



TR-4981: Oracle Active Data Guard Cost Reduction with Amazon FSx ONTAP

NetApp database solutions

NetApp
July 31, 2025

Table of Contents

- TR-4981: Oracle Active Data Guard Cost Reduction with Amazon FSx ONTAP 1
 - Purpose 1
 - Audience 1
 - Solution test and validation environment 2
 - Architecture 2
 - Hardware and software components 2
 - Oracle Data Guard configuration with hypothetical NY to LA DR setup 3
 - Key factors for deployment consideration 3
 - Solution deployment 4
 - Prerequisites for deployment 4
 - Prepare the primary database for Data Guard 5
 - Prepare standby database and activate Data Guard 12
 - Setup Data Guard Broker 21
 - Clone standby database for other use cases 24
 - Where to find additional information 37

TR-4981: Oracle Active Data Guard Cost Reduction with Amazon FSx ONTAP

Allen Cao, Niyaz Mohamed, NetApp

This solution provides overview and details for configuring Oracle Data Guard using AWS FSx ONTAP as standby site Oracle database storage to reduce licensing and operational cost of Oracle Data Guard HA/DR solution in AWS.

Purpose

Oracle Data Guard ensures high availability, data protection, and disaster recovery for enterprise data in a primary database and standby database replication configuration. Oracle Active Data Guard empowers users to access standby databases while data replication is active from the primary database to standby databases. Data Guard is a feature of Oracle Database Enterprise Edition. It does not require separate licensing. On the other hand, Active Data Guard is an Oracle Database Enterprise Edition Option therefore requires separate licensing. Multiple standby databases can receive data replication from a primary database in the Active Data Guard setup. However, each additional standby database requires an Active Data Guard license and extra storage as the size of primary database. The operational costs add up quickly.

If you are keen on cutting back cost of your Oracle database operation and are planning to set up an Active Data Guard in AWS, you should consider an alternative. Instead of Active Data Guard, use Data Guard to replicate from primary database to a single physical standby database on Amazon FSx ONTAP storage. Subsequently, multiple copies of this standby database can be cloned and opened for read/write access to serve many other use cases such as reporting, development, test etc. The net results effectively deliver functionalities of Active Data Guard while eliminating Active Data Guard license and extra storage cost for each additional standby database. In this documentation, we demonstrate how to setup an Oracle Data Guard with your existing primary database in AWS and place physical standby database on Amazon FSx ONTAP storage. The standby database is backed up via snapshot and cloned for read/write access for use cases as desired.

This solution addresses the following use cases:

- Oracle Data Guard between a primary database on any storage in AWS to standby database on Amazon FSx ONTAP storage.
- Clone the standby database while closed for data replication to serve use cases such as reporting, dev, test, etc.

Audience

This solution is intended for the following people:

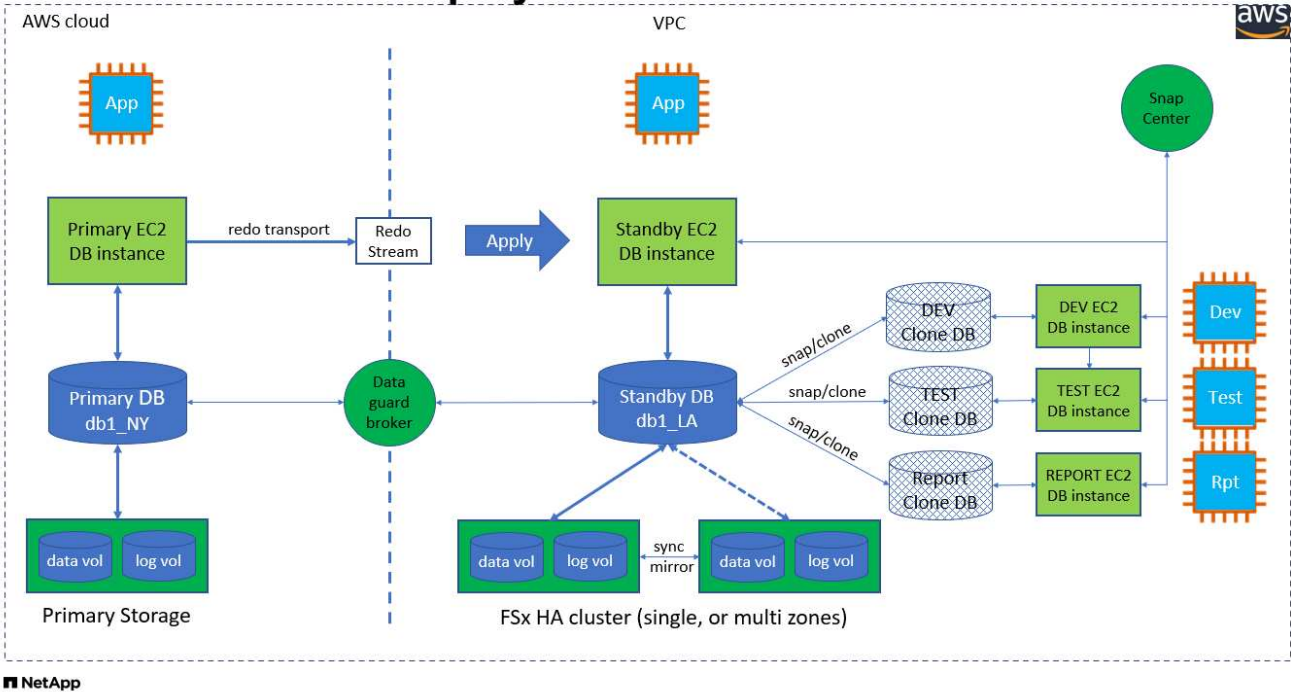
- A DBA who set up Oracle Active Data Guard in AWS for high availability, data protection, and disaster recovery.
- A database solution architect interested in Oracle Active Data Guard configuration in the AWS cloud.
- A storage administrator who manages AWS FSx ONTAP storage that supports Oracle Data Guard.
- An application owner who like to stand up Oracle Data Guard in AWS FSx/EC2 environment.

Solution test and validation environment

The testing and validation of this solution was performed in an AWS FSx ONTAP and EC2 lab environment that might not match the final deployment environment. For more information, see the section [Key factors for deployment consideration](#).

Architecture

Oracle Data Guard Deployment with Amazon FSx for ONTAP



Hardware and software components

Hardware		
FSx ONTAP storage	Current version offered by AWS	One FSx HA cluster in the same VPC and availability zone
EC2 instance for compute	t2.xlarge/4vCPU/16G	Three EC2 T2 xlarge EC2 instances, one as primary DB server, one as standby DB server, and the third as a clone DB server
Software		
RedHat Linux	RHEL-8.6.0_HVM-20220503-x86_64-2-Hourly2-GP2	Deployed RedHat subscription for testing
Oracle Grid Infrastructure	Version 19.18	Applied RU patch p34762026_190000_Linux-x86-64.zip
Oracle Database	Version 19.18	Applied RU patch p34765931_190000_Linux-x86-64.zip

Oracle OPatch	Version 12.2.0.1.36	Latest patch p6880880_190000_Linux-x86-64.zip
---------------	---------------------	--

Oracle Data Guard configuration with hypothetical NY to LA DR setup

Database	DB_UNIQUE_NAME	Oracle Net Service Name
Primary	db1_NY	db1_NY.demo.netapp.com
Physical Standby	db1_LA	db1_LA.demo.netapp.com

Key factors for deployment consideration

- **How Oracle Standby Database FlexClone Works.** AWS FSx ONTAP FlexClone provides shared copies of the same standby database volumes that are writable. The copies of the volumes are actually pointers that link back to original data blocks until a new write initiates on the clone. ONTAP then allocates new storage blocks for the new writes. Any read IOs are serviced by original data blocks under active replication. Thus, the clone are very storage efficient that can be used for many other use cases with minimal and incremental new storage allocation for new write IOs. This provides tremendous storage cost saving by substantially reducing Active Data Guard storage footprint. NetApp recommends to minimize FlexClone activities in the event of database switching over from primary storage to standby FSx storage in order to maintain Oracle performance at high level.
- **Oracle Software Requirements.** In general, a physical standby database must have the same Database Home version as the primary database including Patch Set Exceptions (PSEs), Critical Patch Updates (CPUs), and Patch Set Updates (PSUs), unless an Oracle Data Guard Standby-First Patch Apply process is in progress (as described in My Oracle Support note 1265700.1 at support.oracle.com)
- **Standby Database Directory Structure Considerations.** If possible, the data files, log files, and control files on the primary and standby systems should have the same names and path names and use Optimal Flexible Architecture (OFA) naming conventions. The archival directories on the standby database should also be identical between sites, including size and structure. This strategy allows other operations such as backups, switchovers, and failovers to execute the same set of steps, reducing the maintenance complexity.
- **Force Logging Mode.** To protect against unlogged direct writes in the primary database that cannot be propagated to the standby database, turn on FORCE LOGGING at the primary database before performing data file backups for standby creation.
- **Database Storage Management.** For operational simplicity, Oracle recommends that when you set up Oracle Automatic Storage Management (Oracle ASM) and Oracle Managed Files (OMF) in an Oracle Data Guard configuration that you set it up symmetrically on the primary and standby database(s).
- **EC2 compute instances.** In these tests and validations, we used an AWS EC2 t2.xlarge instance as the Oracle database compute instance. NetApp recommends using a M5 type EC2 instance as the compute instance for Oracle in production deployment because it is optimized for database workload. You need to size the EC2 instance appropriately for the number of vCPUs and the amount of RAM based on actual workload requirements.
- **FSx storage HA clusters single- or multi-zone deployment.** In these tests and validations, we deployed an FSx HA cluster in a single AWS availability zone. For production deployment, NetApp recommends deploying an FSx HA pair in two different availability zones. An FSx cluster is always provisioned in a HA pair that is sync mirrored in a pair of active-passive file systems to provide storage-level redundancy. Multi-zone deployment further enhances high availability in the event of failure in a single AWS zone.

- **FSx storage cluster sizing.** An Amazon FSx ONTAP storage file system provides up to 160,000 raw SSD IOPS, up to 4GBps throughput, and a maximum of 192TiB capacity. However, you can size the cluster in terms of provisioned IOPS, throughput, and the storage limit (minimum 1,024 GiB) based on your actual requirements at the time of deployment. The capacity can be adjusted dynamically on the fly without affecting application availability.

Solution deployment

It is assumed that you already have your primary Oracle database deployed in AWS EC2 environment within a VPC as the starting point for setting up Data Guard. The primary database is deployed using Oracle ASM for storage management. Two ASM disk groups - +DATA and +LOGS are created for Oracle data files, log files, and control file etc. For details on Oracle deployment in AWS with ASM, please refer to following technical reports for help.

- [Oracle Database Deployment on EC2 and FSx Best Practices](#)
- [Oracle Database Deployment and Protection in AWS FSx/EC2 with iSCSI/ASM](#)
- [Oracle 19c in Standalone Restart on AWS FSx/EC2 with NFS/ASM](#)

Your primary Oracle database can be running either on an FSx ONTAP or any other storage of choices within the AWS EC2 ecosystem. The following section provides step-by-step deployment procedures for setting up Oracle Data Guard between a primary EC2 DB instance with ASM storage to a standby EC2 DB instance with ASM storage.

Prerequisites for deployment

Deployment requires the following prerequisites.

1. An AWS account has been set up, and the necessary VPC and network segments have been created within your AWS account.
2. From the AWS EC2 console, you need to deploy minimum three EC2 Linux instances, one as the primary Oracle DB instance, one as standby Oracle DB instance, and an clone target DB instance for reporting, dev, and test etc. See the architecture diagram in the previous section for more details about the environment setup. Also review the AWS [User Guide for Linux instances](#) for more information.
3. From the AWS EC2 console, deploy Amazon FSx ONTAP storage HA clusters to host Oracle volumes that stores the Oracle standby database. If you are not familiar with the deployment of FSx storage, see the documentation [Creating FSx ONTAP file systems](#) for step-by-step instructions.
4. Steps 2 and 3 can be performed using the following Terraform automation toolkit, which creates an EC2 instance named `ora_01` and an FSx file system named `fsx_01`. Review the instruction carefully and change the variables to suit your environment before execution. The template can be easily revised for your own deployment requirements.

```
git clone https://github.com/NetApp-  
Automation/na_aws_fsx_ec2_deploy.git
```



Ensure that you have allocated at least 50G in EC2 instance root volume in order to have sufficient space to stage Oracle installation files.

Prepare the primary database for Data Guard

In this demonstration, we have setup a primary Oracle database called db1 on the primary EC2 DB instance with two ASM disk groups in standalone Restart configuration with data files in ASM disk group +DATA and flash recovery area in ASM disk group +LOGS. Following illustrates the detailed procedures for setting up primary database for Data Guard. All steps should be executed as database owner - oracle user.

1. Primary database db1 configuration on primary EC2 DB instance ip-172-30-15-45. The ASM disk groups can be on any type of storage within EC2 ecosystem.

```
[oracle@ip-172-30-15-45 ~]$ cat /etc/oratab

# This file is used by ORACLE utilities.  It is created by root.sh
# and updated by either Database Configuration Assistant while
# creating
# a database or ASM Configuration Assistant while creating ASM
# instance.

# A colon, ':', is used as the field terminator.  A new line
# terminates
# the entry.  Lines beginning with a pound sign, '#', are comments.
#
# Entries are of the form:
#   $ORACLE_SID:$ORACLE_HOME:<N|Y>:
#
# The first and second fields are the system identifier and home
# directory of the database respectively.  The third field indicates
# to the dbstart utility that the database should , "Y", or should
# not,
# "N", be brought up at system boot time.
#
# Multiple entries with the same $ORACLE_SID are not allowed.
#
#
+ASM:/u01/app/oracle/product/19.0.0/grid:N
db1:/u01/app/oracle/product/19.0.0/db1:N

[oracle@ip-172-30-15-45 ~]$
/u01/app/oracle/product/19.0.0/grid/bin/crsctl stat res -t
-----
-----
Name                Target    State          Server          State
details
-----
-----
Local Resources
-----
```



```

-----
ora.DATA.dg
      ONLINE  ONLINE      ip-172-30-15-45      STABLE
ora.LISTENER.lsnr
      ONLINE  ONLINE      ip-172-30-15-45      STABLE
ora.LOGS.dg
      ONLINE  ONLINE      ip-172-30-15-45      STABLE
ora.asm
      ONLINE  ONLINE      ip-172-30-15-45
Started,STABLE
ora.ons
      OFFLINE OFFLINE      ip-172-30-15-45      STABLE
-----
Cluster Resources
-----
-----
ora.cssd
      1      ONLINE  ONLINE      ip-172-30-15-45      STABLE
ora.dbf1.db
      1      ONLINE  ONLINE      ip-172-30-15-45
Open,HOME=/u01/app/o

racle/product/19.0.0

/db1,STABLE
ora.diskmon
      1      OFFLINE OFFLINE      STABLE
ora.driver.afd
      1      ONLINE  ONLINE      ip-172-30-15-45      STABLE
ora.evmd
      1      ONLINE  ONLINE      ip-172-30-15-45      STABLE
-----
-----

```

2. From sqlplus, enable forced logging on primary.

```
alter database force logging;
```

3. From sqlplus, enable flashback on primary. Flashback allows easy reinstate primary database as a standby after a failover.

```
alter database flashback on;
```

4. Configure redo transport authentication using Oracle password file - create a pwd file on the primary using orapwd utility if not set and copy over to standby database \$ORACLE_HOME/dbs directory.
5. Create standby redo logs on the primary DB with same size as current online log file. Log groups are one more than online log file groups. The primary database can then quickly transition to the standby role and begin receiving redo data, if necessary.

```
alter database add standby logfile thread 1 size 200M;
```

Validate after standby logs addition:

```
SQL> select group#, type, member from v$logfile;
```

GROUP#	TYPE	MEMBER
3	ONLINE	+DATA/DB1/ONLINELOG/group_3.264.1145821513
2	ONLINE	+DATA/DB1/ONLINELOG/group_2.263.1145821513
1	ONLINE	+DATA/DB1/ONLINELOG/group_1.262.1145821513
4	STANDBY	+DATA/DB1/ONLINELOG/group_4.286.1146082751
4	STANDBY	+LOGS/DB1/ONLINELOG/group_4.258.1146082753
5	STANDBY	+DATA/DB1/ONLINELOG/group_5.287.1146082819
5	STANDBY	+LOGS/DB1/ONLINELOG/group_5.260.1146082821
6	STANDBY	+DATA/DB1/ONLINELOG/group_6.288.1146082825
6	STANDBY	+LOGS/DB1/ONLINELOG/group_6.261.1146082827
7	STANDBY	+DATA/DB1/ONLINELOG/group_7.289.1146082835
7	STANDBY	+LOGS/DB1/ONLINELOG/group_7.262.1146082835

11 rows selected.

6. From sqlplus, create a pfile from spfile for editing.

```
create pfile='/home/oracle/initdb1.ora' from spfile;
```

7. Revise the pfile and add following parameters.

```
DB_NAME=db1
DB_UNIQUE_NAME=db1_NY
LOG_ARCHIVE_CONFIG='DG_CONFIG=(db1_NY,db1_LA) '
LOG_ARCHIVE_DEST_1='LOCATION=USE_DB_RECOVERY_FILE_DEST
VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_NAME=db1_NY'
LOG_ARCHIVE_DEST_2='SERVICE=db1_LA ASYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=db1_LA'
REMOTE_LOGIN_PASSWORDFILE=EXCLUSIVE
FAL_SERVER=db1_LA
STANDBY_FILE_MANAGEMENT=AUTO
```

8. From sqlplus, create spfile in ASM +DATA directory from revised pfile in /home/oracle directory.

```
create spfile='+DATA' from pfile='/home/oracle/initdb1.ora';
```

9. Locate the newly created spfile under +DATA disk group(using asmcmd utility if necessary). Use srvctl to modify grid to start database from new spfile as shown below.

```

[oracle@ip-172-30-15-45 db1]$ srvctl config database -d db1
Database unique name: db1
Database name: db1
Oracle home: /u01/app/oracle/product/19.0.0/db1
Oracle user: oracle
Spfile: +DATA/DB1/PARAMETERFILE/spfile.270.1145822903
Password file:
Domain: demo.netapp.com
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Disk Groups: DATA
Services:
OSDBA group:
OSOPER group:
Database instance: db1
[oracle@ip-172-30-15-45 db1]$ srvctl modify database -d db1 -spfile
+DATA/DB1/PARAMETERFILE/spfiledb1.ora
[oracle@ip-172-30-15-45 db1]$ srvctl config database -d db1
Database unique name: db1
Database name: db1
Oracle home: /u01/app/oracle/product/19.0.0/db1
Oracle user: oracle
Spfile: +DATA/DB1/PARAMETERFILE/spfiledb1.ora
Password file:
Domain: demo.netapp.com
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Disk Groups: DATA
Services:
OSDBA group:
OSOPER group:
Database instance: db1

```

10. Modify tnsnames.ora to add db_unique_name for name resolution.

```
# tnsnames.ora Network Configuration File:
/u01/app/oracle/product/19.0.0/db1/network/admin/tnsnames.ora
# Generated by Oracle configuration tools.

db1_NY =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = ip-172-30-15-
45.ec2.internal) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SID = db1)
    )
  )

db1_LA =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = ip-172-30-15-
67.ec2.internal) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SID = db1)
    )
  )

LISTENER_DB1 =
  (ADDRESS = (PROTOCOL = TCP) (HOST = ip-172-30-15-
45.ec2.internal) (PORT = 1521))
```

11. Add data guard service name db1_NY_DGMGRL.demo.netapp for primary database to listener.ora file.

```
#Backup file is /u01/app/oracle/crsdata/ip-172-30-15-45/output/listener.ora.bak.ip-172-30-15-45.oracle line added by Agent
# listener.ora Network Configuration File:
/u01/app/oracle/product/19.0.0/grid/network/admin/listener.ora
# Generated by Oracle configuration tools.
```

```
LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP) (HOST = ip-172-30-15-45.ec2.internal) (PORT = 1521))
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
    )
  )
```

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = db1_NY_DGMGRL.demo.netapp.com)
      (ORACLE_HOME = /u01/app/oracle/product/19.0.0/db1)
      (SID_NAME = db1)
    )
  )
```

```
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER=ON # line added by Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER=ON # line added by Agent
```

1. Shutdown and restart database with srvctl and validate that data guard parameters are now active.

```
srvctl stop database -d db1
```

```
srvctl start database -d db1
```

This completes primary database setup for Data Guard.

Prepare standby database and activate Data Guard

Oracle Data Guard requires OS kernel configuration and Oracle software stacks including patch sets on standby EC2 DB instance to match with primary EC2 DB instance. For easy management and simplicity, the standby EC2 DB instance database storage configuration ideally should match with the primary EC2 DB instance as well, such as the name, number and size of ASM disk groups. Following are detail procedures for setting up the standby EC2 DB instance for Data Guard. All commands should be executed as oracle owner user id.

1. First, review the configuration of the primary database on primary EC2 instance. In this demonstration, we have setup a primary Oracle database called db1 on the primary EC2 DB instance with two ASM disk groups +DATA and +LOGS in standalone Restart configuration. The primary ASM disk groups may be on any type of storage within EC2 ecosystem.
2. Follow procedures in documentation [TR-4965: Oracle Database Deployment and Protection in AWS FSx/EC2 with iSCSI/ASM](#) to install and configure grid and Oracle on standby EC2 DB instance to match with primary database. The database storage should be provisioned and allocated to standby EC2 DB instance from FSx ONTAP with same storage capacity as primary EC2 DB instance.



Stop at step 10 in Oracle database installation section. The standby database will be instantiated from primary database using dbca database duplication function.

3. Once Oracle software is installed and configured, from standby \$ORACLE_HOME dbs directory, copy oracle password from primary database.

```
scp  
oracle@172.30.15.45:/u01/app/oracle/product/19.0.0/db1/dbs/orapwdb1  
.
```

4. Create tnsnames.ora file with following entries.

```
# tnsnames.ora Network Configuration File:
/u01/app/oracle/product/19.0.0/db1/network/admin/tnsnames.ora
# Generated by Oracle configuration tools.

db1_NY =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = ip-172-30-15-
45.ec2.internal) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SID = db1)
    )
  )

db1_LA =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = ip-172-30-15-
67.ec2.internal) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SID = db1)
    )
  )
```

5. Add DB data guard service name to listener.ora file.


```

#Backup file is /u01/app/oracle/crsdata/ip-172-30-15-
67/output/listener.ora.bak.ip-172-30-15-67.oracle line added by
Agent
# listener.ora Network Configuration File:
/u01/app/oracle/product/19.0.0/grid/network/admin/listener.ora
# Generated by Oracle configuration tools.

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP) (HOST = ip-172-30-15-
67.ec2.internal) (PORT = 1521))
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
    )
  )

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = db1_LA_DGMGRL.demo.netapp.com)
      (ORACLE_HOME = /u01/app/oracle/product/19.0.0/db1)
      (SID_NAME = db1)
    )
  )

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER=ON # line added
by Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER=ON # line added
by Agent

```

6. Set oracle home and path.

```
export ORACLE_HOME=/u01/app/oracle/product/19.0.0/db1
```

```
export PATH=$PATH:$ORACLE_HOME/bin
```

7. Use dbca to instantiate standby database from primary database db1.

```

[oracle@ip-172-30-15-67 bin]$ dbca -silent -createDuplicateDB
-gdbName db1 -primaryDBConnectionString ip-172-30-15-
45.ec2.internal:1521/db1_NY.demo.netapp.com -sid db1 -initParams
fal_server=db1_NY -createAsStandby -dbUniqueName db1_LA
Enter SYS user password:

Prepare for db operation
22% complete
Listener config step
44% complete
Auxiliary instance creation
67% complete
RMAN duplicate
89% complete
Post duplicate database operations
100% complete

Look at the log file
"/u01/app/oracle/cfgtoollogs/dbca/db1_LA/db1_LA.log" for further
details.

```

8. Validate duplicated standby database. Newly duplicated standby database open in READ ONLY mode initially.

```

[oracle@ip-172-30-15-67 bin]$ export ORACLE_SID=db1
[oracle@ip-172-30-15-67 bin]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Aug 30 18:25:46
2023
Version 19.18.0.0.0

Copyright (c) 1982, 2022, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.18.0.0.0

SQL> select name, open_mode from v$database;

NAME          OPEN_MODE
-----
DB1           READ ONLY

```

```
SQL> show parameter name
```

NAME	TYPE	VALUE
-----	-----	

cdb_cluster_name	string	
cell_offloadgroup_name	string	
db_file_name_convert	string	
db_name	string	db1
db_unique_name	string	db1_LA
global_names	boolean	FALSE
instance_name	string	db1
lock_name_space	string	
log_file_name_convert	string	
pdb_file_name_convert	string	
processor_group_name	string	

NAME	TYPE	VALUE
-----	-----	

service_names	string	
db1_LA.demo.netapp.com		

```
SQL>
```

```
SQL> show parameter log_archive_config
```

NAME	TYPE	VALUE
-----	-----	

log_archive_config	string	
DG_CONFIG=(db1_NY,db1_LA)		

```
SQL> show parameter fal_server
```

NAME	TYPE	VALUE
-----	-----	

fal_server	string	db1_NY

```
SQL> select name from v$datafile;
```

NAME

+DATA/DB1_LA/DATAFILE/system.261.1146248215
+DATA/DB1_LA/DATAFILE/sysaux.262.1146248231
+DATA/DB1_LA/DATAFILE/undotbs1.263.1146248247
+DATA/DB1_LA/03C5C01A66EE9797E0632D0F1EAC5F59/DATAFILE/system.264.11

```

46248253
+DATA/DB1_LA/03C5C01A66EE9797E0632D0F1EAC5F59/DATAFILE/sysaux.265.11
46248261
+DATA/DB1_LA/DATAFILE/users.266.1146248267
+DATA/DB1_LA/03C5C01A66EE9797E0632D0F1EAC5F59/DATAFILE/undotbs1.267.
1146248269
+DATA/DB1_LA/03C5EFD07C41A1FAE0632D0F1EAC9BD8/DATAFILE/system.268.11
46248271
+DATA/DB1_LA/03C5EFD07C41A1FAE0632D0F1EAC9BD8/DATAFILE/sysaux.269.11
46248279
+DATA/DB1_LA/03C5EFD07C41A1FAE0632D0F1EAC9BD8/DATAFILE/undotbs1.270.
1146248285
+DATA/DB1_LA/03C5EFD07C41A1FAE0632D0F1EAC9BD8/DATAFILE/users.271.114
6248293

```

NAME

```

-----
-----
+DATA/DB1_LA/03C5F0DDF35CA2B6E0632D0F1EAC8B6B/DATAFILE/system.272.11
46248295
+DATA/DB1_LA/03C5F0DDF35CA2B6E0632D0F1EAC8B6B/DATAFILE/sysaux.273.11
46248301
+DATA/DB1_LA/03C5F0DDF35CA2B6E0632D0F1EAC8B6B/DATAFILE/undotbs1.274.
1146248309
+DATA/DB1_LA/03C5F0DDF35CA2B6E0632D0F1EAC8B6B/DATAFILE/users.275.114
6248315
+DATA/DB1_LA/03C5F1C9B142A2F1E0632D0F1EACF21A/DATAFILE/system.276.11
46248317
+DATA/DB1_LA/03C5F1C9B142A2F1E0632D0F1EACF21A/DATAFILE/sysaux.277.11
46248323
+DATA/DB1_LA/03C5F1C9B142A2F1E0632D0F1EACF21A/DATAFILE/undotbs1.278.
1146248331
+DATA/DB1_LA/03C5F1C9B142A2F1E0632D0F1EACF21A/DATAFILE/users.279.114
6248337

```

19 rows selected.

```
SQL> select name from v$controlfile;
```

NAME

```

-----
-----
+DATA/DB1_LA/CONTROLFILE/current.260.1146248209
+LOGS/DB1_LA/CONTROLFILE/current.257.1146248209

```

```
SQL> select name from v$tempfile;
```

NAME

```
-----  
-----  
+DATA/DB1_LA/TEMPFILE/temp.287.1146248371  
+DATA/DB1_LA/03C5C01A66EE9797E0632D0F1EAC5F59/TEMPFILE/temp.288.1146  
248375  
+DATA/DB1_LA/03C5EFD07C41A1FAE0632D0F1EAC9BD8/TEMPFILE/temp.290.1146  
248463  
+DATA/DB1_LA/03C5F0DDF35CA2B6E0632D0F1EAC8B6B/TEMPFILE/temp.291.1146  
248463  
+DATA/DB1_LA/03C5F1C9B142A2F1E0632D0F1EACF21A/TEMPFILE/temp.292.1146  
248463
```

```
SQL> select group#, type, member from v$logfile order by 2, 1;
```

GROUP#	TYPE	MEMBER
1	ONLINE	+LOGS/DB1_LA/ONLINELOG/group_1.259.1146248349
1	ONLINE	+DATA/DB1_LA/ONLINELOG/group_1.280.1146248347
2	ONLINE	+DATA/DB1_LA/ONLINELOG/group_2.281.1146248351
2	ONLINE	+LOGS/DB1_LA/ONLINELOG/group_2.258.1146248353
3	ONLINE	+DATA/DB1_LA/ONLINELOG/group_3.282.1146248355
3	ONLINE	+LOGS/DB1_LA/ONLINELOG/group_3.260.1146248355
4	STANDBY	+DATA/DB1_LA/ONLINELOG/group_4.283.1146248357
4	STANDBY	+LOGS/DB1_LA/ONLINELOG/group_4.261.1146248359
5	STANDBY	+DATA/DB1_LA/ONLINELOG/group_5.284.1146248361
5	STANDBY	+LOGS/DB1_LA/ONLINELOG/group_5.262.1146248363
6	STANDBY	+LOGS/DB1_LA/ONLINELOG/group_6.263.1146248365
6	STANDBY	+DATA/DB1_LA/ONLINELOG/group_6.285.1146248365
7	STANDBY	+LOGS/DB1_LA/ONLINELOG/group_7.264.1146248369
7	STANDBY	+DATA/DB1_LA/ONLINELOG/group_7.286.1146248367

14 rows selected.

```
SQL> select name, open_mode from v$database;
```

NAME	OPEN_MODE
DB1	READ ONLY

- Restart standby database in mount stage and execute following command to activate standby database managed recovery.

```
alter database recover managed standby database disconnect from
session;
```

```
SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup mount;
ORACLE instance started.
```

```
Total System Global Area 8053062944 bytes
Fixed Size                  9182496 bytes
Variable Size              1291845632 bytes
Database Buffers           6744440832 bytes
Redo Buffers                7593984 bytes
```

```
Database mounted.
```

```
SQL> alter database recover managed standby database disconnect from
session;
```

```
Database altered.
```

10. Validate the standby database recovery status. Notice the recovery logmerger in APPLYING_LOG action.

```
SQL> SELECT ROLE, THREAD#, SEQUENCE#, ACTION FROM
V$DATAGUARD_PROCESS;
```

ROLE	THREAD#	SEQUENCE#	ACTION
recovery apply slave	0	0	IDLE
recovery apply slave	0	0	IDLE
recovery apply slave	0	0	IDLE
recovery apply slave	0	0	IDLE
recovery logmerger	1	30	APPLYING_LOG
RFS ping	1	30	IDLE
RFS async	1	30	IDLE
archive redo	0	0	IDLE
archive redo	0	0	IDLE
archive redo	0	0	IDLE
gap manager	0	0	IDLE

ROLE	THREAD#	SEQUENCE#	ACTION
managed recovery	0	0	IDLE
redo transport monitor	0	0	IDLE
log writer	0	0	IDLE
archive local	0	0	IDLE
redo transport timer	0	0	IDLE

16 rows selected.

```
SQL>
```

This completes the Data Guard protection setup for db1 from primary to standby with managed standby recovery enabled.

Setup Data Guard Broker

Oracle Data Guard broker is a distributed management framework that automates and centralizes the creation, maintenance, and monitoring of Oracle Data Guard configurations. Following section demonstrate how to setup Data Guard Broker to manage Data Guard environment.

1. Start data guard broker on both primary and standby databases with following command via sqlplus.

```
alter system set dg_broker_start=true scope=both;
```

2. From primary database, connect to Data Guard Borker as SYSDBA.

```
[oracle@ip-172-30-15-45 db1]$ dgmgrl sys@db1_NY
DGMGRL for Linux: Release 19.0.0.0.0 - Production on Wed Aug 30
19:34:14 2023
Version 19.18.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights
reserved.

Welcome to DGMGRL, type "help" for information.
Password:
Connected to "db1_NY"
Connected as SYSDBA.
```

3. Create and enable Data Guard Broker configuration.


```
DGMGRL> create configuration dg_config as primary database is db1_NY
connect identifier is db1_NY;
Configuration "dg_config" created with primary database "db1_ny"
DGMGRL> add database db1_LA as connect identifier is db1_LA;
Database "db1_la" added
DGMGRL> enable configuration;
Enabled.
DGMGRL> show configuration;

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
db1_ny - Primary database
db1_la - Physical standby database

Fast-Start Failover: Disabled

Configuration Status:
SUCCESS (status updated 28 seconds ago)
```

4. Validate database status within Data Guard Broker management framework.

```
DGMGRL> show database db1_ny;
```

```
Database - db1_ny
```

```
Role:                PRIMARY
Intended State:       TRANSPORT-ON
Instance(s):         db1
```

```
Database Status:
SUCCESS
```

```
DGMGRL> show database db1_la;
```

```
Database - db1_la
```

```
Role:                PHYSICAL STANDBY
Intended State:       APPLY-ON
Transport Lag:        0 seconds (computed 1 second ago)
Apply Lag:            0 seconds (computed 1 second ago)
Average Apply Rate:   2.00 KByte/s
Real Time Query:      OFF
Instance(s):         db1
```

```
Database Status:
SUCCESS
```

```
DGMGRL>
```

In the event of a failure, Data Guard Broker can be used to failover primary database to standby instantaneously.

Clone standby database for other use cases

The key benefit of staging standby database on AWS FSx ONTAP in Data Guard is that it can be FlexCloned to serve many other use cases with minimal additional storage investment. In the following section, we demonstrate how to snapshot and clone the mounted and under recovery standby database volumes on FSx ONTAP for other purposes, such as DEV, TEST, REPORT, etc., using the NetApp SnapCenter tool.

Following are high level procedures to clone a READ/WRITE database from the managed physical standby database in Data Guard using SnapCenter. For detail instructions on how to setup and configure SnapCenter, please refer to [Hybrid Cloud Database Solutions with SnapCenter](#) relevant Oracle sections.

1. We begin with creating a test table and inserting a row into the test table on primary database. We will then validate if the transaction traverse down to standby and finally the clone.

```
[oracle@ip-172-30-15-45 db1]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Aug 31 16:35:53
2023
Version 19.18.0.0.0

Copyright (c) 1982, 2022, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.18.0.0.0

SQL> alter session set container=db1_pdb1;

Session altered.

SQL> create table test(
  2   id integer,
  3   dt timestamp,
  4   event varchar(100));

Table created.

SQL> insert into test values(1, sysdate, 'a test transaction on
primary database db1 and ec2 db host: ip-172-30-15-
45.ec2.internal');

1 row created.

SQL> commit;

Commit complete.
```

```
SQL> select * from test;
```

ID

DT

EVENT

1

31-AUG-23 04.49.29.000000 PM

a test transaction on primary database db1 and ec2 db host: ip-172-30-15-45.ec2.

internal

```
SQL> select instance_name, host_name from v$instance;
```

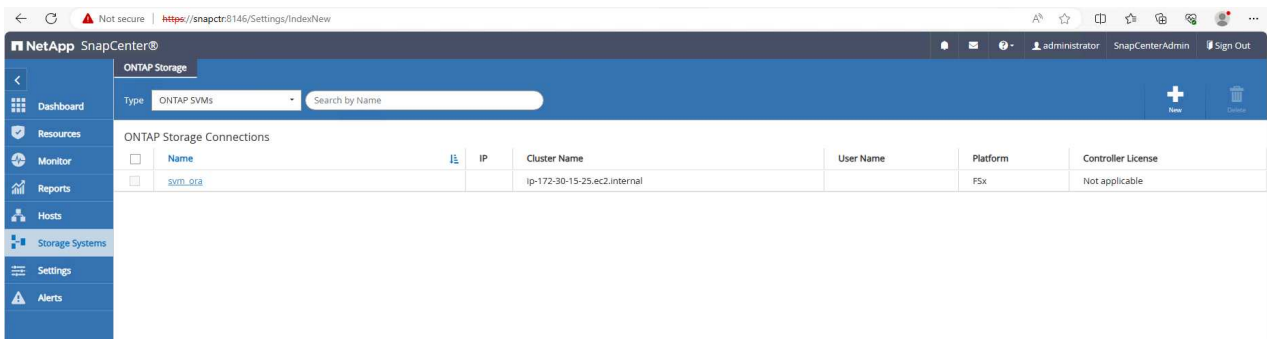
INSTANCE_NAME

HOST_NAME

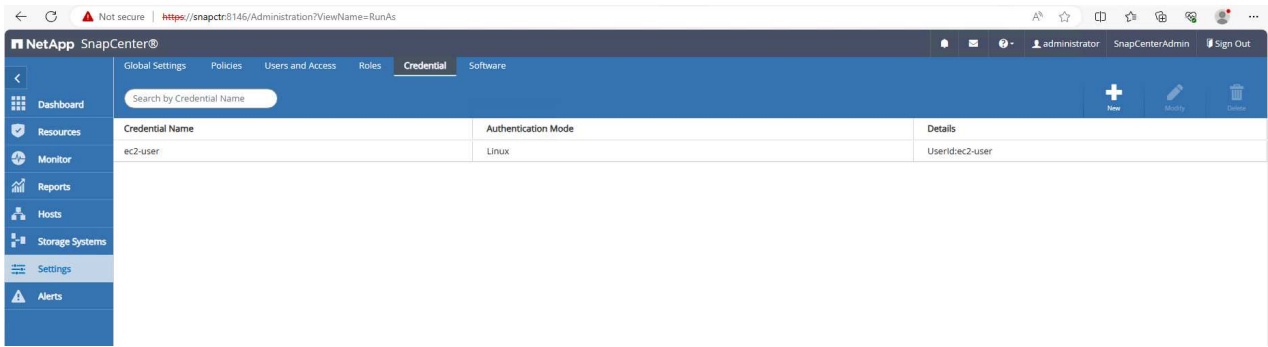
db1

ip-172-30-15-45.ec2.internal

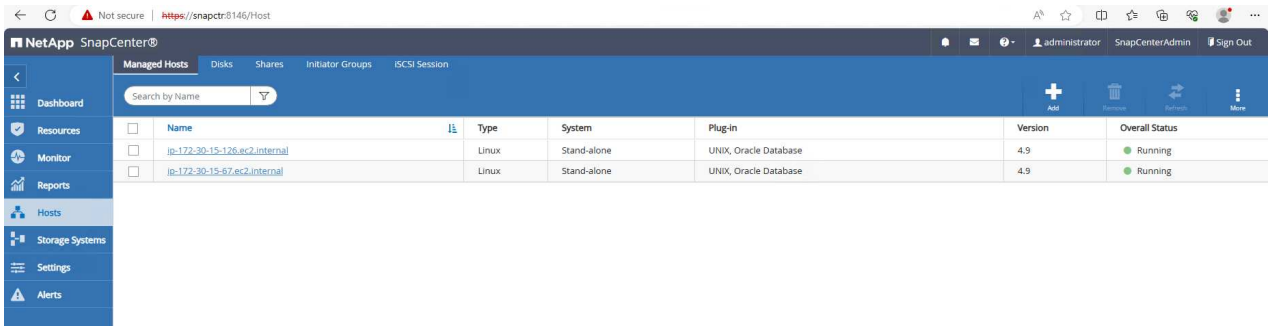
2. Add FSx storage cluster to Storage Systems in SnapCenter with FSx cluster management IP and fsxadmin credential.



3. Add AWS ec2-user to Credential in Settings.

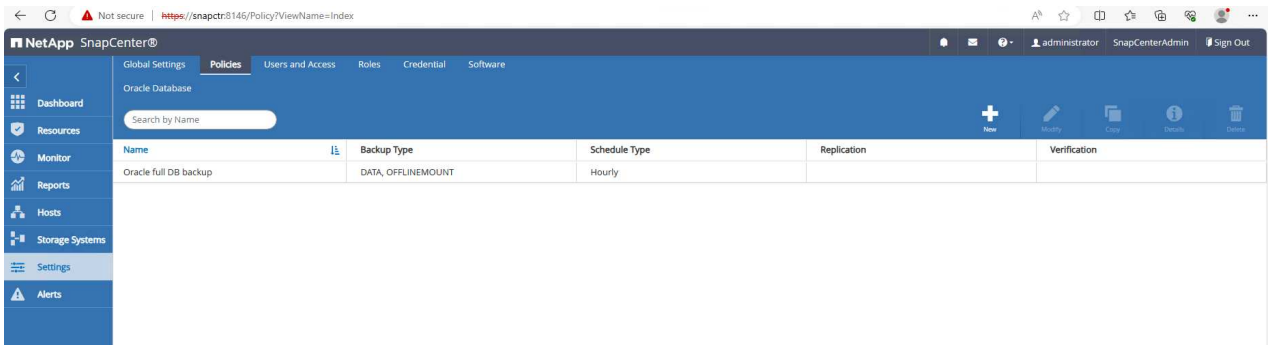


4. Add standby EC2 DB instance and clone EC2 DB instance to **Hosts**.

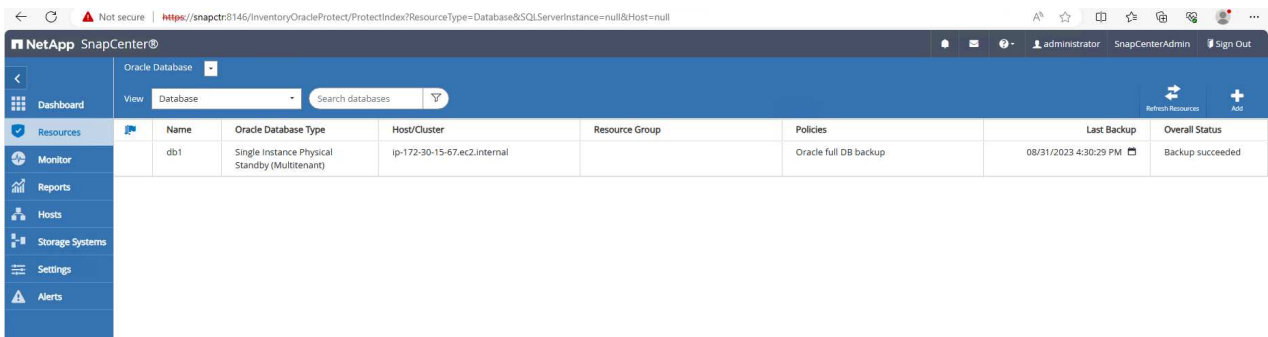


The clone EC2 DB instance should have similar Oracle software stacks installed and configured. In our test case, the grid infrastructure and Oracle 19C installed and configured but no database created.

5. Create a backup policy that is tailored for offline/mount full database backup.



6. Apply backup policy to protect standby database in **Resources** tab.



7. Click on database name to open the database backups page. Select a backup to be used for database clone and click on Clone button to launch clone workflow.

The screenshot shows the NetApp SnapCenter Oracle Database interface. The left sidebar contains navigation icons. The main content area is titled 'db1 Topology'. It includes a 'Manage Copies' section with a '3 Backups' button and a '0 Clones' button. A 'Summary Card' on the right shows '3 Backups', '3 Data Backups', '0 Log Backups', and '0 Clones'. Below this is a 'Primary Backup(s)' section with a search bar and a table of backups.

Backup Name	Count	Type	IF	End Date	Verified	Mounted	RMAN Cataloged	SCN
db1_LA_08-31-2023_17.42.01.6804_0	1	Data		08/31/2023 5:42:29 PM	Unverified	False	Not Cataloged	
db1_LA_08-31-2023_16.30.01.6158_0	1	Data		08/31/2023 4:30:29 PM	Unverified	False	Not Cataloged	
db1_LA_08-31-2023_15.59.09.6092_0	1	Data		08/31/2023 3:59:42 PM	Unverified	False	Not Cataloged	

8. Select Complete Database Clone and name the clone instance SID.

The screenshot shows the 'Clone from db1' dialog box. The 'Name' tab is selected, showing a list of steps: 1 Name, 2 Locations, 3 Credentials, 4 PreOps, 5 PostOps, 6 Notification, and 7 Summary. The 'Complete Database Clone' radio button is selected. The 'Clone SID' field is set to 'db1dev'. The 'Exclude PDBs' field is set to 'Type to find PDBs'. The 'PDB Clone' radio button is unselected. At the bottom right, there are 'Previous' and 'Next' buttons.

9. Select the clone host, which hosts the cloned database from standby DB. Accept the default for data files, control files, and redo logs. Two ASM disk groups will be created on the clone host that are corresponding to the disk groups on standby database.

Clone from db1

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

7 Summary

Select the host to create a clone

Clone host

ip-172-30-15-126.ec2.internal

Datafile locations

+SC_2090922_db1dev

+SC_2342319_db1dev

Reset

Control files

+SC_2090922_db1dev/db1dev/control/control01.ctl

+SC_2090922_db1dev/db1dev/control/control02.ctl

Reset

Redo logs

Group		Size	Unit	Number of files	
RedoGroup 1	<div>×</div>	200	MB	2	<div>+</div>
RedoGroup 2	<div>×</div>	200	MB	2	<div>+</div>
RedoGroup 3	<div>×</div>	200	MB	2	<div>+</div>

Reset

Previous

Next

- No database credentials are needed for OS based authentication. Match Oracle home setting with what is configured on the clone EC2 database instance.

Clone from db1

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

7 Summary

Database Credentials for the clone

Credential name for sys user

None

+

i

ASM instance Credential name

None

+

i

Database port

1521

ASM Port

1521

Oracle Home Settings

i

Oracle Home

/u01/app/oracle/product/19.0.0/dev

Oracle OS User

oracle

Oracle OS Group

oinstall

Previous

Next

11. Change clone database parameters if needed and specify scripts to run before cloen if any.

Clone from db1

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

7 Summary

Specify scripts to run before clone operation

Prescript full path

/var/opt/snapcenter/spl/scripts/

Enter Prescript path

Arguments

Script timeout

60

secs

Database Parameter settings

audit_file_dest	/u01/app/oracle/admin/db1dev_LA/adump	X
audit_trail	DB	X
open_cursors	300	X
pga_aggregate_target	2684354560	X

+

Reset

Previous

Next

12. Enter SQL to run after clone. In the demo, we executed commands to turn off database archive mode for a dev/test/report database.

Clone from db1

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

7 Summary

Until Cancel recovery will be performed for Physical Standby Dataguard/Active Dataguard database.

☒ Create new DBID

☒ Create tempfile for temporary tablespace

☐ Enter SQL queries to apply when clone is created

shutdown immediate ; startup mount ; alter database noarchivelog ; alter database open ;

+

Reset

☐ Enter scripts to run after clone operation

Previous

Next

13. Configure email notification if desired.

Clone from db1

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

7 Summary

Provide email settings

Email preference

Never

From

From email

To

Email to

Subject

Notification

☐ Attach job report

Previous

Next

14. Review the summary, click `Finish` to start the clone.

Clone from db1

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

7 Summary

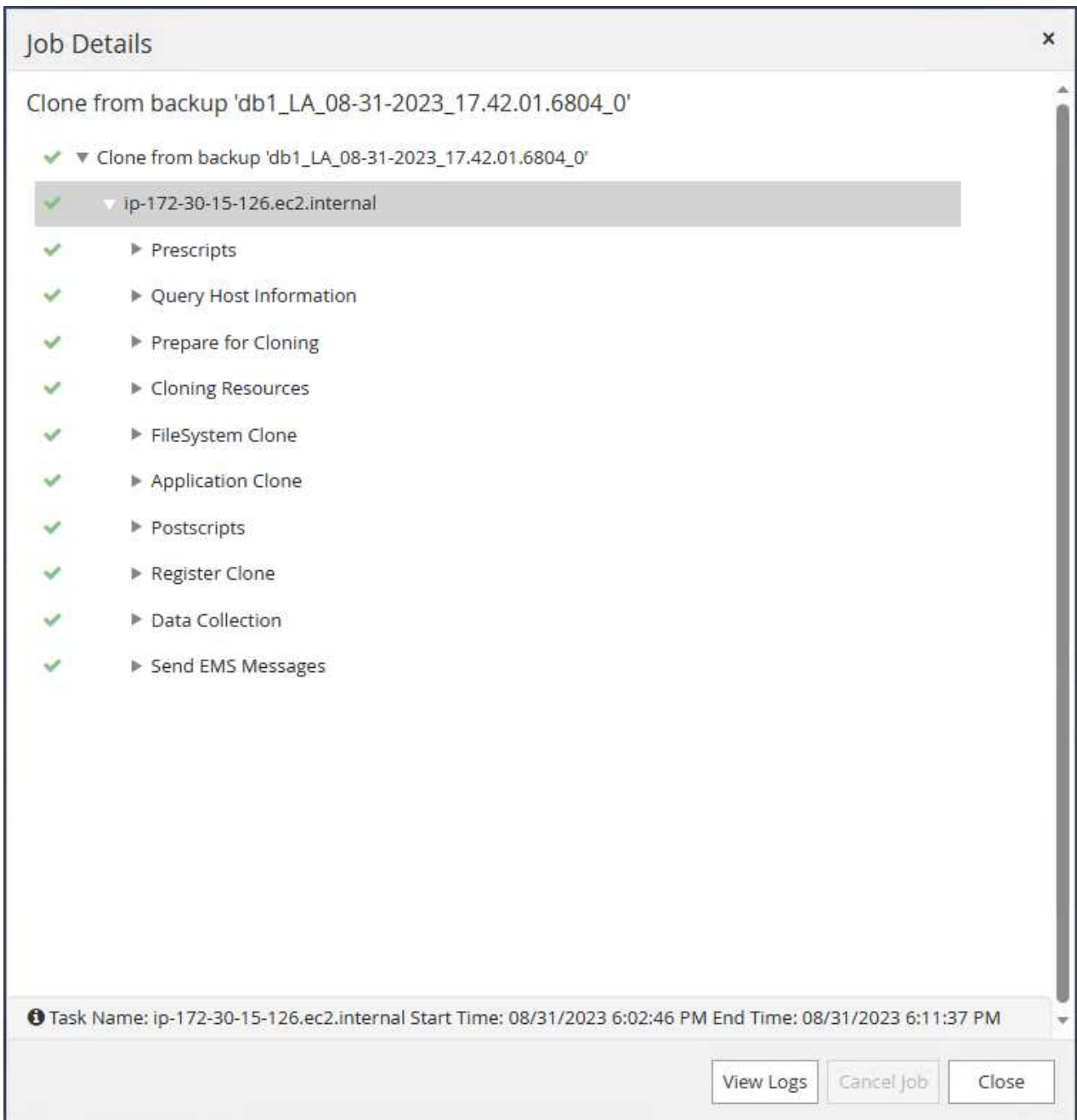
Summary

Clone from backup	db1_LA_08-31-2023_17.42.01.6804_0
Clone SID	db1dev
Clone server	ip-172-30-15-126.ec2.internal
Exclude PDBs	none
Oracle home	/u01/app/oracle/product/19.0.0/dev
Oracle OS user	oracle
Oracle OS group	oinstall
Datafile mountpaths	+SC_2090922_db1dev +SC_2342319_db1dev
Control files	+SC_2090922_db1dev/db1dev/control/control01.ctl +SC_2090922_db1dev/db1dev/control/control02.ctl
Redo groups	RedoGroup =1 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo01_01.log RedoGroup =1 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo01_02.log RedoGroup =2 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo02_01.log RedoGroup =2 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo02_02.log RedoGroup =3 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo03_01.log RedoGroup =3 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo03_02.log RedoGroup =4 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo04_01.log RedoGroup =4 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo04_02.log RedoGroup =5 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo05_01.log RedoGroup =5 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo05_02.log RedoGroup =6 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo06_01.log RedoGroup =6 TotalSize =200 Path =+SC_2090922_db1dev/db1dev/redolog/redo06_02.log

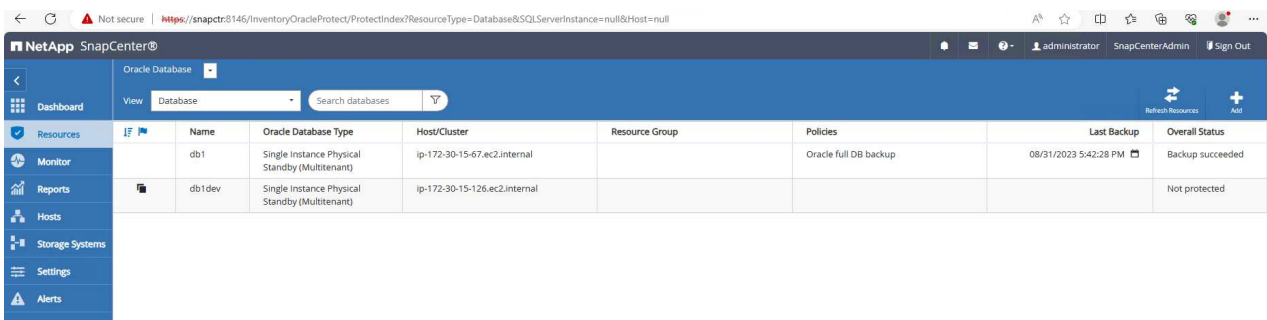
Previous

Finish

- Monitor clone job in Monitor tab. We observed that it took around 8 minutes to clone a database about 300GB in database volume size.



16. Validate the clone database from SnapCenter, which is immediately registered in Resources tab right after clone operation.



17. Query the clone database from clone EC2 instance. We validated that test transaction that occurred in primary database had traversed down to clone database.

```
[oracle@ip-172-30-15-126 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/dev
[oracle@ip-172-30-15-126 ~]$ export ORACLE_SID=db1dev
[oracle@ip-172-30-15-126 ~]$ export PATH=$PATH:$ORACLE_HOME/bin
[oracle@ip-172-30-15-126 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Sep 6 16:41:41 2023
Version 19.18.0.0.0

Copyright (c) 1982, 2022, Oracle. All rights reserved.
```

```
Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.18.0.0.0
```

```
SQL> select name, open_mode, log_mode from v$database;
```

NAME	OPEN_MODE	LOG_MODE
DB1DEV	READ WRITE	NOARCHIVELOG

```
SQL> select instance_name, host_name from v$instance;
```

INSTANCE_NAME	HOST_NAME
db1dev	ip-172-30-15-126.ec2.internal

```
SQL> alter session set container=db1_pdb1;
```

```
Session altered.
```

```
SQL> select * from test;
```

ID	DT	EVENT

```
1
31-AUG-23 04.49.29.000000 PM
a test transaction on primary database db1 and ec2 db host: ip-172-
30-15-45.ec2.
internal

SQL>
```

This completes the clone and validation of a new Oracle database from standby database in Data Guard on FSx storage for DEV, TEST, REPORT or any other use cases. Multiple Oracle databases can be cloned off the same standby database in Data Guard.

Where to find additional information

To learn more about the information described in this document, review the following documents and/or websites:

- Data Guard Concepts and Administration

<https://docs.oracle.com/en/database/oracle/oracle-database/19/sbydb/index.html#Oracle%C2%AE-Data-Guard>

- WP-7357: Oracle Database Deployment on EC2 and FSx Best Practices

[Introduction](#)

- Amazon FSx ONTAP

<https://aws.amazon.com/fsx/netapp-ontap/>

- Amazon EC2

https://aws.amazon.com/pm/ec2/?trk=36c6da98-7b20-48fa-8225-4784bced9843&sc_channel=ps&s_kwcid=AL!4422!3!467723097970!e!!g!!aws%20ec2&ef_id=Cj0KCQiA54KfBhCKARIsAJzSrdqwQrghn6I71jiWzSeaT9Uh1-vY-VfhJixF-xnv5rWwn2S7RqZOTQ0aAh7eEALw_wcB:G:s&s_kwcid=AL!4422!3!467723097970!e!!g!!aws%20ec2

Copyright information

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

Trademark information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.