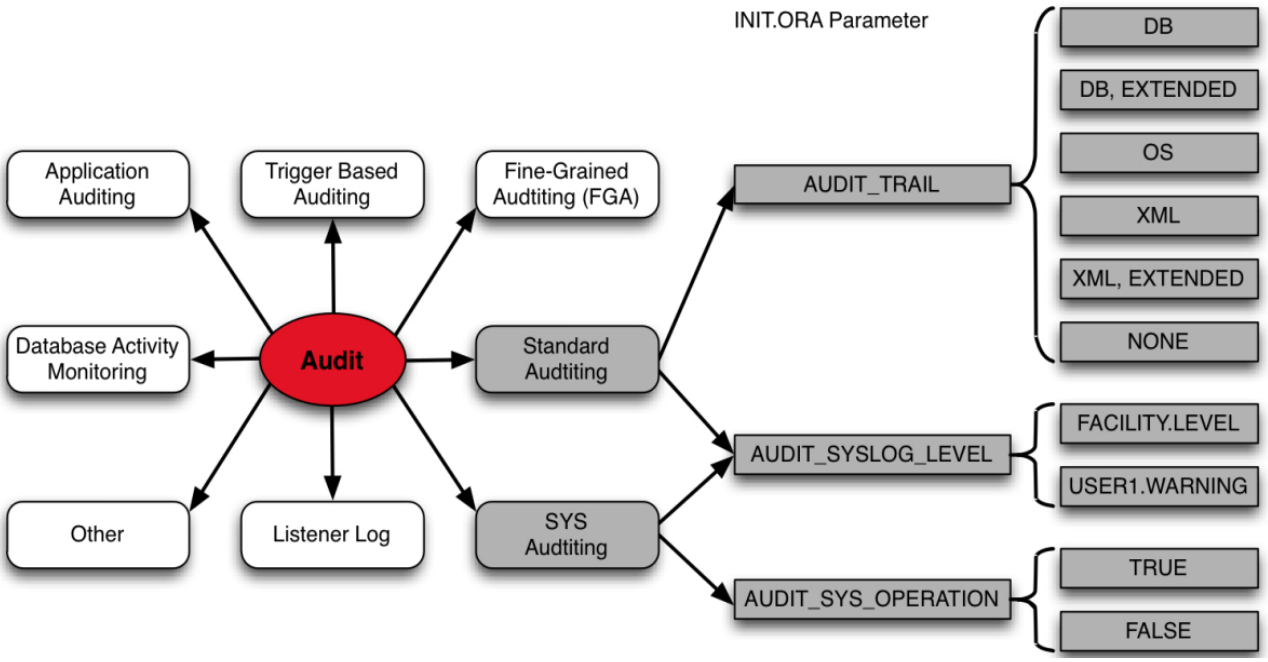
DATABASE VAULT

مقدمه

همانطور که می دانید کاربر sys می تواند همه اطلاعات کاربران دیگر را مشاهده و نیز تغییراتی را بر روی انها اعمال کند این مسئله به لحاظ امنیتی می تواند در برخی شرایط چالش افرین باشد و به همین جهت ممکن است برخی از سازمانها برای حفظ اطلاعات محرمانه ای که دارند، بخواهند بر sys هم نظارت داشته باشند برای کنترل بر این موضوع، می توان از پارامتری به نام audit_sys_operations استفاده کرد و با کمک این پارامتر، بر همه انچه که sys انجام می دهد، نظارت و بازرسی داشت ولی چه فایده که کاربر sys هم می تواند خروجی را ببیند و هم می تواند ان را غیرفعال کند و صدالبته که خروجی این لاگها هم نسبتا بی کیفیت هستند و معمولا در سطح سیستم عامل قابل مشاهده می باشند. برای مدیریت کاربران غیر sys، پارامترها و روشهای دیگری هم وجود دارند که در این نوشتار قصد پرداختن به ان را نداریم و تنها به ارائه شکل زیر بسنده می کنیم:



شایسته تاکید است که پارامتر مذکور تنها امکان نظارت بر کارهای sys را به صورت کاملا ناقض ممکن می سازد و هیچ محدودیتی را برای کاربر sys اعمال نمی کند و حتی کاربر sys می تواند با یکبار راه اندازی مجدد بانک، این قابلیت را غیرفعال کند. مشابه این دغدغه، دغدغه دیگری هم وجود دارد که ان هم کنترل و محدود کردن ادمین بانک اطلاعاتی و ادمین OS در سطح سیستم عامل می باشد چون در صورتی که برای ذخیره

اطلاعات از filesystem استفاده شده باشد، ادمین سیستم عامل و یا ادمین بانک اطلاعاتی(در صورت محدود شدن در داخل بانک) ممکن است با کمک ابزارهایی، از اطلاعات فایلها با خبر شوند. اوراکل برای جلوگیری از این مسئله، ویژگی ای به نام TDE را مطرح کرد و همچنین برای رفع چالش اول که محدود سازی کاربر sys در سطح بانک اطلاعاتی بود، database vault را ارائه کرده است.

در این نوشتار قصد بر ان است تا در مورد database vault به صورت گذرا مطالبی را عنوان کنیم و در نوشتاری دیگر، به فضل الهی به TDE خواهیم پرداخت.

مهمترین وظیفه ای که database vault بر عهده دارد: محدود کردن دسترسی کاربران به شکلی که تنها بتوانند کارهایی که بر عهده انها است را انجام دهند و هیچ دسترسی اضافه ای نداشته باشند و با کمک این ویژگی می توان مشخص کرد که "چه کسی"، "چه وقت" و به "چه شکلی" به داده دسترسی داشته باشد و همه انچه که در ابتدا گفته شد، یعنی محدود کردن دسترسی کاربر sys و کنترل و نظارت بر عملکرد ان، به راحتی از طریق database vault ممکن خواهد بود.

شاید محدود شدن ورود با sysdba و مجبور کردن dba برای استفاده از sysoper به جای sysdba، مثال خوبی در این زمینه باشد که در نسخه های قبلی بلافاصله بعد از نصب DV(database vault) اتفاق می افتاد زیرا که dba می تواند عمده کارهای مدیریتی را از این طریق انجام دهد و در عین حال، می توان ان را محدود کرد تا به همه اطلاعات دسترسی نداشته باشد البته این اتفاق سبب مشکلاتی هم می شد که فرصت طرح ان در این نوشتار وجود ندارد.

در ادامه نحوه نصب database vault اورده شده و نیز در مورد مولفه های ان، مطالبی بیان خواهد شد.

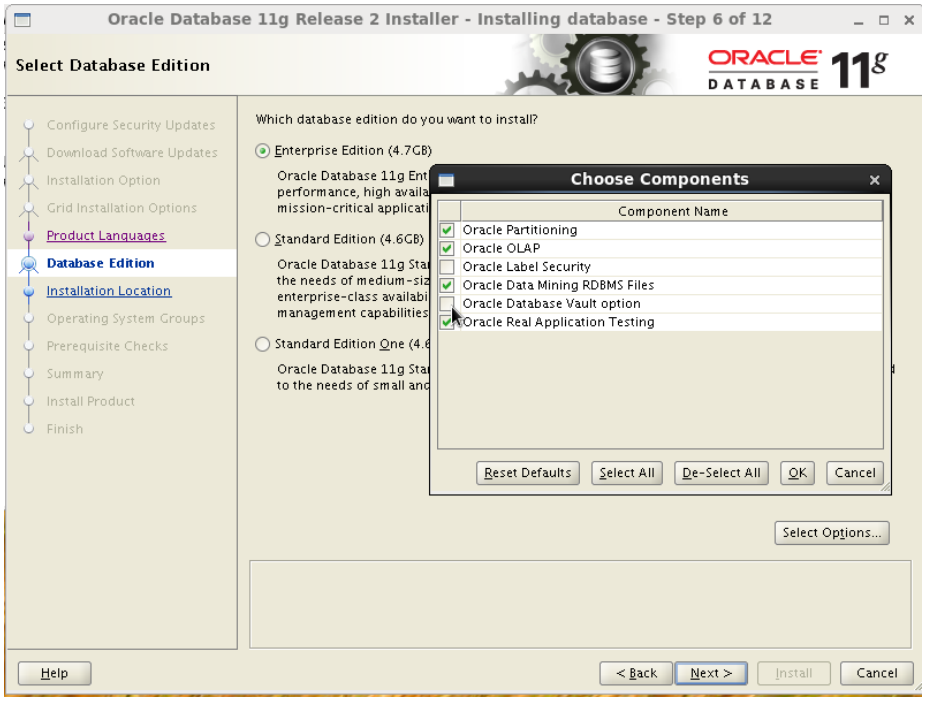
نصب database vault

database vault از نسخه 9iR2 عرضه شد که برای استفاده از آن می بایست نرم افزاری از سایت اوراکل دانلود می شد و با نصب ان نرم افزار از این ویژگی استفاده می شد ولی از اوراکل 11g این ویژگی به همراه نرم افزار اوراکل ارائه می شود و قابل نصب می باشد.

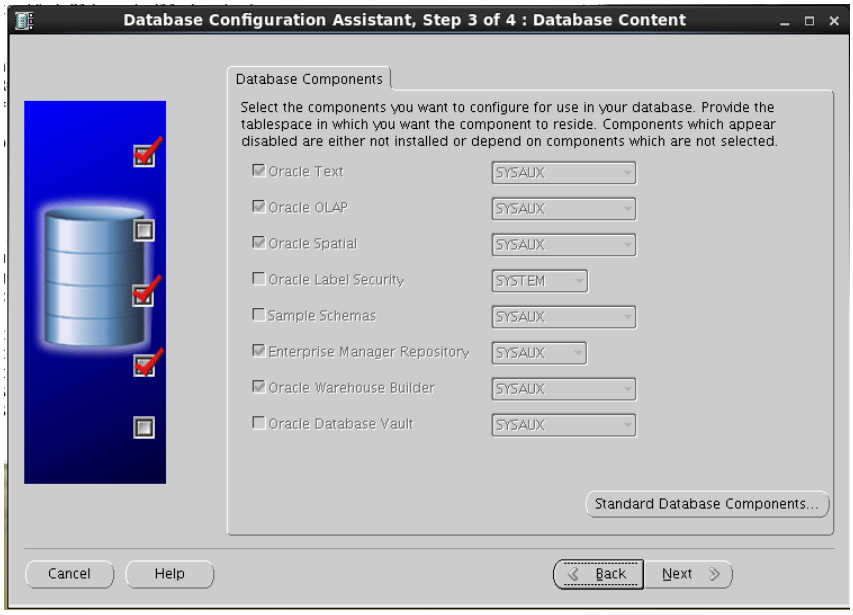
برای نصب database vault، باید در دو مرحله اقدام کرد که یکی در حین نصب نرم افزار اوراکل و دیگری در حین ساخت بانک اطلاعاتی می باشد.

-حین نصب نرم افزار

برای داشتن این ویژگی، باید در مرحله 6 از نصب نرم افزار اوراکل(نسخه 11.2.0.4)، در قسمت select options، database Vault را هم انتخاب کرد که همیشه به همراه انتخاب آن، oracle label security هم انتخاب خواهد شد.



در صورتی که بخواهیم برای بانکی که قبلا نصب شده است، این ویژگی را فعال کنیم، یا باید نصب مجدد انجام داد و یا می بایست از اسکریپت استفاده شود قبل از اجرای این اسکریپت، قابلیت نصب این ویژگی وجود ندارد:



برای فعالسازی این ویژگی، از دستور chopt استفاده می شود که از طریق این دستور، نه تنها این کامپوننت، بلکه می توان کامپوننتهای دیگری را هم فعال و یا غیرفعال کرد.

```
[oracle@hkm4 ~]$ chopt

chopt <enable|disable> <option>

options:
        dm = Oracle Data Mining RDBMS Files
        dv = Oracle Database Vault option
      lbac = Oracle Label Security
      olap = Oracle OLAP
partitioning = Oracle Partitioning
       rat = Oracle Real Application Testing
e.g. chopt enable rat
```

پس برای فعالسازی database vault باید مراحل زیر را طی کرد:

1.ابتدا باید تمامی سرویسها متوقف شوند:

```
emctl stop dbconsole
lsnrctl stop
sqlplus "/as sysdba"
SQL> shut abort
```

2.ویژگی Oracle Label Security فعال شود:

```
[oracle@hkm4 ~]$ chopt enable lbac
Writing to /u01/oracle/11g/install/enable_lbac.log...
/usr/bin/make -f /u01/oracle/11g/rdbms/lib/ins_rdbms.mk lbac_on ORACLE_HOME=/u01/oracle/11g
/usr/bin/make -f /u01/oracle/11g/rdbms/lib/ins_rdbms.mk ioracle ORACLE_HOME=/u01/oracle/11g
```

همانطور که می بینید، خروجی دستور قبلی، شامل دو اسکریپت می باشد که باید هر دو ان را اجرا نمود:

```
[oracle@hkm4 ~]$ /usr/bin/make -f /u01/oracle/11g/rdbms/lib/ins_rdbms.mk lbac_on
ORACLE_HOME=/u01/oracle/11g
```

[oracle@hkm4 ~]$ /usr/bin/make -f /u01/oracle/11g/rdbms/lib/ins_rdbms.mk ioracle ORACLE_HOME=/u01/oracle/11g

3.ویژگی database vault را هم فعال می کنیم:

[oracle@hkm4 ~]$ chopt enable dv

Writing to /u01/oracle/11g/install/enable_dv.log...

/usr/bin/make -f /u01/oracle/11g/rdbms/lib/ins_rdbms.mk dv_on ORACLE_HOME=/u01/oracle/11g

/usr/bin/make -f /u01/oracle/11g/rdbms/lib/ins_rdbms.mk ioracle ORACLE_HOME=/u01/oracle/11g

برای این دستور هم اسکریپتها را اجرا می کنیم:

[oracle@hkm4 ~]$ /usr/bin/make -f /u01/oracle/11g/rdbms/lib/ins_rdbms.mk dv_on ORACLE_HOME=/u01/oracle/11g

[**oracle@hkm4 ~]$** /usr/bin/ar cr /u01/oracle/11g/rdbms/lib/libknlopt.a /u01/oracle/11g/rdbms/lib/kzvidv.o

[oracle@hkm4 ~]$ /usr/bin/make -f /u01/oracle/11g/rdbms/lib/ins_rdbms.mk ioracle ORACLE_HOME=/u01/oracle/11g

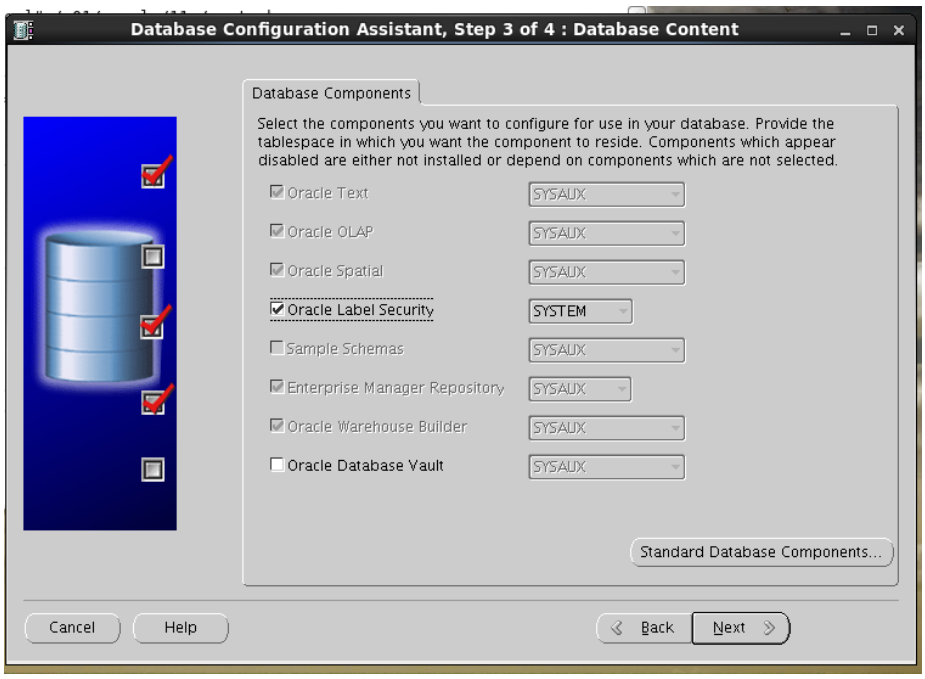
4.در نهایت بانک را راه اندازی مجدد خواهیم کرد:

lsnrctl start

sqlplus "/as sysdba"

startup

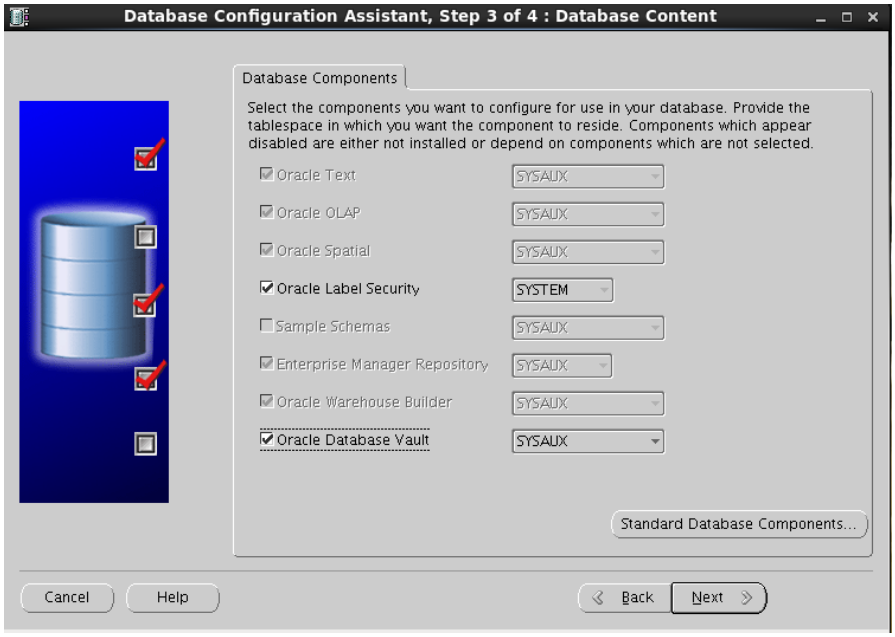
با اجرا مجدد dbca خواهیم دید که قابلیت افزودن دو کامپوننت ممکن شده است:



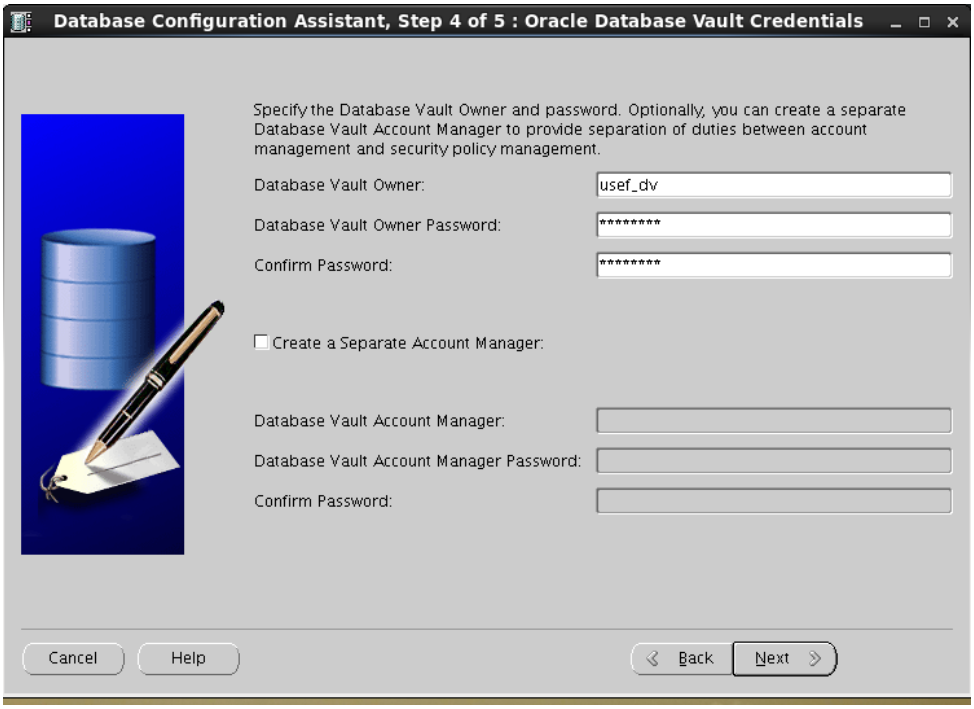
-حین dbca

در صورتی که قابلیت استفاده از این ویژگی در سطح نرم افزار ایجاد شده باشد(با یکی از دو روش مذکور) می توان از طریق dbca و یا از طریق اسکریپت، این کامپوننت را به بانک اضافه کرد که مراحل انجام ان از طریق dbca به صورت زیر می باشد:

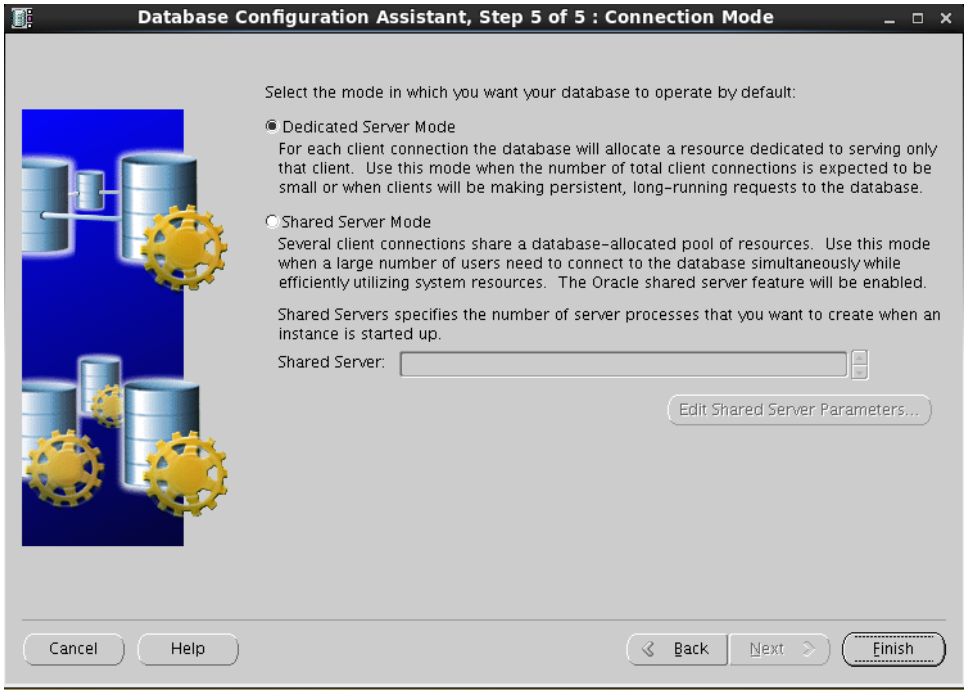
1.گزینه های مربوط به OLS و VAULT را انتخاب می کنیم.



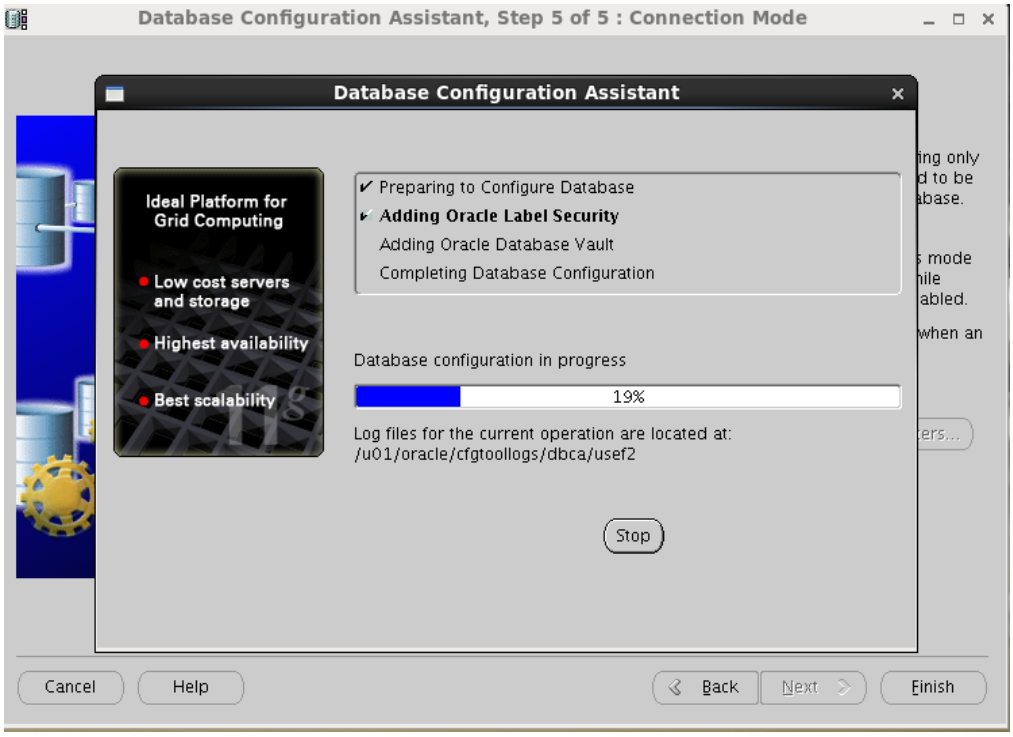
2.کاربری را به عنوان database vault owner معرفی می کنیم که از طریق ان می توان به کنسول مدیریتی DV متصل شد و همچنین در صورتی که می خواهیم قسمتی از مدیریت vault به کاربر دیگری تحت عنوان account manager داده شود، باید در قسمت زیرین، کاربر مورد نظر را معرفی کنیم.



3. از بین دو مود dedicated و shared یکی را انتخاب می کنیم.



4.با انتخاب finish، نصب صورت خواهد پذیرفت.



5.با پرس و جوی زیر خواهیم یافت که این دو ویژگی به بانک اضافه شده اند:

```
SQL> select PARAMETER,VALUE from v$option where PARAMETER in('Oracle Database Vault','Oracle Label Security');

PARAMETER                      VALUE
------------------------------ ----------
Oracle Label Security          TRUE
Oracle Database Vault          TRUE
```

حال با ورود به کاربر sys، قصد داریم تا مجوز dba را به کاربر sysman اهدا کنیم:

```
sqlplus "/as sysdba"

SQL> show user

USER is "SYS"

SQL>  grant dba to sysman;

ORA-47410: Realm violation for GRANT on UNLIMITED TABLESPACE
```

همانطور که می بینید، کاربر sys هم نمی تواند مجوز dba را به کسی بدهد.

به همراه نصب database vault، دو کاربر با نامهای DVSYS وDVF ایجاد می شوند که به صورت پیش فرض هر دو در حالت lock قرار دارند و معمولا به این دو کاربر نیازی نخواهیم داشت همچنین اگر account manager به صورت جداگانه تعریف شود، کاربر دیگری هم ساخته خواهد شد.

به همراه نصب database vault، roleهای جدیدی هم به بانک اضافه می شوند که اسامی انها به این شکل می باشد:

select role from dba_roles l where l.role like '%DV%';

DV_ACCTMGR

DV_ADMIN

DV_AUDIT_CLEANUP

DV_GOLDENGATE_ADMIN

DV_GOLDENGATE_REDO_ACCESS

DV_MONITOR

DV_OWNER

DV_PATCH_ADMIN

DV_PUBLIC

DV_REALM_OWNER

DV_REALM_RESOURCE

DV_SECANALYST

DV_STREAMS_ADMIN

DV_XSTREAM_ADMIN

همانطور که می بیند، این roleها در سه سطح owner، administration و monitoring قابل تقسیم می باشند.

همچنین با نصب DV، مقادیر پارامترهای زیر هم تغییر خواهد کرد:

Parameter	Default	New value
AUDIT_SYS_OPERATIONS	FALSE	TRUE
OS_AUTHENT_PREFIX	ops$	ops$
OS_ROLES	Not configured	FALSE
REMOTE_LOGIN_PASSWORDFILE	EXCLUSIVE	EXCLUSIVE
REMOT_OS_AUTHENT	FALSE	FALSE
REMOTE_OS_ROLES	FALSE	FALSE
SQL92_SECURITY	FALSE	TRUE

دستورات زیر با نصب DV، توسط کاربر sys و system قابل استفاده نخواهند بود(به دلیل command rule تعیین شده):

Alter user - Alter profile - Create profile - Create user - Drop profile - Drop user

از دیگر تاثیرات نصب DV، گرفتن برخی مجوزها از برخی roleها می باشد که لیست ان در جدول زیر امده است:

Role	Privileges that will be revoked
DBA	BECOME USER
	SELECT ANY TRANSACTION
	CREATE ANY JOB
	CREATE EXTERNAL JOB
	EXECUTE ANY PROGRAM
	EXECUTE ANY CLASS
	MANAGE SCHEDULER
	DEQUEUE ANY QUEUE
	ENQUEUE ANY QUEUE
	MANAGE ANY QUEUE
IMP_FULL_DATABASE	BECOME USER
	MANAGE ANY QUEUE
EXECUTE_CATALOG_ROLE	EXECUTE ON DBMS_LOGMNR
	EXECUTE ON DBMS_LOGMNR_D
	EXECUTE ON DBMS_LOGMNR_LOGREP_DICT
	EXECUTE ON DBMS_LOGMNR_SESSION
	EXECUTE ON DBMS_FILE_TRANSFER
PUBLIC user	EXECUTE ON UTL_FILE
SCHEDULER_ADMIN	CREATE ANY JOB
	CREATE EXTERNAL JOB
	EXECUTE ANY PROGRAM
	EXECUTE ANY CLASS
	MANAGE SCHEDULER

البته توصیه می شود تا قبل از نصب DV، از جداول مرتبط با مجوزها، کپی تهیه کنیم:

create table copy_dba_tab_privs as Select * from ***dba_tab_privs***;

create table copy_dba_sys_privs as Select * from ***dba_sys_privs***;

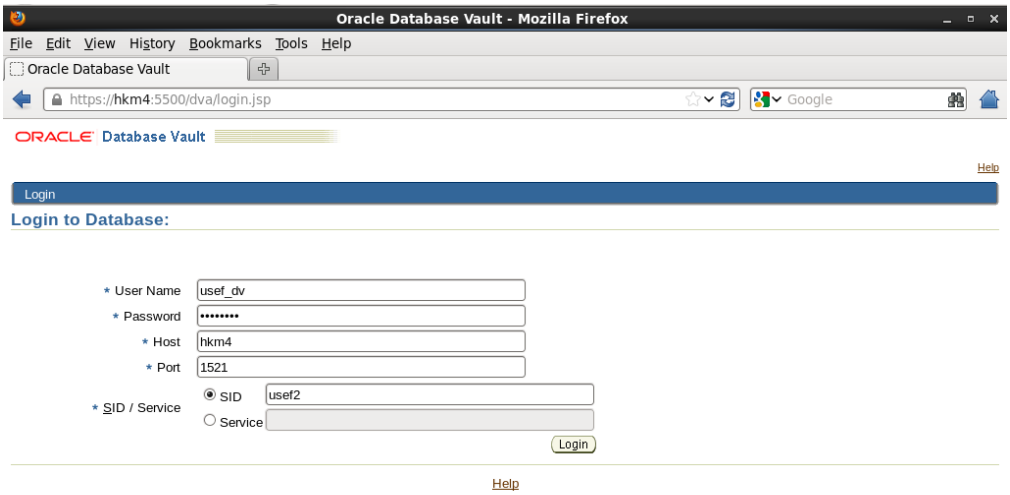
create table copy_dba_role_privs as Select * from ***dba_role_privs***;

شایان ذکر است که در مواردی از قبیل SQL Injection و یا دسترسی غیرمجاز به دیتافایلها در لایه OS، استفاده و وجود DV موثر نخواهد بود و همچنین اگر کسی به OS دسترسی داشته باشد، می تواند این ویژگی را غیرفعال کند.

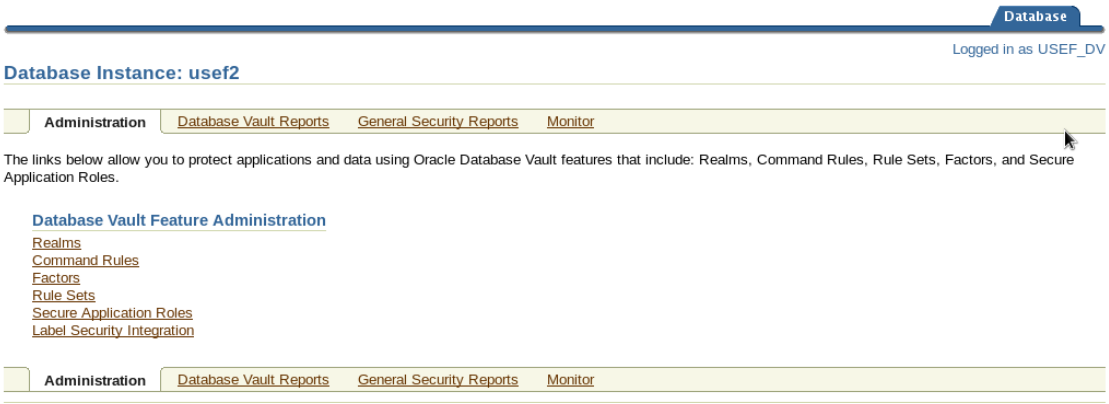
برای مدیریت vault، هم می توان از محیط گرافیکی استفاده کرد و هم می توان از پکیجهایی از قبیل DBMS_MACADM بهره گرفت که استفاده از محیط گرافیکی، به شدت کار را اسان خواهد کرد.

لینک ورود به محیط گرافیکی مربوط به database vault، همانند EM می باشد با این تفاوت که به جای عبارت em باید از dva استفاده کرد:

https://hkm4:5500/dva



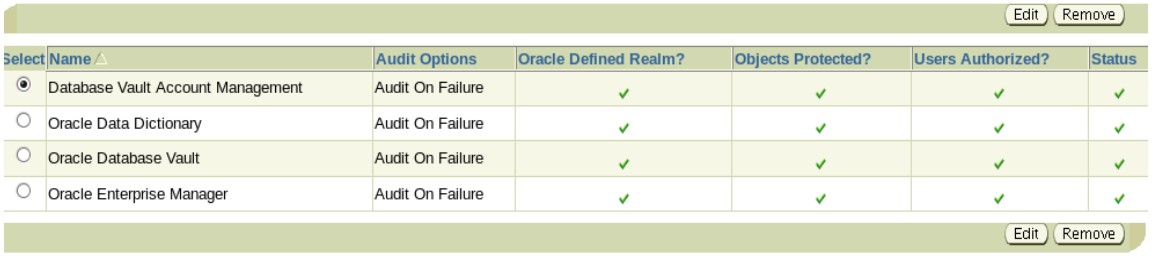
بعد از ورود مشخصات کاربر و سرور، وارد صفحه اصلی مربوط به database vault خواهیم شد:



همانطور که در این صفحه می بینید، database vault شامل چند مولفه می باشد که در ابتدا در مورد این مولفه ها مطالبی را عرضه خواهیم داشت و سپس سناریوهایی را در رابطه با هرکدام، پیاده سازی خواهیم کرد.

Realm: در صورتی که قصد داشته باشیم مجموعه ای از چند اشیا را از دسترس برخی یا همه افراد دور نگه داریم و به اصلاح انها را secure یا امنشان کنیم، می توانیم از Realm استفاده کنیم در این صورت این امکان هم وجود دارد تا تنها با چند کلیک و یا چند دستور، دسترسی به همه این اشیا تعریف شده را برای چند کاربر خاص مجاز کنیم و یا اینکه به صورت کلی، در کسری از ثانیه، همه این محدودیتها را غیرفعال کنیم.

با نصب database vault، چند realm به صورت پیش فرض ایجاد می شوند:



از مهمترین این realmها، Oracle Data Dictionary می باشد که سبب می شود تا محدودیتهای بسیاری برای کاربران ایجاد شود. برای اینکه کاربری را از این محدودیتها مستثنی کنیم، باید نام ان کاربر را در قسمت users authorized اضافه کنیم و همچنین این امکان وجود دارد تا به لیست اشیا امن شده، اشیای دیگری را هم اضافه کنیم که برای انجام این کار، باید از قسمت object protected استفاده کرد.

برای مشاهده لیست realmها می توان از دستور زیر هم استفاده کرد:

sqlplus usef_dv/usef_123

select name , enabled from dvsys.**dba_dv_realm**;

NAME	**ENABLED**
Database Vault Account Management	Y
Oracle Data Dictionary	Y
Oracle Database Vault	Y
Oracle Enterprise Manager	Y

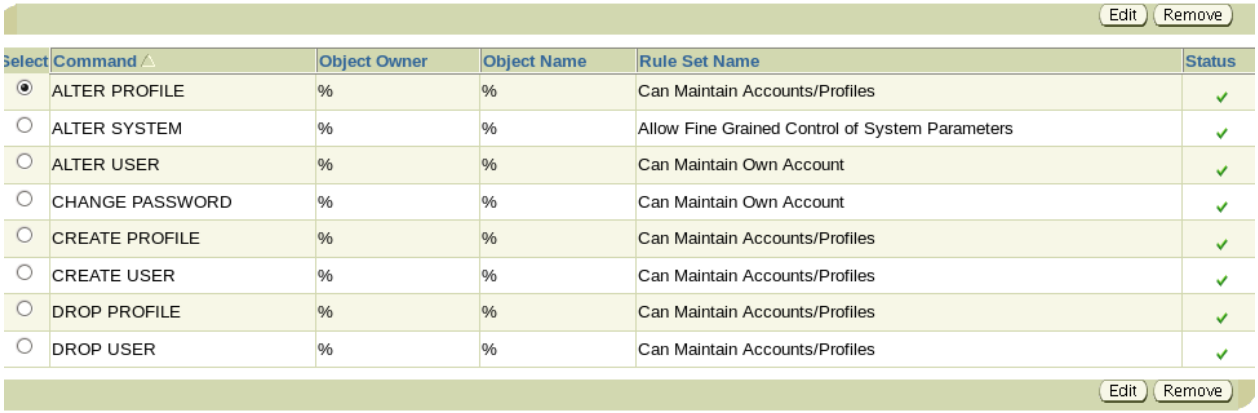
همچنین برای مشاهده لیست قوانین موجود در هر کدام از realmها، می توان از دستور زیر بهره گرفت:

select * from dvsys.dba_dv_realm_object where realm_name='Oracle Data Dictionary';

REALM_NAME	**OWNER**	**OBJECT_NAME**	**OBJECT_TYPE**

Oracle Data Dictionary % DBA ROLE

Command Rule: در صورتی که بخواهیم کاربر و یا کاربران خاصی را از زدن دستورات مشخصی منع و یا مجاز کنیم، می توانیم از Command Rule استفاده کنیم دستورات می توانند DDLای ،DMLای و ... باشند. به صورت پیش فرض، command ruleهای زیر در سیستم وجود دارند:



Edit Remove

Select	Command	Object Owner	Object Name	Rule Set Name	Status
◉	ALTER PROFILE	%	%	Can Maintain Accounts/Profiles	✓
○	ALTER SYSTEM	%	%	Allow Fine Grained Control of System Parameters	✓
○	ALTER USER	%	%	Can Maintain Own Account	✓
○	CHANGE PASSWORD	%	%	Can Maintain Own Account	✓
○	CREATE PROFILE	%	%	Can Maintain Accounts/Profiles	✓
○	CREATE USER	%	%	Can Maintain Accounts/Profiles	✓
○	DROP PROFILE	%	%	Can Maintain Accounts/Profiles	✓
○	DROP USER	%	%	Can Maintain Accounts/Profiles	✓

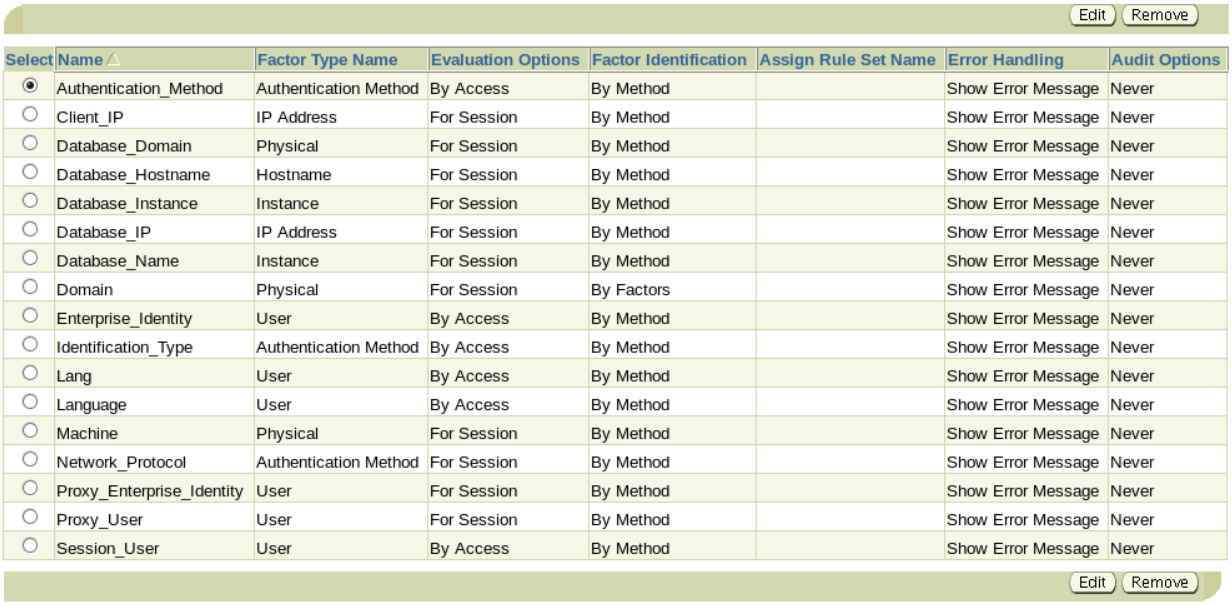
Edit Remove

برای مشاهده لیست command ruleها، از ویوی dba_dv_command_rule استفاده می شود.

```
select * from dvsys.dba_dv_command_rule;
```

COMMAND	RULE_SET_NAME	OBJECT_OWNER	OBJECT_NAME	ENABLED
SELECT	ip	USEF %		Y
DROP USER	Can Maintain Accounts/Profiles	%	%	Y

Factor: با کمک فاکتور می توان محدودیتهای دیگری را از طریق اطلاعات و خصوصیات sessionها لحاظ کنیم از قبیل انکه، "فردی با ip مشخص، تنها حق دسترسی به جدول مشخصی را داشته باشد"، "فردی با ماشین مشخص، نتواند از دستور x استفاده کند" و یا حتی از طریق ساعت و دیگر خصوصیات sessionها محدودیت هایی را ایجاد کنیم. برای انجام این کار، می توان از تابع sys_context استفاده کرد. لیست فاکتورهای پیش فرض در این قسمت دیده می شود:



Edit Remove

Select	Name	Factor Type Name	Evaluation Options	Factor Identification	Assign Rule Set Name	Error Handling	Audit Options
◉	Authentication_Method	Authentication Method	By Access	By Method		Show Error Message	Never
○	Client_IP	IP Address	For Session	By Method		Show Error Message	Never
○	Database_Domain	Physical	For Session	By Method		Show Error Message	Never
○	Database_Hostname	Hostname	For Session	By Method		Show Error Message	Never
○	Database_Instance	Instance	For Session	By Method		Show Error Message	Never
○	Database_IP	IP Address	For Session	By Method		Show Error Message	Never
○	Database_Name	Instance	For Session	By Method		Show Error Message	Never
○	Domain	Physical	For Session	By Factors		Show Error Message	Never
○	Enterprise_Identity	User	By Access	By Method		Show Error Message	Never
○	Identification_Type	Authentication Method	By Access	By Method		Show Error Message	Never
○	Lang	User	By Access	By Method		Show Error Message	Never
○	Language	User	By Access	By Method		Show Error Message	Never
○	Machine	Physical	For Session	By Method		Show Error Message	Never
○	Network_Protocol	Authentication Method	For Session	By Method		Show Error Message	Never
○	Proxy_Enterprise_Identity	User	For Session	By Method		Show Error Message	Never
○	Proxy_User	User	For Session	By Method		Show Error Message	Never
○	Session_User	User	By Access	By Method		Show Error Message	Never

Edit Remove

برای مشاهده لیست فاکتورها، می توان از ویوی DBA_DV_FACTOR استفاده کرد:

SELECT * FROM DVSYS.DBA_DV_FACTOR;

```
SELECT DVF.F$ip_factor FROM DUAL;
```

Rule Set: مجموعه ای از یک یا چند rule می باشد و در زمان استفاده از command rule و realm باید از طریق rule set مشخص کرد که دامنه محدودیتها مربوط به چه زمینه و چه محدوده ای باشد. همچنین می توان از یک rule set استفاده مکرر داشت.

برای مشاهده لیست Rule Setها، از دستور زیر استفاده می شود:

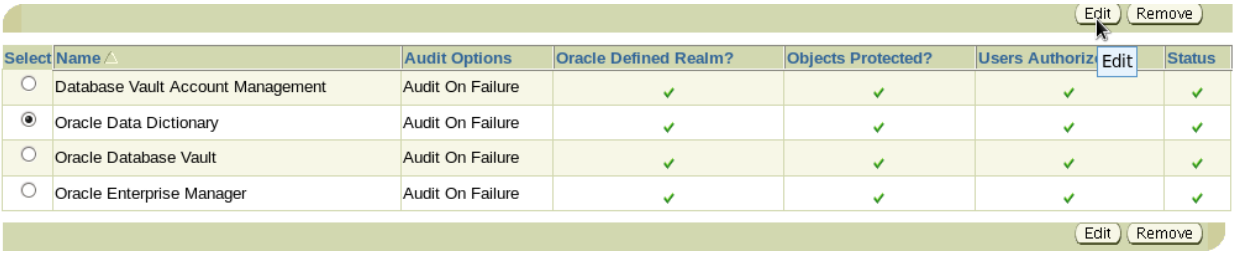
select rule_set_name, enabled from dvsys.dba_dv_rule_set;

RULE_SET_NAME	ENABLED
ip	Y
Enabled	Y
Disabled	Y
Can Maintain Accounts/Profiles	Y

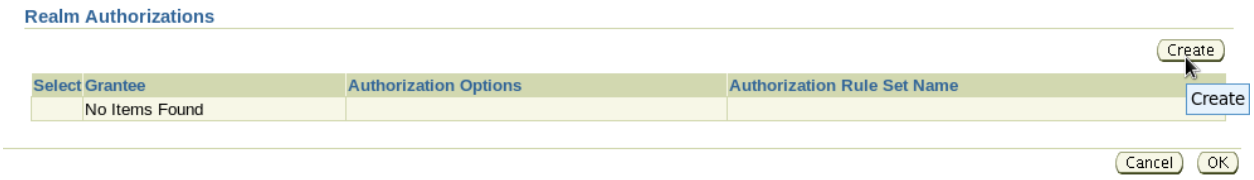
در ادامه از هر یک از این مولفه ها با طرح سناریویی استفاده خواهیم کرد.

مثال اول(realm Authorization): کاربر sys از oracle data dictionary realm مستثنی شود.

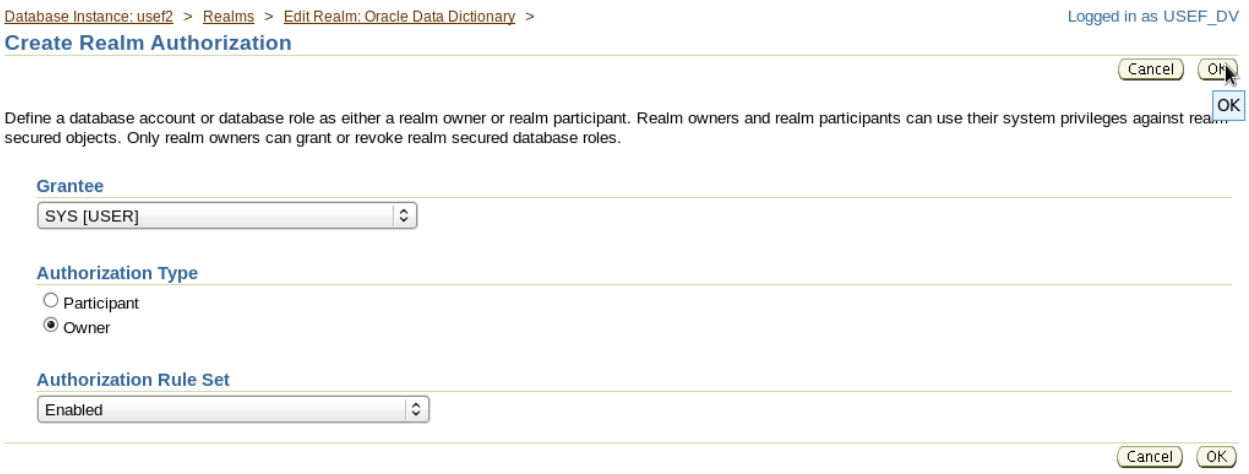
ابتدا وارد قسمت realm شده و oracle data dictionary را ویرایش می کنیم:



در قسمت realm Authorizations بر روی create کلیک می کنیم:



در این قسمت، grantee را به sys تنظیم کرده و برای اینکه هیچ گونه محدودیتی در این زمینه برای کاربر sys اعمال نشود، نوع Authorizations را به owner و rule set را هم به enable تنظیم می کنیم تا این Authorizations برای کاربر sys فعال باشد.

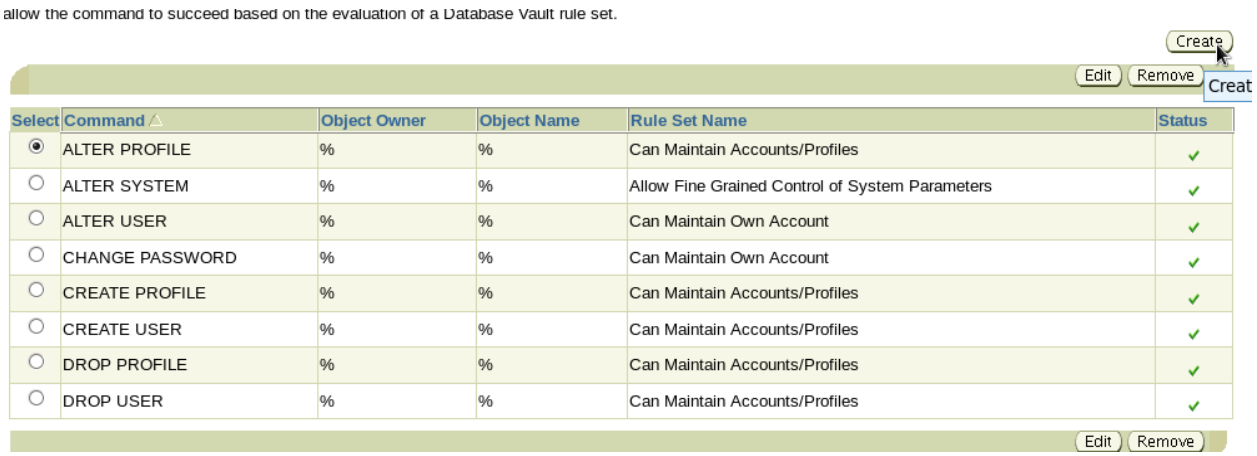


با انجام این مراحل، خواهیم دید که کاربر sys قادر به انجام کارهایی است که بعد از نصب vault محدود شده بود(البته با این کار، قسمتی از تحریم ها برداشته شده است نه همه تحریم ها!!!).

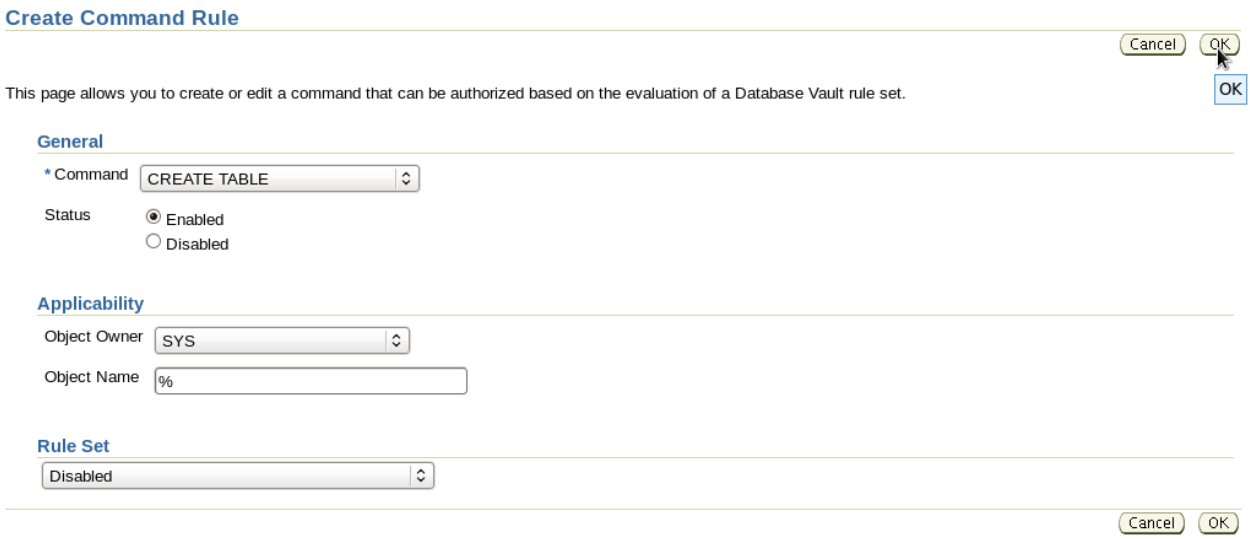
```
SQL> grant dba to sysman;

Grant succeeded.
```

مثال دوم(Command Rule): امکان ساخت جدولی که مالک ان کاربر sys باشد، غیر فعال شود.

این کار باید از طریق command rule انجام شود به همین جهت، وارد قسمت command rule شده و rule جدیدی را ایجاد می کنیم:



سپس در قسمت create command rule، مجوز ساخت جدولی که که مالک ان sys باشد را غیرفعال خواهیم کرد:



در قسمت object_name، % به معنی همه کارکترها می باشد و همانند * در دستور sql، نقش wildcard را ایفا می کند همچنین usef% بیانگر همه اشیایی هست که با usef شروع می شوند. با بازگشت به قسمت command rule، rule جدیدی دیده خواهد شد:

حال با کاربر sys قصد ایجاد جدول tbl را داریم که به خطای زیر برمی خوریم:

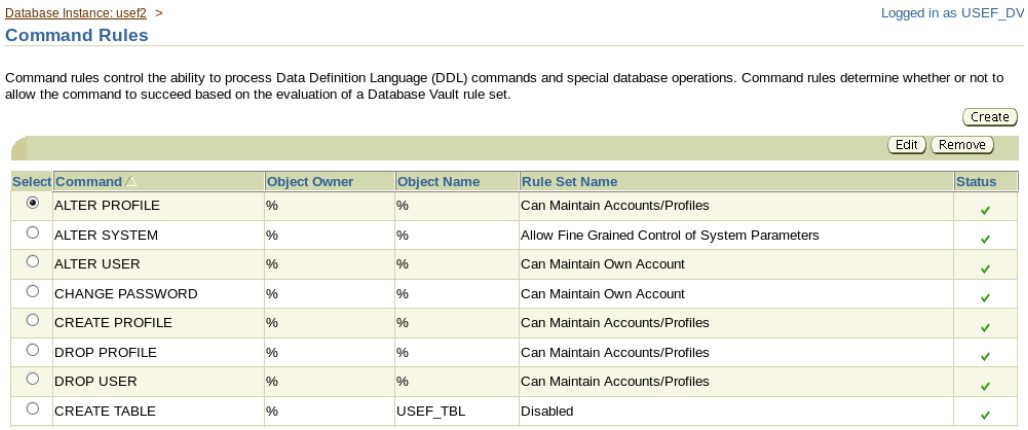
```
SQL>  create table sys.tbl(id number);
```

ORA-47400: **Command Rule violation** for CREATE TABLE on SYS.TBL

نکته 1: با این rule، کاربر sys می تواند برای بقیه کاربران، جدول ایجاد کند و محدودیتی از این نظر ایجاد نشده است.

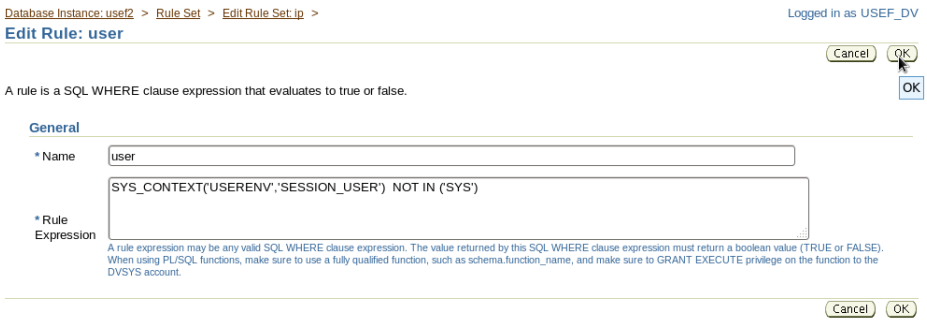
```
SQL>  create table app1.u33(a number);

Table created.
```

نکته 2: در صورتی که object_name را در مثال بالا برابر با مقداری قرار دهیم، سبب می شود تا از ساختن جدولی با ان اسم ممانعت شود. برای مثال، در شکل زیر نشان داده شده که هیچ جدولی با اسم usef_tbl نمی توان ساخت.



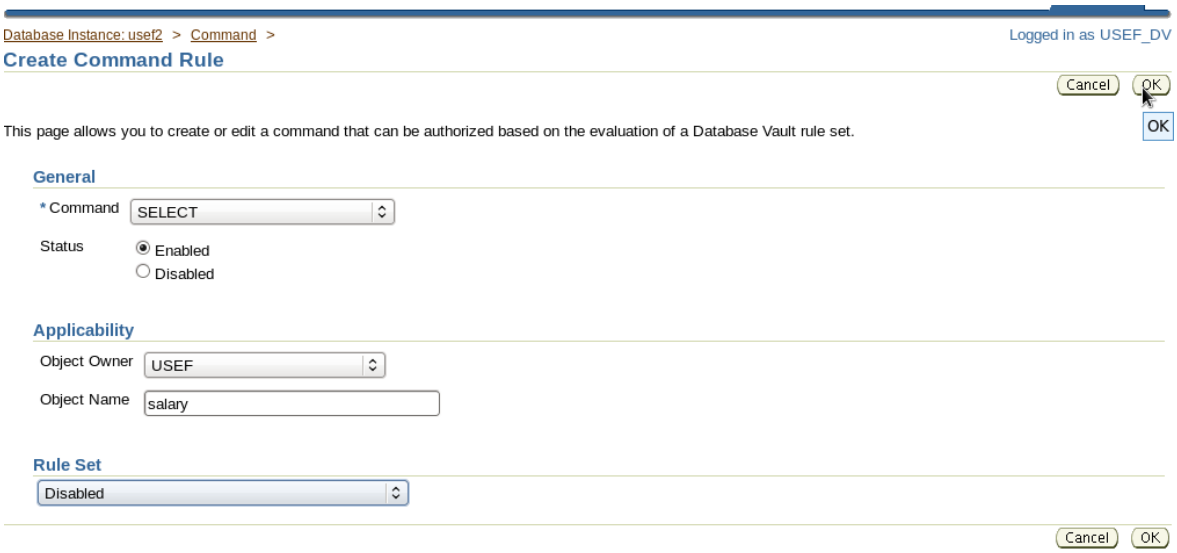
نکته 3: برای جلوگیری از ساخت جدول توسط کاربر sys باید از rule set مناسب استفاده کرد.

```
SYS_CONTEXT('USERENV','SESSION_USER')  NOT IN ('SYS')
```



مثال سوم(Command Rule): امکان خواندن(select) از جدول salary برای همه کاربران حتی sys غیرفعال شود.

این کار با افزودن command rule قابل انجام است.



حال از طریق کاربر sys اقدام به خواندن از جدول usef.salary می کنیم:

SQL> show user

USER is "SYS"

SQL> select * from usef.salary;

ORA-01031: insufficient privileges

مثال چهارم(Realm): دو application در بانک وجود دارند که قرار است هر کدام تنها اطلاعات خودش را ببیند و applicationای حق ندارد اطلاعات دیگری را ببیند.

برای پیاده سازی این مسئله، نیاز است از realm استفاده شود. برای پیاده سازی این سناریو، ابتدا دو کاربر با نامهای App1 و App2 می سازیم و با کمک realm، هر یک را تنها به دیدن اطلاعات خودشان محدود می کنیم.

SQL> create user app1 identified by app1;

SQL> grant dba to app1;

SQL> create user app2 identified by app2;

SQL> grant dba to app2;

SQL> conn **app1**/app1

SQL> create table tbl_app1 as select * from v$tablespace;

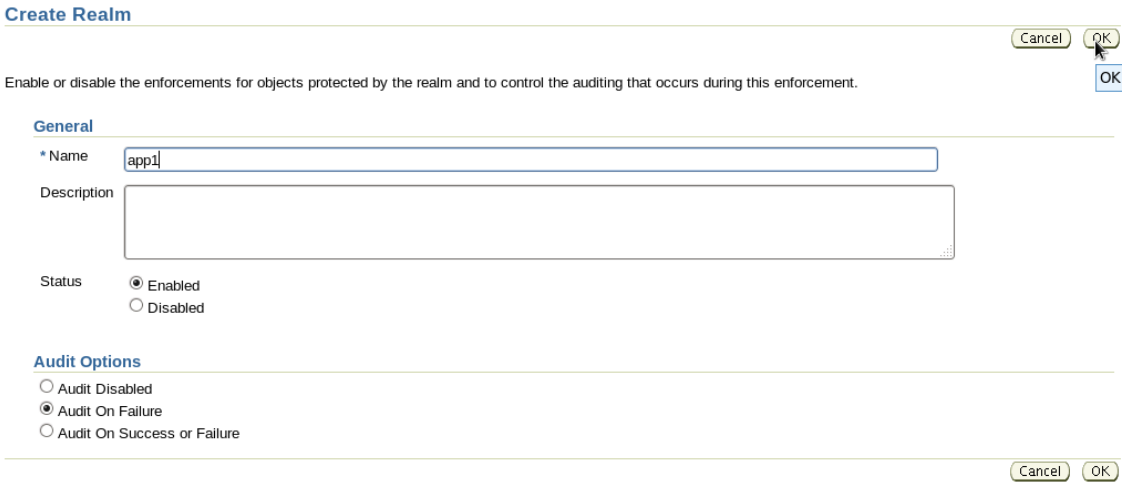
SQL> conn **app2**/app2

SQL> create table tbl_app2 as select * from v$tablespace;

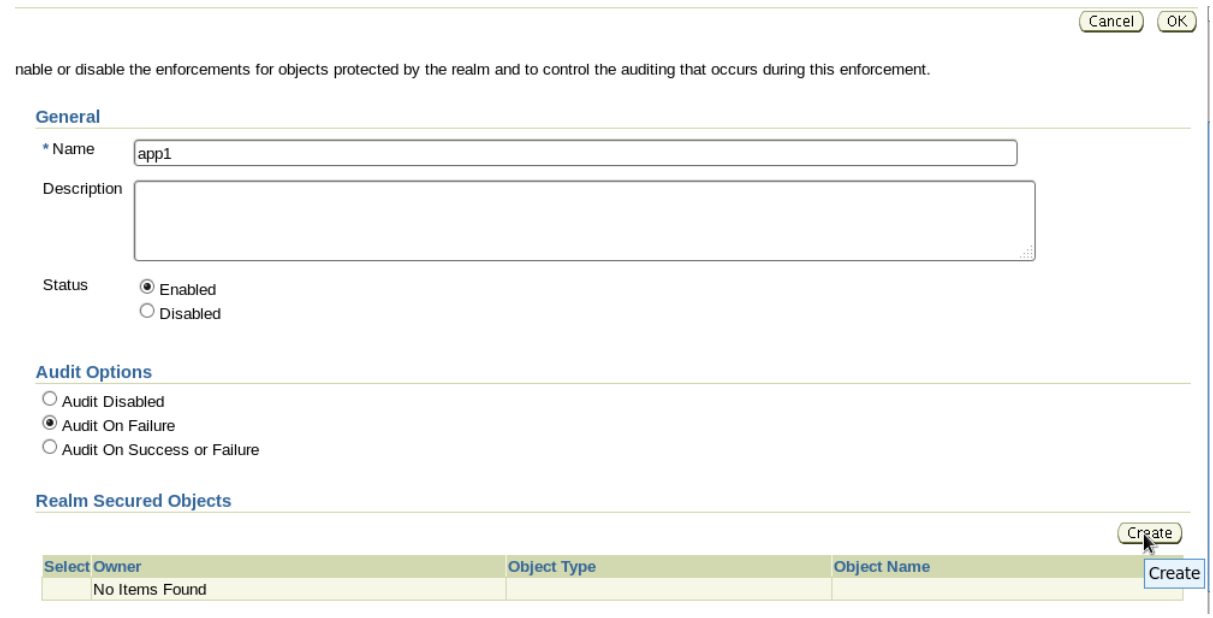
As sysdba:

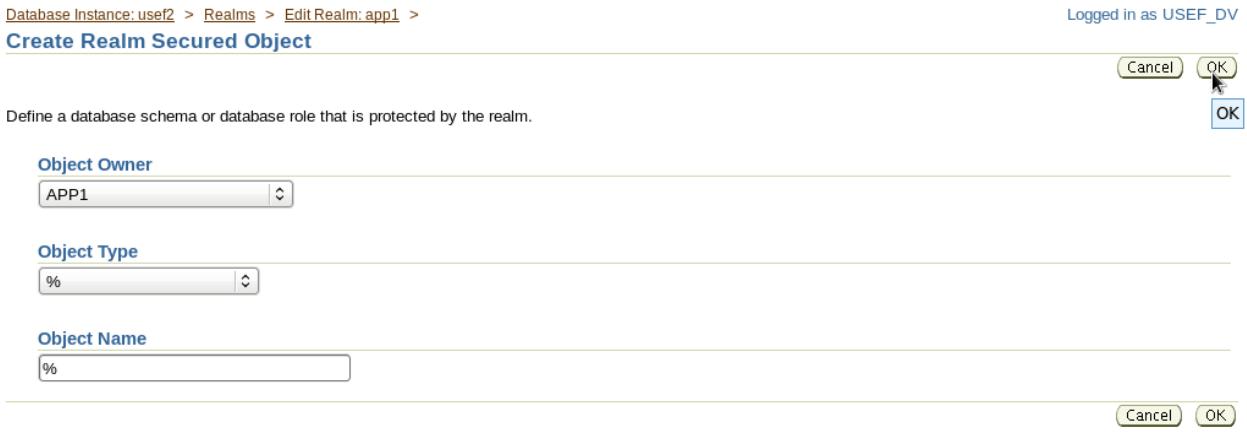
```
SQL> select * from app2.tbl_app2;

     TS# NAME                           INC BIG FLA ENC
---------- ------------------------------ --- --- --- ---
         0 SYSTEM                         YES NO  YES
         1 SYSAUX                         YES NO  YES
         2 UNDOTBS1                       YES NO  YES
         4 USERS                          YES NO  YES
         3 TEMP                           NO  NO  YES
```

حال با کمک realm، سناریوی مورد نظر را انجام می دهیم ابتدا realmای برای app1 ایجاد می کنیم:



با ویرایش این realm، اشیایی که قرار است امن بمانند را مشخص می کنیم:





همانطور که می بینیم، قرار است همه اشیاهای مختص به کاربر App1، از دسترسی همگان خارج شوند.حال تستی را انجام می دهیم که ایا کاربر sys و یا App2 می توانند به اطلاعات App1 دسترسی داشته باشند یا نه؟ App1 چطور؟

As sysdba:

SQL> select * from app1.tbl_app1;

ORA-01031: insufficient privileges

SQL> conn app2/app2

SQL> select * from app1.tbl_app1;

ORA-01031: insufficient privileges

SQL> conn app1/app1

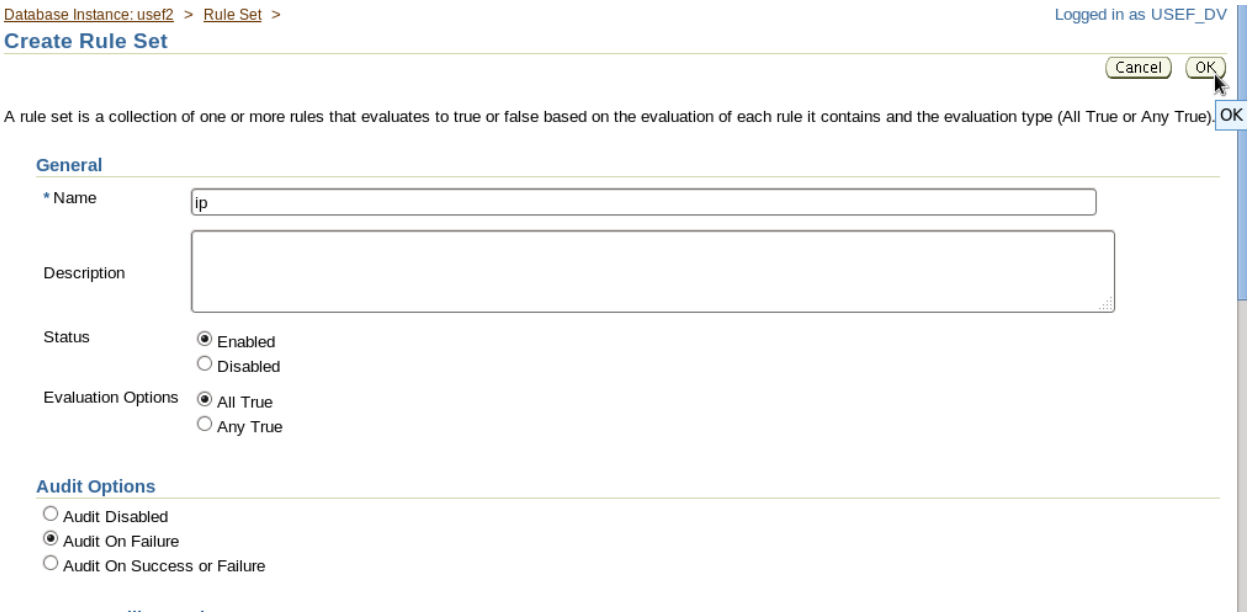
SQL> select * from app1.tbl_app1;

```
    TS# NAME                           INC BIG FLA ENC
------- ------------------------------ --- --- --- ---
      0 SYSTEM                         YES NO  YES
      1 SYSAUX                         YES NO  YES
      2 UNDOTBS1                       YES NO  YES
      4 USERS                          YES NO  YES
      3 TEMP                           NO  NO  YES
```

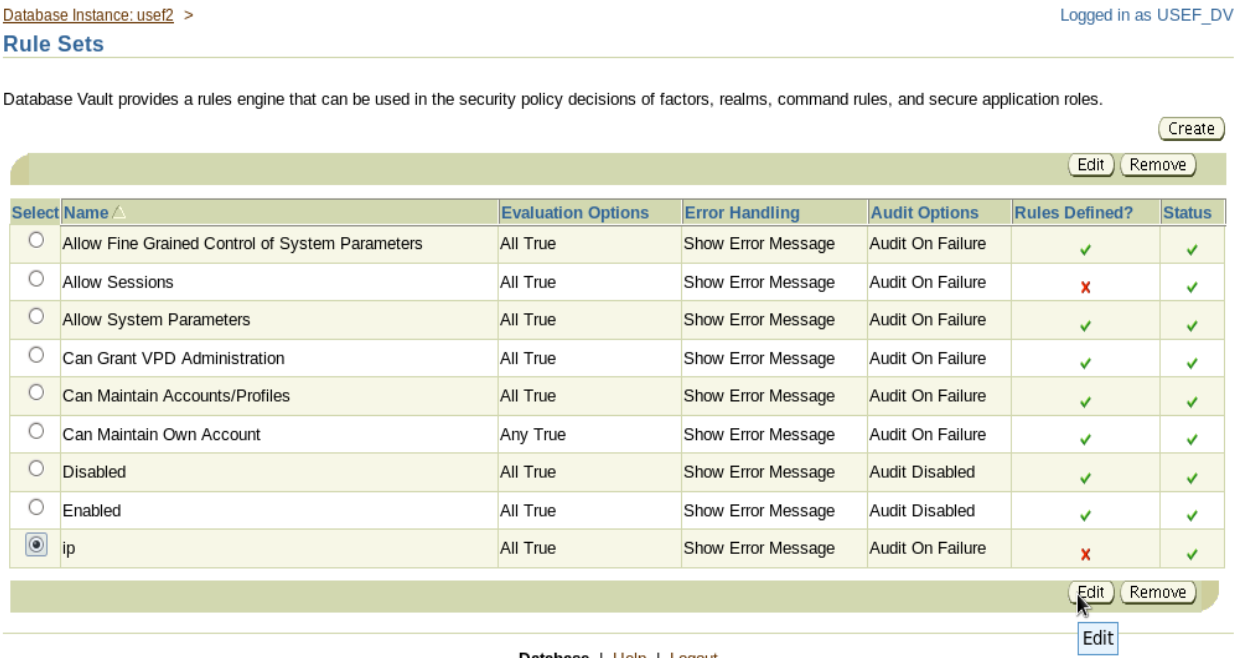
حال می توانیم همین سناریو را برای App2 انجام دهیم.

مثال پنجم(Rule Set - Factor): تنها فردی بتواند از اطلاعات کاربر usef بخواند(select) که از ip ای با شماره 10.47.52.21 وارد بانک شده باشد.

این کار از طریق فاکتور به همراه Rule set و یا Rule set به تنهایی قابل انجام است به همین جهت rule set جدیدی را با نام ip ایجاد می کنیم:



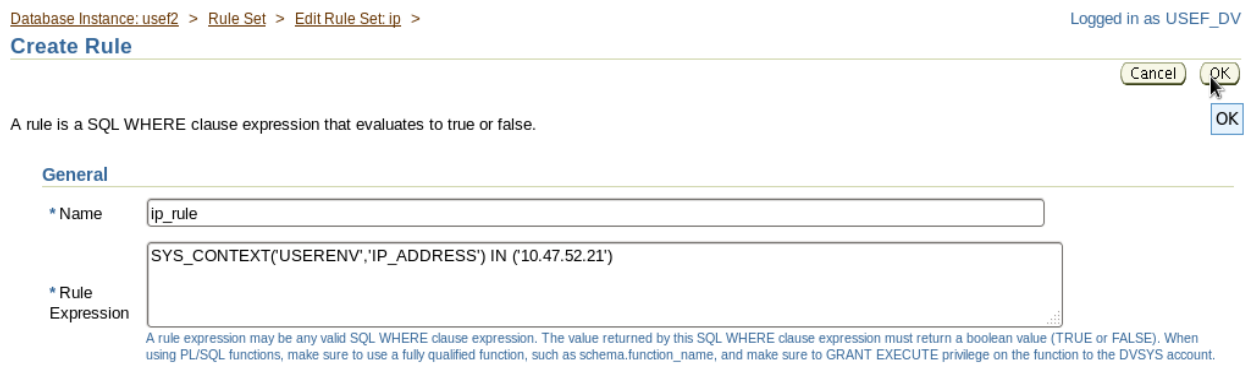
سپس Rule set مورد نظر را ویرایش می کنیم:



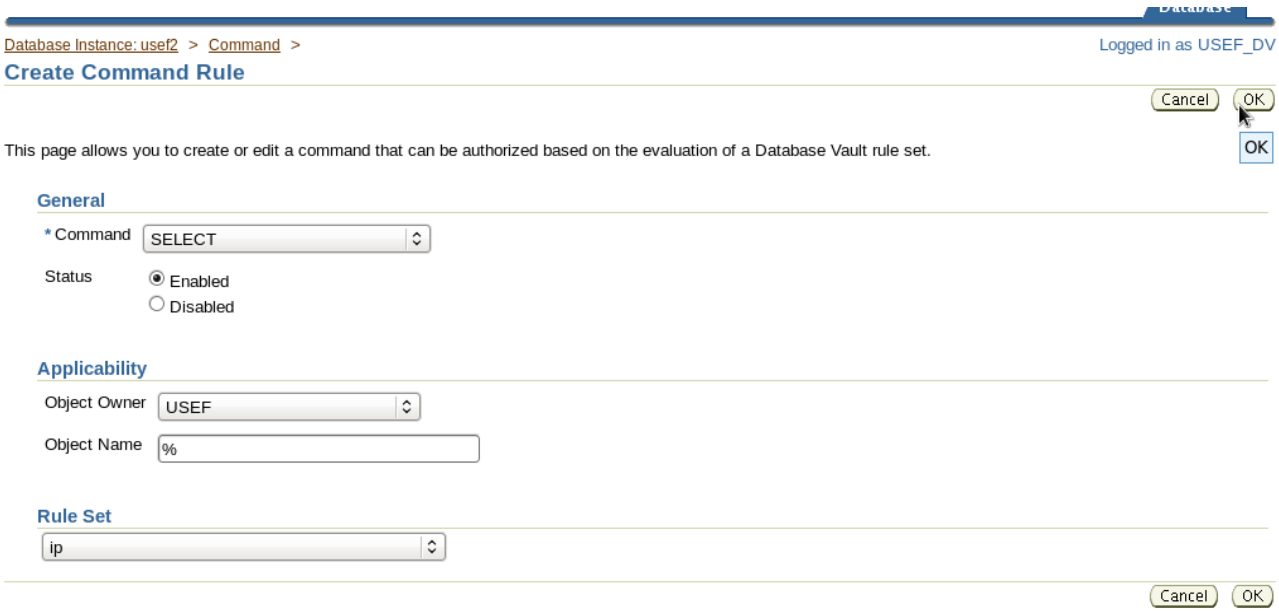
در قسمت rule associated، می توانیم فاکتوری را اضافه کنیم و یا rule جدیدی را تعریف کنیم که به همین جهت، create را انتخاب می کنیم:

در قسمت create rule، با کمک application context پیش فرض بانک که USERENV می باشد، شرط مورد نظر را قرار می دهیم.

SYS_CONTEXT('USERENV','IP_ADDRESS') IN ('10.47.52.21')



در نهایت به طریقی که قبلا اورده شد، command rule جدیدی را ایجاد می کنیم و در ان، rule set را به ip تنظیم می کنیم تا تنها ip مورد نظر بتواند اطلاعات کاربر usef را بخواند.



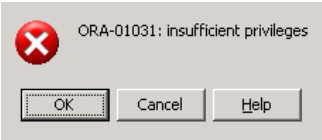
تست زیر نشان می دهد که اطلاعات کاربر usef به کاربرانی که از ip غیر مجاز وارد شده اند، نشان داده نمی شود:

Conn usef/usef

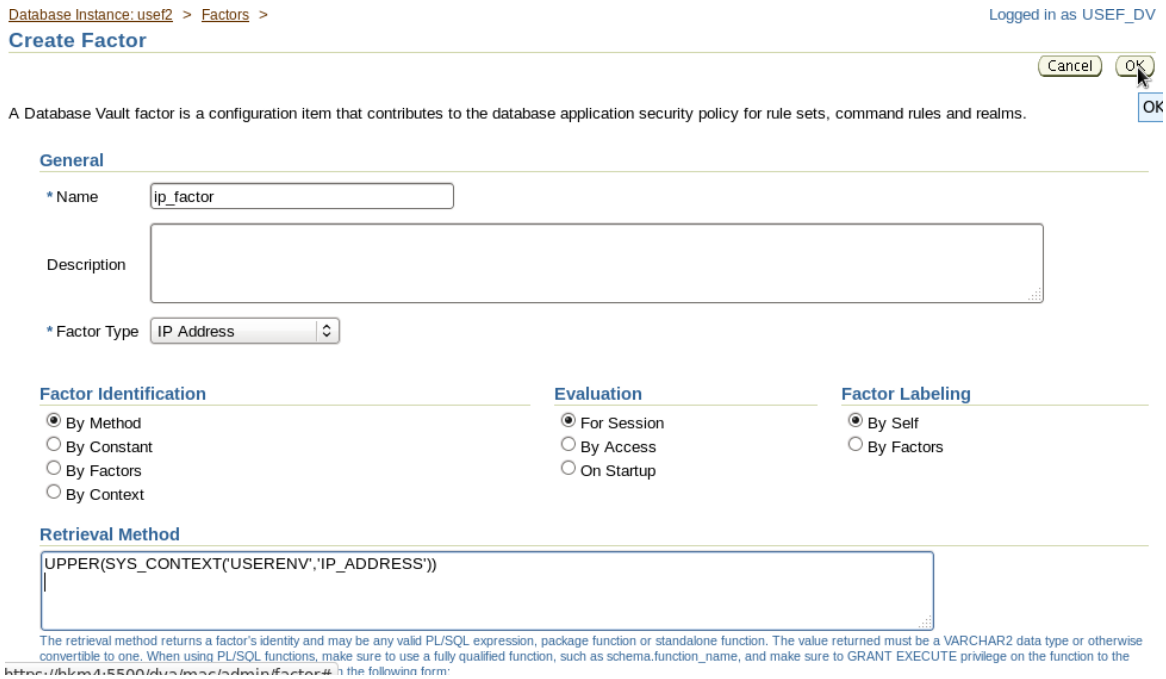
select get_factor('client_ip') from dual;

10.43.136.34

select * from usef.salary;



همچنین در صورتی که بخواهیم این کار با کمک Factor انجام شود، باید ابتدا یک فاکتور ایجاد کنیم:

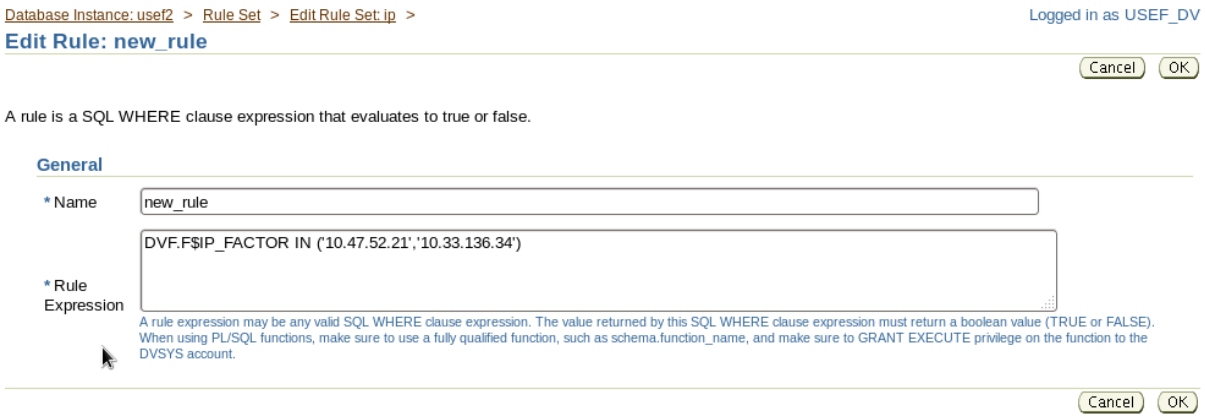


که در این صورت یک تابع با F$ip_factor برای کاربر DVF ساخته می شود که با اجرای ان، عبارت قید شده در قسمت Retrieval Method برخواهد گشت.

```
SELECT DVF.F$ip_factor FROM DUAL;
```

10.47.52.21

حال می توانیم با کمک این فاکتور، شرط مورد نظر را اعمال کنیم:



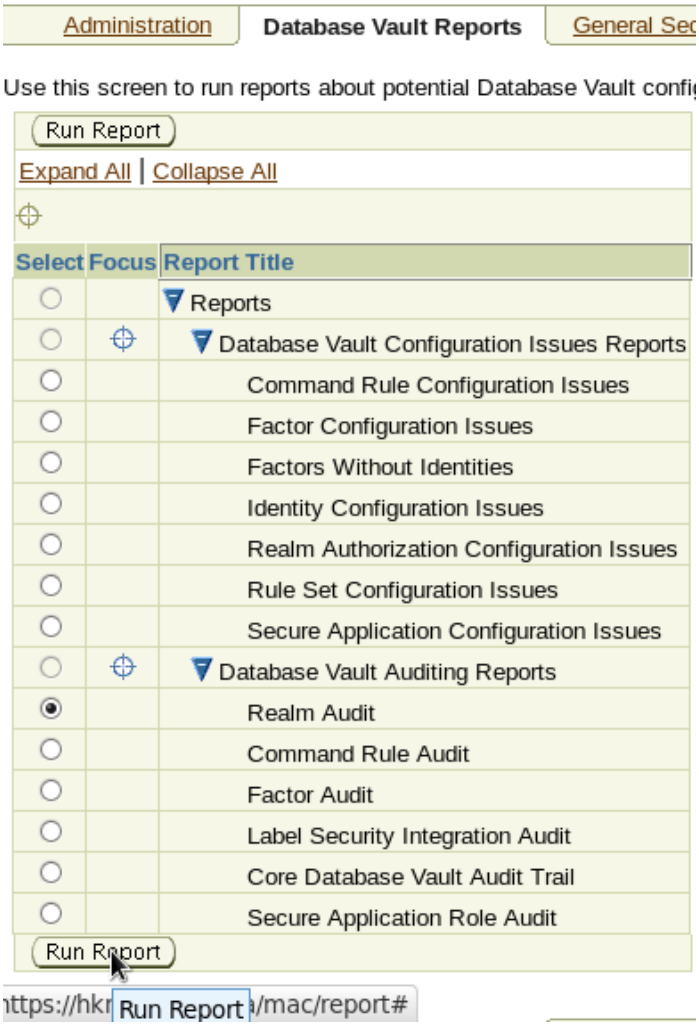
و در نهایت از این Rule Set در Command Rule مورد نظر استفاده کنیم.

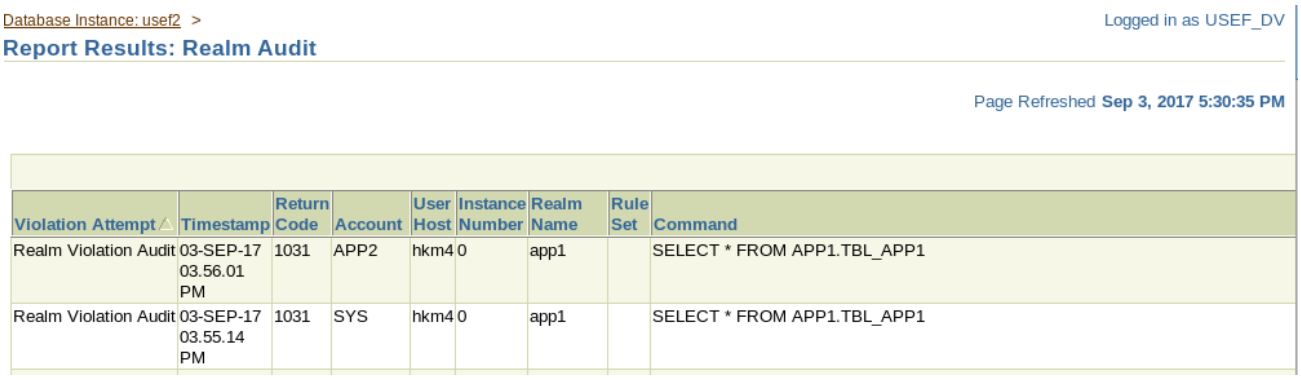
DATABASE VAULT REPORT

از دیگر ویژگی های مفید database vault، قابلیت مانیتورینگ ان می باشد که می توان از طریق ان، مشاهده کرد که "چه کسی" با کمک "چه مولفه ای" موفق به انجام "چه کاری" شده است.

DV به دو شیوه امکان نظارت و مانیتورینگ را فراهم می کند:

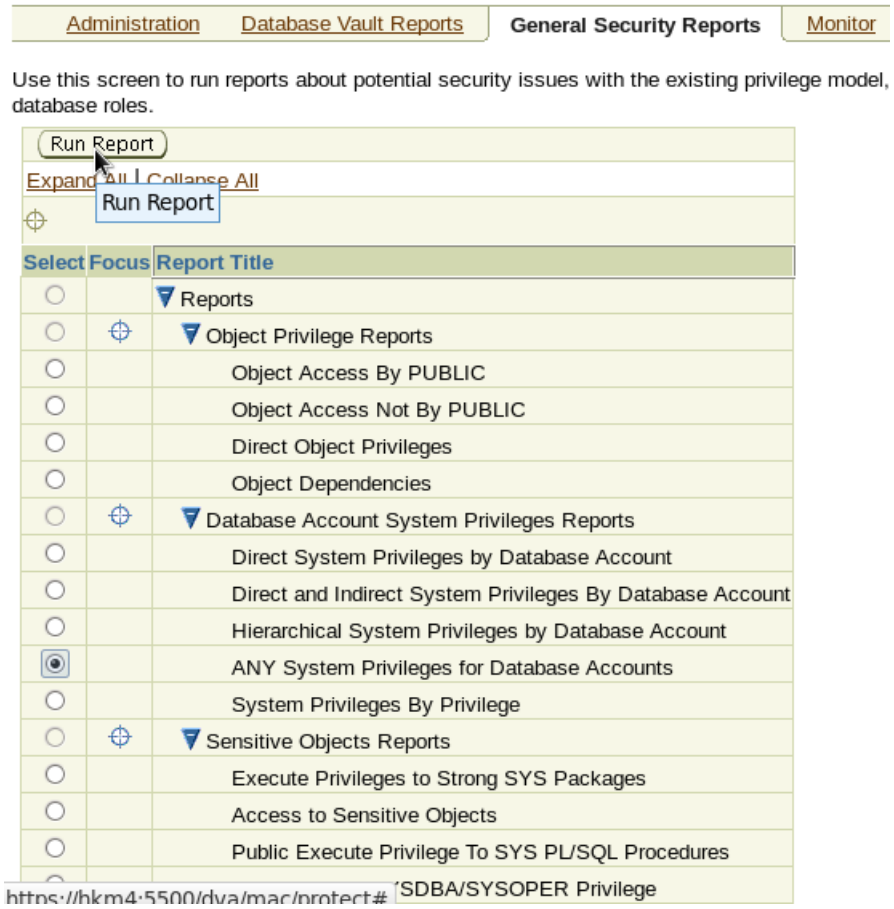
1.database Vault reports : در این بخش اطلاعاتی را مورد مولفه های مختلف اعم از realm، factor و ... را مشاهده خواهیم کرد. برای مثال میتوان با مراجعه به این قسمت مشاهده کرد که realmها مانع از انجام چه کاری شده اند:

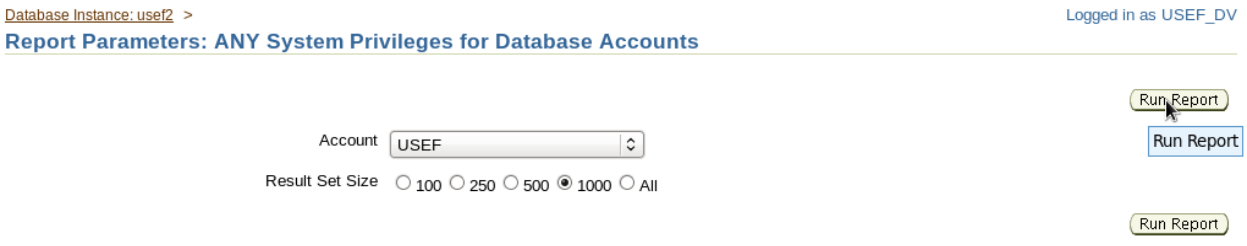


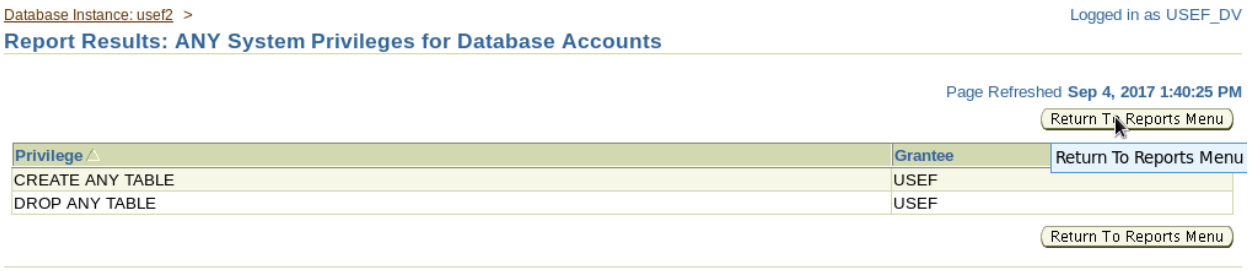


همانطور که در خروجی می بینید، کاربر app2 و sys قصد اجرای دستوری را داشته اند که به خاطر وجود realmای به نام app1، موفق به دسترسی ان اطلاعات نشده اند.

2.General Security reports : در این بخش گزارشی در مورد مسائل امنیتی عمومی از قبیل system privilegeها و object privilegeها ارائه می شود. برای مثال، شکل زیر نشان می دهد که کاربر usef دو مجوز create any table و drop any table را دریافت کرده است.







حذف database vault

برای حذف database vault می توان مراحل زیر را طی نمود:

1.ابتدا همه سرویسها را متوقف می کنیم:

sqlplus "/as sysdba"

shut abort

2.با دستور chopt، این ویژگی را غیرفعال می کنیم:

[oracle@hkm6 ~]$ chopt disable dv

Writing to /u01/oracle/11g44/install/disable_dv.log...

/usr/bin/make -f /u01/oracle/11g44/rdbms/lib/ins_rdbms.mk dv_off ORACLE_HOME=/u01/oracle/11g44

/usr/bin/make -f /u01/oracle/11g44/rdbms/lib/ins_rdbms.mk ioracle ORACLE_HOME=/u01/oracle/11g44

اجرای اسکریپت ایجاد شده:

[oracle@hkm6 ~]$ /usr/bin/make -f /u01/oracle/11g44/rdbms/lib/ins_rdbms.mk dv_off ORACLE_HOME=/u01/oracle/11g44

/usr/bin/ar cr /u01/oracle/11g44/rdbms/lib/libknlopt.a /u01/oracle/11g44/rdbms/lib/kzvndv.o

[oracle@hkm6 ~]$ /usr/bin/ar cr /u01/oracle/11g44/rdbms/lib/libknlopt.a /u01/oracle/11g44/rdbms/lib/kzvndv.o

[oracle@hkm6 ~]$ /usr/bin/make -f /u01/oracle/11g44/rdbms/lib/ins_rdbms.mk ioracle ORACLE_HOME=/u01/oracle/11g44

3.حال با راه اندازی مجدد بانک اطلاعاتی، تمامی اطلاعات مرتبط با DV را پاک می کنیم و تنظیمات بانک را به قبل از راه اندازی DV بر می گردانیم:

@?/rdbms/admin/dvremov.sql

drop user dbvowner cascade;

drop user dbvacctmgr cascade;

```
grant BECOME USER to DBA;
grant SELECT ANY TRANSACTION to DBA;
grant CREATE ANY JOB to DBA;
grant CREATE EXTERNAL JOB to DBA;
grant EXECUTE ANY PROGRAM to DBA;
grant EXECUTE ANY CLASS to DBA;
grant MANAGE SCHEDULER to DBA;
grant DEQUEUE ANY QUEUE to DBA;
grant ENQUEUE ANY QUEUE to DBA;
grant MANAGE ANY QUEUE to DBA;
grant BECOME USER to IMP_FULL_DATABASE;
grant MANAGE ANY QUEUE to IMP_FULL_DATABASE;
grant DBA to INFA_ADMIN;
grant EXECUTE ON DBMS_LOGMNR to EXECUTE_CATALOG_ROLE;
grant EXECUTE ON DBMS_LOGMNR_D to EXECUTE_CATALOG_ROLE;
grant EXECUTE ON DBMS_LOGMNR_LOGREP_DICT to EXECUTE_CATALOG_ROLE;
grant EXECUTE ON DBMS_LOGMNR_SESSION to EXECUTE_CATALOG_ROLE;
grant EXECUTE ON DBMS_FILE_TRANSFER to EXECUTE_CATALOG_ROLE;
grant EXECUTE ON UTL_FILE to PUBLIC;
grant CREATE ANY JOB to SCHEDULER_ADMIN;
grant CREATE EXTERNAL JOB to SCHEDULER_ADMIN;
grant EXECUTE ANY PROGRAM to SCHEDULER_ADMIN;
grant EXECUTE ANY CLASS to SCHEDULER_ADMIN;
grant MANAGE SCHEDULER to SCHEDULER_ADMIN;
ALTER SYSTEM SET AUDIT_SYS_OPERATIONS=FALSE SCOPE=SPFILE sid='*';
ALTER SYSTEM SET RECYCLEBIN='ON' SCOPE=SPFILE sid='*';
ALTER SYSTEM SET SQL92_SECURITY=FALSE SCOPE=SPFILE sid='*';
```