



Eusko Jaurlaritzaren Informatika Elkarte
Sociedad Informática del Gobierno Vasco

EncryptSys For Oracle:

Manual de usuario

Fecha: 20/09/2007

Referencia:

EJIE S.A.
Mediterráneo, 3
Tel. 945 01 73 00*
Fax. 945 01 73 01
01010 Vitoria-Gasteiz
Posta-kutxatila / Apartado: 809
01080 Vitoria-Gasteiz
www.ejie.es

Control de documentación

Título de documento: EncryptSys For Oracle. Manual de usuario.

Histórico de versiones

Código:

Versión: 1.0

Fecha:

Resumen de cambios:

Cambios producidos desde la última versión

Primera versión.

Control de difusión

Responsable: Ander Martínez

Aprobado por: Ander Martínez

Firma:

Fecha: 20/09/2007

Distribución:

Referencias de archivo

Autor: Consultoría de áreas de conocimiento

Nombre archivo: EncryptSys For Oracle. Manual de usuario v1.0

Localización:

Contenido

	Capítulo/sección	Página
1	Introducción	4
2	Conceptos básicos	4
3	Procedimiento de ofuscación	5
4	Utilidad práctica	6

1 Introducción

El presente documento describe cuales son los procesos de configuración que deben realizarse para la correcta explotación de EncryptSys For Oracle.

Se describen entonces todas las posibles peticiones de gestión de EncryptSys For Oracle que deben ser tramitadas por el servicio de implantación.

2 Conceptos básicos

EncryptSys For Oracle es una herramienta flexible para dar solución al cifrado de columnas en una base de datos. EncryptSys For Oracle es una solución completa del cifrado de la base de datos que no permite un algoritmo reversible.

EncryptSys For Oracle permite:

- * Ahorro de tiempo y costes.
- * Una Solución Fácil de utilizar en el cifrado de la Base de datos
- * Ofuscación sin algoritmo reversible
- * Transparente
- * Flexible

Las principales características de EncryptSys For Oracle son:

- * Cifrado de columna a nivel de la base de datos
- * Acceso a los recursos de la seguridad que facilitan la protección eficaz de los datos
- * Controles y mecanismos que proporcionan la autenticación, el cifrado, y la integridad de datos.

El paquete PKG_ENCRYPTSYS es capaz de encriptar campos tipo VARCHAR, VARCHAR2, NUMBER, DATE y LONG; con lo que cubre un amplio espectro de los tipos de campos normalmente usados en las BD.

3 Procedimiento de ofuscación

Todas las peticiones de encriptación de información se aconseja que se efectúen siempre preferiblemente fuera de los horarios de trabajo habituales, y se efectúe obligatoriamente una copia de seguridad de la información existente antes de efectuar cualquier modificación previa sobre el SGBDR.

Asistencia técnica deberá proporcionar un script debidamente cumplimentado.

Se facilita un ejemplo de Script de Ejecución (encriptar-ejemplo.sql), desde el cual, cambiando los parámetros necesarios, pueden lanzarse la encriptación/ofuscación de diferentes campos y tablas.

Este Script será modificado y enviado por Asistencia Técnica, y para la modificación se tendrán en cuenta las propias necesidades de encriptación/ofuscación necesarias dentro del propio proyecto. Si por ejemplo, se desea ofuscar una única tabla con 2 de sus campos, la modificación del script puede quedar como a continuación se detalla (ENCRIPtar_S73A.sql que viene acompañado de su encryp_s73a.sh correspondiente):

```
CONNECT S73A/S73A@EDE18;
--SPOOL log_encriptar.log;
-----
-- Se desactivan objetos de la BD:
-----
-- 1º Desactivar Claves foráneas
ALTER TABLE S73APRUEBA01 DISABLE CONSTRAINT FKCOD;
-- 2º Desactivar Primary Keys
ALTER TABLE S73APRUEBA02 DISABLE CONSTRAINT S73APRUEBA02_PK;
--Desactivar Triggers
--ALTER TRIGGER NOMBRE_TRIGGER DISABLE;
--Desactivar TODOS los Trigger de una tabla
--ALTER TABLE NOMBRE_TABLA DISABLE ALL TRIGGERS;
--CONNECT MHSYS/&MHSYS_PASS@&SID_BD;
--CONNECT S73A/S73A@EDE18;
SET FEEDBACK ON;
SET HEADING ON;
SET LINESIZE 120;
SET PAGESIZE 240;
SET TRIMSPOOL ON;
SET SERVEROUTPUT ON;
-----
-- Se informan las tres variabes para cada encriptación y se ejecuta:
-----
DEFINE ESQUEMA = 'S73A' (CHAR);
DEFINE TABLA = 'S73APRUEBA01' (CHAR);
DEFINE COLUMNA = 'NUMBER01' (CHAR);
EXECUTE PKG_ENCRYPTSYS.P_MAIN('&ESQUEMA','&TABLA','&COLUMNA');
-----
-- Se repite la encriptación por tabla/columna tantas veces como sea necesario:
-----
DEFINE TABLA = 'S73APRUEBA01' (CHAR);
DEFINE COLUMNA = 'LONG01' (CHAR);
EXECUTE PKG_ENCRYPTSYS.P_MAIN('&ESQUEMA','&TABLA','&COLUMNA');
-----
-- Se repite la encriptación por tabla/columna tantas veces como sea necesario:
-----
DEFINE TABLA = 'S73APRUEBA01' (CHAR);
DEFINE COLUMNA = 'DATE01' (CHAR);
EXECUTE PKG_ENCRYPTSYS.P_MAIN('&ESQUEMA','&TABLA','&COLUMNA');
-----
-- Se activan objetos de la BD:
-----
```

```
-- 1º Activar Primary Keys
ALTER TABLE S73APRUEBA02 ENABLE CONSTRAINT S73APRUEBA02_PK;
-- 2º Activar Claves foráneas
ALTER TABLE S73APRUEBA01 ENABLE CONSTRAINT FKCOD;
--Desactivar Trigger
--ALTER TRIGGER NOMBRE_TRIGGER ENABLE;
--Desactivar TODOS los Trigger de una tabla
--ALTER TABLE NOMBRE_TABLA ENABLE ALL TRIGGERS;
--SPOOL OFF;

EXIT;
```

4 Utilidad práctica

La utilización de los scripts de ofuscación tendrá como objetivo la carga en entornos de Pruebas y Desarrollo de datos reales de las aplicaciones, para las consiguientes pruebas de carga. La ofuscación permitirá “ocultar” datos sensibles, según hayan sido definidos por la aplicación de la LOPD.

De esta forma, se podrían utilizar por ejemplo datos de pacientes de Sanidad en Desarrollo, ocultando datos como su nombre y apellidos, sin ir más lejos. Para ello, bastaría con ofuscar, mediante el procedimiento anterior dichos campos en BD de desarrollo una vez cargados los datos de producción.

Por supuesto, hay que tener en cuenta que la ofuscación crea combinaciones aleatorias de fechas, números y caracteres, lo que puede provocar que un campo que fuese un e-mail, por ejemplo, ya no tenga el formato de un e-mail válido tras la ofuscación.

Hay que estudiar bien cuáles son por tanto los campos a ofuscar y cómo la ofuscación puede afectar a las validaciones existentes en la aplicación.