise Edition, no se requiere licencia de opción. Active Data Guard es un superconjunto de Data Guard hereda todas las capacidades de Data Guard.

Una de las grandes ventajas de Active Data Guard 19c es la mejor capacidad para realizar lecturas intensivas sin conexión aplicaciones al modo de espera. Ahora también es posible emitir DML ocasional contra el modo de espera base de datos, por lo que ahora es una base de datos de informes completamente funcional. Esto aprovecha el retorno de la inversión ya que la base de datos primaria se utiliza de forma óptima y los recursos del sistema de recuperación ante desastres se utilizan de forma óptima.

Se puede obtener más información sobre Oracle Active Data Guard 19c desde el siguiente enlace:

Enlace: https://www.oracle.com/technetwork/database/availability/dg-adg-technical-overview-wp-5347548.pdf

En cuanto a la trazabilidad y el crecimiento repentino que puedan tener cada una de las tablas de Oracle en función de las necesidades de cada organización, se han implementado tablas con particiones híbridas permitiendo la gestión de una tabla entre particiones dentro de la base de datos y también fuera de la base de datos, en el caso externo con acceso de lectura.

Pude complementarse la información a través del siguiente enlace:

Enlace: https://oracle-base.com/articles/19c/hybrid-partitioned-tables-19c

2. Fundamentos de la seguridad en las bases de datos

La autorización es el proceso de determinar si un usuario autenticado, dispone de acceso a la información y permisos que está solicitando. Este proceso se realiza íntegramente dentro de Oracle 19c, consultando los permisos asociados a una identidad concreta. En este sentido, existen distintos tipos de permisos que pueden ser otorgados.

- **Permisos primarios:** Aquellos que se otorgan directamente al identificador de autorización.
- **Permisos secundarios:** Aquellos que se otorgan a grupos y roles de los cuales es miembro un identificador de autorización.
- **Permisos públicos:** Aquellos que se otorgan a la entidad PUBLIC.
- **Permisos basados en contexto:** Aquellos que se otorgan a un rol de contexto de confianza.

Estos permisos se pueden otorgar a los usuarios en varios niveles o categorías:

- **Autorización a nivel de sistema:** Son las autoridades que realizan labores de administración. Hay varios usuarios con roles diferenciados.
- El usuario SYS es administrador de sistema. Su contraseña debe ser cambiada sobre la predeterminada del fabricante. No conviene crear objetos dentro de su esquema.
- El usuario SYSTEM a cargo del control del sistema, éste posee el rol DBA y también debe cambiar su contraseña por defecto. En su esquema se pueden crear tablas y vistas de administración.
- Los usuarios SYSBACKUP, SYSDG, SYSKM, y SYSRAC se crean automáticamente en la instalación para facilitar las labores de administración.
- El usuario SYSBACKUP facilita las operaciones de copia de seguridad y recuperación de Oracle Recovery Manager (RMAN) ya sea desde RMAN o SQL * Plus.
- El usuario SYSDG facilita las operaciones de Data Guard. El usuario puede realizar operaciones con Data Guard Broker o con la interfaz de línea de comandos DGMGRL.

- El usuario SYSKM facilita las operaciones del almacén de claves de cifrado transparente de datos.
- El usuario SYSRAC facilita las operaciones de Oracle Real Application Clusters (Oracle RAC) al conectarse a la base de datos a través del agente Clusterware contra las utilidades de Oracle RAC como SRVCTL.
- El privilegio administrativo de SYSRAC no se puede otorgar a los usuarios de la base de datos y no se admite en un archivo de contraseña. El privilegio administrativo de SYSRAC solo lo usa el agente de Oracle de Oracle Clusterware para conectarse a la base de datos mediante la autenticación del sistema operativo.

Los usuarios a los que se les ha otorgado el privilegio del sistema CREATE USER pueden crear cuentas de usuario, incluidas las cuentas de usuario que se utilizarán como usuarios proxy. Debido a que el privilegio del sistema CREATE USER es un privilegio poderoso, un administrador de base de datos o un administrador de seguridad suele ser el único usuario que tiene este privilegio del sistema. Si se desea crear usuarios que tengan el privilegio de crear usuarios, se puede incluir la cláusula WITH ADMIN OPTION en la declaración GRANT.

- **Autorización a nivel de base de datos:** Oracle 19c posee 79 roles predefinidos durante la instalación. Puede consultarse la definición de éstos en:

  Enlace: https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/configuring-privilege-and-role-authorization.html

  Independientemente de éstos, se pueden generar otros roles, con la sentencia create role a los que posteriormente se le pueden asignar permisos específicos. Por ello, una auditoría de permisos en la base de datos debe ser dinámica y no restringirse únicamente a los roles y permisos generados en una instalación. El fabricante provee al menos 30 vistas de administración de roles para facilitar estas tareas. Las autoridades con permisos sobre grant and revoke privilege, pueden asignar y revocar los permisos a los usuarios y roles.

- **Autorización a nivel de objeto:** La autorización a nivel de objeto implica la verificación de privilegios cuando se realiza una operación concreta sobre un objeto específico.

Autorización basada en contenido: Una forma de autorizar el acceso basado en contenido son las vistas. Las vistas permiten controlar qué columnas o filas de una tabla pueden ser leídas por usuarios específicos. Por otro lado, el fabricante Oracle a través Oracle Label Security, permite que el control de acceso llegue a filas específicas (etiquetadas) de una base de datos. Con Oracle Label Security implementado, los usuarios con diferentes niveles de privilegios tienen automáticamente (o están excluidos) el derecho a ver o modificar filas de datos etiquetadas. La Guía del administrador de Oracle Label Security describe cómo utilizar Oracle Label Security para proteger datos confidenciales. Explica los conceptos básicos detrás de la seguridad basada en etiquetas y proporciona ejemplos para mostrar cómo se usa.

Otro componente importante a la hora de definir la seguridad de un gestor de bases de datos es el cifrado, tanto de los datos en tránsito como de los datos en reposo. Oracle 19c ofrece diferentes opciones de cifrado de los datos y de transporte. que se exponen más adelante en el presente documento.

La información de este apartado puede completarse desde:

Enlace: https://docs.oracle.com/en/database/oracle/oracle-database/19/admin/getting-started-with-database-administration.html

3. Implementación segura de la base de datos

En este punto se establece unas recomendaciones sobre la instalación del producto Oracle 19c orientadas a la mayoría de los casos posibles de uso de cada organización dependiendo del sistema.

Los comandos descritos a continuación deberán ser adecuados según el entorno y el sistema sobre el cual se estén ejecutando, atendiendo a los requisitos mínimos marcados por el fabricante.

El objetivo de los pasos descritos a continuación es que sean repetidos después de realizar una actualización de software sobre el motor de base de datos de Oracle 19c.

Para realizar la instalación de Oracle 19c primero se deben instalar los prerrequisitos necesarios por medio del siguiente comando:

```
dnf install y https://yum.oracle.com/repo/
OracleLinux/OL8/baseos/latest/x86_64/getPackage/
oracle-database-preinstall-19c-1.0-1.el8.x86_64.rpm
```

Al ejecutar el comando, el sistema arrojará una ventana de salida similar a la siguiente imagen:

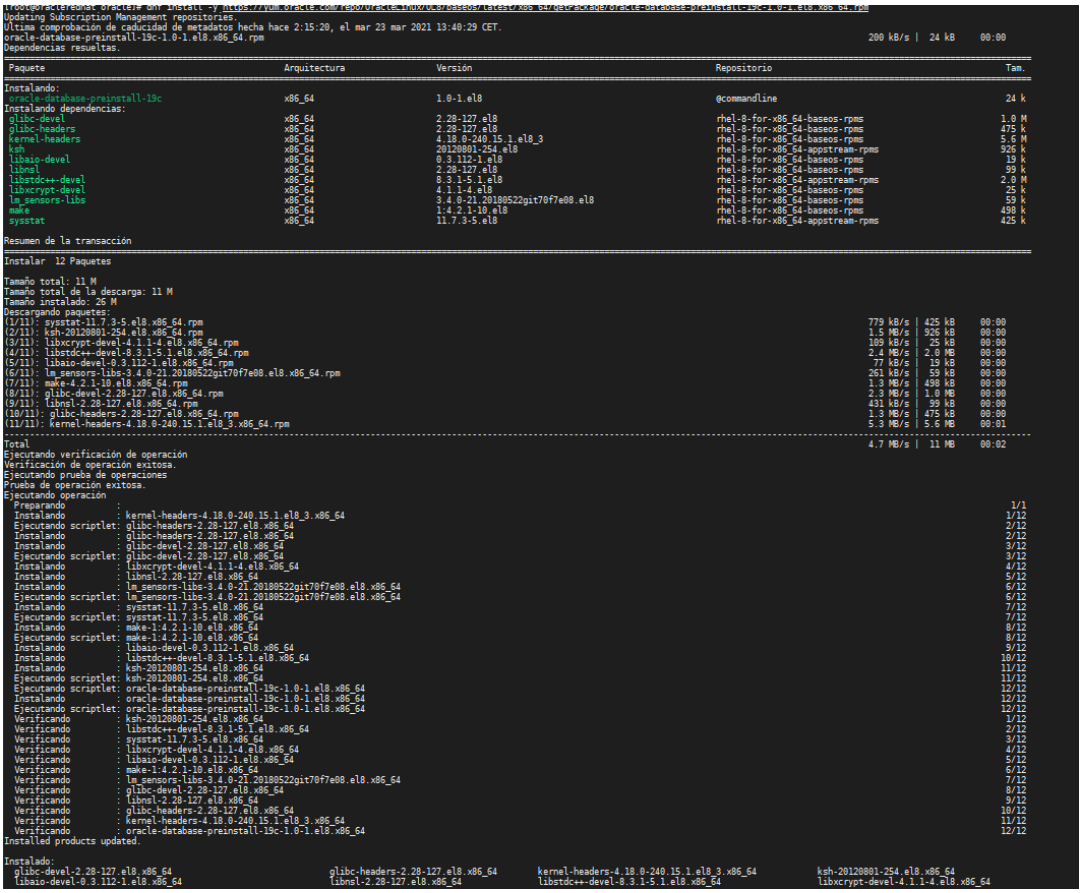


Ilustración 1 – Comando Instalación Prerrequisitos Oracle

Una vez finalizado, se debe descargar el software de instalación de Oracle e instalarlo con el siguiente comando:

```
rpm -i oracle-database-ee-19c-1.0-1.x86_64.rpm
```

Al ejecutar el comando, el sistema arrojará información similar a la siguiente:

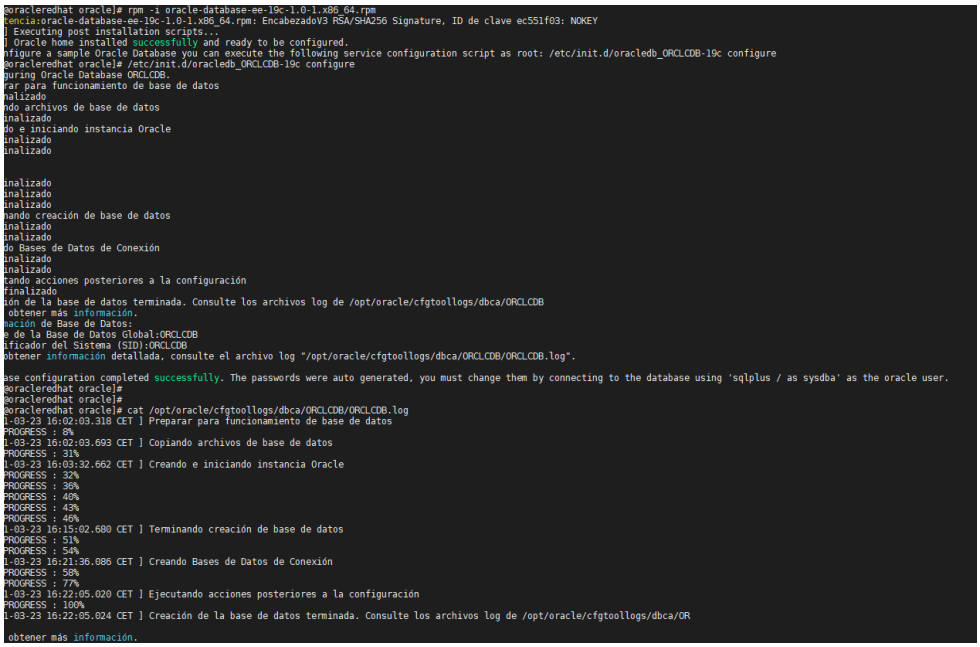


Ilustración 2 - Comando Software de Instalación Oracle

Nota: Puede obtener más información sobre la instalación en el documento del fabricante con título **"database-installation-guide-linux.pdf"**. Tenga en cuenta que algunos documentos del fabricante pueden necesitar inicio de sesión en su plataforma.

Debido a la nueva versión de Oracle, existen 2 nuevas opciones a la hora de crear una base de datos durante la instalación. Por ello, se deberán tener en consideración las siguientes opciones dependiendo de las necesidades de la organización:

NON-CDB

Base de datos semejante a las versiones anteriores 9.x, 10.x o 11.x

CDB

Base de datos "contenedor" destinada al almacenamiento de las bases de datos "pluggables".

Dicha base de datos CDB habilita la opción "multitenant" de la versión 19c, lo que posibilita que, sobre este contenedor, se creen varias bases de datos "pluggables" compartiendo los metadatos que alberga la base de datos contenedora o "CDB".

La creación de una CDB apenas difiere de una creación de otra base de datos de versiones anteriores.

Al arrancar el DBCA ofrece las siguientes opciones que se muestras a continuación:

- **Crear una CDB junto con una pluggable database ("PDB").**
- **Crear una CDB en modo avanzado, lo que permite la creación de la CDB vacía.**

Por diseño, puede conectar rápidamente una PDB a una CDB, desconectar la PDB de la CDB y luego conectar esta PDB a una CDB diferente. También puede clonar PDB mientras estén disponibles.

Puede consultar la documentación sobre las distintas arquitecturas en los siguientes enlaces:

Enlace: https://docs.oracle.com/en/database/oracle/oracle-database/18/rilin/deciding-between-multitenant-container-databases-and-non-cdbs-in-oracle-rac.html

Enlace: https://docs.oracle.com/en/database/oracle/oracle-database/19/multi/introduction-to-the-multitenant-architecture.html

3. Implementación segura de la base de datos

Después de la instalación del producto o parcheado del mismo, se debe revisar el estado de la solución y revisar la documentación del fabricante, ya que puede necesitarse bastionar objetos previamente bastionados o existir objetos nuevos después de la instalación.

A nivel de software se deben realizar las siguientes tareas de cumplimiento periódicamente:

- Mantener la versión del motor actualizada.
- Mantener las versiones de cualquier software extra al motor actualizada, por ejemplo Apex o cualquier otro producto que pueda modificar o incorporar objetos del servidor de base datos. Existen múltiples productos que dan de alta roles, permisos, packages, etc. En este punto la seguridad del motor de base de datos debe ser revisada de nuevo.
- Verificar que las cuentas de usuario **ORA_DBA** no sean root en el sistema operativo.
- Revisar las vulnerabilidades de cada componente perteneciente a la instalación. Se pueden consultar las vulnerabilidades conocidas (CVEs) por componente (CPE) en portales como el NIST.
- En caso de que se publiquen vulnerabilidades, y no se haya corregido por Oracle, se debe reportar este hecho a los responsables superiores de seguridad.
- Limpiar los ficheros temporales después de instalación de producto o parche (TMP_DIR, TMPDIR, TEMP, TMP…).

4. Configuración segura de la base de datos

A continuación, se ofrecen una serie de recomendaciones para reforzar la seguridad de la base de datos Oracle 19c una vez realizado el proceso de instalación.

4.1 Control de acceso

Diseñar unos controles de acceso adecuados y ajustados a las necesidades de explotación de los datos por parte de usuarios y herramientas es fundamental para reducir los riesgos de exfiltración o accesos no autorizados. La mayoría de las amenazas se engloban en esta categoría y se minimizan o directamente se eliminan manteniendo unos controles estrictos.

El acceso a una instancia o a una base de datos requiere que el usuario se autentique. Oracle proporciona distintos protocolos de autenticación como se ha expuesto en el punto 2 del documento.

Se recomienda hacer uso de mecanismos robustos de autenticación como SERVER, LDAP o Kerberos y evitar hacer uso de autenticación CLIENT, sobre todo en aquellos entornos donde no se puede garantizar la seguridad del cliente.

Se recomienda seguir el principio de mínimo nivel de privilegios, donde solo se permita a los usuarios acceder a la información y hacer las acciones que realmente necesitan, minimizando la superficie de exposición.

Diseñar unos controles de acceso adecuados y ajustados a las necesidades de explotación de los datos por parte de usuarios y herramientas es fundamental para reducir los riesgos de exfiltración o accesos no autorizados.

Se recomienda revisar, y si es necesario, revocar aquellos permisos de usuarios o grupos que no los necesitan.

En escenarios donde se almacenen datos sensibles, se recomienda, además, **revisar los privilegios**, establecer controles de acceso granulares como celda, columna o fila, a fin de evitar el acceso a datos sensibles desde entornos poco confiables, se puede acudir al punto 4.4.1 del presente documento para aplicarlo.

De forma predeterminada, un DBA tiene **acceso a cualquier tabla** en su instancia de base de datos. Esto supone un riesgo, sobre todo si la cuenta se ha vulnerado o se producen abusos en el uso de estos privilegios. Se recomienda revocar los privilegios de acceso a los datos del DBA si realmente no tiene la necesidad de acceder a dichos datos.

Se recomienda comprobar que no se ha otorgado acceso PUBLIC a ninguna base de datos.

Un usuario no autorizado puede acceder a información que reside en tablas del sistema si no se han protegido adecuadamente. Se recomienda revisar y proteger las tablas y vistas importantes del sistema como las contenedoras del código plsql: **All_source**, **dba_source** o las objetos **ALL_OBJECTS** , **DBA_OBJECTS**.

También, recomienda asignar privilegios a través de un modelo de roles, evitando la asignación directa a usuarios. Sin olvidar posteriormente asignar los roles a los usuarios específicos o concretos que puedan identificar quién hace qué.

Además, se recomienda usar los controles del sistema operativo para evitar que los administradores del sistema operativo obtengan demasiado acceso.

Por otro lado, **se recomienda asignar permisos de tipo DBA solo a través de un rol, y controlar el acceso a este rol mediante contextos de confianza**. Esto permite restringir el acceso únicamente a conexiones originadas desde equipos de confianza.

Así mismo, se recomienda revocar el privilegio de crear bases de datos a todos los usuarios, excepto el usuario DBA.

El Listener uno de los componentes con mayor probabilidad de ser susceptible a ataques, principalmente por ataque de denegación de servicio distribuido (DDoS). Es por ello que, los componentes de este servicio deben ser asegurados y auditados.

También, recomienda asignar privilegios a través de un modelo de roles, evitando la asignación directa a usuarios.

4. Configuración segura de la base de datos

A continuación, se exponen una serie de **recomendaciones sobre la configuración de seguridad** del servicio.

a. **Se debe aplicar medidas de seguridad sobre los accesos a los ficheros del servicio**: **lsnrctl**, **listener.ora**, **sqlnet.ora** y **tnslnsr**. Tanto **lsnrctl** como **tnslnsr** son ejecutables que deben tener permisos 0700.

Atendiendo a estos ficheros se debe configurar de manera segura el acceso al servicio. Se debe cambiar el nombre del SID por defecto, permitiendo sólo autentificaciones locales para su administración.

Para configurar dichos parámetros se pueden seguir los pasos indicados a continuación:

```
LOCAL_OS_AUTHENTICATION_ = ON , ADMIN_RESTRICTIONS_
LISTENER=ON
```

Se pueden ejecutar los siguientes comandos de auditoria sobre el servicio para capturar los posibles ataques de fuerza bruta que pueda recibir:

```
set current_listener <listener name>
set  log_directory  <oracle_home  path>/network/
admin
set log_file <sid name>.log
set log_status on
save_config
```

También se deben revisar los permisos de acceso al servicio, usando un nombre único para cada servicio. Se debe habilitar la auditoría del Listener para cotejar los valores con las siguientes marcas para identificar posibles ataques en vivo:

Mensaje
TNS-01169
TNS-01189
TNS-01190
TNS-12508
ORA-12525
ORA-28040
ORA-12170

b. **Es conveniente establecer log de paquetes de red.**

Dentro del fichero "**Listener.ora**" se deben configurar los siguientes parámetros como al menos el siguiente nivel de log.

- "SEC_PROTOCOL_ERROR_TRACE_ACTION" por "TRACE", "LOG o "ALERT"
- "SEC_PROTOCOL_ERROR_FURTHER_ACTION" por "DROP,3".

c. **Se deben editar y cambiar los puertos por defecto del fabricante editando el fichero "Listener.ora" o usando la utilidad "Netmgr".**

- Puerto 1521, 1522 por defecto de TNS del Listener.
- Puerto 1575 por defecto de Oracle Names Server.
- Puerto 1630 por defecto de Oracle Connection Manager – client connections
- Puerto 1830 por defecto de Oracle Connection Manager – administrative connections.
- Puerto 2483 por defecto el puerto para TNS en protocolo TCP/IP.
- Puerto 2484 por defecto de TNS en protocolo TCP/IP con SSL.

d. **Se debe establecer el parámetro "INBOUND_CONNECT_TIMEOUT" a 60 en los ficheros ".ora" para evitar ataques DDoS.**

e. **Se debe configurar la entrada en el fichero "Sqlnet.ora" de listas blancas y negras de IPs y rangos con acceso al servidor (Valid Node Checking).**

Puede tomar los siguientes valores de ejemplo:

```
tcp.validnode_checking = yes
tcp.invited_nodes = (x.x.x.x | name, x.x.x.x | name)
tcp.excluded_nodes=( x.x.x.x | name, x.x.x.x | name)
```

f. **Se debe cifrar el tráfico SQL entre clientes y servidor.**

En el fichero "**Sqlnet.ora**" se deben configurar las entradas required como obligatorias:

```
SQLNET.ENCRYPTION_SERVER = [accepted | rejected |
requested | required ]
SQLNET.ENCYRPTION_TYPES_SERVER = (algorithm name)
```

Desde el lado del cliente se deben configurar las entradas **required** como obligatorias:

```
SQLNET.ENCRYPTION_CLIENT = [ accepted | rejected |
requested | required ]
SQLNET.ENCYRPTION_TYPES_CLIENT = ( algorithm name
)
```

Se puede verificar la configuración con el comando:

```
SELECT NETWORK_SERVICE_BANNER FROM V$SESSION_
CONNECT_INFO;
```

Nota: Puede consultar más información sobre las configuraciones de seguridad mencionadas anteriormente en los siguientes enlaces:

https://www.integrigy.com/files/Integrigy_Oracle_Listener_TNS_Security.pdf

https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/

https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/keeping-your-oracle-database-secure.html

4.2 Auditoría

La auditoría es un componente fundamental en el refuerzo de la seguridad de un entorno informático, especialmente en entornos multi usuario, donde existe una necesidad de conocer las acciones realizadas por cada uno de los usuarios.

El registro de las acciones no deseadas o accesos no autorizados a los datos y el análisis posterior mejora los niveles de control de acceso a los datos y la prevención de accesos no autorizados, accesos malintencionados o configuraciones inadecuadas.

La **supervisión** del acceso de usuarios individuales y de aplicaciones, incluidas las acciones de administración del sistema, puede proporcionar un registro histórico de la actividad en sus sistemas de base de datos.

La auditoría de Oracle 19c genera y mantiene evidencias de auditoría para una serie de eventos de base de datos predefinidos. Los registros generados en una tabla o fichero de registro de auditoría y su análisis puede revelar patrones de uso que identificarían el mal uso del sistema. Una vez identificado, se pueden tomar acciones para reducir o eliminar dicho uso indebido del sistema. Se puede usar el comando create audit policy para generar el registro sobre el evento que se desea auditar. También se pueden definir acciones a auditar sobre el objeto como podrían ser leer de una tabla o ejecutar una función. Se pueden revisar todos los objetos auditables y sobre qué acciones en el enlace:

Enlace: https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/CREATE-AUDIT-POLICY-Unified-Auditing.html

Se pueden definir acciones a auditar sobre el objeto como podrían ser leer de una tabla o ejecutar una función.

4. Configuración segura de la base de datos

La función de auditoría permite auditar tanto a nivel de instancia como a nivel de base de datos individual, registrando de forma independiente todas las actividades en registros separados para cada una.

Cabe destacar que si se desea auditar y/o bastionar los accesos a los registros de una tabla, se debe verificar que todos los accesos a vistas, vistas materializadas, sinónimos o posibles salidas en fichero vía ETLs basados en los registros de dicha tabla sean igualmente auditados y/o bastionados.

Además, Oracle 19c incorpora herramientas de filtrado, políticas de seguridad y auditorías sobre todas las peticiones entrantes y salientes al motor de Base de datos con **"AVDF Oracle Audit Vault"** y **"Database Firewall"**. Se deben generar políticas mediante el WASS (Web Application Acceleration and Security Policy) antes de crear las reglas del WAF de Oracle.

En el caso de una de las bases de datos esté en desarrollo continuo se debe considerar la aplicación de la metodología OSSA para la aplicación de seguridad en bases de datos en construcción y pruebas.

Una vez configuradas las políticas WASS, se deben crear las reglas del WAF con los siguientes parámetros a modo de recomendación estándar:

- **AccessRules.** Se deben configurar los valores **ALLOW, DETECT**, y **BLOCK** de la política WASS.
- **AddressRateLimiting.** Limitación en entero de peticiones de una dirección IP.
- **CachingRules.** Reglas de cacheo para acceso a una aplicación web.
- **Captchas.** Configuración de captchas para evitar el acceso de bots.
- **CustomProtectionRules.** Reglas de bloqueo OCIDs y acciones permitidas.

- **DeviceFingerprintChallenge.** Reglas de denegación de enumeración al motor por bots usando técnicas de fingerprint.
- **GoodBots.** Lista blanca de bots con acceso al servidor web.
- **HumanInteractionChallenge.** Lista de interacciones humanas como movimientos de ratón, tiempos de reacción, scroll de páginas, etc para identificación de bots.
- **JsChallenge.** Lista de opciones de configuración de peticiones Javascript para bloqueo de bots.
- **Origin.** Key contenedora dentro de los Origins definidos en el WASS Policy.
- **OriginGroups.** Grupo origen del objeto Origin que pretende acceder definido en el WASS.
- **ProtectionRules.** Lista de reglas de protección y su descripción.
- **ProtectionSettings.** Lista de opciones a aplicar a la protectionRules.
- **ThreatFeeds.** Acciones a aplicar cuando se detecta tráfico malicioso.
- **Whitelists.** Lista blanca de direcciones IP que pueden traspasar el Firewall.

4.3 Medidas de protección de comunicaciones

El fabricante permite cifrar las comunicaciones a nivel de sockets o de capa de transporte (TLS). Se puede **revisar las ventajas e inconvenientes de cada opción** en el siguiente enlace:

Enlace: https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/configuring-network-data-encryption-and-integrity.html

Oracle Database proporciona cifrado e integridad de la red de datos nativos para garantizar que los datos estén seguros mientras viajan por la red.

Oracle Database es compatible con el algoritmo de cifrado Estándar de procesamiento de información federal (FIPS), Estándar de cifrado avanzado (AES). También es posible cifrar con triple DES.

Los algoritmos que el fabricante marca para el cifrado de red nativa en desuso y no deben usarse son: **DES, DES40, 3DES112, 3DES168, RC4_40, RC4_56, RC4_128 y RC4_256**. Los algoritmos mejorados en esta versión con el parche 2118136.2 son : AES128, AES192 y AES256.

Para elevar la seguridad del cifrado de red nativa se deben configurar los ficheros sql.ora en los clientes eliminando las siguientes entradas sí y solo sí existen:

```
SQLNET.ENCRYPTION_TYPES_CLIENT

SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT
```

Se deben configurar los ficheros sql.ora en los servidores eliminando las siguientes entradas sí y solo sí existen:

```
SQLNET.ENCRYPTION_TYPES_SERVER

SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER
```

Se debe configurar en el servidor el fichero sql.ora con los parámetros:

```
SQLNET.ENCRYPTION_SERVER = REQUIRED

SQLNET.ENCRYPTION_TYPES_SERVER = (AES256)

SQLNET.CRYPTO_CHECKSUM_SERVER = REQUIRED

SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER = (SHA512)

SQLNET.ALLOW_WEAK_CRYPTO_CLIENTS = FALSE
```

Se debe configurar cada cliente en el fichero sql.ora incorporando las siguientes entradas:

```
SQLNET.ENCRYPTION_CLIENT = REQUIRED

SQLNET.ENCRYPTION_TYPES_CLIENT = (AES256)

SQLNET.CRYPTO_CHECKSUM_CLIENT = REQUIRED

SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT = (SHA512)

SQLNET.ALLOW_WEAK_CRYPTO = FALSE
```

Nota: Puede consultar más información en los siguientes en los enlaces:

https://ittutorial.org/oracle-19c-network-encryption/

https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/release-changes.html

https://docs.oracle.com/en/database/oracle/oracle-database/19/asoag/configuring-transparent-data-encryption.html

https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/configuring-network-data-encryption-and-integrity.html

4.4 Medidas de protección de información

Las medidas de protección de la información incluyen tanto aquellas que se configuran o implementan en el entorno del servidor de bases de datos, como en el entorno del sistema operativo que ejecuta el servidor.

Oracle 19c permite generar claves de cifrado y cifrar las bases de datos. También permite cifrar objetos específicos como tablas, columna de tabla o celdas. Es responsabilidad de la organización conocer y asegurar los datos más sensibles. Este hecho depende del contenido de la información de las distintas bases de datos. No todos los datos poseen la misma criticidad y corresponde a la organización categorizar primero la información, y posteriormente asegurar el acceso a ella en función de la sensibilidad del dato.

Las vistas de administración de objetos cifrados como los permisos de estos objetos deben ser revisados. Solo los administradores deben poder ejecutar "**Select"** sobre estas vistas. Se puede asignar los permisos con el comando "**GRANT SELECT ON vista TO user**";

Las vistas que arrojarán información de objetos cifrados son:

Vista
ALL_ENCRYPTED_COLUMNS
DBA_ENCRYPTED_COLUMNS
USER_ENCRYPTED_COLUMNS
DBA_TABLESPACE_USAGE_METRICS
V$CLIENT_SECRETS
V$DATABASE_KEY_INFO
V$ENCRYPTED_TABLESPACES
V$ENCRYPTION_KEYS
V$ENCRYPTION_WALLET
V$WALLET

a. Se debe considerar la opción "Generar Claves de Cifrado" en el motor con el comando:

```
ADMINISTER KEY MANAGEMENT CREATE KEY [USING TAG 'tag']
[FORCE KEYSTORE]
IDENTIFIED BY [EXTERNAL STORE | keystore_password]
[WITH BACKUP [USING 'backup_identifier']];
```

b. **Se deben cifrar las bases de datos Oracle con el siguiente comando.**

```
ALTER TABLESPACE SYSTEM ENCRYPTION Type ENCRYPT
OPTIONS;
```

c. **La clave puede activarse con el siguiente comando:**

```
ADMINISTER KEY MANAGEMENT CREATE KEY USING TAG
OPTIONS;
```

El diccionario de objetos del motor de base de datos es una de las fuentes de ataque común al motor de base de datos. Si un atacante consigue el acceso al motor de base de datos podría dar de alta o modificar objetos de la base de datos. Es necesario conocer que objetos y con qué código si procede, son objetos lícitos de la base de datos. También es básico inventariar en que fechas se han ido creando y modificando los objetos.

Oracle 19c posee objetos de “Machine Learning” que crean índices de manera automática para optimizar el rendimiento del motor. Por ello los objetos de índices deben ser excluidos del análisis de inventariado.

- **Control inventariado de objetos. Cada vez que se produzca un cambio o subida de versión de desarrollo a producción, se debe documentar y fechar el contenido de objetos del motor, así como su código “PLSQL”.**

- **Se deben revisar los permisos de los sinónimos del motor. Cabe destacar revisar la seguridad del sinónimo creado.**

- **De igual manera que en el punto anterior, se deben asegurar los accesos y contenido de las vistas materializadas.**

Oracle permite accesos a otros servidores de base de datos vinculándolos. La seguridad de estos accesos y sus datos expuestos deben tener la misma consideración que la de los datos propios del motor Oracle.

A menudo las aplicaciones generan tablas temporales desde el código. Por ello, se recomienda el uso de “Global Temporary”, para que el acceso a estos datos sólo sea posible desde la conexión activa que la genera y la tabla sea destruida al final de la ejecución del código “PLSQL”.

Se recomienda el uso de “Global Temporary”, para que el acceso a estos datos sólo sea posible desde la conexión activa que la general.

Se debe revisar la seguridad sobre peticiones de incorporación de clases Java u otros objetos ajenos a los incorporados por el fabricante. Oracle permite la incorporación de clases ajenas al motor que pueden comprometer la seguridad del producto.

Se debe revisar la seguridad de todos los objetos contenedores de código “PLSQL” desarrollados para la explotación de los aplicativos como, procedimientos almacenados, funciones y packages. Especialmente sensibles son los objetos contenedor (procedures, functions y packages) de ejecución de SQL dinámico como execute inmmediate. Si estos objetos no se han parametrizado correctamente pueden ser objeto de ataques SQL dinámico.

Se debe cifrar el código de todos los objetos contenedores de código "PLSQL" desarrollados para la explotación de los aplicativos: procedimientos almacenados, funciones y packages. De esta manera un usuario que edite el objeto no podrá ver su contenido. Se pueden usar las vistas "**all_source**" y/o "**dba_source**" para ver rápidamente el código de estos objetos.

Las cuentas predeterminadas de usuario son un claro vector de ataque sobre la solución. Por ello, éstas deben cumplir unos criterios de seguridad para minimizar su exposición y posible explotación.

La seguridad de las cuentas de usuario se recomienda que cumplan con los siguientes criterios.

- **Segregación de Privilegios y Mínima Exposición: Solo se deben dar permisos a los objetos a los que se deba tener acceso.**
- **Se debe tener especial cuidado con los privilegios concedidos con la cláusula "ANY" que concede los privilegios a todos los objetos del mismo tipo. En Oracle 19c existen 148 posibles instrucciones de asignación de permisos a objetos que incluyen dicha cláusula ANY. Además, existen otros 84 posibles comandos de asignación de permisos genéricos que otorgan autorización a un conjunto de objetos.**

A continuación, se detallan las pautas de seguridad de contraseñas de las cuentas de usuario recomendadas:

- Contener al menos **12 caracteres**.
- Incluir **mayúsculas**, al menos dos.
- Contener al menos **dos minúsculas**.
- Contener al menos **dos números**.
- Contener al menos **dos caracteres especiales**.

Las cuentas predeterminadas de usuario son un claro vector de ataque sobre la solución.

- **No contener el nombre del usuario**.
- Configurar los siguientes parámetros:
  - **PASSWORD_REUSE_TIME:** Número de días durante los cuales no se puede reutilizar una contraseña.
  - **PASSWORD_REUSE_MAX:** Número de contraseñas que se deben usar antes de poder reutilizar la primera.
- Adicionalmente sobre las contraseñas se recomienda configurar los siguientes elementos:
  - **PASSWORD_LIFE_TIME:** Es el parámetro que define el número de días antes de que caduque la contraseña.
  - **PASSWORD_GRACE_TIME:** Es el parámetro que define el número de días máximo, tras la expiración de la contraseña, que el usuario tiene para cambiar su contraseña al vencimiento antes de que se rechacen todas las conexiones.
- Bloqueos de cuentas. Se recomienda configurar las variables:
  - **FAILED_LOGIN_ATTEMPTS:** Número de intentos fallidos de inicio de sesión permitidos antes de que se bloquee la cuenta de usuario.
  - **PASSWORD_LOCK_TIME:** Número de días que se bloqueará una cuenta después de una serie de intentos fallidos de inicio de sesión.
- Es de especial interés limitar el tiempo de sesión de cuenta para cuentas no pertenecientes a servidor aplicación con un valor aproximado de 90 minutos.

- Las cuentas de usuario no pertenecientes a servidores de aplicaciones pueden ser bloqueadas tras **dos intentos fallidos**.

- Tiempo de **cierre de sesión por inactividad 20 minutos** para cuentas no pertenecientes a servidor de aplicación.

- **Intentos fallidos** de login antes de bloqueo de cuenta **igual o inferior a 6**.

- Máximo tiempo de **vida de una contraseña** antes de forzar a cambiarla **180 días**.

- Modificar **cuenta SYS**:

  Esta cuenta puede realizar todas las funciones administrativas. Todas las tablas base (subyacentes) y las vistas del diccionario de datos de base de datos se almacenan en el esquema SYS. Estas tablas base y vistas son fundamentales para el funcionamiento de Oracle Database.

  Para mantener la integridad del diccionario de datos, las tablas del esquema SYS solo son manipuladas por la base de datos.

  Por otro lado, nunca deben ser modificadas por ningún usuario o administrador de base de datos. No se debe crear ninguna tabla en el esquema SYS. Al usuario SYS se le concede el privilegio SYSDBA, que permite al usuario realizar tareas administrativas de alto nivel, como copia de seguridad y recuperación.

  Las contraseñas pueden ser modificadas con el siguiente comando:

```
ALTER USER SYS IDENTIFIED BY "nueva contraseña";
```

Nota: Las nuevas contraseñas deben de cumplir con los requisitos mínimos de complejidad.

- Modificar **cuenta SYSTEM**.

  Esta cuenta puede realizar todas las funciones administrativas excepto copias de seguridad y recuperación, y actualización de la base de datos. Es cierto que esta cuenta puede utilizarse para realizar tareas administrativas diarias, pero Oracle recomienda encarecidamente la creación de cuentas de usuarios con nombre para administrar la base de datos Oracle para permitir el seguimiento de la actividad de la base de datos.

  Se puede modificar la contraseña del usuario SYSTEM desde "SQLPLUS" de la siguiente forma:

  ```
  ALTER USER SYSTEM IDENTIFIED BY "nueva contraseña ";
  ```

  Nota: Las nuevas contraseñas deben de cumplir con los requisitos mínimos de complejidad.

- Se recomienda configurar cuentas de usuario específicas para los servidores de aplicaciones.

- Las cuentas de usuarios deben ser nominativas para poder garantizar la trazabilidad de las distintas acciones ejecutadas en el motor. No deben usarse cuentas genéricas asociadas a los distintos roles sino cuentas que identifiquen unívocamente al autor de cualquier cambio.

- Es conveniente disponer un certificado de usuario para cada cuenta con acceso al motor.

- Se valora positivamente establecer un doble factor de autentificación al motor de base de datos para las cuentas de servidor de aplicación, con accesos como Google Authenticator u otras redes sociales (Social Sign-In Authentication), esto es recomendable para los servidores de aplicaciones donde se van a conectar un número indeterminado de usuarios.

4. Configuración segura de la base de datos

Con frecuencia después de realizar una instalación se restauran las bases de datos a explotar y se ejecutan scripts que pueden modificar permisos asociados a las cuentas de usuarios, roles, permisos de lectura, modificación, eliminación de objetos, etc.

Por tanto, antes y después de restaurar una base de datos en el motor, se debe comprobar las siguientes vistas del diccionario.

- Se deben verificar todos los roles existentes "**DBA_ROLES**" por si se han generados roles nuevos.
- Se deben comprobar los usuarios con asociación a estos roles "**DBA_TAB_PRIVS**".
- Se deben comprobar los privilegios de sistema asociales a roles de sistema y sus cuentas asociadas "**DBA_SYS_PRIVS**".
- Cabe tener en cuenta los usuarios APEX.

  Nota: APEX es el interfaz web para gestionar los espacios de trabajo. Se genera un usuario "ADMIN" con la misma contraseña que la de la cuenta system. Debe revisarse y cambiarse la contraseña de esta cuenta acorde a los requisitos mínimos de complejidad
- Las cuentas de usuarios deben asociarse a tipos o roles. Los roles o tipos de cuentas definidos por el fabricante como mínimos son:
- **Usuarios habituales de bases de datos:** Normalmente están restringidos a su esquema que contiene sus tablas, vistas, índices y procedimientos almacenados. Si los piratas informáticos piratean sus cuentas, no solo podrían ver / actualizar datos dentro del esquema de usuario, pero también acceder a objetos en otros esquemas que el usuario puede tener autorización para acceder.

Cuentas de aplicaciones: Son las cuentas de la base de datos que se utilizan para ejecutar sus aplicaciones, tanto comercial y de cosecha propia. Estas cuentas son similares a sus cuentas de usuario de base de datos habituales, pero dado que las aplicaciones deben funcionar 24 horas al día, 7 días a la semana, sus contraseñas a menudo se almacenan en varios servidores de nivel medio. El compromiso en estas cuentas de la base de datos puede provocar la pérdida de datos para toda la aplicación, incluidos los datos sobre los usuarios finales.

Administradores de aplicaciones: Estas cuentas se utilizan para administrar, parchear y actualizar su aplicación, y por lo tanto tener acceso completo a todos los datos y los procedimientos almacenados utilizados para la aplicación.

Analistas de datos o usuarios de inteligencia empresarial: Estos usuarios suelen tener acceso de lectura sin restricciones al esquema de aplicación sin pasar por los controles de acceso a nivel de aplicación.

Administradores de bases de datos (DBA): Son responsables de una amplia variedad de tareas para la base de datos incluida la gestión del rendimiento, el diagnóstico y el ajuste, la actualización y el parcheo, el inicio de la base de datos y apagado, y respaldo de la base de datos. Su acceso a la base de datos altamente privilegiado también le da acceso a cualquier dato confidencial contenidos en la base de datos (registros personales, de salud, de finanzas corporativas, etc.) aunque ese acceso no es necesario para realizar tareas de DBA. Los administradores de bases de datos tienen acceso a la gestión de cuentas y por lo tanto, suelen tener plena confianza en sus organizaciones. Estas cuentas de usuario suelen ser objeto de ataques.

Administradores de seguridad: Muchas organizaciones tienen administradores de bases de datos especializados que también tienen la responsabilidad de administradores de seguridad, incluida la gestión de cuentas de usuario, la gestión de claves de cifrado y gestión de auditoría.

Oracle recomienda que en ningún caso se genere una cuenta asociada a la vez al rol DBA y al rol de Administración de Seguridad. Se debe **generar dos cuentas nominativas distintas** si debe entregar las credenciales de ambos roles a una misma persona física para mejorar la gestión de la segregación de roles

Todas las cuentas deben tener asignación por defecto a espacios de trabajo distintos a “SYSTEM”.

- Las cuentas de usuarios asignadas a servidores de aplicaciones **no deben tener cuotas**.

- Los accesos a los objetos cuyo propietario sea "**SYS**" (objetos de administrador de sistema), "**DBA_**"(objetos de administrador de base de datos), "**USER_**"(tablas de permisos y roles de usuarios) deben ser revisados y sólo cuentas pertenecientes a perfiles "**SYS**" o "**DBA**" deben tener acceso a estos objetos. Si posteriormente se requiere el acceso de una cuenta a un objeto concreto, este hecho se debe analizar y documentar.

- Los permisos de roles sobre el catálogo "**SELECT_CATALOG_ROLE**", "**EXECUTE_CATALOG_ROLE**", "**DELETE_CATALOG_ROLE**", "**RECOVERY_CATALOG_OWNER**" deben ser revisados.

4.4.1 Row and column access control

Los datos sensibles pueden cifrarse a nivel de registro, columna, fila e incluso celda.

Se aconseja cifrar las tablas y/o columnas con datos más sensibles.

a. Se puede generar una columna cifrada con el comando "**ENCRYPT**" en el "ddl" de la creación de tabla como en el siguiente ejemplo:

```
CREATE TABLE nombre_tabla
  (campo_a VARCHAR2(11),
  Campo_cifrado VARCHAR2(16) ENCRYPT NO SALT);
```

b. Se pueden cifrar los procedimientos "packages", "funciones", "all Source". Esto hace que una persona con acceso al código del objeto "PLSQL" no pueda ver su código.

Se puede utilizar el siguiente comando:

```
wrap iname=input_file [ oname=output_file ]
```

c. Considere la opción de cifrado de discos, particiones a nivel de sistema operativo, con la opción de cifrar los datos de una celda especialmente sensible con la utilidad "**DBMS_CRYPTO.SQL**" con la siguiente pieza de código:

```
DECLARE
         input_string           VARCHAR2(16)  :=
'tigertigertigert';
    raw_input        RAW(128) :=
UTL_RAW.CAST_TO_RAW(CONVERT(input_
string,'AL32UTF8','US7ASCII'));
    key_string       VARCHAR2(8)  := 'scottsco';
    raw_key          RAW(128) :=
UTL_RAW.CAST_TO_RAW(CONVERT(key_
string,'AL32UTF8','US7ASCII'));
    encrypted_raw    RAW(2048);
    encrypted_string VARCHAR2(2048);
    decrypted_raw    RAW(2048);
    decrypted_string VARCHAR2(2048);
-- Begin testing Encryption:
BEGIN
          dbms_output.put_line('>  Input  String
: ' ||
            CONVERT(UTL_RAW.CAST_TO_VARCHAR2(raw_
input),'US7ASCII','AL32UTF8'));
     dbms_output.put_line('> ========= BEGIN TEST
Encrypt =========');
    encrypted_raw := dbms_crypto.Encrypt(
        src => raw_input,
        typ => DBMS_CRYPTO.DES_CBC_PKCS5,
```

```
        key => raw_key);
           dbms_output.put_line('> Encrypted hex
value              : ' ||
          rawtohex(UTL_RAW.CAST_TO_RAW(encrypted_
raw)));
decrypted_raw := dbms_crypto.Decrypt(
        src => encrypted_raw,
        typ => DBMS_CRYPTO.DES_CBC_PKCS5,
        key => raw_key);
    decrypted_string :=
      CONVERT(UTL_RAW.CAST_TO_VARCHAR2(decrypted_
raw),'US7ASCII','AL32UTF8');
dbms_output.put_line('> Decrypted string output
: ' ||
        decrypted_string);
if input_string = decrypted_string THEN
     dbms_output.put_line('> String DES Encyption
and Decryption successful');
END if;
dbms_output.put_line('');
dbms_output.put_line('> ========= BEGIN TEST Hash
=========');
    encrypted_raw := dbms_crypto.Hash(
        src => raw_input,
        typ => DBMS_CRYPTO.HASH_SH1);
dbms_output.put_line('> Hash value of input string
: ' ||
          rawtohex(UTL_RAW.CAST_TO_RAW(encrypted_
raw)));
dbms_output.put_line('> ========= BEGIN TEST Mac
=========');
    encrypted_raw := dbms_crypto.Mac(
        src => raw_input,
        typ => DBMS_CRYPTO.HMAC_MD5,
        key => raw_key);
dbms_output.put_line('>  Message  Authentication
Code      : ' ||
          rawtohex(UTL_RAW.CAST_TO_RAW(encrypted_
raw)));
dbms_output.put_line('');
dbms_output.put_line('> End of DBMS_CRYPTO tests
');
END;
/
```

4.4.2 Label based access control

Se ha implementado seguridad basada en firmas para localizadores de LOB. Los tipos de datos LOB CLOB, NLOB o BLOB se utilizan para guardar ficheros o grandes campos de texto de hasta 4GB de capacidad.

Los tipos de datos LOB CLOB, NLOB o BLOB se utilizan para guardar ficheros o grandes campos de texto.

Cabe hacer hincapié en cómo almacenar la clave de firma LOB en formato cifrado, la base de datos o PDB debe tener un almacén de claves TDE abierto. A partir de esta versión, se permite configurar la seguridad basada en firmas para localizadores de objetos grandes.

Estas claves de firma LOB pueden ser cifradas con el siguiente comando:

```
ALTER DATABASE DICTIONARY ENCRYPT CREDENTIALS
```

Los algoritmos TDE permitidos en Oracle 19c son:

- **Advanced Encryption Standard (AES) 128, 192, 256 bits**
- **Triple Data Encryption Standard (TDES) 168 bits.**

Nota: Para más información puede consultarse el siguiente enlace:

https://docs.oracle.com/en/database/oracle/oracle-database/19/asoag/configuring-transparent-data-encryption.html

Aquellas medidas de seguridad con las que cuenta el análisis de privilegios vienen heredadas desde versiones anteriores.

Nota: Para obtener más información acerca de la seguridad sobre el análisis de privilegios visite las guías de seguridad de Oracle Database Vault y Oracle Database Security Guide.

https://docs.oracle.com/en/database/oracle/oracle-database/19/dvadm/index.html

https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/index.html

Se recomiendan establecer roles para conceder y renovar privilegios administrativos desde cualquier esquema. Los grupos SYSOPER y SYSBACKUP pueden segmentarse a los esquemas que defina el administrador.

Será conveniente admitir los enlaces de autenticación simple y capa de seguridad (SASL) y Seguridad de la capa de transporte (TLS) para las conexiones de Microsoft Active Directory.

Siguiendo con la conectividad de Oracle, en esta versión de producto sería necesario contar con Oracle Native Encryption y autenticación SSL para los diferentes usuarios conectados simultáneamente.

Se recomienda hacer uso de la nueva compatibilidad con la coincidencia parcial de DN (nombre de dominio) basada en nombres de host para la coincidencia de certificados de servidor agregando así un doble factor de autenticación entre el cliente y el servidor

Por otra parte, se establecerán opciones de auditoría en instrucciones SQL de nivel superior. Dicha característica de instrucciones de nivel superior de auditoría unificada se corresponde a una auditaría sobre aquellas modificaciones de diccionario de objetos ejecutadas por los usuarios, así se permite auditar las actividades de usuario de nivel superior (o usuario directo) en la base de datos, pero sin recopilar datos de auditoría de actividad de usuario indirectos. Esto agrega una capa más seguridad a la hora de buscar trazas sobre posibles incidentes.

En esta versión de producto sería necesario contar con Oracle Native Encryption y autenticación SSL para los diferentes usuarios conectados simultáneamente.

Nota: Se puede obtener más información ara más información pueden consultarse las guías de seguridad en los enlaces:

https://docs.oracle.com/en/database/oracle/oracle-database/19/asoag/

https://docs.oracle.com/en/database/oracle/oracle-database/19/dbseg/

Dependiendo de los datos almacenados en cada base de datos del motor, Oracle 19c posee servicios de auto análisis de datos basados en inteligencia artificial y patrones de datos asociados a la configuración regional de la instalación. La herramienta busca datos como nombres, tarjetas de crédito, sueldos, importes direcciones personales, etc. y permite asegurar el acceso a los mismos.

Muchas veces, los administradores de bases de datos no poseen la información necesaria sobre cuáles son los datos más sensibles en cada base de datos para asegurar adecuadamente la información. El componente de Oracle 19c "Label Security" permite la clasificación de datos a nivel de fila y proporciona mediación de acceso, lista para usar en función de la clasificación de datos y la autorización de la etiqueta del usuario o la autorización de seguridad.

El software "Oracle Label Security" se instala de forma predeterminada, pero no se habilita automáticamente. Puede habilitar Oracle Label Security en SQL* Plus o mediante el Asistente de configuración de la base de datos de Oracle (DBCA).

El administrador predeterminado de Oracle Label Security es el usuario "**LBACSYS**". Para administrar "Oracle Label Security", puede utilizar un conjunto de paquetes "PL / SQL" y funciones independientes en el nivel de línea de comandos o "Oracle Enterprise Manager Cloud Control". Para obtener más información sobre las políticas de "Oracle Label Security", se puede consultar "**ALL_SA_***", "**DBA_SA_***", "**SA_***" o "**USER_SA_**".

Los objetos y utilidades para categorizar los datos son los siguientes:

Paquete	Propósito
SA_SYDBA	Para crear, modificar y eliminar políticas de seguridad de etiquetas de Oracle.
SA_COMPONENTS	Para definir los niveles, compartimentos y grupos de la política.
SA_LABEL_ADMIN	Para realizar funciones administrativas de políticas de etiquetas estándar, como la creación de etiquetas.
SA_POLICY_ADMIN	Para aplicar políticas a esquemas y tablas.
SA_USER_ADMIN	Gestionar autorizaciones de usuarios para niveles, compartimentos y grupos, así como privilegios de unidad de programa. También para administrar los privilegios de los usuarios.
SA_AUDIT_ADMIN	Gestionar autorizaciones de usuarios para niveles, compartimentos y grupos, así como privilegios de unidad de programa. También para administrar los privilegios de los usuarios.
SA_SESSION	Gestionar autorizaciones de usuarios para niveles, compartimentos y grupos, así como privilegios de unidad de programa. También para administrar los privilegios de los usuarios.
SA_UTL	Para configurar opciones para auditar tareas administrativas y uso de privilegios.

“Enterprise Manager” ofrece un entorno gráfico de descubrimiento y categorización de datos sensibles.

Se debe categorizar y documentar la información contenida en las bases de datos independientemente del nivel de configuración de seguridad que se aplique. Se debe conocer que información se almacena, dónde y la sensibilidad de la misma.

4.5 Políticas de backup

En ocasiones una deficiente política de protección de las copias de seguridad permite el acceso no autorizado a datos que han dejado de estar protegidos por la seguridad del servidor.

Si los datos almacenados en copias de seguridad se dejan desprotegidos, pueden ser accedidos directamente desde el servicio de backup.

Se recomienda cifrar todos los ficheros de backup e imágenes de archivo, independientemente del medio donde se almacenen.

Se recomienda garantizar que la restauración de cualquier copia de seguridad debe requerir un acceso controlado a la clave de cifrado y debe ser auditado, tanto el acceso como la propia restauración.

Se deben mantener las prácticas recomendadas por el fabricante de copias de seguridad.

Se recomienda realizar copias de seguridad periódicas. Al menos también debe generarse una copia de seguridad incremental semanal en domingo con conservación de cuatro semanas. También debe generarse una copia incremental cada día uno de mes, conservándose los doce últimos meses. También debe generarse una copia de seguridad anual, conservándose durante cinco años.

Se deben almacenar copias de seguridad en lugares distintos a la ubicación física del servidor de producción.

Se recomienda la ejecución de pruebas de recuperación periódicamente al igual que ejecutar periódicamente simulacros de recuperación frente a desastres.

Debe generarse una copia de seguridad incremental diaria que debe conservarse durante siete días.

5. Otras consideraciones

A continuación, se exponen algunas buenas prácticas de generales independientemente de la versión del producto. El fabricante incorpora documentación sobre el producto que se han referenciado en el presente documento y complementan la información de éste. Algunas funcionalidades del fabricante pueden necesitar la adquisición de licencias independientes de propia del motor de base de datos. Las capturas de pantalla incorporadas en el presente documento fueron realizadas sobre un sistema operativo Rhell 8, por lo tanto, la salida en pantalla alguno de los pasos puede no corresponder exactamente a la correspondiente en otra versión de sistema operativo.

Se recomienda activar el redireccionamiento Active Data Guard.

Oracle 19c incorpora esta funcionalidad extra como base de datos reflejada de recuperación ante desastres. Además, se permite el uso de la base de datos reflejada para explotación de datos en modo lectura como explotación de informes u otros procesos con acceso de lectura y denegación de escritura. Transacciones de lectura pueden ser redirigidas al espejo. Esto permite tener una copia de seguridad en tiempo real en una ubicación física distinta a la base de datos original.

Oracle 19c incorpora tablas de partición híbrida que pueden residir físicamente en ubicaciones físicamente distintas del resto de tablas. Se debe aplicar partición híbrida en aquellas tablas que deban residir en una ubicación física aplicando una configuración de seguridad mayor sobre los accesos físicos donde residan dichas tablas.

Se recomienda almacenar los datos en sistemas de discos redundantes RAID 1 o 5 por ejemplo.

Se recomienda replicar los ficheros logs y redo logs en ubicaciones físicamente distintas. Son ficheros que almacenan los cambios del log de transacciones especialmente sensibles ante análisis de cambios de datos.

En el caso de una de las bases de datos esté en desarrollo continuo se debe considerar la aplicación de la metodología OSSA para la aplicación de seguridad en bases de datos en construcción y pruebas.

- **Se recomienda generar alarmas (IAM policies) de consumo y utilización del motor de BD.**
- **Se recomienda cifrar discos a nivel de controladora.**
- **Se recomienda documentar los procedimientos de cambios en el motor de BD, así como las diferentes tareas de administración.**
- **Se recomienda configurar clúster de alta disponibilidad.**
- **Se recomienda los discos viejos o estropeados que hayan contenido información crítica deben ser borrados magnéticamente antes de ser finalmente descartados.**
- **Se recomienda verificar los permisos de los ficheros del motor de base de datos y rutas de copias de seguridad.**

Autenticación: es el proceso por el cual un sistema verifica la

6. Glosario

identidad de un usuario. En Oracle 19c, este proceso se realiza fuera del entorno de la aplicación, a través de un módulo de autenticación. Mediante distintos módulos que incorpora Oracle, se puede hacer uso de protocolos de autenticación como LDAP, OS,TNS o Kerberos. Habitualmente la autenticación de los usuarios la realiza el sistema operativo o un servidor externo.

Autorización: es el proceso de determinar si un usuario autenticado, dispone de acceso a la información y permisos que está solicitando. Este proceso se realiza íntegramente dentro de Oracle 19c, consultando los permisos asociados a una identidad concreta.

Cifrado nativo de Oracle 19c: El cifrado nativo de Oracle 19c proporciona capacidad de cifrado incorporada para proteger las imágenes de copia de seguridad de la base de datos y los archivos clave de la base de datos de accesos no autorizados mientras se encuentran en un medio de almacenamiento externo. El cifrado es un componente clave en la protección de datos fuera de línea.

TLS: Transport Layer Security es un protocolo de comunicaciones cuyo principal objetivo es proporcionar privacidad e integridad de datos entre dos aplicaciones que se comunican. El protocolo está compuesto por dos capas: el protocolo de registro TLS y el protocolo de negociación (handshake) TLS. Durante la negociación TLS, se utiliza un algoritmo de clave pública para intercambiar de forma segura firmas digitales y claves de cifrado entre un cliente y un servidor. La información de identidad y la clave se utilizan para establecer una conexión segura para la sesión entre el cliente y el servidor. Una vez establecida la sesión segura, la transmisión de datos entre el cliente y el servidor se cifra mediante un algoritmo simétrico, como AES.

Active Data Guard: es una de las soluciones de replicación de BBDD de Oracle. Oracle Active Data Guard es una evolución de la anterior con mejoras de disponibilidad, rendimiento y protección.

Oracle Label Security : es una funcionalidad que registra y aplica el permiso de acceso a los datos según los códigos del proyecto, las regiones o las clasificaciones de datos. Estos controles reducen el riesgo de acceso no autorizado a datos confidenciales y ayudan a demostrar el cumplimiento normativo.

AVDF: Oracle Audit Vault and Database Firewall es una solución de

monitorización y filtrado de actividad en la base de datos. Incorpora agentes de recopilación de datos de las auditorías, firewall de base de datos, herramientas de análisis e informes.

Global Temporary: tipo de tabla de Oracle cuya temporalidad puede ser definida a nivel de transacción (los datos existen mientras se realiza la transacción) o a nivel de sesión (los datos existen mientras dura la sesión). Los datos en una tabla temporal son propios y privativos de la sesión Oracle que la está utilizando.

DBCA: DataBase Computer Assistant. Software que facilita la creación de bases de datos Oracle. Requiere que el software de Oracle para gestionar bases de datos esté instalado.

DBUA: DataBase Computer Assistant. Asistente para actualizar bases de datos.

CDB: Base de datos contenedor de objetos ya sean esquemas, otros objetos de esquema u otros objetos.

PDB: Base de datos agrupable (Pluggable) que se comporta como un contenedor más, dentro de la arquitectura de un CDB, compone una colección de objetos independiente de otros pdbs con sus propios datafiles.

OPatch: utilidad basada en Java que permite la aplicación y reversión de parches al software de Oracle.

Oracle Enterprise Manager: es una plataforma de administración que proporciona un único cuadro de mando para administrar todas las bases de datos de Oracle.

SSH: Secure Shell (SSH) es un protocolo para el inicio de sesión remoto seguro y otros servicios de red segura a través de una red insegura. SSH se puede utilizar como base para una serie de servicios de red seguros ya que proporciona un cifrado robusto, autenticación del servidor y protección de la integridad. También proporciona compresión de datos. SSH se usa durante la instalación para configurar los nodos miembros del clúster, y SSH se usa después de la instalación por los asistentes de configuración, Oracle Enterprise Manager, Opatch y otras características.

RCAC: Row and Column Access Control. Permite controlar el acceso

a una tabla a nivel de fila, de columna o en ambos y se puede utilizar para complementar el modelo de privilegios de tabla, garantizando que la información esté protegida adecuadamente y que los usuarios solo tienen acceso al subconjunto de datos que se requieren para realizar sus tareas laborales y cumplir con normativas y regulaciones específicas.

Machine Learning: Oracle Machine Learning es una funcionalidad ofrecida en el producto SQL Developer del fabricante que descubre patrones y puede ofrecer conocimientos en los datos almacenados.
OUI: Oracle Universal Installer. Instalador universal de Oracle, herramienta para instalar software de Oracle.

LBAC: Label Based Access Control. Es un modelo de seguridad que está destinado principalmente a aplicaciones gubernamentales o con grados de clasificación conocidos, ya que requiere que los datos y los usuarios se clasifiquen con un conjunto fijo de reglas que se implementan.

DBA: Database Administrator.

MV: Materialized View. Una vista materializada en Oracle es un objeto de base de datos que contiene los resultados de una consulta. Son copias locales de datos ubicados de forma remota o se utilizan para crear tablas de resumen basadas en agregaciones de datos de una tabla. Las vistas materializadas, que almacenan datos basados en tablas remotas, también se conocen como instantáneas, el motor de la base de datos puede devolver los datos de una vista materializada para mejorar el rendimiento. Los datos constan de resultados precalculados de las tablas que se indican en la definición de la vista materializada.

FIPS: Federal Information Processing Standards. La Publicación 140-2 del Estándar federal de procesamiento de información (FIPS) es un estándar gubernamental de Estados Unidos que define los requisitos mínimos de seguridad para módulos criptográficos en productos de tecnología de la información, tal como se define en la sección 5131 de la Ley de reforma de la administración de tecnologías de la información de 1996.

7. Tabla resumen de medidas de refuerzo de la seguridad

ÁMBITO	NUM.	MEDIDA	MOTIVO
IMPLEMENTACIÓN SEGURA	1	En sistemas Unix o Linux, se recomienda especificar diferentes nombres de usuario a los creados de forma predeterminada.	Evitar el uso de nombres predeterminados para planificar ataques a la base de datos.
	2	En sistemas Windows, se recomienda cambiar esta configuración predeterminada y especificar unos nombres de usuarios distintos para cada función.	Evitar el uso de nombres predeterminados para planificar ataques a la base de datos.
	3	Se recomienda crear identificadores de usuarios propietarios de instancias específicos para cada instancia, añadiéndolo solo como miembro del grupo propietario de la instancia y no usarlo en ningún otro grupo.	Permite disponer de un mayor control en el número de usuarios y grupos que pueden modificar la instancia.
	4	Durante la instalación, se recomienda hacer uso de contraseñas robustas que cumplan las directivas de seguridad de la organización.	Minimiza la posibilidad de ataques por fuerza bruta.

7. Tabla resumen de medidas de refuerzo de la seguridad

ÁMBITO	NUM.	MEDIDA	MOTIVO
CONTROL DE ACCESO	5	Se recomienda hacer uso de mecanismos robustos de autenticación y comunicación como SERVER, LDAP TLS o Kerberos y evitar hacer uso de autenticación CLIENT, sobre todo en aTTquellos entornos donde no se puede garantizar la seguridad del cliente.	Mejorar la seguridad y confiabilidad de los mecanismos de autenticación.
	6	Se recomienda seguir el principio de mínimo nivel de privilegios, donde solo se permita a los usuarios acceder a la información y hacer las acciones que realmente necesitan.	Minimizar la superficie de exposición.
	7	Se recomienda revisar y, si es necesario, revocar aquellos permisos de usuarios o grupos que no los necesitan.	Minimizar la superficie de exposición.
	8	En escenarios donde se almacenen datos sensibles, se recomienda, además, revisar los privilegios, establecer controles de acceso granulares.	Evitar el acceso a los roles sensibles desde entornos poco confiables.
	9	Se recomienda revocar los privilegios de acceso a los datos del DBA si realmente no tiene la necesidad de acceder a dichos datos.	De forma predeterminada, un DBA tiene acceso a cualquier tabla en su instancia de base de datos. Esto supone un riesgo, sobre todo si la cuenta se ha vulnerado o se producen abusos en el uso de estos privilegios.
	10	Se recomienda comprobar que no se ha otorgado acceso PUBLIC a ninguna base de datos.	Minimizar la superficie de exposición.
	11	Se recomienda revisar y proteger las tablas y vistas importantes del sistema como ALL_OBJECTS, ALL_SOURCE.	Un usuario no autorizado puede acceder a información que reside en tablas del sistema si no se han protegido adecuadamente.
	12	Se recomienda asignar privilegios a través de un modelo de roles, evitando la asignación directa a usuarios.	Mejorar el control y mantenimiento de los privilegios de acceso.
	13	Se recomienda usar los controles de acceso del sistema operativo.	Evitar que los administradores del sistema operativo obtengan demasiado acceso.
	14	Se recomienda asignar permisos de tipo DBA solo a través de un rol, y controlar el acceso a este rol mediante contextos de confianza.	Permite restringir el acceso únicamente a conexiones originadas desde equipos de confianza.
	15	Se recomienda revocar el privilegio de crear bases de datos a todos los usuarios, excepto el DBA.	Minimizar la superficie de exposición.

7. Tabla resumen de medidas de refuerzo de la seguridad

ÁMBITO	NUM.	MEDIDA	MOTIVO
AUDITORÍA	16	Se recomienda revisar las necesidades de registro de eventos de auditoría y seleccionar únicamente aquellos eventos importantes para la organización o los que estén relacionados con la seguridad del sistema.	Controlar la información de auditoría generada, evitando datos no relevantes y problemas de almacenamiento que pueden derivar en pérdida de evidencias relevantes.
	17	Se recomienda crear un rol AUDITOR y otorgar los privilegios necesarios para leer y administrar los eventos de auditoría.	Controlar quién y cómo se puede acceder a la información de auditoría.
	18	Se recomienda controlar el acceso al rol AUDITOR mediante contextos de confianza.	Permite restringir el acceso únicamente a conexiones originadas desde equipos de confianza.
	19	Se recomienda evitar que los ficheros de auditoría generados puedan ser copiados, modificados o eliminados directamente por el administrador del sistema operativo o por otro usuario no autorizado de la plataforma.	Evitar la exfiltración de datos o el acceso a información sensible de auditoría saltándose los mecanismos de seguridad de la base de datos.
	20	Se recomienda hacer uso de un servicio centralizador de registros de auditorías.	Unificación de diversos orígenes de auditoría, facilitar la correlación de logs y evitar la pérdida o manipulación de evidencias.
	21	Se recomienda cifrar los registros de autoría almacenados en el disco (datos en reposo), tanto en el servidor de bases de datos, como en el servicio centralizador de registros, en caso de disponer de uno.	Evitar la exfiltración de datos o el acceso a información sensible de auditoría saltándose los mecanismos de seguridad de la base de datos.
	22	Se recomienda auditar todas las acciones del DBA.	Mantener un registro de auditoría de las acciones administrativas que puedan comprometer el sistema.
	23	Se recomienda auditar el acceso de los usuarios, en particular aquellos que tengan acceso a los datos sensibles.	Mantener un registro de auditoría de las acciones de los usuarios.
	24	Se recomienda auditar todos los accesos a las tablas importantes.	Mantener un registro de auditoría de las acciones que puedan comprometer el sistema.
	25	Se recomienda auditar los objetos del esquema SYS.	Mantener un registro de auditoría de estos objetos permitirá tener un control sobre los cambios los cambios realizados en objetos como tablas, vistas, índices, etc…
	26	Se recomienda auditar todos los intentos de crear bases de datos.	Mantener un registro de auditoría de las acciones administrativas que puedan comprometer el sistema.

7. Tabla resumen de medidas de refuerzo de la seguridad

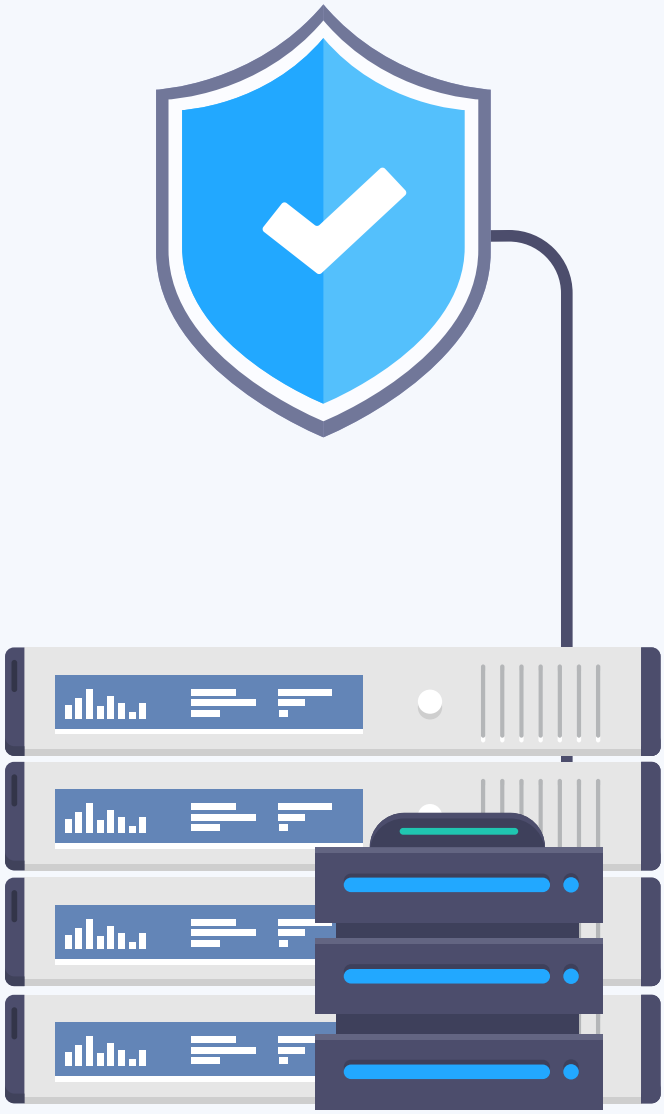
ÁMBITO	NUM.	MEDIDA	MOTIVO
PROTECCIÓN DE COMUNICACIONES	27	Se recomienda hacer uso del cifrado con los algoritmos seguros expuestos en la capa de comunicaciones.	Evitar la captura de datos en tránsito a través de la red.
	28	Se recomienda no cifrar con los algoritmos marcados como obsoletos por el fabricante.	El fabricante marca como algoritmos obsoletos los siguientes algoritmos: DES, DES40, 3DES112, 3DES168, RC4_40, RC4_56, RC4_128 y RC4_256, por lo que no deben ser utilizados.
	29	Se recomienda utilizar conjuntos de algoritmos de cifrado robustos avalados por el Centro Criptológico Nacional.	Evitar la explotación de vulnerabilidades en algoritmos débiles u obsoletos.
	30	Se recomienda verificar que se dispone de una versión reciente de Oracle 19c	Las versiones más antiguas, hacen uso de algoritmos de cifrado débiles o vulnerables que no deben ser utilizados.
	31	Se recomienda instalar el parche de soporte de algoritmos avanzados del presente documento.	Oracle 19c disponen del parche 2118136.2 para instalar cifrados avanzados.
	32	Para habilitar TLS 1.2 (SSl 3.0) en Oracle 19c, se recomienda hacer uso de certificados emitidos por una entidad de certificación de confianza.	Permite validar correctamente la cadena de emisión del certificado y por lo tanto su confianza.
	33	Se recomienda revisar y configurar los puertos utilizados por todas las instancias del servidor utilizando el archivo de servicios para asignar el nombre del servicio en el archivo de configuración del administrador de la base de datos del servidor a su número de puerto.	Minimizar la superficie de exposición, habilitando únicamente los puertos de comunicaciones necesarios.
	34	Se recomienda configurar el WAF de Oracle.	El Firewall propio del fabricante le permite establecer reglas específicas adaptadas cada entorno.
PROTECCIÓN DE LA INFORMACIÓN	35	Se recomienda diseñar y hacer uso de políticas de acceso granular a registros o columna(RCAC) en aquellos entornos donde existan una regulaciones o normativas que cumplir y el acceso a los datos tenga que realizarse según el contexto de quien lo solicita.	Cumplir con el principio de “necesidad de conocer”.

7. Tabla resumen de medidas de refuerzo de la seguridad

ÁMBITO	NUM.	MEDIDA	MOTIVO
PROTECCIÓN DE LA INFORMACIÓN	36	Se recomienda hacer uso de LBAC a nivel de registros cuando se maneje información sensible o clasificada relacionada con entidades del gobierno.	Cumplir con el principio de "necesidad de conocer".
	37	Se recomienda hacer uso de LBAC a nivel de registros cuando las siguientes afirmaciones son ciertas: –Se conoce el grado de clasificación de los datos. –La clasificación de los datos se puede representar por una o varias etiquetas de seguridad LBAC. –Las reglas de autorización se pueden enlazar a los componentes de la etiqueta de seguridad.	Cumplir con el principio de "necesidad de conocer".
	38	Se recomienda LBAC a nivel de columna cuando: –Se requiere proteger columnas sensibles de accesos no autorizados a los dueños de la tabla o incluso al DBA. –Se requiere proteger tablas completas de accesos no autorizados a los dueños de la tabla o incluso al DBA.	Cumplir con el principio de "necesidad de conocer".
	39	Independientemente de los controles de acceso que se implementen, se recomienda hacer uso de mecanismos de cifrado en reposo de los datos, tablas, ficheros de auditoría y archivos de respaldo a nivel del sistema operativo.	Evitar el acceso no autorizado a la información sensible fuera del ámbito de protección de la base de datos.
BACKUP	40	Se recomienda cifrar todos los ficheros de backup e imágenes de archivo, independientemente del medio donde se almacenen.	Evitar el acceso no autorizado a las copias de seguridad.
	41	Se recomienda garantizar que la restauración de cualquier copia de seguridad debe requerir un acceso controlado a la clave de cifrado y debe ser auditado, tanto el acceso como la propia restauración.	Evitar el acceso no autorizado a las copias de seguridad y registrar cualquier tipo de acceso mediante una auditoría.

centro criptológico nacional

centro criptológico nacional



www.ccn.cni.es

www.ccn-cert.cni.es

oc.ccn.cni.es