TR-4981: Kostensenkung für Oracle Active Data Guard mit Amazon FSx ONTAP

NetApp database solutions

NetApp
August 18, 2025

Inhalt

TR-4981: Kostensenkung für Oracle Active Data Guard mit Amazon FSx ONTAP

Allen Cao, Niyaz Mohamed, NetApp

Diese Lösung bietet einen Überblick und Details zur Konfiguration von Oracle Data Guard unter Verwendung von AWS FSx ONTAP als Oracle-Datenbankspeicher am Standby-Standort, um die Lizenz- und Betriebskosten der Oracle Data Guard HA/DR-Lösung in AWS zu senken.

Zweck

Oracle Data Guard gewährleistet hohe Verfügbarkeit, Datenschutz und Notfallwiederherstellung für Unternehmensdaten in einer primären Datenbank- und Standby-Datenbankreplikationskonfiguration. Oracle Active Data Guard ermöglicht Benutzern den Zugriff auf Standby-Datenbanken, während die Datenreplikation von der primären Datenbank zu den Standby-Datenbanken aktiv ist. Data Guard ist eine Funktion der Oracle Database Enterprise Edition. Es ist keine separate Lizenzierung erforderlich. Andererseits ist Active Data Guard eine Option der Oracle Database Enterprise Edition und erfordert daher eine separate Lizenzierung. Mehrere Standby-Datenbanken können im Active Data Guard-Setup Datenreplikationen von einer primären Datenbank empfangen. Allerdings erfordert jede zusätzliche Standby-Datenbank eine Active Data Guard-Lizenz und zusätzlichen Speicherplatz in der Größe der primären Datenbank. Die Betriebskosten summieren sich schnell.

Wenn Sie die Kosten für Ihren Oracle-Datenbankbetrieb senken möchten und planen, einen Active Data Guard in AWS einzurichten, sollten Sie eine Alternative in Betracht ziehen. Verwenden Sie anstelle von Active Data Guard Data Guard, um von der primären Datenbank auf eine einzelne physische Standby-Datenbank im Amazon FSx ONTAP Speicher zu replizieren. Anschließend können mehrere Kopien dieser Standby-Datenbank geklont und für Lese-/Schreibzugriff geöffnet werden, um viele andere Anwendungsfälle wie Berichterstellung, Entwicklung, Test usw. zu erfüllen. Die Nettoergebnisse liefern effektiv die Funktionen von Active Data Guard, während gleichzeitig die Lizenz- und zusätzlichen Speicherkosten für Active Data Guard für jede zusätzliche Standby-Datenbank entfallen. In dieser Dokumentation zeigen wir, wie Sie einen Oracle Data Guard mit Ihrer vorhandenen primären Datenbank in AWS einrichten und eine physische Standby-Datenbank auf dem Amazon FSx ONTAP -Speicher platzieren. Die Standby-Datenbank wird per Snapshot gesichert und für den Lese-/Schreibzugriff für gewünschte Anwendungsfälle geklont.

Diese Lösung ist für die folgenden Anwendungsfälle geeignet:

- Oracle Data Guard zwischen einer primären Datenbank auf einem beliebigen Speicher in AWS und einer Standby-Datenbank auf einem Amazon FSx ONTAP -Speicher.
- Klonen Sie die Standby-Datenbank, während sie für die Datenreplikation geschlossen ist, um Anwendungsfälle wie Berichterstellung, Entwicklung, Test usw. zu unterstützen.

Publikum

Diese Lösung ist für folgende Personen gedacht:

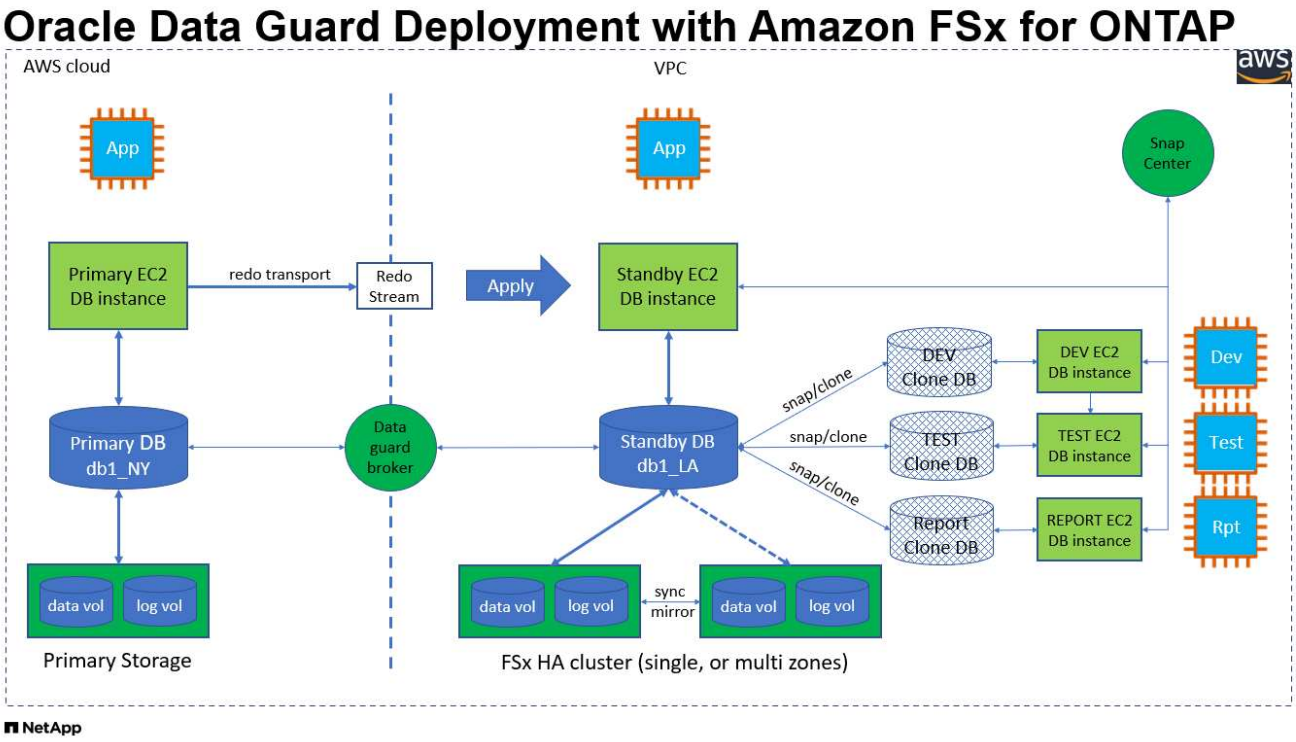
- Ein DBA, der Oracle Active Data Guard in AWS für hohe Verfügbarkeit, Datenschutz und Notfallwiederherstellung eingerichtet hat.
- Ein Datenbanklösungsarchitekt, der an der Konfiguration von Oracle Active Data Guard in der AWS-Cloud interessiert ist.

- Ein Speicheradministrator, der AWS FSx ONTAP Speicher verwaltet, der Oracle Data Guard unterstützt.
- Ein Anwendungsbesitzer, der Oracle Data Guard in einer AWS FSx/EC2-Umgebung einsetzen möchte.

Test- und Validierungsumgebung für Lösungen

Das Testen und Validieren dieser Lösung wurde in einer AWS FSx ONTAP und EC2-Laborumgebung durchgeführt, die möglicherweise nicht der endgültigen Bereitstellungsumgebung entspricht. Weitere Informationen finden Sie im Abschnitt Wichtige Faktoren für die Bereitstellungsüberlegungen .

Architektur



Hardware- und Softwarekomponenten

Hardware		
FSx ONTAP Speicher	Aktuelle von AWS angebotene Version	Ein FSx HA-Cluster in derselben VPC und Verfügbarkeitszone
EC2-Instanz für Compute	t2.xlarge/4vCPU/16G	Drei EC2 T2 xlarge EC2-Instanzen, eine als primärer DB-Server, eine als Standby-DB-Server und die dritte als Klon-DB-Server
Software		
RedHat Linux	RHEL-8.6.0_HVM-20220503-x86_64-2-Hourly2-GP2	RedHat-Abonnement zum Testen bereitgestellt
Oracle Grid-Infrastruktur	Version 19.18	RU-Patch p34762026_190000_Linux-x86-64.zip angewendet

Oracle-Datenbank	Version 19.18	RU-Patch p34765931_190000_Linux-x86-64.zip angewendet
Oracle OPatch	Version 12.2.0.1.36	Neuester Patch p6880880_190000_Linux-x86-64.zip

Oracle Data Guard-Konfiguration mit hypothetischem DR-Setup von NY nach LA

Datenbank	DB_EINDEUTIGER_NAME	Oracle Net-Dienstname
Primär	db1_NY	db1_NY.demo.netapp.com
Physischer Standby	db1_LA	db1_LA.demo.netapp.com

Wichtige Faktoren für die Bereitstellungsüberlegungen

- **So funktioniert die Oracle Standby-Datenbank FlexClone .** AWS FSx ONTAP FlexClone bietet gemeinsam genutzte Kopien derselben Standby-Datenbank-Volumes, die beschreibbar sind. Bei den Kopien der Datenträger handelt es sich tatsächlich um Zeiger, die auf die ursprünglichen Datenblöcke verweisen, bis ein neuer Schreibvorgang auf dem Klon beginnt. ONTAP weist dann neue Speicherblöcke für die neuen Schreibvorgänge zu. Alle Lese-E/As werden von Originaldatenblöcken unter aktiver Replikation bedient. Daher sind die Klone sehr speichereffizient und können für viele andere Anwendungsfälle mit minimaler und inkrementeller neuer Speicherzuweisung für neue Schreib-E/As verwendet werden. Dies ermöglicht enorme Speicherkosteneinsparungen durch eine deutliche Reduzierung des Active Data Guard-Speicherbedarfs. NetApp empfiehlt, die FlexClone -Aktivitäten im Falle einer Datenbankumschaltung vom Primärspeicher auf den Standby-FSx-Speicher zu minimieren, um die Oracle-Leistung auf einem hohen Niveau zu halten.
- **Oracle-Softwareanforderungen.** Im Allgemeinen muss eine physische Standby-Datenbank dieselbe Datenbank-Home-Version wie die primäre Datenbank haben, einschließlich Patch Set Exceptions (PSEs), Critical Patch Updates (CPUs) und Patch Set Updates (PSUs), es sei denn, ein Oracle Data Guard Standby-First Patch Apply-Prozess ist im Gange (wie in My Oracle Support-Hinweis 1265700.1 unter"support.oracle.com"
- **Überlegungen zur Verzeichnisstruktur der Standby-Datenbank.** Wenn möglich, sollten die Datendateien, Protokolldateien und Steuerdateien auf dem Primär- und Standby-System dieselben Namen und Pfadnamen haben und die Namenskonventionen der Optimal Flexible Architecture (OFA) verwenden. Die Archivverzeichnisse in der Standby-Datenbank sollten zwischen den Sites ebenfalls identisch sein, einschließlich Größe und Struktur. Diese Strategie ermöglicht es, dass bei anderen Vorgängen wie Sicherungen, Umschaltungen und Failovers die gleichen Schritte ausgeführt werden, wodurch die Komplexität der Wartung reduziert wird.
- **Protokollierungsmodus erzwingen.** Um sich vor nicht protokollierten direkten Schreibvorgängen in der Primärdatenbank zu schützen, die nicht an die Standbydatenbank weitergegeben werden können, aktivieren Sie FORCE LOGGING in der Primärdatenbank, bevor Sie Datendateisicherungen für die Standby-Erstellung durchführen.
- **Datenbankspeicherverwaltung.** Aus Gründen der betrieblichen Vereinfachung empfiehlt Oracle, dass Sie Oracle Automatic Storage Management (Oracle ASM) und Oracle Managed Files (OMF) in einer Oracle Data Guard-Konfiguration symmetrisch auf der/den primären und Standby-Datenbank(en) einrichten.
- **EC2-Recheninstanzen.** Bei diesen Tests und Validierungen haben wir eine AWS EC2 t2.xlarge-Instanz als Oracle-Datenbank-Compute-Instanz verwendet. NetApp empfiehlt die Verwendung einer EC2-Instanz vom Typ M5 als Compute-Instanz für Oracle in der Produktionsbereitstellung, da diese für die Datenbank-

Workload optimiert ist. Sie müssen die EC2-Instanz entsprechend der Anzahl der vCPUs und der RAM-Menge entsprechend den tatsächlichen Arbeitslastanforderungen dimensionieren.

- **FSx-Speicher-HA-Cluster, Bereitstellung in einer oder mehreren Zonen.** Bei diesen Tests und Validierungen haben wir einen FSx HA-Cluster in einer einzelnen AWS-Verfügbarkeitszone bereitgestellt. Für die Produktionsbereitstellung empfiehlt NetApp die Bereitstellung eines FSx HA-Paares in zwei verschiedenen Verfügbarkeitszonen. Ein FSx-Cluster wird immer in einem HA-Paar bereitgestellt, das in einem Paar aktiv-passiver Dateisysteme synchron gespiegelt wird, um Redundanz auf Speicherebene bereitzustellen. Durch die Bereitstellung in mehreren Zonen wird die Hochverfügbarkeit im Falle eines Ausfalls in einer einzelnen AWS-Zone weiter verbessert.
- **Größenbestimmung des FSx-Speicherclusters.** Ein Amazon FSx ONTAP Speicherdateisystem bietet bis zu 160.000 rohe SSD-IOPS, bis zu 4 GBps Durchsatz und eine maximale Kapazität von 192 TiB. Sie können die Größe des Clusters jedoch hinsichtlich der bereitgestellten IOPS, des Durchsatzes und des Speicherlimits (mindestens 1.024 GiB) basierend auf Ihren tatsächlichen Anforderungen zum Zeitpunkt der Bereitstellung anpassen. Die Kapazität kann dynamisch im laufenden Betrieb angepasst werden, ohne die Anwendungsverfügbarkeit zu beeinträchtigen.

Lösungsbereitstellung

Es wird davon ausgegangen, dass Sie Ihre primäre Oracle-Datenbank bereits in einer AWS EC2-Umgebung innerhalb einer VPC als Ausgangspunkt für die Einrichtung von Data Guard bereitgestellt haben. Die primäre Datenbank wird mit Oracle ASM zur Speicherverwaltung bereitgestellt. Für Oracle-Datendateien, Protokolldateien und Steuerdateien usw. werden zwei ASM-Datenträgergruppen erstellt – +DATA und +LOGS. Weitere Informationen zur Oracle-Bereitstellung in AWS mit ASM finden Sie in den folgenden technischen Berichten.

- "Best Practices für die Oracle-Datenbankbereitstellung auf EC2 und FSx"
- "Bereitstellung und Schutz von Oracle-Datenbanken in AWS FSx/EC2 mit iSCSI/ASM"
- "Oracle 19c im Standalone-Neustart auf AWS FSx/EC2 mit NFS/ASM"

Ihre primäre Oracle-Datenbank kann entweder auf einem FSx ONTAP oder einem anderen Speicher Ihrer Wahl innerhalb des AWS EC2-Ökosystems ausgeführt werden. Der folgende Abschnitt enthält schrittweise Bereitstellungsverfahren zum Einrichten von Oracle Data Guard zwischen einer primären EC2-DB-Instance mit ASM-Speicher und einer Standby-EC2-DB-Instance mit ASM-Speicher.

Voraussetzungen für die Bereitstellung

Für die Bereitstellung sind die folgenden Voraussetzungen erforderlich.

1. Ein AWS-Konto wurde eingerichtet und die erforderlichen VPC- und Netzwerksegmente wurden innerhalb Ihres AWS-Kontos erstellt.
2. Von der AWS EC2-Konsole aus müssen Sie mindestens drei EC2-Linux-Instanzen bereitstellen, eine als primäre Oracle-DB-Instanz, eine als Standby-Oracle-DB-Instanz und eine geklonte Ziel-DB-Instanz für Berichterstellung, Entwicklung und Tests usw. Weitere Einzelheiten zur Einrichtung der Umgebung finden Sie im Architekturdiagramm im vorherigen Abschnitt. Überprüfen Sie auch die AWS"Benutzerhandbuch für Linux-Instanzen" für weitere Informationen.
3. Stellen Sie über die AWS EC2-Konsole Amazon FSx ONTAP Storage HA-Cluster bereit, um Oracle-Volumes zu hosten, auf denen die Oracle-Standby-Datenbank gespeichert ist. Wenn Sie mit der Bereitstellung von FSx-Speicher nicht vertraut sind, lesen Sie die Dokumentation"Erstellen von FSx ONTAP Dateisystemen" für schrittweise Anleitungen.
4. Die Schritte 2 und 3 können mit dem folgenden Terraform-Automatisierungstoolkit durchgeführt werden, das eine EC2-Instanz namens `ora_01` und ein FSx-Dateisystem namens `fsx_01` . Lesen Sie die Anweisung sorgfältig durch und ändern Sie die Variablen vor der Ausführung entsprechend Ihrer Umgebung. Die Vorlage lässt sich problemlos an Ihre eigenen Einsatzanforderungen anpassen.

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

Stellen Sie sicher, dass Sie im Stammvolume der EC2-Instanz mindestens 50 GB zugewiesen haben, um ausreichend Speicherplatz für die Bereitstellung der Oracle-Installationsdateien zu haben.

Vorbereiten der primären Datenbank für Data Guard

In dieser Demonstration haben wir eine primäre Oracle-Datenbank namens db1 auf der primären EC2-DB-Instance mit zwei ASM-Datenträgergruppen in einer eigenständigen Neustartkonfiguration mit Datendateien in der ASM-Datenträgergruppe +DATA und einem Flash-Wiederherstellungsbereich in der ASM-Datenträgergruppe +LOGS eingerichtet. Im Folgenden werden die detaillierten Verfahren zum Einrichten der primären Datenbank für Data Guard veranschaulicht. Alle Schritte sollten als Datenbankbesitzer – Oracle-Benutzer – ausgeführt werden.

1. Primäre Datenbank db1-Konfiguration auf der primären EC2-DB-Instance IP-172-30-15-45. Die ASM-Datenträgergruppen können sich auf jedem Speichertyp innerhalb des EC2-Ökosystems befinden.

```
[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.db1.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. Aktivieren Sie in SQLPlus die erzwungene Protokollierung auf dem Primärserver.

```
alter database force logging;
```

3. Aktivieren Sie von SQLPlus aus Flashback auf dem Primärserver. Flashback ermöglicht die einfache Wiederherstellung der primären Datenbank als Standby nach einem Failover.

```
alter database flashback on;
```

4. Konfigurieren Sie die Redo-Transportauthentifizierung mithilfe der Oracle-Kennwortdatei. Erstellen Sie eine Kennwortdatei auf dem Primärserver mithilfe des Dienstprogramms orapwd, falls diese nicht festgelegt ist, und kopieren Sie sie in das Verzeichnis $ORACLE_HOME/dbs der Standby-Datenbank.
5. Erstellen Sie Standby-Redo-Protokolle auf der primären Datenbank mit derselben Größe wie die aktuelle Online-Protokolldatei. Protokollgruppen sind eine Gruppe mehr als Online-Protokolldateigruppen. Die primäre Datenbank kann dann schnell in die Standby-Rolle wechseln und bei Bedarf mit dem Empfang von Redo-Daten beginnen.

```
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. Erstellen Sie aus SQLPlus eine P-Datei aus der SP-Datei zur Bearbeitung.

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

7. Überarbeiten Sie die P-Datei und fügen Sie die folgenden Parameter hinzu.

```
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. Erstellen Sie von sqlplus aus eine SP-Datei im ASM +DATA-Verzeichnis aus der überarbeiteten P-Datei im Verzeichnis /home/oracle.

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

9. Suchen Sie die neu erstellte SP-Datei unter der Datenträgergruppe +DATA (ggf. mit dem Dienstprogramm asmcmd). Verwenden Sie srvctl, um das Raster so zu ändern, dass die Datenbank aus einer neuen SP-Datei gestartet wird, wie unten gezeigt.

```
[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. Ändern Sie tnsnames.ora, um db_unique_name für die Namensauflösung hinzuzufügen.

```
# 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. Fügen Sie der Datei listener.ora den Data Guard-Dienstnamen db1_NY_DGMGRL.demo.netapp für die primäre Datenbank hinzu.

```
#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. Fahren Sie die Datenbank mit srvctl herunter und starten Sie sie neu. Überprüfen Sie, ob die Data Guard-Parameter jetzt aktiv sind.

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

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

Damit ist die Einrichtung der primären Datenbank für Data Guard abgeschlossen.

Standby-Datenbank vorbereiten und Data Guard aktivieren

Oracle Data Guard erfordert eine Betriebssystemkernelkonfiguration und Oracle-Software-Stacks einschließlich Patch-Sets auf der Standby-EC2-DB-Instance, um mit der primären EC2-DB-Instance übereinzustimmen. Zur Vereinfachung der Verwaltung und Vereinfachung sollte die Datenbankspeicherkonfiguration der Standby-EC2-DB-Instance idealerweise auch mit der primären EC2-DB-Instance übereinstimmen, beispielsweise hinsichtlich Name, Anzahl und Größe der ASM-Datenträgergruppen. Im Folgenden finden Sie detaillierte Verfahren zum Einrichten der Standby-EC2-DB-Instance für Data Guard. Alle Befehle sollten mit der Benutzer-ID des Oracle-Eigentümers ausgeführt werden.

1. Überprüfen Sie zunächst die Konfiguration der primären Datenbank auf der primären EC2-Instanz. In dieser Demonstration haben wir eine primäre Oracle-Datenbank namens db1 auf der primären EC2-DB-Instance mit zwei ASM-Datenträgergruppen +DATA und +LOGS in einer eigenständigen Neustartkonfiguration eingerichtet. Die primären ASM-Datenträgergruppen können sich auf jedem Speichertyp innerhalb des EC2-Ökosystems befinden.
2. Befolgen Sie die Verfahren in der Dokumentation"TR-4965: Bereitstellung und Schutz von Oracle-Datenbanken in AWS FSx/EC2 mit iSCSI/ASM" um Grid und Oracle auf der Standby-EC2-DB-Instance zu installieren und zu konfigurieren, damit sie mit der primären Datenbank übereinstimmen. Der Datenbankspeicher sollte bereitgestellt und der Standby-EC2-DB-Instance von FSx ONTAP mit derselben Speicherkapazität wie die primäre EC2-DB-Instance zugewiesen werden.

   Stoppen Sie bei Schritt 10 in `Oracle database installation` Abschnitt. Die Standby-Datenbank wird mithilfe der Datenbankduplizierungsfunktion von dbca aus der Primärdatenbank instanziiert.

3. Sobald die Oracle-Software installiert und konfiguriert ist, kopieren Sie aus dem Standby-DB-Verzeichnis $ORACLE_HOME das Oracle-Passwort aus der primären Datenbank.

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

4. Erstellen Sie die Datei tnsnames.ora mit den folgenden Einträgen.

```
# 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. Fügen Sie der Datei listener.ora den Namen des DB-Data-Guard-Dienstes hinzu.

```
#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. Legen Sie Oracle Home und Pfad fest.

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

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

7. Verwenden Sie dbca, um eine Standby-Datenbank aus der primären Datenbank db1 zu instanziieren.

```
[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. Validieren Sie die duplizierte Standby-Datenbank. Neu duplizierte Standby-Datenbank wird zunächst im NUR-LESEN-Modus geöffnet.

```
[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
```

9. Starten Sie die Standby-Datenbank neu in `mount` Führen Sie die Phase aus und führen Sie den folgenden Befehl aus, um die verwaltete Wiederherstellung der Standby-Datenbank zu aktivieren.

```
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. Überprüfen Sie den Wiederherstellungsstatus der Standby-Datenbank. Beachten Sie die `recovery logmerger` In `APPLYING_LOG` Aktion.

```
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>
```

Damit ist die Einrichtung des Data Guard-Schutzes für db1 vom Primär- zum Standby-System mit aktivierter verwalteter Standby-Wiederherstellung abgeschlossen.

Data Guard Broker einrichten

Oracle Data Guard Broker ist ein verteiltes Verwaltungsframework, das die Erstellung, Wartung und Überwachung von Oracle Data Guard-Konfigurationen automatisiert und zentralisiert. Der folgende Abschnitt zeigt, wie Sie Data Guard Broker einrichten, um die Data Guard-Umgebung zu verwalten.

1. Starten Sie den Data Guard Broker sowohl auf der primären als auch auf der Standby-Datenbank mit dem folgenden Befehl über SQLPlus.

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

2. Stellen Sie von der primären Datenbank aus als SYSDBA eine Verbindung zum Data Guard Borker her.

```
[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. Erstellen und aktivieren Sie die Data Guard Broker-Konfiguration.

```
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. Überprüfen Sie den Datenbankstatus im Data Guard Broker-Verwaltungsframework.

```
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>
```

Im Falle eines Fehlers kann Data Guard Broker verwendet werden, um ein sofortiges Failover der primären Datenbank auf den Standby-Modus durchzuführen.

Klonen Sie die Standby-Datenbank für andere Anwendungsfälle

Der Hauptvorteil der Staging-Standby-Datenbank auf AWS FSx ONTAP in Data Guard besteht darin, dass sie mit FlexCloned viele andere Anwendungsfälle mit minimaler zusätzlicher Speicherinvestition bedienen kann. Im folgenden Abschnitt zeigen wir, wie Sie mit dem NetApp SnapCenter -Tool einen Snapshot der gemounteten und wiederherzustellenden Standby-Datenbankvolumes auf FSx ONTAP für andere Zwecke wie DEV, TEST, REPORT usw. erstellen und klonen.

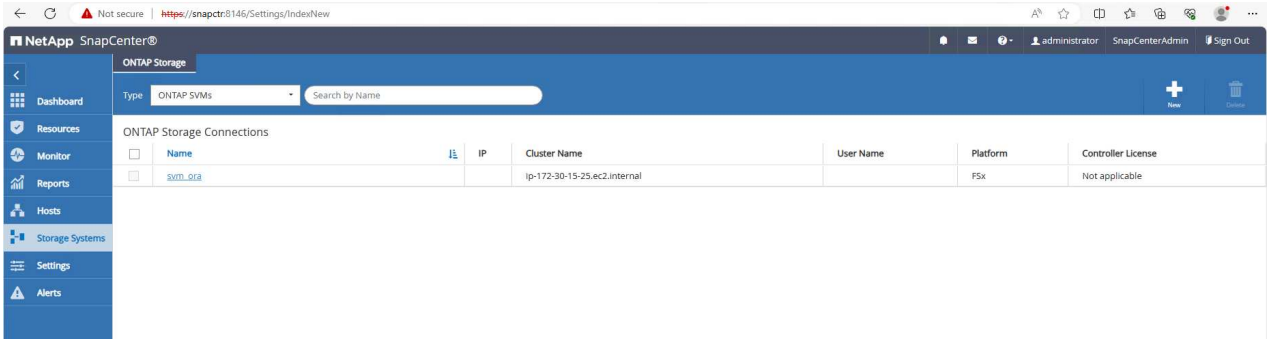
Im Folgenden finden Sie allgemeine Verfahren zum Klonen einer LESE-/SCHREIB-Datenbank aus der verwalteten physischen Standby-Datenbank in Data Guard mithilfe von SnapCenter. Ausführliche Anweisungen zum Einrichten und Konfigurieren von SnapCenter finden Sie unter"Hybrid Cloud-Datenbanklösungen mit SnapCenter" relevante Oracle-Abschnitte.

1. Wir beginnen mit der Erstellung einer Testtabelle und dem Einfügen einer Zeile in die Testtabelle in der Primärdatenbank. Wir werden dann überprüfen, ob die Transaktion zum Standby und schließlich zum Klon durchläuft.

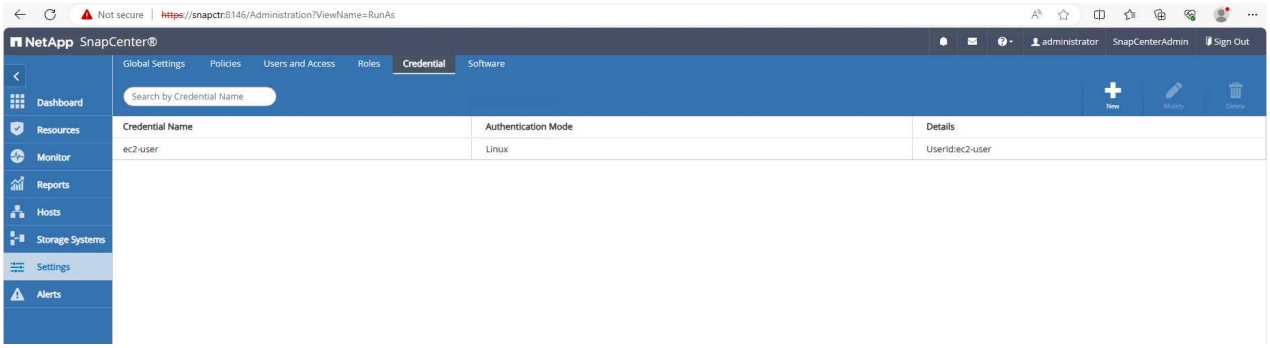
```
[oracle@ip-172-30-15-45 db1]$ sqlplus / as sysdba

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
```

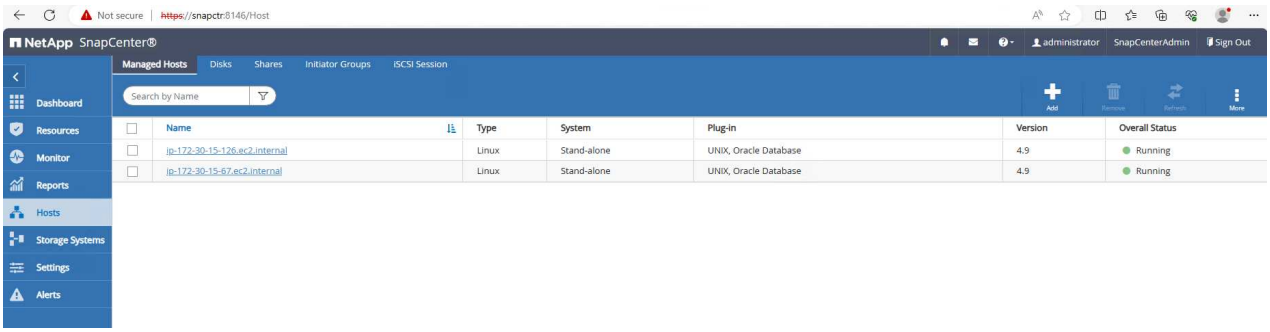
2. FSx-Speichercluster hinzufügen zu `Storage Systems` in SnapCenter mit FSx-Cluster-Management-IP und fsxadmin-Anmeldeinformationen.



3. Fügen Sie AWS ec2-user hinzu zu `Credential` In `Settings` .

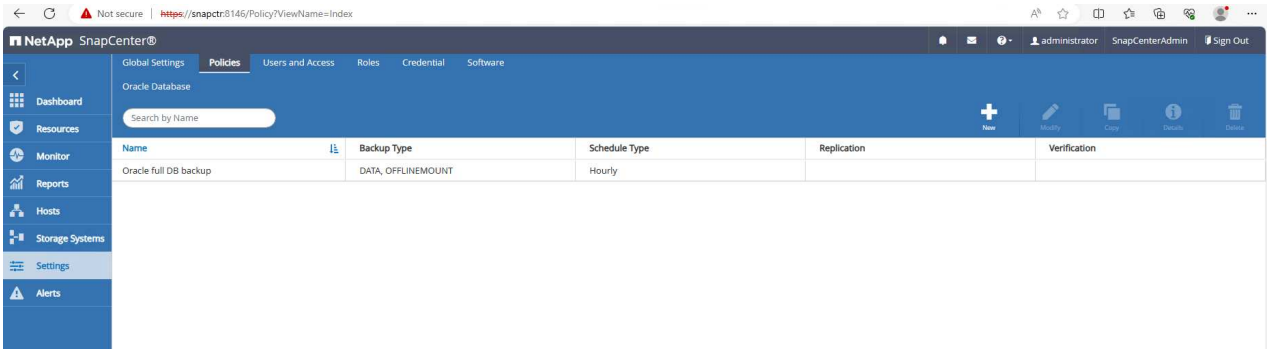


4. Fügen Sie eine Standby-EC2-DB-Instance hinzu und klonen Sie die EC2-DB-Instance in `Hosts` .

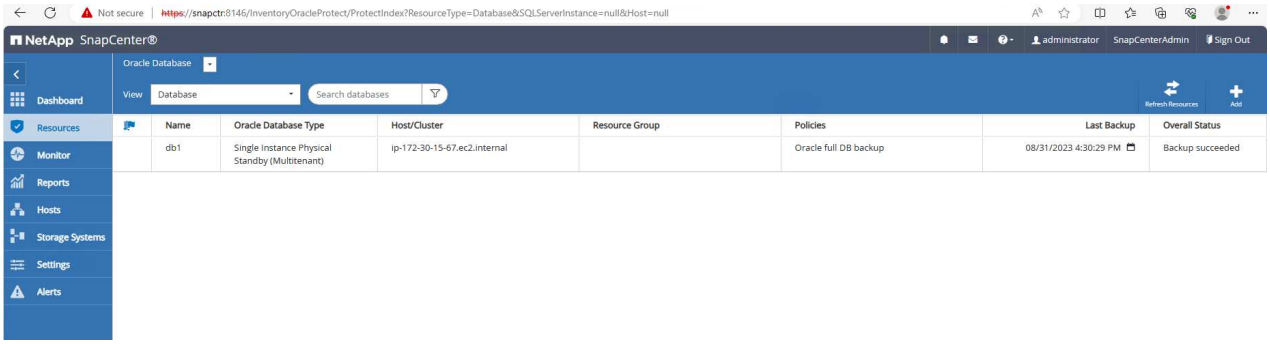


Auf der geklonten EC2-DB-Instance sollten ähnliche Oracle-Software-Stacks installiert und konfiguriert sein. In unserem Testfall wurden die Grid-Infrastruktur und Oracle 19C installiert und konfiguriert, aber keine Datenbank erstellt.

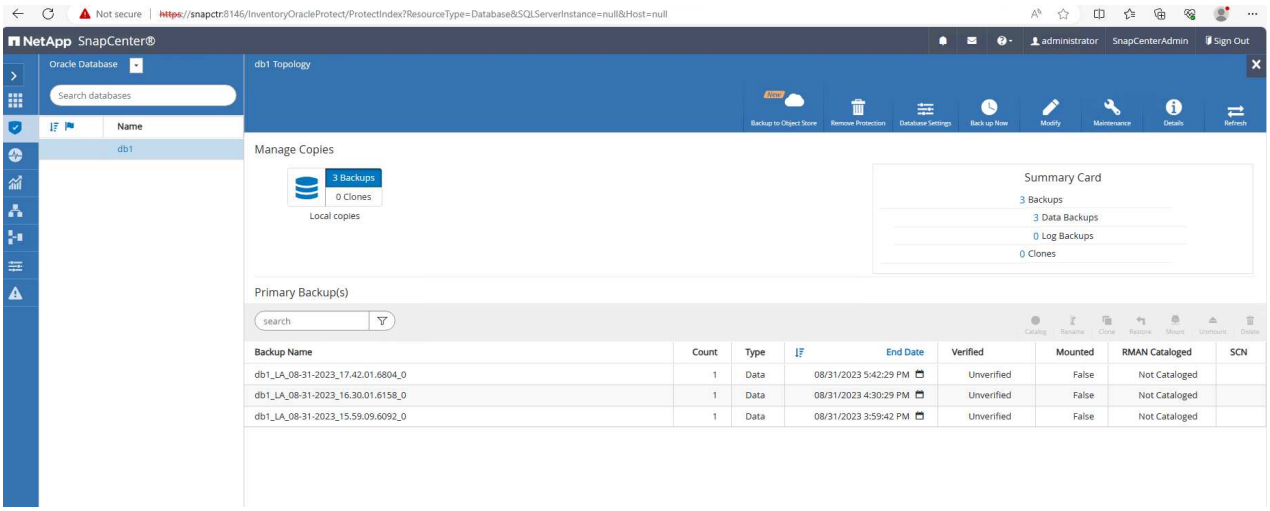
5. Erstellen Sie eine Sicherungsrichtlinie, die auf die Offline-/Mount-Volldatenbanksicherung zugeschnitten ist.



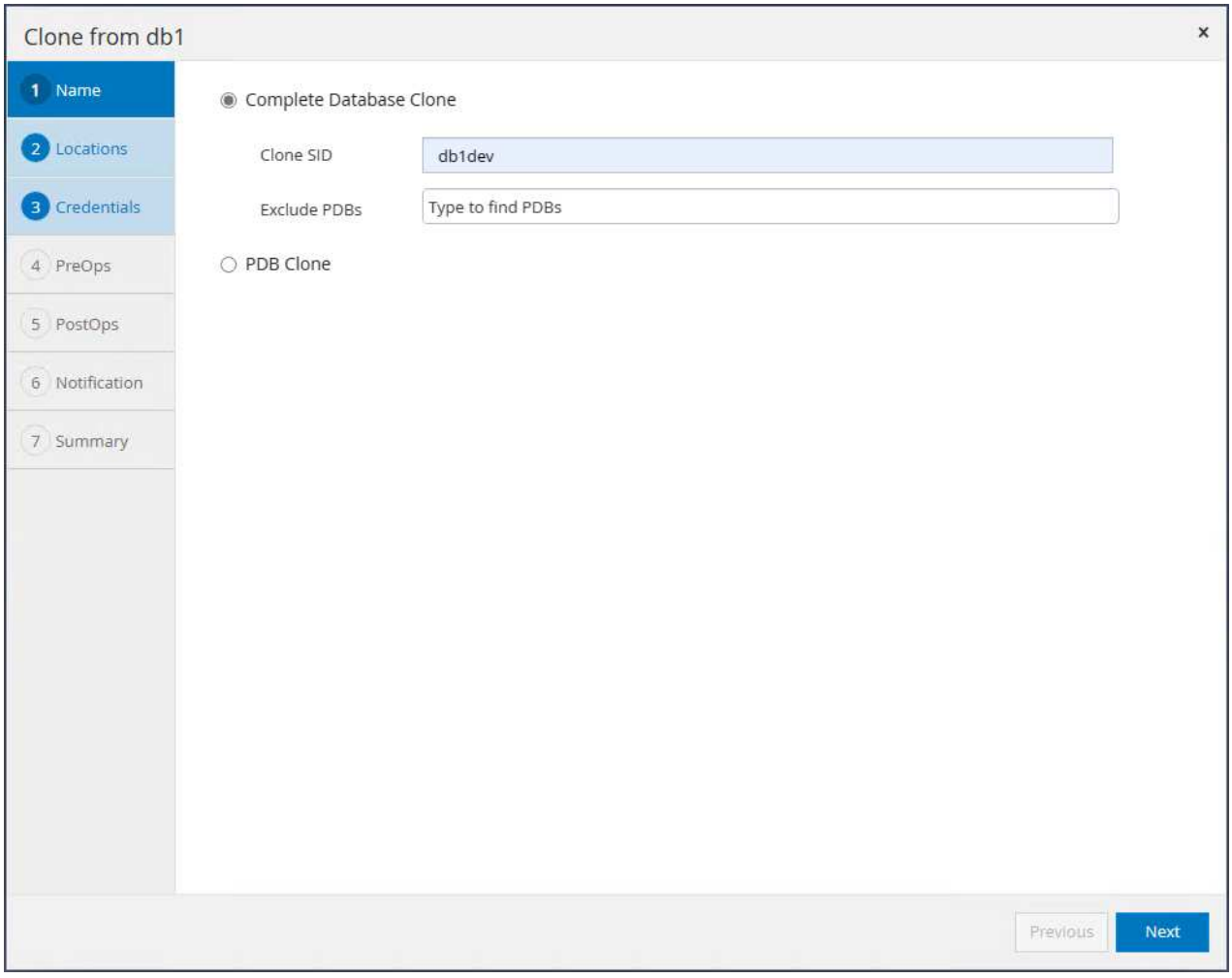
6. Wenden Sie die Sicherungsrichtlinie an, um die Standby-Datenbank zu schützen in `Resources` Tab.



7. Klicken Sie auf den Datenbanknamen, um die Seite mit den Datenbanksicherungen zu öffnen. Wählen Sie eine Sicherung aus, die zum Klonen der Datenbank verwendet werden soll, und klicken Sie auf `Clone` Schaltfläche, um den Klon-Workflow zu starten.

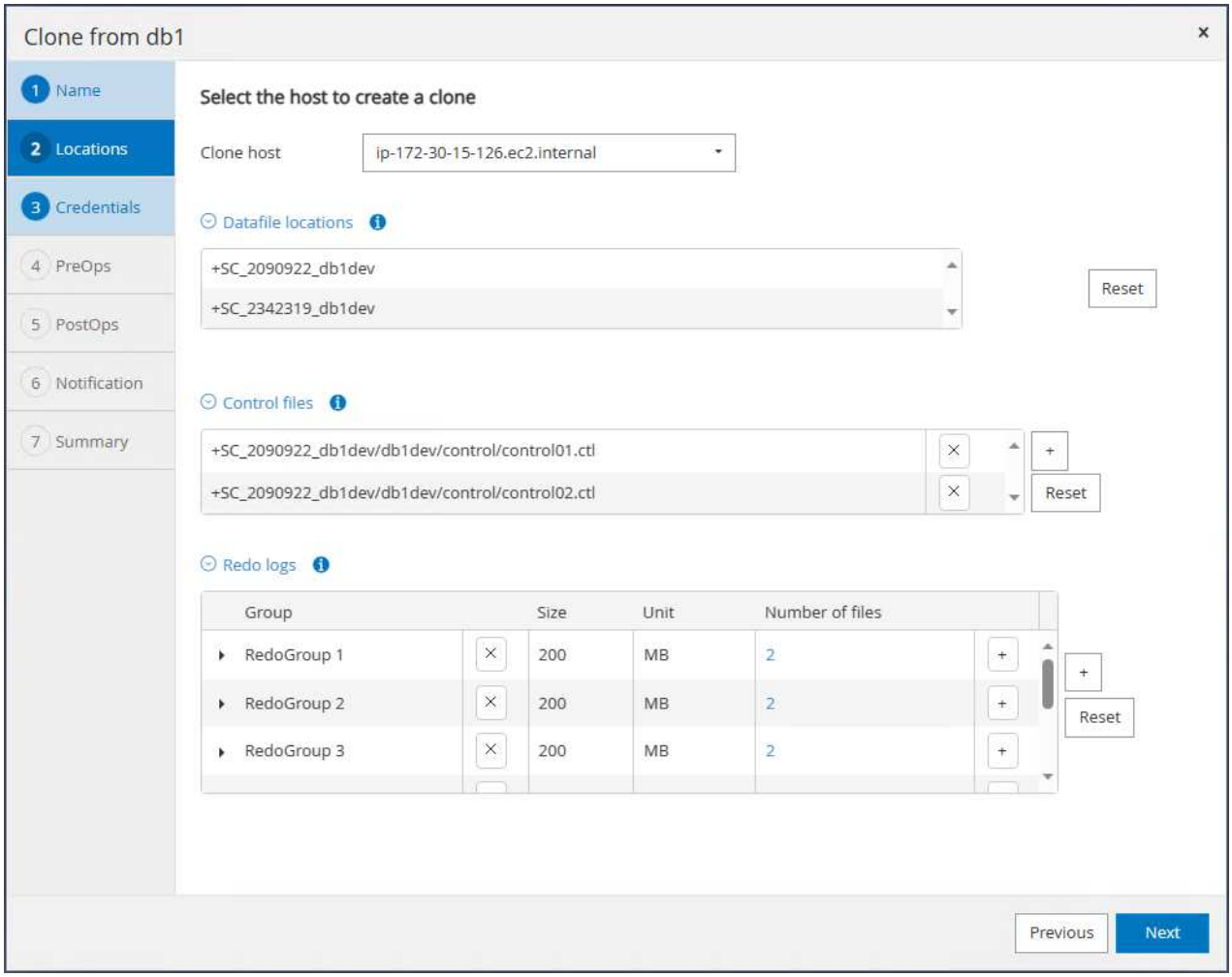


8. Wählen `Complete Database Clone` und benennen Sie die SID der Kloninstanz.

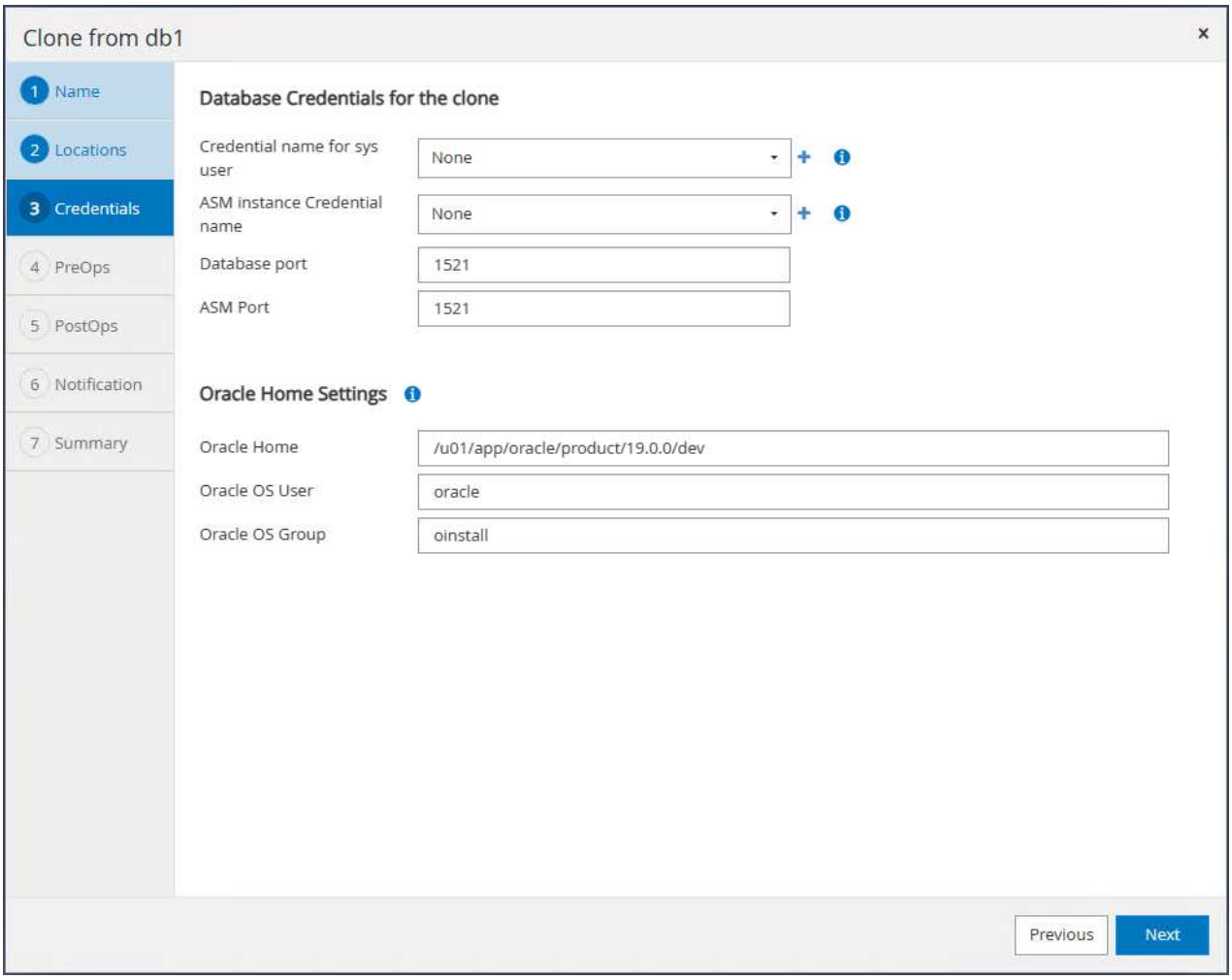


9. Wählen Sie den Klonhost aus, der die geklonte Datenbank aus der Standby-DB hostet. Akzeptieren Sie die Standardeinstellungen für Datendateien, Steuerdateien und Redo-Protokolle. Auf dem

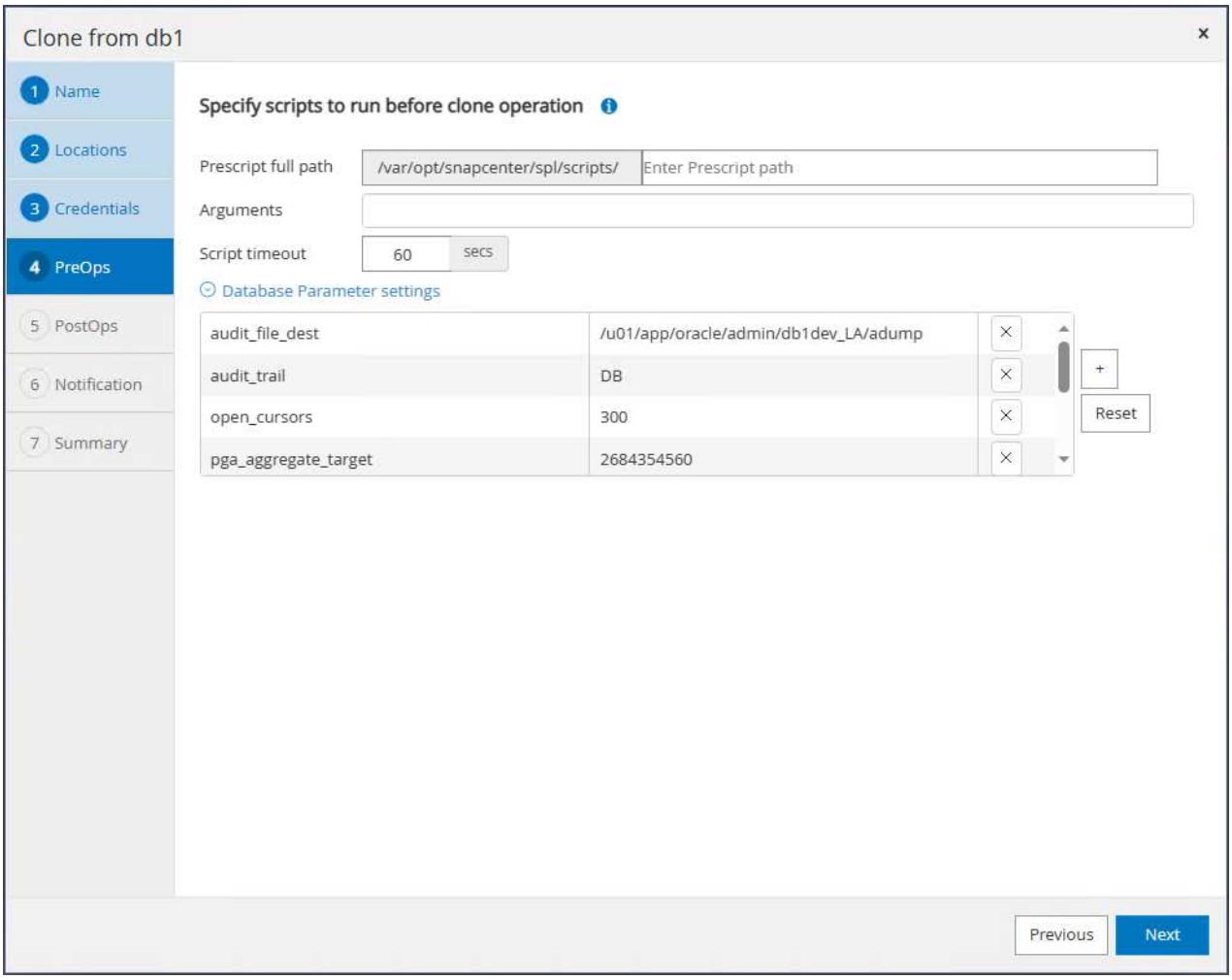
Klonhost werden zwei ASM-Datenträgergruppen erstellt, die den Datenträgergruppen der Standby-Datenbank entsprechen.



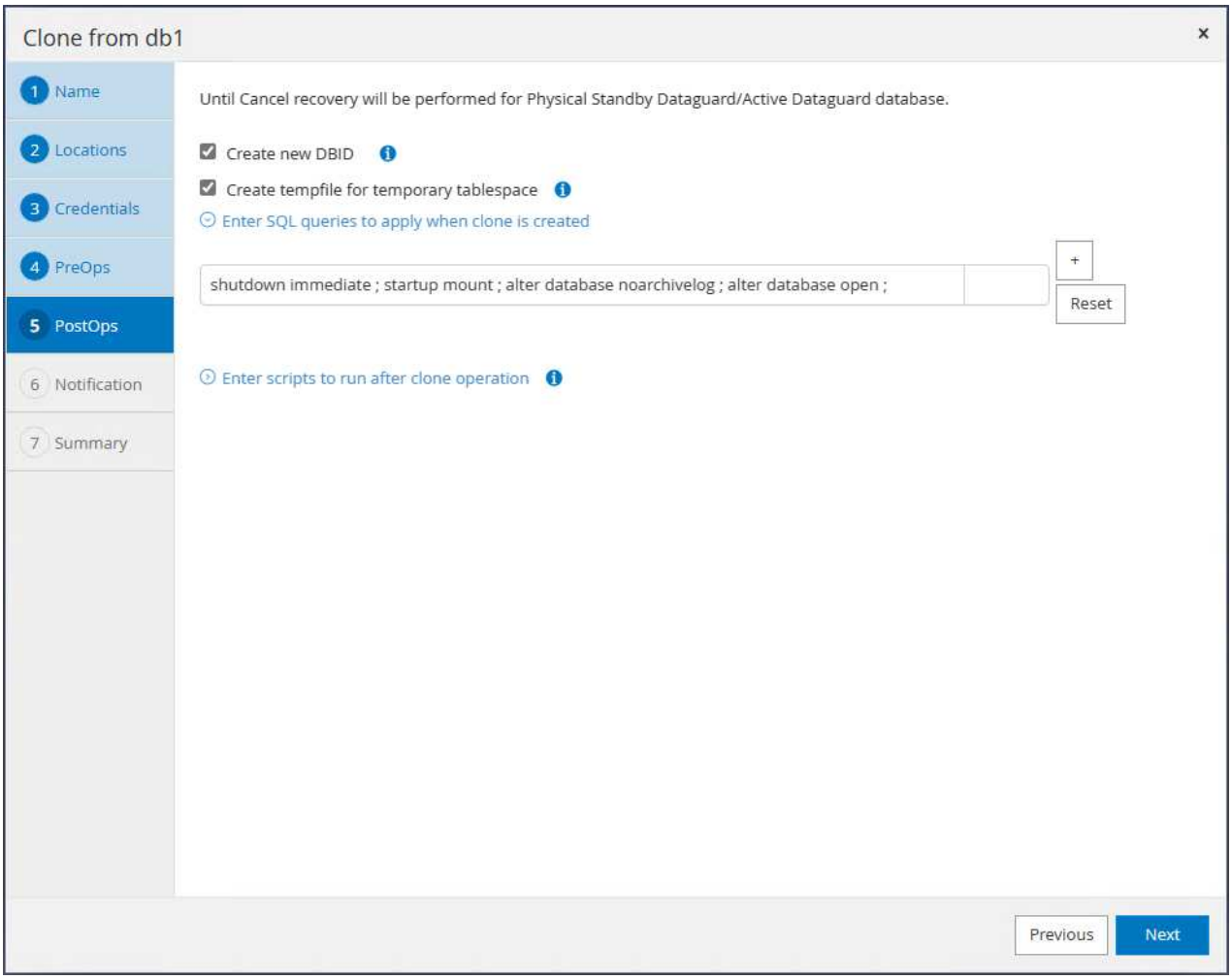
10. Für die betriebssystembasierte Authentifizierung sind keine Datenbankanmeldeinformationen erforderlich. Passen Sie die Oracle-Home-Einstellung an die Konfiguration auf der geklonten EC2-Datenbankinstanz an.



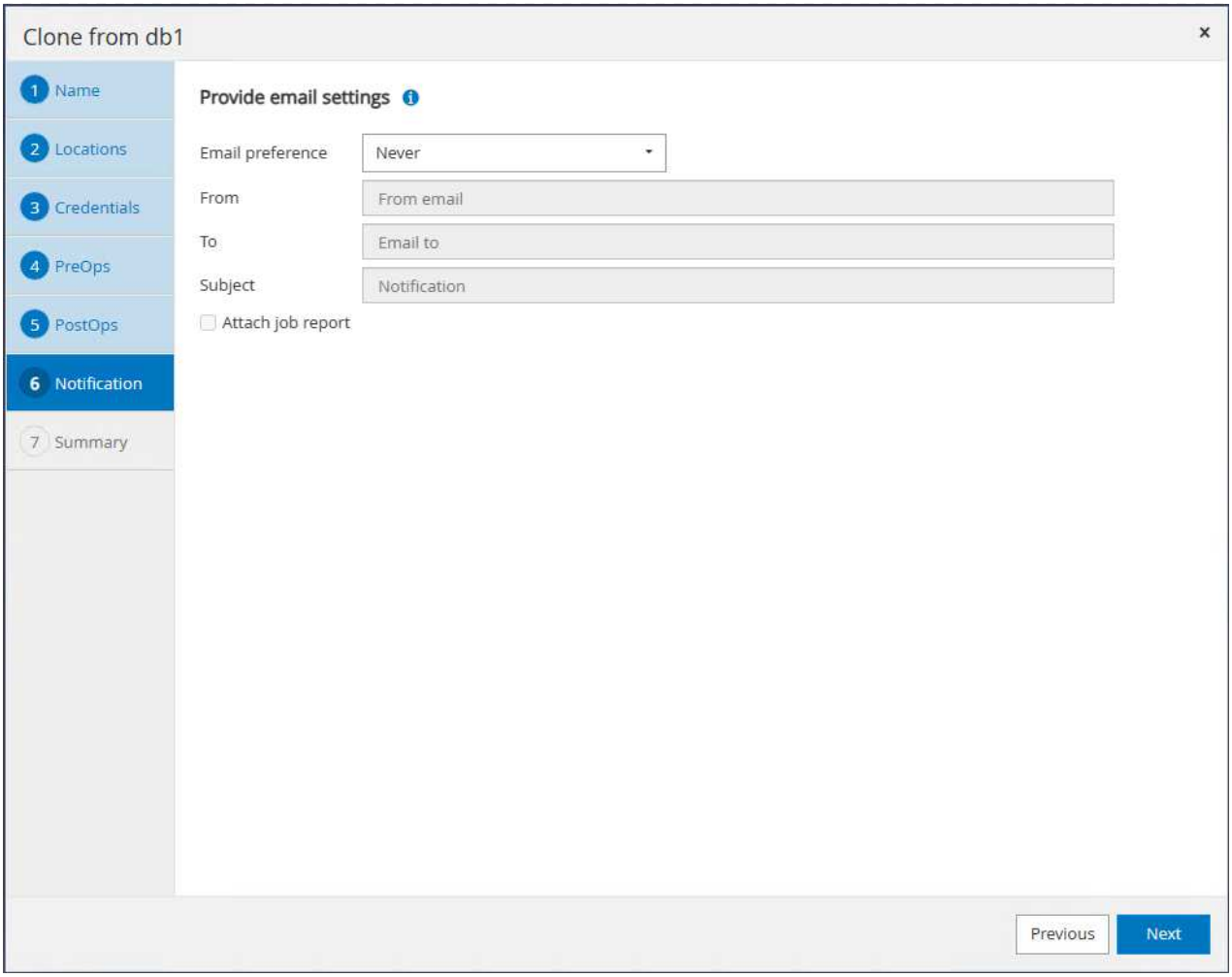
11. Ändern Sie bei Bedarf die Parameter der Klondatenbank und geben Sie ggf. vor dem Klonen auszuführende Skripts an.



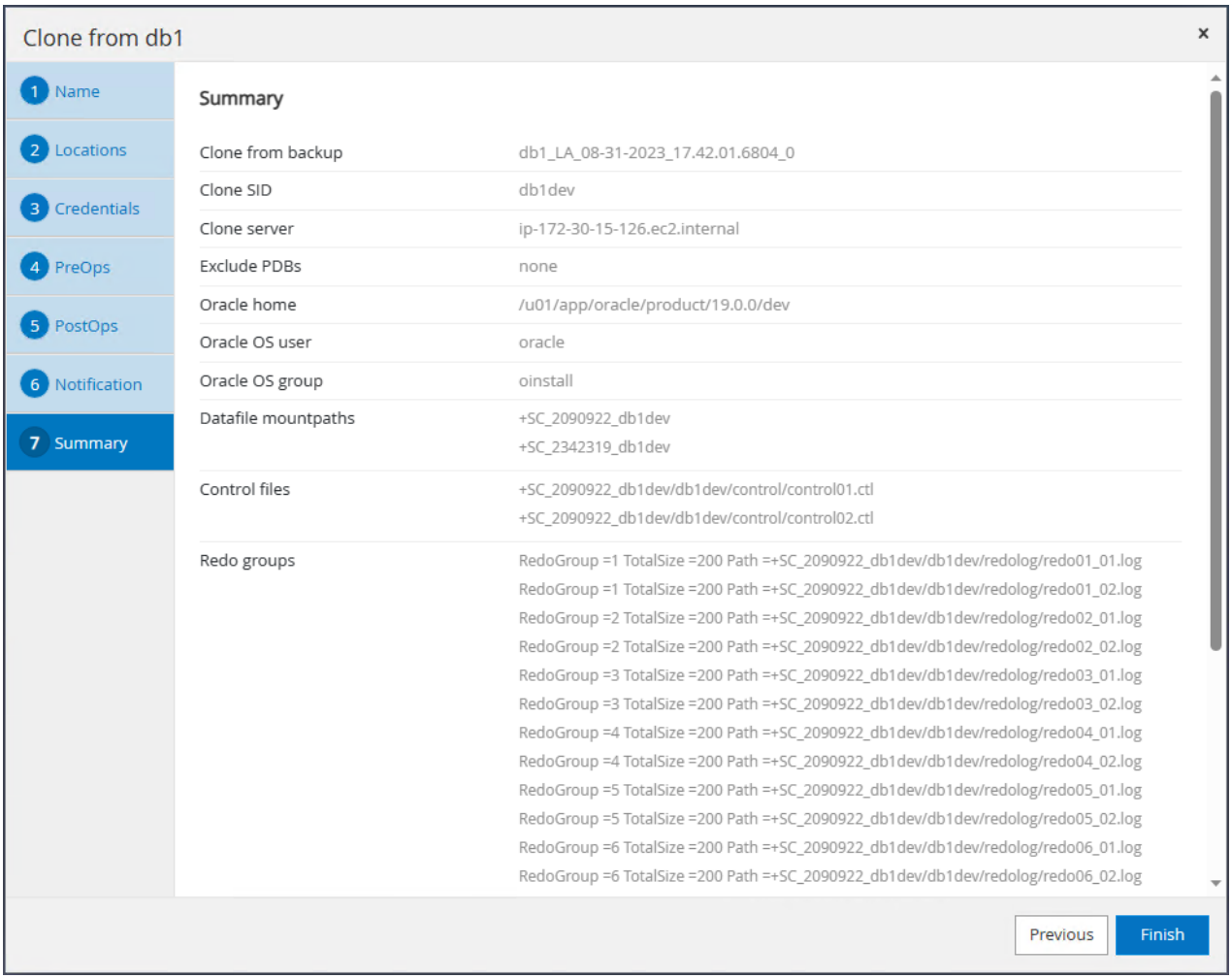
12. Geben Sie SQL ein, das nach dem Klonen ausgeführt werden soll. In der Demo haben wir Befehle ausgeführt, um den Datenbankarchivmodus für eine Dev/Test/Report-Datenbank zu deaktivieren.



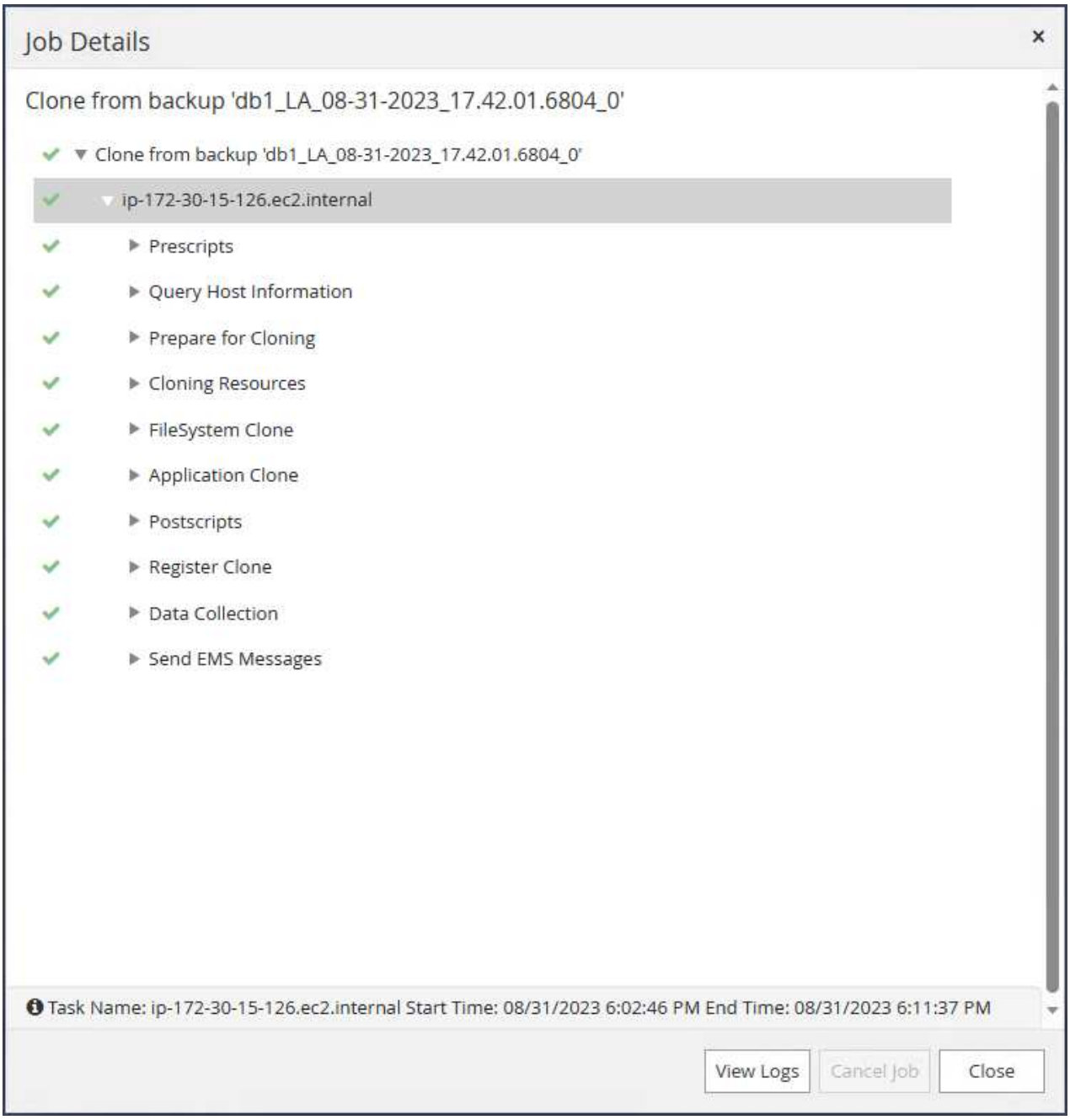
13. Konfigurieren Sie bei Bedarf die E-Mail-Benachrichtigung.



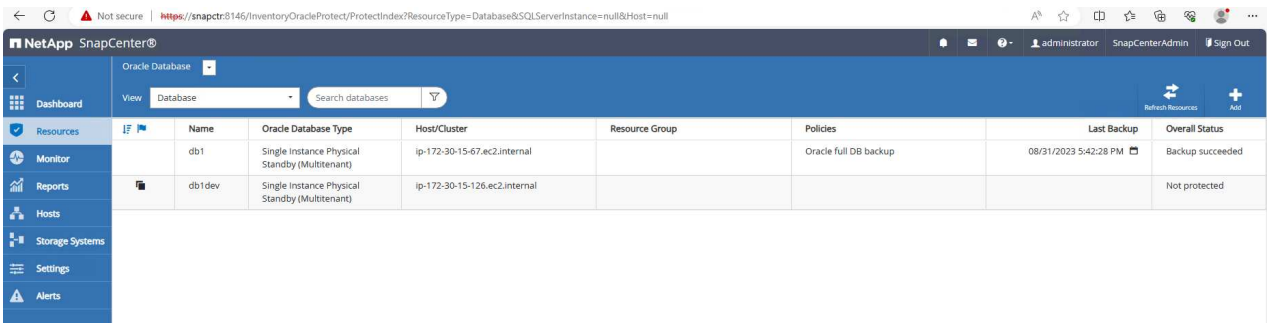
14. Überprüfen Sie die Zusammenfassung, klicken Sie auf `Finish` um den Klon zu starten.



15. Überwachen Sie den Klonjob in `Monitor` Tab. Wir haben festgestellt, dass das Klonen einer Datenbank mit einer Datenbankvolumengröße von etwa 300 GB etwa 8 Minuten dauerte.



16. Validieren Sie die Klondatenbank von SnapCenter, die sofort in registriert ist `Resources` Registerkarte direkt nach dem Klonvorgang.



17. Fragen Sie die Klondatenbank von der EC2-Kloninstanz ab. Wir haben bestätigt, dass die in der primären Datenbank aufgetretene Testtransaktion bis zur Klondatenbank durchgedrungen ist.

```
[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>
```

Damit ist das Klonen und die Validierung einer neuen Oracle-Datenbank aus der Standby-Datenbank in Data Guard auf dem FSx-Speicher für DEV, TEST, REPORT oder andere Anwendungsfälle abgeschlossen. In Data Guard können mehrere Oracle-Datenbanken aus derselben Standby-Datenbank geklont werden.

Wo Sie weitere Informationen finden

Weitere Informationen zu den in diesem Dokument beschriebenen Informationen finden Sie in den folgenden Dokumenten und/oder auf den folgenden Websites:

- Data Guard-Konzepte und -Administration

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

- WP-7357: Best Practices für die Bereitstellung von Oracle-Datenbanken auf EC2 und FSx

  "Einführung"

- 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