Oracle 数据库

NetApp Solutions

NetApp
April 12, 2024

This PDF was generated from https://docs.netapp.com/zh-cn/netapp-solutions/databases/azure_ora_anf_inc_merge.html on April 12, 2024. Always check docs.netapp.com for the latest.

目录

Oracle 数据库 . . . 1
TR-4990：《使用ANF上的增量合并快速恢复Oracle VLDB》 . . . 1
TR-4987：《在采用NFS的Azure NetApp Files上简化的自动化Oracle部署》 . . . 59
TR-4986：《在采用iSCSI的Amazon FSx ONTAP上简化的自动化Oracle部署》 . . . 77
TR-4983：《在采用iSCSI的NetApp ASA上简化的自动化Oracle部署》 . . . 95
TR-4979：《在AWS上的VMware Cloud中使用子系统装载的FSx ONTAP简化的自行管理Oracle》 . . . 111
TR-4981：《借助Amazon FSx ONTAP降低Oracle Active Data Guard成本》 . . . 177
TR-4973：《在AWS FSx ONTAP上使用增量合并快速恢复和克隆Oracle VLDB》 . . . 211
TR-4974：《Oracle 19c在使用NFS/ASM的AWS FSX/EC2上独立重新启动》 . . . 291
TR-4965：《使用iSCSI/ASM在AWS FSX/EC2中部署和保护Oracle数据库》 . . . 317
在AWS EC2和FSX上部署Oracle数据库最佳实践 . . . 348
Azure NetApp Files 上的Oracle数据库部署和保护 . . . 402
NVA-1155：《基于FlexPod 数据中心且采用Cisco UCS和基于FC的NetApp AFF A800的Oracle 19c RAC数据库—设计和部署指南》 . . . 442
TR-4250：SAP与基于UNIX的Oracle和NFS以及适用于SAP 3.4的NetApp集群模式Data ONTAP和SnapManager . . . 442
部署 Oracle 数据库 . . . 443
解决方案概述 . . . 463
TR-4794：《基于NetApp EF系列的Oracle数据库》 . . . 485

Oracle 数据库

TR-4990：《使用ANF上的增量合并快速恢复Oracle VLDB》

NetApp公司Allen Cao、Niyaz Mohamed

目的

使用Oracle Recovery Manager (RMAN)备份工具在Oracle中恢复超大型数据库(VLDB)可能是一项极具挑战性的任务。发生故障时从备份介质还原数据库的过程可能会非常耗时、从而会延迟数据库恢复、并可能显著影响服务级别协议(Service Level Agreement、SLA)。但是、从10g开始、Oracle引入了RMAN功能、允许用户在DB服务器主机上的其他磁盘存储上创建Oracle数据库数据文件的暂存映像副本。这些映像副本可以每天使用RMAN进行增量更新。如果发生故障、数据库管理员(Database Administrator、DBA)可以快速将Oracle数据库从故障介质切换到映像副本、而无需进行完整的数据库介质恢复。结果是SLA得到了大幅改进、尽管成本是所需数据库存储的两倍。

如果您对VLDB的SLA非常感兴趣、并考虑将Oracle数据库迁移到Azure等公共云、则可以使用Microsoft Azure NetApp Files (ANF)等资源设置类似的数据库保护结构、以便暂存备用数据库映像副本。在本文档中、我们将演示如何从ANF容量池配置和导出NFS文件系统、以便挂载到Oracle数据库服务器上、暂存备用数据库副本、以便在主存储发生故障时快速恢复。

此解决方案 可解决以下使用情形：

- Oracle VLDB映像副本通过RMAN在Microsoft ANF容量池存储以外的NFS挂载点上进行增量合并。
- 在同一Azure数据库服务器VM发生故障时快速恢复Oracle VLDB。
- 在备用Azure数据库服务器VM发生故障时快速恢复Oracle VLDB。

audience

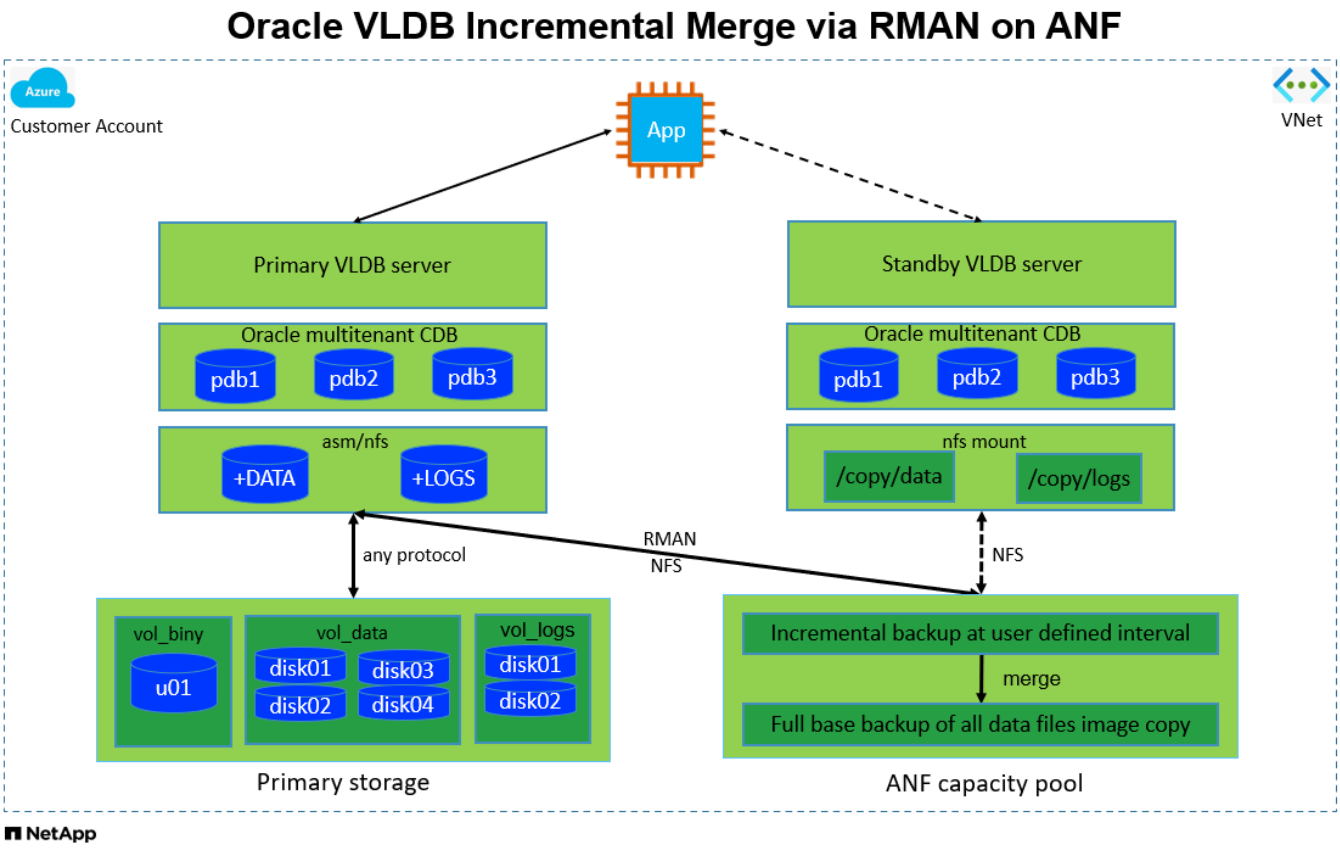
此解决方案 适用于以下人员：

- 在Azure中通过RMAN设置Oracle VLDB映像副本增量合并以加快数据库恢复的数据库提供商。
- 在Azure公共云中测试Oracle工作负载的数据库解决方案架构师。
- 负责管理部署到ANF容量池存储的Oracle数据库的存储管理员。
- 希望在Azure云环境中设置Oracle数据库的应用程序所有者。

解决方案 测试和验证环境

此解决方案的测试和验证是在Microsoft ANF容量池存储和Azure VM计算环境中执行的、这些环境可能与最终部署环境不匹配。有关详细信息，请参见一节 [Key Factors for Deployment Consideration]。

架构



硬件和软件组件

* 硬件 *		
ANF存储	Microsoft提供的当前版本	具有高级服务级别的2 TiB ANF容量池存储
适用于数据库服务器的Azure VM	standard_B4ms—4个vCPU、16 GiB	2个VM、一个用作主数据库服务器、另一个用作备用
软件		
RedHat Linux	RHEL Linux 8.6 (LVM)- x64 Gen2	已部署RedHat订阅以进行测试
Oracle 数据库	版本19.18	已应用RU修补程序p34765931_190000_Linux-x86-64.zip
Oracle OPatch	版本12.2.0.1.36	最新修补程序p6880880_190000_Linux-x86-64.zip
NFS	版本 3.0	已启用Oracle DNFS

部署注意事项的关键因素

- *用于RMAN增量合并的Oracle VLDB存储布局。*在我们的测试和验证中、用于Oracle增量备份和合并的NFS卷是从一个ANF容量池中分配的、该容量池每个卷具有100 TiB的容量限制、总容量限制为1000 TiB。对于超过阈值的部署、可以将多个卷和ANF容量池与多个NFS挂载点并行连接、以提供更高的容量。

- 使用**RMAN**增量合并的**Oracle**可恢复性。 RMAN增量备份和合并通常根据RTO和RPO目标以用户定义的频率执行。如果主数据存储和/或归档日志完全丢失、则可能会发生数据丢失。Oracle数据库可以恢复到ANF数据库备份映像副本提供的最后一次增量备份。为了最大限度地减少数据丢失、可以在ANF NFS挂载点上设置Oracle闪存恢复区域、并将归档日志与数据库映像副本一起备份到ANF NFS挂载。
- *在ANF NFS文件系统之外运行Oracle VLDB。*与用于数据库备份的其他批量存储不同、Microsoft ANF是支持云的生产级存储、可提供高级别的性能和存储效率。Oracle VLDB从主存储切换到ANF NFS文件系统上的映像副本后、可以在解决主存储故障的同时保持较高的数据库性能。您可以放心地知道、主存储故障不会影响用户应用程序体验。
- * Azure计算实例。*在这些测试和验证中、我们使用Standard"标准B4ms Azure VM作为Oracle数据库服务器。还有其他Azure VM可能已经过优化、更适合数据库工作负载。此外、您还需要根据实际工作负载要求、根据vCPU数量和RAM量相应地调整Azure VM的大小。
- * ANF容量池服务级别。* ANF容量池提供三种服务级别：标准、高级、超级。默认情况下、自动QoS会对容量池中创建的卷执行适用场景操作、从而限制卷上的吞吐量。可以根据容量池大小和服务级别手动调整卷上的吞吐量。
- **DNFS**配置。 DNFS内置在Oracle内核中、众所周知、在将Oracle部署到NFS存储时、它可以显著提高Oracle数据库性能。DNFS打包到Oracle二进制文件中、但默认情况下不启用。对于NFS上的任何Oracle数据库部署、都应启用此功能。对于VLDB的多个ANF容量池部署、应正确配置指向不同ANF容量池存储的DNFS多路径。

解决方案 部署

我们假定您已将Oracle VLDB部署在VNet中的Azure云环境中。如果您需要有关在Azure中部署Oracle的帮助、请参阅以下技术报告以获取帮助。

- "使用NFS在Azure NetApp Files上简化、自动化的Oracle部署"
- "Azure NetApp Files 上的Oracle数据库部署和保护"

Oracle VLDB可以运行在ANF存储上、也可以运行在Azure云生态系统中的任何其他可选存储上。下一节介绍了为Oracle VLDB的映像副本设置RMAN增量合并的分步部署过程、该副本暂存在ANF存储的NFS挂载中。

部署的前提条件

部署需要满足以下前提条件。

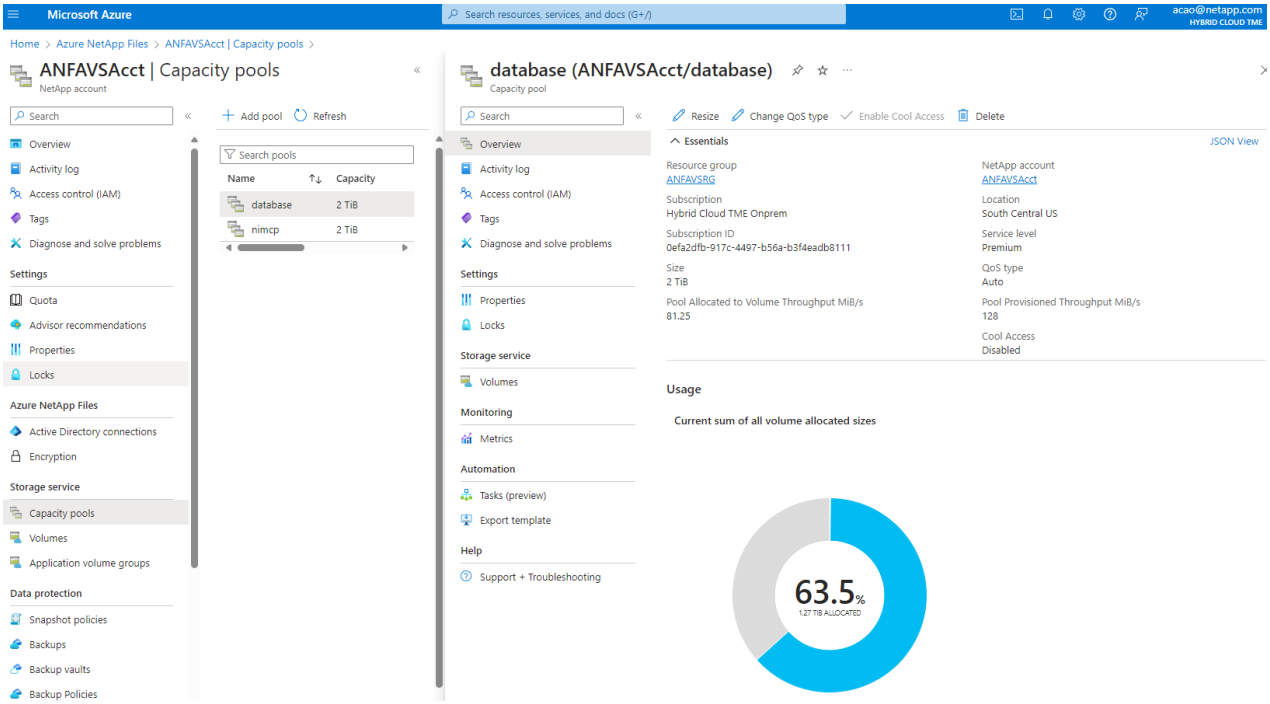
1. 已设置Azure帐户、并且已在Azure帐户中创建必要的Azure vNet和网段。
2. 从Azure门户控制台中、您必须部署两个Azure VM实例、一个用作主Oracle数据库服务器、另一个用作可选备用数据库服务器。有关环境设置的详细信息、请参见上一节中的架构图。另请查看 "Azure虚拟机系列" 有关详细信息 …
3. 从Azure门户控制台中、部署ANF存储以托管用于存储Oracle数据库备用映像副本的NFS卷。如果您不熟悉ANF的部署、请参见文档 "快速入门：设置 Azure NetApp Files 并创建 NFS 卷" 了解分步说明。

确保您已在Azure VM根卷中至少分配128 G、以便有足够的空间来暂存Oracle安装文件。

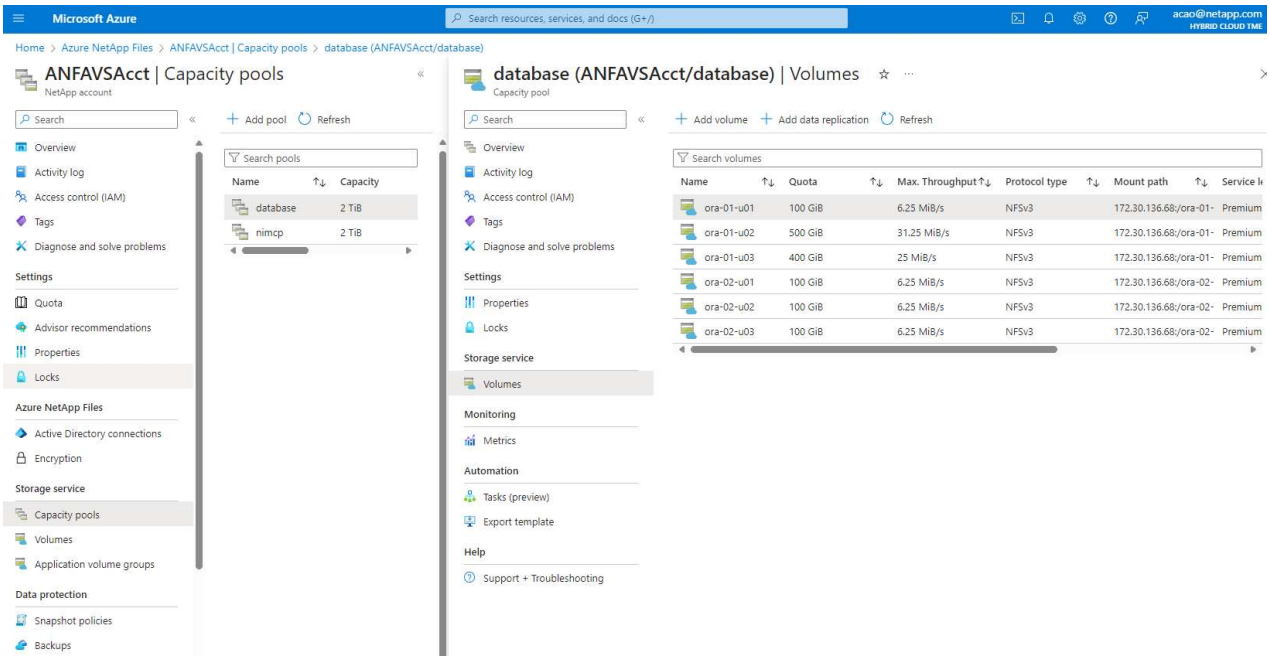
配置和导出要挂载到**Oracle**主**VLDB**服务器上的**NFS**卷

在本节中、我们将介绍通过Azure门户控制台从ANF容量池配置NFS卷。如果设置了多个ANF容量池来容纳数据库的大小、请对其他ANF容量池重复上述过程。

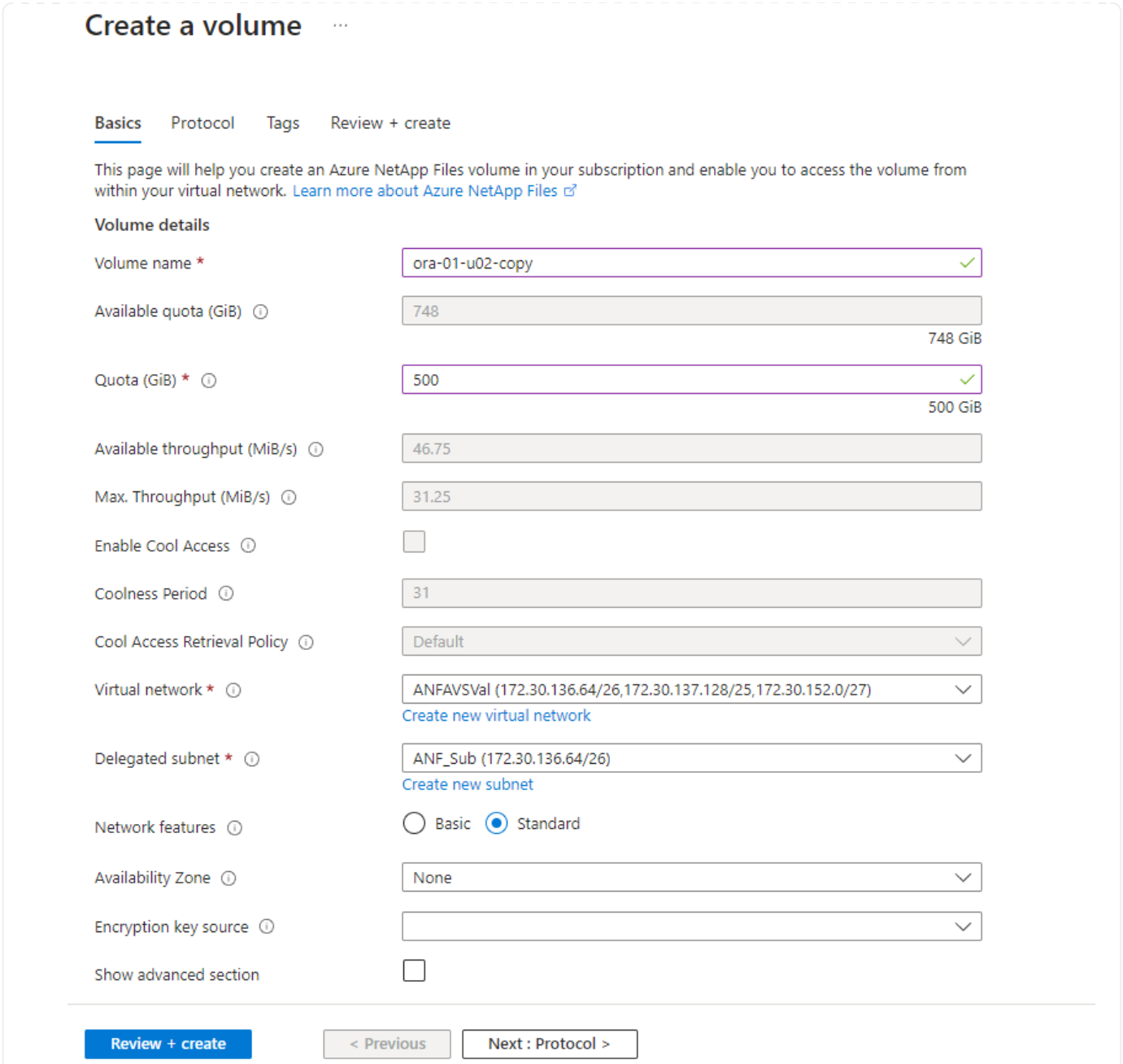
1. 首先、从Azure门户控制台导航到用于暂存Oracle VLDB映像副本的ANF容量池。



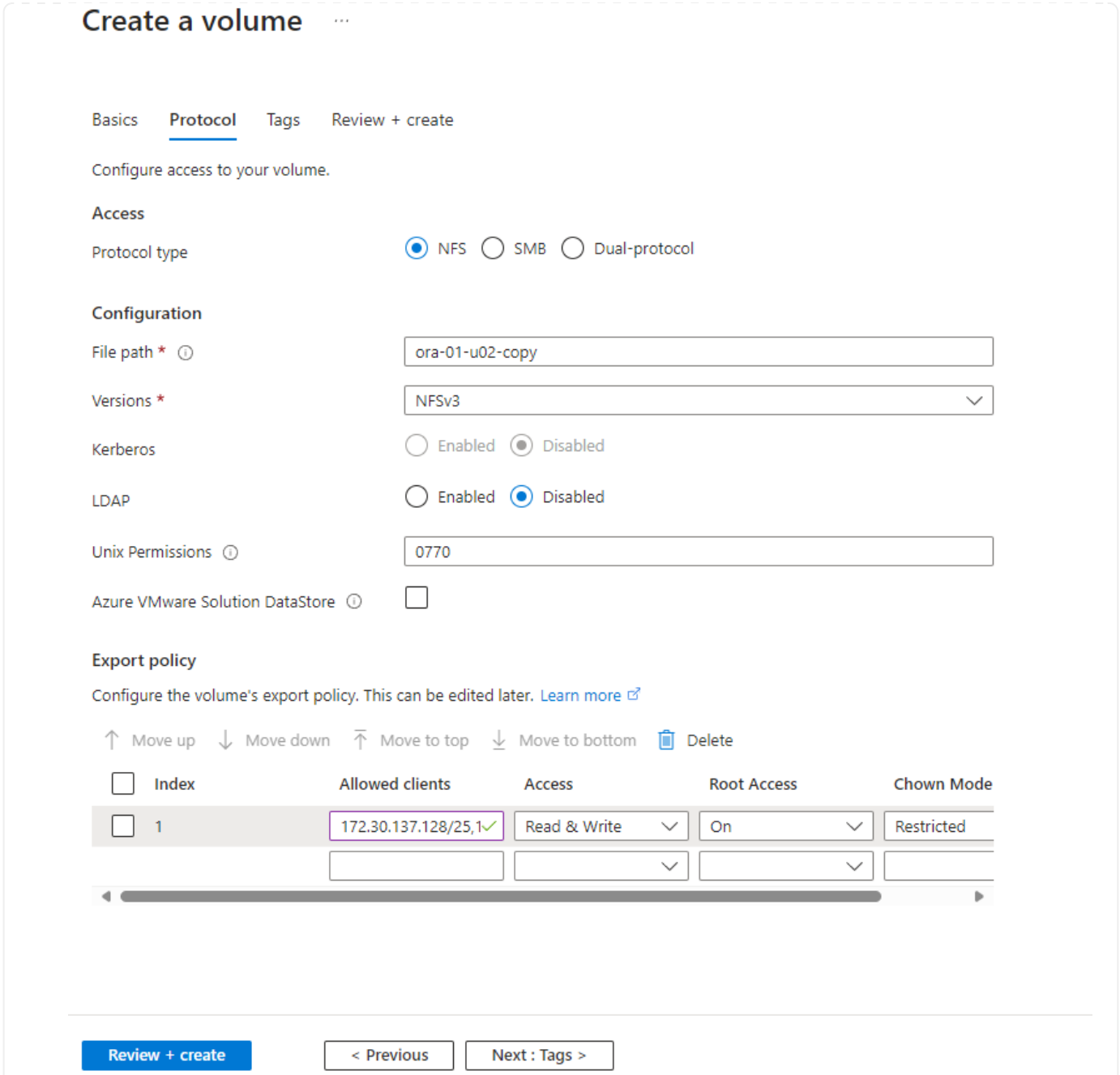
2. 从选定容量池- `database`下、单击 `Volumes` 然后、 `Add volume` 以启动添加卷工作流。



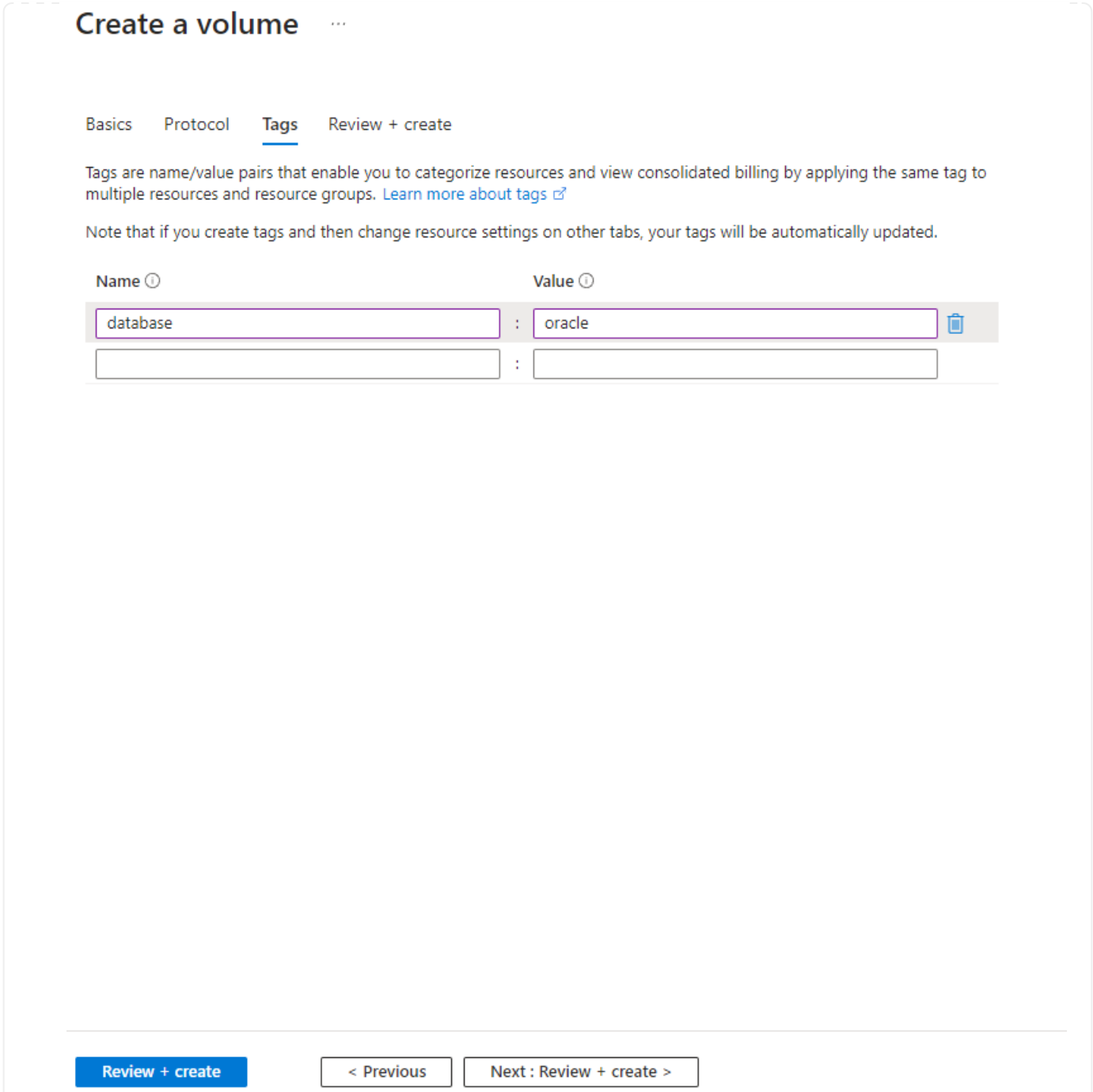
3. 填写 `Volume name`， `Quota`， `Virtual network`，和 `Delegated subnet` 移动到 `Protocol` 页面。



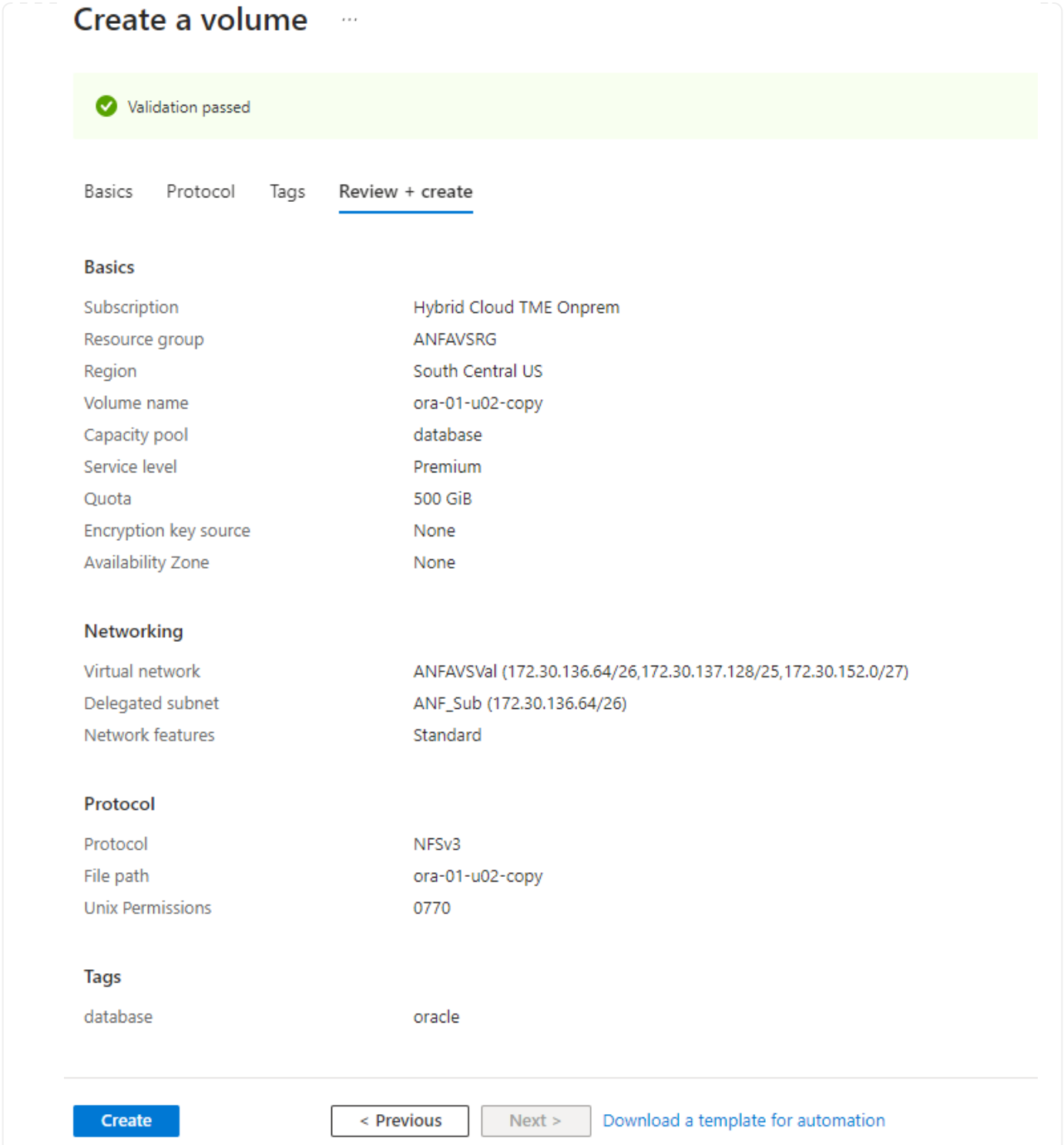
4. 记下文件路径、输入允许的客户端CIDR范围、然后启用 `Root Access` 卷。


Create a volume
Basics
Protocol
Tags
Review + create
Configure access to your volume.
Access
Protocol type
NFS
SMB
Dual-protocol
Configuration
File path *
ora-01-u02-copy
Versions *
NFSv3
Kerberos
Enabled
Disabled
LDAP
Enabled
Disabled
Unix Permissions
0770
Azure VMware Solution DataStore
Export policy
Configure the volume's export policy. This can be edited later. Learn more
Move up
Move down
Move to top
Move to bottom
Delete
Index
Allowed clients
Access
Root Access
Chown Mode
1
172.30.137.128/25,1
Read & Write
On
Restricted
Review + create
< Previous
Next : Tags >

5. 根据需要添加卷标记。


Create a volume
Basics
Protocol
Tags
Review + create
Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. Learn more about tags
Note that if you create tags and then change resource settings on other tabs, your tags will be automatically updated.
Name
Value
database
oracle
Review + create
< Previous
Next : Review + create >

6. 查看并创建卷。



7. 以具有sudo权限的用户身份登录到主Oracle VLDB服务器、然后挂载从ANF存储导出的NFS卷。根据需要更改ANF NFS服务器IP地址和文件路径。可以从ANF卷控制台页面检索ANF NFS服务器IP地址。

```
sudo mkdir /nfsanf
```

```
sudo mount 172.30.136.68:/ora-01-u02-copy /nfsanf -o
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=262144,wsize=262144,noin
tr
```

8. 将挂载点所有权更改为oracle：oisntall、根据需要更改为Oracle用户名和主组。

```
sudo chown oracle:oinstall /nfsanf
```

将**Oracle RMAN**增量合并设置为**ANF**上的映像副本

RMAN增量合并会在每个增量备份/合并间隔持续更新暂存数据库数据文件映像副本。数据库备份的映像副本将与您执行增量备份/合并的频率相同。因此、在确定RMAN增量备份和合并的频率时、应考虑数据库性能、RTO和RPO目标。

1. 以Oracle用户身份登录到主Oracle VLDB服务器。
2. 在挂载点/nfsanf下创建oracopy目录、用于存储Oracle闪存恢复区域的Oracle数据文件映像副本和归档日志目录。

```
mkdir /nfsanf/oracopy
```

```
mkdir /nfsanf/archlog
```

3. 通过sqlplus登录到Oracle数据库、启用块更改跟踪以加快增量备份、如果Oracle闪存恢复区域当前位于主存储上、则将其更改为ANF NFS挂载。这样可以将RMAN默认控制文件/spfile自动备份和归档日志备份到ANF NFS挂载以进行恢复。

```
sqlplus / as sysdba
```

从sqlplus提示符处、执行以下命令。

```
alter database enable block change tracking using file
'/nfsanf/oracopy/bct_ntap1.ctf'
```

```
alter system set db_recovery_file_dest='/nfsanf/archlog/'
scope=both;
```

预期输出：

```
[oracle@ora-01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Mar 20 16:44:21
2024
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 database enable block change tracking using file
'/nfsanf/oracopy/bct_ntap1.ctf';

Database altered.

SQL> alter system set db_recovery_file_dest='/nfsanf/archlog/'
scope=both;

System altered.

SQL>
```

4. 创建RMAN备份和增量合并脚本。该脚本会为并行RMAN备份和合并分配多个通道。首次执行将生成初始完整基线映像副本。在完整运行中、它会首先清除保留窗口之外的过时备份、以保持暂存区域干净。然后、它会在合并和备份之前切换当前日志文件。增量备份会在合并后进行、以便数据库映像副本会在当前数据库状态后经过一个备份/合并周期。可以反转合并和备份顺序、以便根据用户的偏好加快恢复速度。RMAN脚本可以集成到一个简单的shell脚本中、以便从主数据库服务器上的crontab执行。确保在RMAN设置中打开控制文件自动备份。

```
vi /home/oracle/rman_bkup_merge.cmd

Add following lines:

RUN
{
  allocate channel c1 device type disk format '/nfsanf/oracopy/%U';
  allocate channel c2 device type disk format '/nfsanf/oracopy/%U';
  allocate channel c3 device type disk format '/nfsanf/oracopy/%U';
  allocate channel c4 device type disk format '/nfsanf/oracopy/%U';
  delete obsolete;
  sql 'alter system archive log current';
  recover copy of database with tag 'OraCopyBKUPonANF_level_0';
  backup incremental level 1 copies=1 for recover of copy with tag
'OraCopyBKUPonANF_level_0' database;
}
```

5. 在主Oracle VLDB服务器上、以Oracle用户身份本地登录到RMAN、无论是否具有RMAN目录。在此演示中、我们不会连接到RMAN目录。

```
rman target / nocatalog;

output:

[oracle@ora-01 ~]$ rman target / nocatalog

Recovery Manager: Release 19.0.0.0.0 - Production on Wed Mar 20
16:54:24 2024
Version 19.18.0.0.0

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

connected to target database: NTAP1 (DBID=2441823937)
using target database control file instead of recovery catalog
```

6. 从RMAN提示符处、执行该脚本。首次执行时创建基线数据库映像副本、后续执行时合并并增量更新基线映像副本。下面介绍了如何执行该脚本以及典型输出。设置通道数、使其与主机上的CPU核匹配。

```
RMAN> @/home/oracle/rman_bkup_merge.cmd

RMAN> RUN
2> {
3>   allocate channel c1 device type disk format
```

```
'/nfsanf/oracopy/%U';
4>   allocate channel c2 device type disk format
'/nfsanf/oracopy/%U';
5>   allocate channel c3 device type disk format
'/nfsanf/oracopy/%U';
6>   allocate channel c4 device type disk format
'/nfsanf/oracopy/%U';
7>   delete obsolete;
8>   sql 'alter system archive log current';
9>   recover copy of database with tag 'OraCopyBKUPonANF_level_0';
10>   backup incremental level 1 copies=1 for recover of copy with
tag 'OraCopyBKUPonANF_level_0' database;
11> }
allocated channel: c1
channel c1: SID=142 device type=DISK

allocated channel: c2
channel c2: SID=277 device type=DISK

allocated channel: c3
channel c3: SID=414 device type=DISK

allocated channel: c4
channel c4: SID=28 device type=DISK

RMAN retention policy will be applied to the command
RMAN retention policy is set to redundancy 1
Deleting the following obsolete backups and copies:
Type                 Key    Completion Time    Filename/Handle
-------------------- ------ ------------------ --------------------
Backup Set           1      18-MAR-24
  Backup Piece       1      18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163958359__04h19dg
r_.bkp
Backup Set           2      18-MAR-24
  Backup Piece       2      18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163961675__07l1m2l
g_.bkp
Backup Set           3      18-MAR-24
  Backup Piece       3      18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163962888__08p6y7l
x_.bkp
Backup Set           4      18-MAR-24
  Backup Piece       4      18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163963796__09k8g1m
4_.bkp
```

```
Backup Set           5       18-MAR-24
  Backup Piece       5       18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163964697__0bd3tqg
3_.bkp
Backup Set           6       18-MAR-24
  Backup Piece       6       18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163965895__0chx6mz
t_.bkp
Backup Set           7       18-MAR-24
  Backup Piece       7       18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163966806__0dbyx34
4_.bkp
Backup Set           8       18-MAR-24
  Backup Piece       8       18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163968012__0fgvg80
5_.bkp
Backup Set           9       18-MAR-24
  Backup Piece       9       18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163968919__0g9x5t1
v_.bkp
Backup Set           10      18-MAR-24
  Backup Piece       10      18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163969821__0h4rfdz
j_.bkp
Backup Set           11      18-MAR-24
  Backup Piece       11      18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163971026__0j8o4wk
8_.bkp
Backup Set           12      18-MAR-24
  Backup Piece       12      18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163971931__0k3pnn2
o_.bkp
Backup Set           13      18-MAR-24
  Backup Piece       13      18-MAR-24
/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163972835__0kyg92t
1_.bkp
deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163963796__
09k8g1m4_.bkp RECID=4 STAMP=1163963804
deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163962888__
08p6y7lx_.bkp RECID=3 STAMP=1163962897
deleted backup piece
backup piece
```

```
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163961675__
07l1m2lg_.bkp RECID=2 STAMP=1163961683
deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163958359__
04h19dgr_.bkp RECID=1 STAMP=1163958361
deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163964697__
0bd3tqg3_.bkp RECID=5 STAMP=1163964705
deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163965895__
0chx6mzt_.bkp RECID=6 STAMP=1163965906
deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163966806__
0dbyx344_.bkp RECID=7 STAMP=1163966814
deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163968012__
0fgvg805_.bkp RECID=8 STAMP=1163968018
deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163968919__
0g9x5t1v_.bkp RECID=9 STAMP=1163968926
deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163969821__
0h4rfdzj_.bkp RECID=10 STAMP=1163969827
Deleted 3 objects

deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163971026__
0j8o4wk8_.bkp RECID=11 STAMP=1163971032
Deleted 3 objects

deleted backup piece
backup piece
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163971931__
0k3pnn2o_.bkp RECID=12 STAMP=1163971938
Deleted 3 objects

deleted backup piece
backup piece
```

```
handle=/u03/orareco/NTAP1/autobackup/2024_03_18/o1_mf_s_1163972835__
0kyg92t1_.bkp RECID=13 STAMP=1163972837
Deleted 4 objects


sql statement: alter system archive log current

Starting recover at 20-MAR-24
no copy of datafile 1 found to recover
no copy of datafile 3 found to recover
no copy of datafile 4 found to recover
.
.
no copy of datafile 31 found to recover
no copy of datafile 32 found to recover
Finished recover at 20-MAR-24

Starting backup at 20-MAR-24
no parent backup or copy of datafile 1 found
no parent backup or copy of datafile 3 found
no parent backup or copy of datafile 4 found
.
.
no parent backup or copy of datafile 19 found
no parent backup or copy of datafile 20 found
channel c1: starting datafile copy
input datafile file number=00021
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_01.dbf
channel c2: starting datafile copy
input datafile file number=00022
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_02.dbf
channel c3: starting datafile copy
input datafile file number=00023
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_03.dbf
channel c4: starting datafile copy
input datafile file number=00024
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_04.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-22_0g2m6brl tag=ORACOPYBKUPONANF_LEVEL_0 RECID=4
STAMP=1164132108
channel c2: datafile copy complete, elapsed time: 01:06:39
channel c2: starting datafile copy
input datafile file number=00025
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_05.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-24_0i2m6brl tag=ORACOPYBKUPONANF_LEVEL_0 RECID=5
```

```
STAMP=1164132121
channel c4: datafile copy complete, elapsed time: 01:06:45
channel c4: starting datafile copy
input datafile file number=00026
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_06.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-23_0h2m6brl tag=ORACOPYBKUPONANF_LEVEL_0 RECID=6
STAMP=1164132198
channel c3: datafile copy complete, elapsed time: 01:08:05
channel c3: starting datafile copy
input datafile file number=00027
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_07.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-21_0f2m6brl tag=ORACOPYBKUPONANF_LEVEL_0 RECID=7
STAMP=1164132248
channel c1: datafile copy complete, elapsed time: 01:08:57
channel c1: starting datafile copy
input datafile file number=00028
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_08.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-25_0j2m6fol tag=ORACOPYBKUPONANF_LEVEL_0 RECID=9
STAMP=1164136123
channel c2: datafile copy complete, elapsed time: 01:06:46
channel c2: starting datafile copy
input datafile file number=00029
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_09.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-26_0k2m6fot tag=ORACOPYBKUPONANF_LEVEL_0 RECID=8
STAMP=1164136113
channel c4: datafile copy complete, elapsed time: 01:06:36
channel c4: starting datafile copy
input datafile file number=00030
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_10.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-27_0l2m6frc tag=ORACOPYBKUPONANF_LEVEL_0 RECID=10
STAMP=1164136293
channel c3: datafile copy complete, elapsed time: 01:08:10
channel c3: starting datafile copy
input datafile file number=00031
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_11.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-28_0m2m6fsu tag=ORACOPYBKUPONANF_LEVEL_0 RECID=11
STAMP=1164136333
channel c1: datafile copy complete, elapsed time: 01:07:52
channel c1: starting datafile copy
input datafile file number=00032
```

```
name=/u02/oradata/NTAP1/NTAP1_pdb1/soe_12.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-29_0n2m6jlr tag=ORACOPYBKUPONANF_LEVEL_0 RECID=12
STAMP=1164140082
channel c2: datafile copy complete, elapsed time: 01:06:01
channel c2: starting datafile copy
input datafile file number=00001
name=/u02/oradata/NTAP1/system01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-30_0o2m6jlr tag=ORACOPYBKUPONANF_LEVEL_0 RECID=13
STAMP=1164140190
channel c4: datafile copy complete, elapsed time: 01:07:49
channel c4: starting datafile copy
input datafile file number=00003
name=/u02/oradata/NTAP1/sysaux01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-1_0r2m6nhk tag=ORACOPYBKUPONANF_LEVEL_0 RECID=14
STAMP=1164140240
channel c2: datafile copy complete, elapsed time: 00:02:38
channel c2: starting datafile copy
input datafile file number=00004
name=/u02/oradata/NTAP1/undotbs01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-4_0t2m6nml tag=ORACOPYBKUPONANF_LEVEL_0 RECID=15
STAMP=1164140372
channel c2: datafile copy complete, elapsed time: 00:02:15
channel c2: starting datafile copy
input datafile file number=00011
name=/u02/oradata/NTAP1/NTAP1_pdb1/undotbs01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-3_0s2m6nl1 tag=ORACOPYBKUPONANF_LEVEL_0 RECID=16
STAMP=1164140377
channel c4: datafile copy complete, elapsed time: 00:03:01
channel c4: starting datafile copy
input datafile file number=00010
name=/u02/oradata/NTAP1/NTAP1_pdb1/sysaux01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-32_0q2m6jsi tag=ORACOPYBKUPONANF_LEVEL_0 RECID=17
STAMP=1164140385
channel c1: datafile copy complete, elapsed time: 01:07:29
channel c1: starting datafile copy
input datafile file number=00014
name=/u02/oradata/NTAP1/NTAP1_pdb2/sysaux01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SOE_FNO-31_0p2m6jrb tag=ORACOPYBKUPONANF_LEVEL_0 RECID=18
STAMP=1164140406
```

```
channel c3: datafile copy complete, elapsed time: 01:08:31
channel c3: starting datafile copy
input datafile file number=00018
name=/u02/oradata/NTAP1/NTAP1_pdb3/sysaux01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-10_0v2m6nqs tag=ORACOPYBKUPONANF_LEVEL_0 RECID=19
STAMP=1164140459
channel c4: datafile copy complete, elapsed time: 00:01:26
channel c4: starting datafile copy
input datafile file number=00006
name=/u02/oradata/NTAP1/pdbseed/sysaux01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-14_102m6nr3 tag=ORACOPYBKUPONANF_LEVEL_0 RECID=20
STAMP=1164140468
channel c1: datafile copy complete, elapsed time: 00:01:22
channel c1: starting datafile copy
input datafile file number=00009
name=/u02/oradata/NTAP1/NTAP1_pdb1/system01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-11_0u2m6nqs tag=ORACOPYBKUPONANF_LEVEL_0 RECID=21
STAMP=1164140471
channel c2: datafile copy complete, elapsed time: 00:01:33
channel c2: starting datafile copy
input datafile file number=00013
name=/u02/oradata/NTAP1/NTAP1_pdb2/system01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-18_112m6nrt tag=ORACOPYBKUPONANF_LEVEL_0 RECID=22
STAMP=1164140476
channel c3: datafile copy complete, elapsed time: 00:00:57
channel c3: starting datafile copy
input datafile file number=00017
name=/u02/oradata/NTAP1/NTAP1_pdb3/system01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-6_122m6nti tag=ORACOPYBKUPONANF_LEVEL_0 RECID=23
STAMP=1164140488
channel c4: datafile copy complete, elapsed time: 00:00:25
channel c4: starting datafile copy
input datafile file number=00005
name=/u02/oradata/NTAP1/pdbseed/system01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-13_142m6ntp tag=ORACOPYBKUPONANF_LEVEL_0 RECID=24
STAMP=1164140532
channel c2: datafile copy complete, elapsed time: 00:01:06
channel c2: starting datafile copy
input datafile file number=00008
name=/u02/oradata/NTAP1/pdbseed/undotbs01.dbf
```

```
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-17_152m6nts tag=ORACOPYBKUPONANF_LEVEL_0 RECID=25
STAMP=1164140539
channel c3: datafile copy complete, elapsed time: 00:01:03
channel c3: starting datafile copy
input datafile file number=00015
name=/u02/oradata/NTAP1/NTAP1_pdb2/undotbs01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-9_132m6ntm tag=ORACOPYBKUPONANF_LEVEL_0 RECID=26
STAMP=1164140541
channel c1: datafile copy complete, elapsed time: 00:01:13
channel c1: starting datafile copy
input datafile file number=00019
name=/u02/oradata/NTAP1/NTAP1_pdb3/undotbs01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-5_162m6nuc tag=ORACOPYBKUPONANF_LEVEL_0 RECID=27
STAMP=1164140541
channel c4: datafile copy complete, elapsed time: 00:00:41
channel c4: starting datafile copy
input datafile file number=00007 name=/u02/oradata/NTAP1/users01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-8_172m6nvr tag=ORACOPYBKUPONANF_LEVEL_0 RECID=28
STAMP=1164140552
channel c2: datafile copy complete, elapsed time: 00:00:16
channel c2: starting datafile copy
input datafile file number=00012
name=/u02/oradata/NTAP1/NTAP1_pdb1/users01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-15_182m6nvs tag=ORACOPYBKUPONANF_LEVEL_0 RECID=30
STAMP=1164140561
channel c3: datafile copy complete, elapsed time: 00:00:24
channel c3: starting datafile copy
input datafile file number=00016
name=/u02/oradata/NTAP1/NTAP1_pdb2/users01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-7_1a2m6o01 tag=ORACOPYBKUPONANF_LEVEL_0 RECID=29
STAMP=1164140560
channel c4: datafile copy complete, elapsed time: 00:00:16
channel c4: starting datafile copy
input datafile file number=00020
name=/u02/oradata/NTAP1/NTAP1_pdb3/users01.dbf
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-19_192m6nvv tag=ORACOPYBKUPONANF_LEVEL_0 RECID=31
STAMP=1164140564
channel c1: datafile copy complete, elapsed time: 00:00:21
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
```

```
USERS_FNO-12_1b2m6o0e tag=ORACOPYBKUPONANF_LEVEL_0 RECID=32
STAMP=1164140564
channel c2: datafile copy complete, elapsed time: 00:00:02
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-16_1c2m6o0k tag=ORACOPYBKUPONANF_LEVEL_0 RECID=34
STAMP=1164140565
channel c3: datafile copy complete, elapsed time: 00:00:01
output file name=/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-20_1d2m6o0k tag=ORACOPYBKUPONANF_LEVEL_0 RECID=33
STAMP=1164140565
channel c4: datafile copy complete, elapsed time: 00:00:01
Finished backup at 20-MAR-24

Starting Control File and SPFILE Autobackup at 20-MAR-24
piece
handle=/nfsanf/archlog/NTAP1/autobackup/2024_03_20/o1_mf_s_116414056
5__5g56ypks_.bkp comment=NONE
Finished Control File and SPFILE Autobackup at 20-MAR-24
released channel: c1
released channel: c2
released channel: c3
released channel: c4

RMAN> **end-of-file**

RMAN>
```

7. 在备份后列出数据库映像副本、以观察是否已在ANF NFS挂载点中创建数据库映像副本。

```
RMAN> list copy of database tag 'OraCopyBKUPonANF_level_0';

List of Datafile Copies
========================

Key     File S Completion Time Ckp SCN    Ckp Time        Sparse
------- ---- - --------------- ---------- --------------- ------
14      1    A 20-MAR-24       4161498    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-1_0r2m6nhk
        Tag: ORACOPYBKUPONANF_LEVEL_0

16      3    A 20-MAR-24       4161568    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-3_0s2m6nl1
        Tag: ORACOPYBKUPONANF_LEVEL_0
```

```
15      4    A 20-MAR-24       4161589    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-4_0t2m6nml
        Tag: ORACOPYBKUPONANF_LEVEL_0

27      5    A 20-MAR-24       2379694    18-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-5_162m6nuc
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

23      6    A 20-MAR-24       2379694    18-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-6_122m6nti
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

29      7    A 20-MAR-24       4161872    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-7_1a2m6o01
        Tag: ORACOPYBKUPONANF_LEVEL_0

28      8    A 20-MAR-24       2379694    18-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-8_172m6nvr
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

26      9    A 20-MAR-24       4161835    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-9_132m6ntm
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

19      10   A 20-MAR-24       4161784    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-10_0v2m6nqs
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

21      11   A 20-MAR-24       4161780    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-11_0u2m6nqs
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1
```

```
32      12   A 20-MAR-24       4161880    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-12_1b2m6o0e
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

24      13   A 20-MAR-24       4161838    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-13_142m6ntp
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 4, PDB Name: NTAP1_PDB2

20      14   A 20-MAR-24       4161785    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-14_102m6nr3
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 4, PDB Name: NTAP1_PDB2

30      15   A 20-MAR-24       4161863    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-15_182m6nvs
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 4, PDB Name: NTAP1_PDB2

34      16   A 20-MAR-24       4161884    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-16_1c2m6o0k
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 4, PDB Name: NTAP1_PDB2

25      17   A 20-MAR-24       4161841    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-17_152m6nts
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 5, PDB Name: NTAP1_PDB3

22      18   A 20-MAR-24       4161810    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-18_112m6nrt
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 5, PDB Name: NTAP1_PDB3

31      19   A 20-MAR-24       4161869    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-19_192m6nvv
        Tag: ORACOPYBKUPONANF_LEVEL_0
```

```
        Container ID: 5, PDB Name: NTAP1_PDB3

33      20   A 20-MAR-24       4161887    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-20_1d2m6o0k
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 5, PDB Name: NTAP1_PDB3

7       21   A 20-MAR-24       4152514    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
21_0f2m6brl
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

4       22   A 20-MAR-24       4152518    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
22_0g2m6brl
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

6       23   A 20-MAR-24       4152522    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
23_0h2m6brl
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

5       24   A 20-MAR-24       4152529    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
24_0i2m6brl
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

9       25   A 20-MAR-24       4156120    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
25_0j2m6fol
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

8       26   A 20-MAR-24       4156130    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
26_0k2m6fot
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

10      27   A 20-MAR-24       4156159    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
```

```
27_0l2m6frc
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

11      28   A 20-MAR-24       4156183    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
28_0m2m6fsu
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

12      29   A 20-MAR-24       4158795    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
29_0n2m6jlr
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

13      30   A 20-MAR-24       4158803    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
30_0o2m6jlr
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

18      31   A 20-MAR-24       4158871    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
31_0p2m6jrb
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

17      32   A 20-MAR-24       4158886    20-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
32_0q2m6jsi
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1
```

8. 通过Oracle RMAN命令提示符报告架构、以观察当前VLDB数据文件是否位于主存储上。

```
RMAN> report schema;

Report of database schema for database with db_unique_name NTAP1

List of Permanent Datafiles
===========================
File Size(MB) Tablespace           RB segs Datafile Name
---- -------- -------------------- ------- ------------------------
1    1060     SYSTEM               YES
```

```
/u02/oradata/NTAP1/system01.dbf
3    1000     SYSAUX               NO
/u02/oradata/NTAP1/sysaux01.dbf
4    695      UNDOTBS1             YES
/u02/oradata/NTAP1/undotbs01.dbf
5    400      PDB$SEED:SYSTEM      NO
/u02/oradata/NTAP1/pdbseed/system01.dbf
6    440      PDB$SEED:SYSAUX      NO
/u02/oradata/NTAP1/pdbseed/sysaux01.dbf
7    5        USERS                NO
/u02/oradata/NTAP1/users01.dbf
8    235      PDB$SEED:UNDOTBS1    NO
/u02/oradata/NTAP1/pdbseed/undotbs01.dbf
9    410      NTAP1_PDB1:SYSTEM    YES
/u02/oradata/NTAP1/NTAP1_pdb1/system01.dbf
10   520      NTAP1_PDB1:SYSAUX    NO
/u02/oradata/NTAP1/NTAP1_pdb1/sysaux01.dbf
11   580      NTAP1_PDB1:UNDOTBS1  YES
/u02/oradata/NTAP1/NTAP1_pdb1/undotbs01.dbf
12   5        NTAP1_PDB1:USERS     NO
/u02/oradata/NTAP1/NTAP1_pdb1/users01.dbf
13   410      NTAP1_PDB2:SYSTEM    YES
/u02/oradata/NTAP1/NTAP1_pdb2/system01.dbf
14   500      NTAP1_PDB2:SYSAUX    NO
/u02/oradata/NTAP1/NTAP1_pdb2/sysaux01.dbf
15   235      NTAP1_PDB2:UNDOTBS1  YES
/u02/oradata/NTAP1/NTAP1_pdb2/undotbs01.dbf
16   5        NTAP1_PDB2:USERS     NO
/u02/oradata/NTAP1/NTAP1_pdb2/users01.dbf
17   410      NTAP1_PDB3:SYSTEM    YES
/u02/oradata/NTAP1/NTAP1_pdb3/system01.dbf
18   500      NTAP1_PDB3:SYSAUX    NO
/u02/oradata/NTAP1/NTAP1_pdb3/sysaux01.dbf
19   235      NTAP1_PDB3:UNDOTBS1  YES
/u02/oradata/NTAP1/NTAP1_pdb3/undotbs01.dbf
20   5        NTAP1_PDB3:USERS     NO
/u02/oradata/NTAP1/NTAP1_pdb3/users01.dbf
21   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_01.dbf
22   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_02.dbf
23   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_03.dbf
24   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_04.dbf
25   31744    NTAP1_PDB1:SOE       NO
```

```
/u02/oradata/NTAP1/NTAP1_pdb1/soe_05.dbf
26   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_06.dbf
27   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_07.dbf
28   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_08.dbf
29   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_09.dbf
30   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_10.dbf
31   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_11.dbf
32   31744    NTAP1_PDB1:SOE       NO
/u02/oradata/NTAP1/NTAP1_pdb1/soe_12.dbf

List of Temporary Files
=======================
File Size(MB) Tablespace           Maxsize(MB) Tempfile Name
---- -------- -------------------- ----------- --------------------
1    123      TEMP                 32767
/u02/oradata/NTAP1/temp01.dbf
2    123      PDB$SEED:TEMP        32767
/u02/oradata/NTAP1/pdbseed/temp012024-03-18_16-07-32-463-PM.dbf
3    31744    NTAP1_PDB1:TEMP      32767
/u02/oradata/NTAP1/NTAP1_pdb1/temp01.dbf
4    123      NTAP1_PDB2:TEMP      32767
/u02/oradata/NTAP1/NTAP1_pdb2/temp01.dbf
5    123      NTAP1_PDB3:TEMP      32767
/u02/oradata/NTAP1/NTAP1_pdb3/temp01.dbf
6    31744    NTAP1_PDB1:TEMP      31744
/u02/oradata/NTAP1/NTAP1_pdb1/temp02.dbf

RMAN>
```

9. 验证从操作系统NFS挂载点复制的数据库映像。

```
[oracle@ora-01 ~]$ ls -l /nfsanf/oracopy
total 399482176
-rw-r----- 1 oracle oinstall    11600384 Mar 20 21:44 bct_ntap1.ctf
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 18:03 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-21_0f2m6brl
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 18:01 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-22_0g2m6brl
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 18:03 data_D-
```

```
NTAP1_I-2441823937_TS-SOE_FNO-23_0h2m6brl
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 18:02 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-24_0i2m6brl
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 19:08 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-25_0j2m6fol
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 19:08 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-26_0k2m6fot
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 19:11 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-27_0l2m6frc
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 19:12 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-28_0m2m6fsu
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 20:14 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-29_0n2m6jlr
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 20:16 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-30_0o2m6jlr
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 20:20 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-31_0p2m6jrb
-rw-r----- 1 oracle oinstall 33286004736 Mar 20 20:19 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-32_0q2m6jsi
-rw-r----- 1 oracle oinstall   545267712 Mar 20 20:20 data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-10_0v2m6nqs
-rw-r----- 1 oracle oinstall   524296192 Mar 20 20:21 data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-14_102m6nr3
-rw-r----- 1 oracle oinstall   524296192 Mar 20 20:21 data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-18_112m6nrt
-rw-r----- 1 oracle oinstall  1048584192 Mar 20 20:19 data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-3_0s2m6nl1
-rw-r----- 1 oracle oinstall   461381632 Mar 20 20:21 data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-6_122m6nti
-rw-r----- 1 oracle oinstall  1111498752 Mar 20 20:17 data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-1_0r2m6nhk
-rw-r----- 1 oracle oinstall   429924352 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-13_142m6ntp
-rw-r----- 1 oracle oinstall   429924352 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-17_152m6nts
-rw-r----- 1 oracle oinstall   419438592 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-5_162m6nuc
-rw-r----- 1 oracle oinstall   429924352 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-9_132m6ntm
-rw-r----- 1 oracle oinstall   608182272 Mar 20 20:21 data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-11_0u2m6nqs
-rw-r----- 1 oracle oinstall   246423552 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-15_182m6nvs
-rw-r----- 1 oracle oinstall   246423552 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-19_192m6nvv
-rw-r----- 1 oracle oinstall   728768512 Mar 20 20:19 data_D-
```

```
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-4_0t2m6nml
-rw-r----- 1 oracle oinstall   246423552 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-8_172m6nvr
-rw-r----- 1 oracle oinstall     5251072 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-USERS_FNO-12_1b2m6o0e
-rw-r----- 1 oracle oinstall     5251072 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-USERS_FNO-16_1c2m6o0k
-rw-r----- 1 oracle oinstall     5251072 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-USERS_FNO-20_1d2m6o0k
-rw-r----- 1 oracle oinstall     5251072 Mar 20 20:22 data_D-
NTAP1_I-2441823937_TS-USERS_FNO-7_1a2m6o01
[oracle@ora-01 ~]$
```

至此、Oracle VLDB备用映像副本备份和合并的设置完成。

将**Oracle VLDB**切换到映像副本、以便快速恢复

如果因主存储问题描述发生故障(例如数据丢失或损坏)、则可以快速将数据库切换到ANF NFS挂载上的映像副本、并将其恢复到当前状态、而无需还原数据库。消除介质还原可显著加快VLDB的数据库恢复速度。此使用情形假定Oracle VLDB数据库服务器完好无损、并且数据库控制文件、归档日志和当前日志均可用于恢复。

1. 在切换之前、以Oracle用户身份登录到Azure主VLDB服务器主机并创建测试表。

```
[oracle@ora-01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Mar 21 15:13:52
2024
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> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 NTAP1_PDB1                     READ WRITE NO
         4 NTAP1_PDB2                     READ WRITE NO
         5 NTAP1_PDB3                     READ WRITE NO
SQL> alter session set container=ntap1_pdb1;

Session altered.

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

Table created.

SQL> insert into test values(1, sysdate, 'test oracle incremental
merge switch to copy');

1 row created.

SQL> commit;

Commit complete.
```

```
SQL> select * from test;

        ID
----------
DT
---------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------
------------
         1
21-MAR-24 03.15.03.000000 PM
test oracle incremental merge switch to copy
```

2. 通过关闭中止数据库、然后在挂载阶段启动Oracle来模拟故障。

```
SQL> shutdown abort;
ORACLE instance shut down.
SQL> startup mount;
ORACLE instance started.

Total System Global Area 6442449688 bytes
Fixed Size                  9177880 bytes
Variable Size            1325400064 bytes
Database Buffers         5100273664 bytes
Redo Buffers                7598080 bytes
Database mounted.
SQL> exit
```

3. 作为Oracle用户、通过RMAN连接到Oracle数据库、以切换要复制的数据库。

```
[oracle@ora-01 ~]$ rman target / nocatalog

Recovery Manager: Release 19.0.0.0.0 - Production on Thu Mar 21
15:20:58 2024
Version 19.18.0.0.0

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

connected to target database: NTAP1 (DBID=2441823937, not open)
using target database control file instead of recovery catalog
```

```
RMAN> switch database to copy;

datafile 1 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-1_0r2m6nhk"
datafile 3 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-3_0s2m6nl1"
datafile 4 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-4_0t2m6nml"
datafile 5 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-5_162m6nuc"
datafile 6 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-6_122m6nti"
datafile 7 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-USERS_FNO-7_1a2m6o01"
datafile 8 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-8_172m6nvr"
datafile 9 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-9_132m6ntm"
datafile 10 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-10_0v2m6nqs"
datafile 11 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-11_0u2m6nqs"
datafile 12 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-USERS_FNO-12_1b2m6o0e"
datafile 13 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-13_142m6ntp"
datafile 14 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-14_102m6nr3"
datafile 15 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-15_182m6nvs"
datafile 16 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-USERS_FNO-16_1c2m6o0k"
datafile 17 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-17_152m6nts"
datafile 18 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-18_112m6nrt"
datafile 19 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-19_192m6nvv"
datafile 20 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-USERS_FNO-20_1d2m6o0k"
datafile 21 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-21_0f2m6brl"
datafile 22 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-22_0g2m6brl"
datafile 23 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-23_0h2m6brl"
```

```
datafile 24 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-24_0i2m6brl"
datafile 25 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-25_0j2m6fol"
datafile 26 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-26_0k2m6fot"
datafile 27 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-27_0l2m6frc"
datafile 28 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-28_0m2m6fsu"
datafile 29 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-29_0n2m6jlr"
datafile 30 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-30_0o2m6jlr"
datafile 31 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-31_0p2m6jrb"
datafile 32 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-32_0q2m6jsi"
```

4. 恢复并打开数据库、使其从上次增量备份恢复到最新状态。

```
RMAN> recover database;

Starting recover at 21-MAR-24
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=392 device type=DISK
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup
set
destination for restore of datafile 00009: /nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-9_0q1sd7cm
destination for restore of datafile 00023: /nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-23_041sd6s5
destination for restore of datafile 00027: /nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-27_081sd70i
destination for restore of datafile 00031: /nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-31_0c1sd74u
destination for restore of datafile 00034: /nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-34_0f1sd788
channel ORA_DISK_1: reading from backup piece
/nfsanf/oracopy/321sfous_98_1_1
channel ORA_DISK_1: piece handle=/nfsanf/oracopy/321sfous_98_1_1
tag=ORACOPYBKUPONANF_LEVEL_0
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:01
```

```
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup
set
destination for restore of datafile 00010: /nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-10_0k1sd7bb
destination for restore of datafile 00021: /nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-21_021sd6pv
destination for restore of datafile 00025: /nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-25_061sd6uc
.
.
.
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup
set
destination for restore of datafile 00016: /nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-USERS_FNO-16_121sd7dn
channel ORA_DISK_1: reading from backup piece
/nfsanf/oracopy/3i1sfov0_114_1_1
channel ORA_DISK_1: piece handle=/nfsanf/oracopy/3i1sfov0_114_1_1
tag=ORACOPYBKUPONANF_LEVEL_0
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:01
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup
set
destination for restore of datafile 00020: /nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-USERS_FNO-20_131sd7do
channel ORA_DISK_1: reading from backup piece
/nfsanf/oracopy/3j1sfov0_115_1_1
channel ORA_DISK_1: piece handle=/nfsanf/oracopy/3j1sfov0_115_1_1
tag=ORACOPYBKUPONANF_LEVEL_0
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:01

starting media recovery
media recovery complete, elapsed time: 00:00:01

Finished recover at 21-MAR-24

RMAN> alter database open;

Statement processed

RMAN>
```

5. 在恢复后从sqlplus检查数据库结构、观察所有VLDB数据文件(控制、临时和当前日志文件除外)现在都已切换到ANF NFS文件系统上的副本。

```
SQL> select name from v$datafile
  2  union
  3  select name from v$tempfile
  4  union
  5  select name from v$controlfile
  6  union
  7* select member from v$logfile
SQL> /

NAME
--------------------------------------------------------------------------------
------------
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-21_0f2m6brl
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-22_0g2m6brl
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-23_0h2m6brl
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-24_0i2m6brl
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-25_0j2m6fol
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-26_0k2m6fot
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-27_0l2m6frc
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-28_0m2m6fsu
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-29_0n2m6jlr
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-30_0o2m6jlr
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-31_0p2m6jrb

NAME
--------------------------------------------------------------------------------
------------
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-32_0q2m6jsi
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSAUX_FNO-10_0v2m6nqs
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSAUX_FNO-14_102m6nr3
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSAUX_FNO-18_112m6nrt
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSAUX_FNO-3_0s2m6nl1
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSAUX_FNO-6_122m6nti
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSTEM_FNO-13_142m6ntp
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSTEM_FNO-17_152m6nts
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSTEM_FNO-1_0r2m6nhk
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSTEM_FNO-5_162m6nuc
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSTEM_FNO-9_132m6ntm

NAME
--------------------------------------------------------------------------------
------------
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-UNDOTBS1_FNO-
```

```
11_0u2m6nqs
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-UNDOTBS1_FNO-
15_182m6nvs
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-UNDOTBS1_FNO-
19_192m6nvv
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-UNDOTBS1_FNO-4_0t2m6nml
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-UNDOTBS1_FNO-8_172m6nvr
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-USERS_FNO-12_1b2m6o0e
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-USERS_FNO-16_1c2m6o0k
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-USERS_FNO-20_1d2m6o0k
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-USERS_FNO-7_1a2m6o01
/u02/oradata/NTAP1/NTAP1_pdb1/temp01.dbf
/u02/oradata/NTAP1/NTAP1_pdb1/temp02.dbf

NAME
--------------------------------------------------------------------------
------------
/u02/oradata/NTAP1/NTAP1_pdb2/temp01.dbf
/u02/oradata/NTAP1/NTAP1_pdb3/temp01.dbf
/u02/oradata/NTAP1/control01.ctl
/u02/oradata/NTAP1/pdbseed/temp012024-03-18_16-07-32-463-PM.dbf
/u02/oradata/NTAP1/temp01.dbf
/u03/orareco/NTAP1/control02.ctl
/u03/orareco/NTAP1/onlinelog/redo01.log
/u03/orareco/NTAP1/onlinelog/redo02.log
/u03/orareco/NTAP1/onlinelog/redo03.log

42 rows selected.
```

6. 从SQL plus中、检查切换到复制之前插入的测试表的内容。

```
SQL> alter session set container=ntap1_pdb1;

Session altered.

SQL> select * from test;

        ID
----------
DT
---------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------
------------
         1
21-MAR-24 03.15.03.000000 PM
test oracle incremental merge switch to copy


SQL>
```

7. 您可以在ANF NFS挂载中长时间运行Oracle VLDB、同时保持预期的性能级别。修复主存储问题描述后、您可以通过反转增量备份合并过程并将停机时间降至最低来回滚到该主存储LUN。

从映像副本到备用数据库服务器的**Oracle VLDB**恢复

如果发生故障、并且主存储和主数据库服务器主机均丢失、则无法从原始服务器执行恢复。但是、ANF NFS文件系统上提供的Oracle数据库备份映像副本非常方便。您可以使用备份映像副本将主数据库快速恢复到备用数据库服务器(如果有)。在本节中、我们将展示此类恢复的分步过程。

1. 插入一行以测试我们之前为Oracle VLDB还原到备用主机验证创建的表。

```
SQL> insert into test values(2, sysdate, 'test recovery on a new
Azure VM host with image copy on ANF');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from test;

        ID
----------
DT
---------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------
------------
         1
21-MAR-24 03.15.03.000000 PM
test oracle incremental merge switch to copy

         2
22-MAR-24 02.22.06.000000 PM
test recovery on a new Azure VM host with image copy on ANF

        ID
----------
DT
---------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------
------------


SQL>
```

2. 以Oracle用户身份运行RMAN增量备份并合并、以将事务转储到ANF NFS挂载上的备份集。

```
[oracle@ip-172-30-15-99 ~]$ rman target / nocatalog

Recovery Manager: Release 19.0.0.0.0 - Production on Tue May 30
17:26:03 2023
Version 19.18.0.0.0

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

connected to target database: NTAP1 (DBID=2441823937)
using target database control file instead of recovery catalog

RMAN> @rman_bkup_merge.cmd
```

3. 关闭主VLDB服务器主机、以模拟存储和数据库服务器主机完全故障。
4. 在操作系统和版本相同的备用数据库服务器ora-02上、应将操作系统内核作为主VLDB服务器主机进行修补。此外、使用纯软件选项在备用数据库服务器上安装和配置了相同版本的Oracle和修补程序。
5. 将Oracle环境配置为类似于主VLDB服务器ora_01、例如oratab和Oracle用户.bash_profile等 最好将这些文件备份到ANF NFS挂载点。
6. 然后、将ANF NFS文件系统上的Oracle数据库备份映像副本挂载到备用数据库服务器上进行恢复。以下步骤演示了流程详细信息。

   以azueruser身份创建挂载点。

```
sudo mkdir /nfsanf
```

   以azureuser身份挂载用于存储Oracle VLDB备份映像副本的NFS卷。

```
sudo mount 172.30.136.68:/ora-01-u02-copy /nfsanf -o
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=262144,wsize=262144,noin
tr
```

7. 验证ANF NFS挂载点上的Oracle数据库备份映像副本。

```
[oracle@ora-02 ~]$ ls -ltr /nfsanf/oracopy/
total 400452728
-rw-r-----. 1 oracle oinstall   461381632 Mar 21 23:47 data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-6_242m9oan
-rw-r-----. 1 oracle oinstall   419438592 Mar 21 23:49 data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-5_282m9oem
```

```
-rw-r-----. 1 oracle oinstall   246423552 Mar 21 23:49 data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-8_292m9oem
-rw-r-----. 1 oracle oinstall    21438464 Mar 22 14:35
2h2mbccv_81_1_1
-rw-r-----. 1 oracle oinstall    17956864 Mar 22 14:35
2i2mbcd0_82_1_1
-rw-r-----. 1 oracle oinstall    17956864 Mar 22 14:35
2j2mbcd1_83_1_1
-rw-r-----. 1 oracle oinstall    15245312 Mar 22 14:35
2k2mbcd3_84_1_1
-rw-r-----. 1 oracle oinstall     1638400 Mar 22 14:35
2m2mbcdn_86_1_1
-rw-r-----. 1 oracle oinstall    40042496 Mar 22 14:35
2l2mbcdn_85_1_1
-rw-r-----. 1 oracle oinstall    21856256 Mar 22 14:35
2n2mbcdo_87_1_1
-rw-r-----. 1 oracle oinstall     3710976 Mar 22 14:35
2o2mbcdv_88_1_1
-rw-r-----. 1 oracle oinstall     3416064 Mar 22 14:35
2p2mbcdv_89_1_1
-rw-r-----. 1 oracle oinstall     2596864 Mar 22 14:35
2r2mbce0_91_1_1
-rw-r-----. 1 oracle oinstall     2531328 Mar 22 14:35
2s2mbce1_92_1_1
-rw-r-----. 1 oracle oinstall     4718592 Mar 22 14:35
2v2mbce2_95_1_1
-rw-r-----. 1 oracle oinstall     4243456 Mar 22 14:35
302mbce2_96_1_1
-rw-r-----. 1 oracle oinstall       57344 Mar 22 14:35
312mbce3_97_1_1
-rw-r-----. 1 oracle oinstall       57344 Mar 22 14:35
322mbce3_98_1_1
-rw-r-----. 1 oracle oinstall       57344 Mar 22 14:35
332mbce3_99_1_1
-rw-r-----. 1 oracle oinstall   608182272 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-11_202m9o22
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-30_1q2m9k7a
-rw-r-----. 1 oracle oinstall   555753472 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-10_212m9o52
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-26_1m2m9g9j
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-27_1n2m9gcg
-rw-r-----. 1 oracle oinstall   429924352 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-9_252m9oc5
```

```
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-22_1i2m9cap
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-23_1j2m9cap
-rw-r-----. 1 oracle oinstall     5251072 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-USERS_FNO-12_2d2m9ofs
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-28_1o2m9gd4
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-31_1r2m9kfk
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-29_1p2m9ju6
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-32_1s2m9kgg
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-25_1l2m9g3u
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-24_1k2m9cap
-rw-r-----. 1 oracle oinstall 33286004736 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SOE_FNO-21_1h2m9cap
-rw-r-----. 1 oracle oinstall  1121984512 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-1_1t2m9nij
-rw-r-----. 1 oracle oinstall  1142956032 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-3_1u2m9nog
-rw-r-----. 1 oracle oinstall   728768512 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-4_1v2m9nu6
-rw-r-----. 1 oracle oinstall   534781952 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-14_222m9o53
-rw-r-----. 1 oracle oinstall   534781952 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-18_232m9oa8
-rw-r-----. 1 oracle oinstall   429924352 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-13_262m9oca
-rw-r-----. 1 oracle oinstall   246423552 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-15_2a2m9of6
-rw-r-----. 1 oracle oinstall   429924352 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-17_272m9oel
-rw-r-----. 1 oracle oinstall     5251072 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-USERS_FNO-7_2c2m9ofn
-rw-r-----. 1 oracle oinstall     5251072 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-USERS_FNO-16_2e2m9og8
-rw-r-----. 1 oracle oinstall   246423552 Mar 22 15:31 data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-19_2b2m9ofn
-rw-r-----. 1 oracle oinstall     5251072 Mar 22 15:32 data_D-
NTAP1_I-2441823937_TS-USERS_FNO-20_2f2m9og8
-rw-r-----. 1 oracle oinstall    76546048 Mar 22 15:37
362mbft5_102_1_1
```

```
-rw-r-----. 1 oracle oinstall    14671872 Mar 22 15:37
392mbg1i_105_1_1
-rw-r-----. 1 oracle oinstall    79462400 Mar 22 15:37
372mbftb_103_1_1
-rw-r-----. 1 oracle oinstall      917504 Mar 22 15:37
3a2mbg23_106_1_1
-rw-r-----. 1 oracle oinstall   428498944 Mar 22 15:37
352mbfst_101_1_1
-rw-r-----. 1 oracle oinstall    88702976 Mar 22 15:37
382mbftm_104_1_1
-rw-r-----. 1 oracle oinstall     5021696 Mar 22 15:37
3b2mbg2b_107_1_1
-rw-r-----. 1 oracle oinstall      278528 Mar 22 15:38
3c2mbg2f_108_1_1
-rw-r-----. 1 oracle oinstall      278528 Mar 22 15:38
3d2mbg2i_109_1_1
-rw-r-----. 1 oracle oinstall      425984 Mar 22 15:38
3f2mbg2m_111_1_1
-rw-r-----. 1 oracle oinstall      442368 Mar 22 15:38
3g2mbg2q_112_1_1
-rw-r-----. 1 oracle oinstall      278528 Mar 22 15:38
3j2mbg37_115_1_1
-rw-r-----. 1 oracle oinstall      270336 Mar 22 15:38
3k2mbg3a_116_1_1
-rw-r-----. 1 oracle oinstall       57344 Mar 22 15:38
3l2mbg3f_117_1_1
-rw-r-----. 1 oracle oinstall       57344 Mar 22 15:38
3n2mbg3k_119_1_1
-rw-r-----. 1 oracle oinstall       57344 Mar 22 15:38
3m2mbg3g_118_1_1
-rw-r-----. 1 oracle oinstall    11600384 Mar 22 15:52 bct_ntap1.ctf
[oracle@ora-02 ~]$
```

8. 验证ANF NFS挂载上可用于恢复的Oracle归档日志、并记下最后一个日志文件日志顺序号。在本例中、此值为10。我们的恢复点最高为日志顺序编号11。

```
[oracle@ora-02 ~]$ ls -ltr
/nfsanf/archlog/NTAP1/archivelog/2024_03_22
total 1429548
-r--r-----. 1 oracle oinstall 176650752 Mar 22 12:00
o1_mf_1_2__9m198x6t_.arc
-r--r-----. 1 oracle oinstall  17674752 Mar 22 14:34
o1_mf_1_3__9vn701r5_.arc
-r--r-----. 1 oracle oinstall 188782080 Mar 22 15:20
o1_mf_1_4__9y6gn5co_.arc
-r--r-----. 1 oracle oinstall 183638016 Mar 22 15:21
o1_mf_1_5__9y7p68s6_.arc
-r--r-----. 1 oracle oinstall 193106944 Mar 22 15:21
o1_mf_1_6__9y8ygtss_.arc
-r--r-----. 1 oracle oinstall 179439104 Mar 22 15:22
o1_mf_1_7__9ybjdp55_.arc
-r--r-----. 1 oracle oinstall 198815232 Mar 22 15:23
o1_mf_1_8__9yctxjgy_.arc
-r--r-----. 1 oracle oinstall 185494528 Mar 22 15:24
o1_mf_1_9__9yfrj0b1_.arc
-r--r-----. 1 oracle oinstall 134470144 Mar 22 15:29
o1_mf_1_10__9yomybbc_.arc
[oracle@ora-02 ~]$
```

9. 作为Oracle用户、将oracle_home变量设置为备用数据库服务器ora-02上的当前Oracle安装、将oracle_sid设置为主Oracle实例SID。在本示例中、此值为NTAP1。

```
[oracle@ora-02 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/NTAP2
[oracle@ora-02 ~]$ export ORACLE_SID=NTAP1
[oracle@ora-02 ~]$ export PATH=$PATH:$ORACLE_HOME/bin
```

10. 以Oracle用户身份、在$oracle_HOME/dbs目录中创建一个通用Oracle init文件、并配置适当的管理目录。最重要的是、拥有Oracle `flash recovery area` 指向主Oracle VLDB服务器中定义的ANF NFS挂载路径。 `flash recovery area` 第节介绍了配置 `Setup Oracle RMAN incremental merge to image copy on ANF`。将Oracle控制文件设置为ANF NFS文件系统。

```
vi $ORACLE_HOME/dbs/initNTAP1.ora
```

包含以下示例条目：

```
*.audit_file_dest='/u01/app/oracle/admin/NTAP1/adump'
*.audit_trail='db'
*.compatible='19.0.0'
*.control_files=('/nfsanf/oracopy/NTAP1.ctl')
*.db_block_size=8192
*.db_create_file_dest='/nfsanf/oracopy/'
*.db_domain='solutions.netapp.com'
*.db_name='NTAP1'
*.db_recovery_file_dest_size=85899345920
*.db_recovery_file_dest='/nfsanf/archlog/'
*.diagnostic_dest='/u01/app/oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=NTAP1XDB)'
*.enable_pluggable_database=true
*.local_listener='LISTENER'
*.nls_language='AMERICAN'
*.nls_territory='AMERICA'
*.open_cursors=300
*.pga_aggregate_target=1024m
*.processes=320
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_target=10240m
*.undo_tablespace='UNDOTBS1'
```

如果出现差异、应将上述init文件替换为从主Oracle VLDB服务器还原的备份init文件。

11. 以Oracle用户身份启动RMAN、以便在备用数据库服务器主机上运行Oracle恢复。首先、在中启动Oracle实例 `nomount` 状态。

```
[oracle@ora-02 ~]$ rman target / nocatalog

Recovery Manager: Release 19.0.0.0.0 - Production on Fri Mar 22
16:02:55 2024
Version 19.18.0.0.0

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

connected to target database (not started)

RMAN> startup nomount;

Oracle instance started

Total System Global Area   10737418000 bytes

Fixed Size                     9174800 bytes
Variable Size               1577058304 bytes
Database Buffers            9126805504 bytes
Redo Buffers                  24379392 bytes
```

12. 设置数据库ID。数据库ID可从ANF NFS挂载点上映像副本的Oracle文件名中检索。

```
RMAN> set dbid = 2441823937;

executing command: SET DBID
```

13. 从自动备份还原控制文件。如果启用了Oracle控制文件和spfile自动备份、则它们会在每个增量备份和合并周期中进行备份。如果有多个副本可用、则会还原最新备份。

```
RMAN> restore controlfile from autobackup;

Starting restore at 22-MAR-24
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=2 device type=DISK

recovery area destination: /nfsanf/archlog/
database name (or database unique name) used for search: NTAP1
channel ORA_DISK_1: AUTOBACKUP
/nfsanf/archlog/NTAP1/autobackup/2024_03_22/o1_mf_s_1164296325__9z77
zyxb_.bkp found in the recovery area
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20240322
channel ORA_DISK_1: restoring control file from AUTOBACKUP
/nfsanf/archlog/NTAP1/autobackup/2024_03_22/o1_mf_s_1164296325__9z77
zyxb_.bkp
channel ORA_DISK_1: control file restore from AUTOBACKUP complete
output file name=/nfsanf/oracopy/NTAP1.ctl
Finished restore at 22-MAR-24
```

14. 将init文件从spfile还原到/tmp文件夹、以便稍后更新参数文件以与主VLDB匹配。

```
RMAN> restore spfile to pfile '/tmp/archive/initNTAP1.ora' from
autobackup;

Starting restore at 22-MAR-24
using channel ORA_DISK_1

recovery area destination: /nfsanf/archlog/
database name (or database unique name) used for search: NTAP1
channel ORA_DISK_1: AUTOBACKUP
/nfsanf/archlog/NTAP1/autobackup/2024_03_22/o1_mf_s_1164296325__9z77
zyxb_.bkp found in the recovery area
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20240322
channel ORA_DISK_1: restoring spfile from AUTOBACKUP
/nfsanf/archlog/NTAP1/autobackup/2024_03_22/o1_mf_s_1164296325__9z77
zyxb_.bkp
channel ORA_DISK_1: SPFILE restore from AUTOBACKUP complete
Finished restore at 22-MAR-24
```

15. 挂载控制文件并验证数据库备份映像副本。

```
RMAN> alter database mount;

released channel: ORA_DISK_1
```

```
Statement processed

RMAN> list copy of database tag 'ORACOPYBKUPONANF_LEVEL_0';

List of Datafile Copies
=======================

Key     File S Completion Time Ckp SCN    Ckp Time        Sparse
------- ---- - -------------- ---------- --------------- ------
82      1    A 22-MAR-24       4598427    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-1_1t2m9nij
        Tag: ORACOPYBKUPONANF_LEVEL_0

83      3    A 22-MAR-24       4598423    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-3_1u2m9nog
        Tag: ORACOPYBKUPONANF_LEVEL_0

84      4    A 22-MAR-24       4598431    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-4_1v2m9nu6
        Tag: ORACOPYBKUPONANF_LEVEL_0

58      5    A 21-MAR-24       2379694    18-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-5_282m9oem
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

52      6    A 21-MAR-24       2379694    18-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-6_242m9oan
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

90      7    A 22-MAR-24       4598462    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-7_2c2m9ofn
        Tag: ORACOPYBKUPONANF_LEVEL_0

59      8    A 21-MAR-24       2379694    18-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-8_292m9oem
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED
```

```
71      9    A 22-MAR-24       4598313    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-9_252m9oc5
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

68      10   A 22-MAR-24       4598308    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-10_212m9o52
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

66      11   A 22-MAR-24       4598304    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-11_202m9o22
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

74      12   A 22-MAR-24       4598318    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-12_2d2m9ofs
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

86      13   A 22-MAR-24       4598445    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-13_262m9oca
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 4, PDB Name: NTAP1_PDB2

85      14   A 22-MAR-24       4598437    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-14_222m9o53
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 4, PDB Name: NTAP1_PDB2

87      15   A 22-MAR-24       4598454    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-15_2a2m9of6
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 4, PDB Name: NTAP1_PDB2

89      16   A 22-MAR-24       4598466    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-16_2e2m9og8
        Tag: ORACOPYBKUPONANF_LEVEL_0
```

```
        Container ID: 4, PDB Name: NTAP1_PDB2

91      17   A 22-MAR-24       4598450    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSTEM_FNO-17_272m9oel
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 5, PDB Name: NTAP1_PDB3

88      18   A 22-MAR-24       4598441    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
SYSAUX_FNO-18_232m9oa8
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 5, PDB Name: NTAP1_PDB3

92      19   A 22-MAR-24       4598458    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
UNDOTBS1_FNO-19_2b2m9ofn
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 5, PDB Name: NTAP1_PDB3

93      20   A 22-MAR-24       4598470    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-
USERS_FNO-20_2f2m9og8
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 5, PDB Name: NTAP1_PDB3

81      21   A 22-MAR-24       4598318    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
21_1h2m9cap
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

72      22   A 22-MAR-24       4598304    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
22_1i2m9cap
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

73      23   A 22-MAR-24       4598308    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
23_1j2m9cap
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

80      24   A 22-MAR-24       4598313    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
```

```
24_1k2m9cap
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

79      25   A 22-MAR-24       4598318    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
25_1l2m9g3u
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

69      26   A 22-MAR-24       4598304    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
26_1m2m9g9j
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

70      27   A 22-MAR-24       4598308    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
27_1n2m9gcg
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

75      28   A 22-MAR-24       4598313    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
28_1o2m9gd4
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

77      29   A 22-MAR-24       4598318    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
29_1p2m9ju6
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

67      30   A 22-MAR-24       4598304    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
30_1q2m9k7a
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1

76      31   A 22-MAR-24       4598308    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
31_1r2m9kfk
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1
```

```
78      32   A 22-MAR-24       4598313    22-MAR-24       NO
        Name: /nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-
32_1s2m9kgg
        Tag: ORACOPYBKUPONANF_LEVEL_0
        Container ID: 3, PDB Name: NTAP1_PDB1
```

16. 切换要复制的数据库、以便在不还原数据库的情况下运行恢复。

```
RMAN> switch database to copy;

Starting implicit crosscheck backup at 22-MAR-24
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=12 device type=DISK
Crosschecked 33 objects
Finished implicit crosscheck backup at 22-MAR-24

Starting implicit crosscheck copy at 22-MAR-24
using channel ORA_DISK_1
Crosschecked 31 objects
Finished implicit crosscheck copy at 22-MAR-24

searching for all files in the recovery area
cataloging files...
cataloging done

List of Cataloged Files
=======================
File Name:
/nfsanf/archlog/NTAP1/autobackup/2024_03_20/o1_mf_s_1164140565__5g56
ypks_.bkp
File Name:
/nfsanf/archlog/NTAP1/autobackup/2024_03_22/o1_mf_s_1164296325__9z77
zyxb_.bkp

datafile 1 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-1_1t2m9nij"
datafile 3 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-3_1u2m9nog"
datafile 4 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-4_1v2m9nu6"
datafile 5 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-5_282m9oem"
datafile 6 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-6_242m9oan"
datafile 7 switched to datafile copy "/nfsanf/oracopy/data_D-
```

```
NTAP1_I-2441823937_TS-USERS_FNO-7_2c2m9ofn"
datafile 8 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-8_292m9oem"
datafile 9 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-9_252m9oc5"
datafile 10 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-10_212m9o52"
datafile 11 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-11_202m9o22"
datafile 12 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-USERS_FNO-12_2d2m9ofs"
datafile 13 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-13_262m9oca"
datafile 14 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-14_222m9o53"
datafile 15 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-15_2a2m9of6"
datafile 16 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-USERS_FNO-16_2e2m9og8"
datafile 17 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSTEM_FNO-17_272m9oel"
datafile 18 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SYSAUX_FNO-18_232m9oa8"
datafile 19 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-UNDOTBS1_FNO-19_2b2m9ofn"
datafile 20 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-USERS_FNO-20_2f2m9og8"
datafile 21 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-21_1h2m9cap"
datafile 22 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-22_1i2m9cap"
datafile 23 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-23_1j2m9cap"
datafile 24 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-24_1k2m9cap"
datafile 25 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-25_1l2m9g3u"
datafile 26 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-26_1m2m9g9j"
datafile 27 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-27_1n2m9gcg"
datafile 28 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-28_1o2m9gd4"
datafile 29 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-29_1p2m9ju6"
datafile 30 switched to datafile copy "/nfsanf/oracopy/data_D-
```

```
NTAP1_I-2441823937_TS-SOE_FNO-30_1q2m9k7a"
datafile 31 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-31_1r2m9kfk"
datafile 32 switched to datafile copy "/nfsanf/oracopy/data_D-
NTAP1_I-2441823937_TS-SOE_FNO-32_1s2m9kgg"
```

17. 在闪存恢复区域运行Oracle恢复、直到最后一个可用归档日志为止。

```
RMAN> run {
2> set until sequence=11;
3> recover database;
4> }

executing command: SET until clause

Starting recover at 22-MAR-24
using channel ORA_DISK_1

starting media recovery

archived log for thread 1 with sequence 4 is already on disk as file
/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_4__9y6gn5co_.arc
archived log for thread 1 with sequence 5 is already on disk as file
/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_5__9y7p68s6_.arc
archived log for thread 1 with sequence 6 is already on disk as file
/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_6__9y8ygtss_.arc
archived log for thread 1 with sequence 7 is already on disk as file
/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_7__9ybjdp55_.arc
archived log for thread 1 with sequence 8 is already on disk as file
/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_8__9yctxjgy_.arc
archived log for thread 1 with sequence 9 is already on disk as file
/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_9__9yfrj0b1_.arc
archived log for thread 1 with sequence 10 is already on disk as
file
/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_10__9yomybbc_.ar
c
archived log file
name=/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_4__9y6gn5co
_.arc thread=1 sequence=4
archived log file
name=/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_5__9y7p68s6
_.arc thread=1 sequence=5
archived log file
name=/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_6__9y8ygtss
_.arc thread=1 sequence=6
```

```
archived log file
name=/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_7__9ybjdp55
_.arc thread=1 sequence=7
archived log file
name=/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_8__9yctxjgy
_.arc thread=1 sequence=8
archived log file
name=/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_9__9yfrj0b1
_.arc thread=1 sequence=9
archived log file
name=/nfsanf/archlog/NTAP1/archivelog/2024_03_22/o1_mf_1_10__9yomybb
c_.arc thread=1 sequence=10
media recovery complete, elapsed time: 00:01:17
Finished recover at 22-MAR-24

RMAN> exit


Recovery Manager complete.
```

要加快恢复速度、请使用recovery _parlism参数启用并行会话、或者在恢复命令中指定并行程度以进行数据库恢复： `RECOVER DATABASE PARALLEL (DEGREE d INSTANCES DEFAULT);`。通常、并行度应等于主机上的CPU核数。

18. 退出RMAN、以Oracle用户身份通过sqlplus登录到Oracle、以便在恢复不完整后打开数据库并重置日志。

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

NAME      OPEN_MODE
--------- --------------------
NTAP1     MOUNTED

SQL> select instance_name, host_name from v$instance;

INSTANCE_NAME
----------------
HOST_NAME
----------------------------------------------------------------
NTAP1
ora-02


SQL>
```

```
SQL> select member from v$logfile;

MEMBER
--------------------------------------------------------------------------------
------------
/u03/orareco/NTAP1/onlinelog/redo03.log
/u03/orareco/NTAP1/onlinelog/redo02.log
/u03/orareco/NTAP1/onlinelog/redo01.log

SQL> alter database rename file
'/u03/orareco/NTAP1/onlinelog/redo01.log' to
'/nfsanf/oracopy/redo01.log';

Database altered.

SQL> alter database rename file
'/u03/orareco/NTAP1/onlinelog/redo02.log' to
'/nfsanf/oracopy/redo02.log';

Database altered.

SQL> alter database rename file
'/u03/orareco/NTAP1/onlinelog/redo03.log' to
'/nfsanf/oracopy/redo03.log';

Database altered.

SQL> alter database open resetlogs;

Database altered.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 NTAP1_PDB1                     READ WRITE NO
         4 NTAP1_PDB2                     READ WRITE NO
         5 NTAP1_PDB3                     READ WRITE NO
```

19. 验证已还原到新主机的数据库结构、以及在主VLDB发生故障之前插入的测试行。

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

NAME
--------------------------------------------------------------------------------
```

```
------------
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSTEM_FNO-1_1t2m9nij
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSAUX_FNO-3_1u2m9nog
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-UNDOTBS1_FNO-4_1v2m9nu6
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSTEM_FNO-5_282m9oem
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSAUX_FNO-6_242m9oan
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-USERS_FNO-7_2c2m9ofn
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-UNDOTBS1_FNO-8_292m9oem
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSTEM_FNO-9_252m9oc5
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSAUX_FNO-10_212m9o52
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-UNDOTBS1_FNO-
11_202m9o22
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-USERS_FNO-12_2d2m9ofs

NAME
--------------------------------------------------------------------
------------
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSTEM_FNO-13_262m9oca
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSAUX_FNO-14_222m9o53
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-UNDOTBS1_FNO-
15_2a2m9of6
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-USERS_FNO-16_2e2m9og8
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSTEM_FNO-17_272m9oel
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SYSAUX_FNO-18_232m9oa8
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-UNDOTBS1_FNO-
19_2b2m9ofn
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-USERS_FNO-20_2f2m9og8
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-21_1h2m9cap
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-22_1i2m9cap
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-23_1j2m9cap

NAME
--------------------------------------------------------------------
------------
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-24_1k2m9cap
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-25_1l2m9g3u
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-26_1m2m9g9j
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-27_1n2m9gcg
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-28_1o2m9gd4
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-29_1p2m9ju6
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-30_1q2m9k7a
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-31_1r2m9kfk
/nfsanf/oracopy/data_D-NTAP1_I-2441823937_TS-SOE_FNO-32_1s2m9kgg

31 rows selected.
```

```
SQL> select member from v$logfile;

MEMBER
--------------------------------------------------------------------------------
------------
/nfsanf/oracopy/redo03.log
/nfsanf/oracopy/redo02.log
/nfsanf/oracopy/redo01.log

SQL> select name from v$controlfile;

NAME
--------------------------------------------------------------------------------
------------
/nfsanf/oracopy/NTAP1.ctl

SQL> alter session set container=ntap1_pdb1;

Session altered.

SQL> select * from test;

        ID
----------
DT
--------------------------------------------------------------------------------
-------
EVENT
--------------------------------------------------------------------------------
------------
         1
21-MAR-24 03.15.03.000000 PM
test oracle incremental merge switch to copy

         2
22-MAR-24 02.22.06.000000 PM
test recovery on a new Azure VM host with image copy on ANF
```

20. 删除无效的临时文件并将新的临时文件添加到临时表空间。

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

NAME
--------------------------------------------------------------------------------
------------
```

```
/u02/oradata/NTAP1/NTAP1_pdb1/temp01.dbf
/u02/oradata/NTAP1/NTAP1_pdb1/temp02.dbf

SQL> alter tablespace temp add tempfile
'/nfsanf/oracopy/ntap1_pdb1_temp01.dbf' size 100M;

Tablespace altered.

SQL> select name from v$tempfile;

NAME
--------------------------------------------------------------------------------
------------
/u02/oradata/NTAP1/NTAP1_pdb1/temp01.dbf
/u02/oradata/NTAP1/NTAP1_pdb1/temp02.dbf
/nfsanf/oracopy/ntap1_pdb1_temp01.dbf

SQL> alter database tempfile
'/u02/oradata/NTAP1/NTAP1_pdb1/temp01.dbf' offline;

Database altered.

SQL> alter database tempfile
'/u02/oradata/NTAP1/NTAP1_pdb1/temp01.dbf' drop;

Database altered.

SQL> alter database tempfile
'/u02/oradata/NTAP1/NTAP1_pdb1/temp02.dbf' offline;

Database altered.

SQL> alter database tempfile
'/u02/oradata/NTAP1/NTAP1_pdb1/temp02.dbf' drop;

Database altered.

SQL> select name from v$tempfile;

NAME
--------------------------------------------------------------------------------
------------
/nfsanf/oracopy/ntap1_pdb1_temp01.dbf

SQL>
```

21. 其他恢复后任务

```
- Add ANF NFS mount to fstab so that the NFS file system will be
mounted when DB server host rebooted.

        As azureuser, sudo vi /etc/fstab and add following entry:

        172.30.136.68:/ora-01-u02-copy       /nfsanf        nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=262144,wsize=262144,noin
tr  0       0

- Update the Oracle init file from primary databse init file backup
that is restored to /tmp/archive and create spfile as needed.
```

这样、Oracle VLDB数据库便可从ANF NFS文件系统上的备份映像副本恢复到备用数据库服务器主机。

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- RMAN：合并增量备份策略(文档ID 745798.1)

  "https://support.oracle.com/knowledge/Oracle%20Database%20Products/745798_1.html"

- RMAN备份和恢复用户指南

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/bradv/getting-started-rman.html"

- Azure NetApp Files

  "https://azure.microsoft.com/en-us/products/netapp"

TR-4987：《在采用NFS的Azure NetApp Files上简化的自动化Oracle部署》

NetApp公司Allen Cao、Niyaz Mohamed

目的

在云中运行性能密集型和延迟敏感型Oracle工作负载可能会带来挑战。借助Azure NetApp Files (ANF)、企业业务部门(LOB)和存储专业人员可以轻松迁移和运行要求苛刻的Oracle工作负载、而无需更改代码。Azure NetApp Files广泛用作各种情形下的底层共享文件存储服务、例如、在内部部署或将Oracle数据库迁移(迁移)到Azure等情形下。

本文档演示了如何使用Azure NetApp Files自动化通过NFS挂载简化Oracle数据库在Oracle中的部署。Oracle数据库部署在启用了Oracle DNFS协议的容器数据库(CDB)和可插拔数据库(PDB)配置中、以提高性能。此外、可

以使用自动化PDB重新定位方法将内部Oracle单实例数据库(PDB)迁移到Azure中新部署的容器数据库中、同时最大限度地减少服务中断。此外、还提供了有关使用Azure云中的NetApp SnapCenter UI工具快速备份、还原和克隆Oracle数据库的信息。

此解决方案 可解决以下使用情形：

- 在Azure NetApp Files上自动部署Oracle容器数据库
- 在内部未命中和Azure云之间自动迁移Oracle数据库

audience

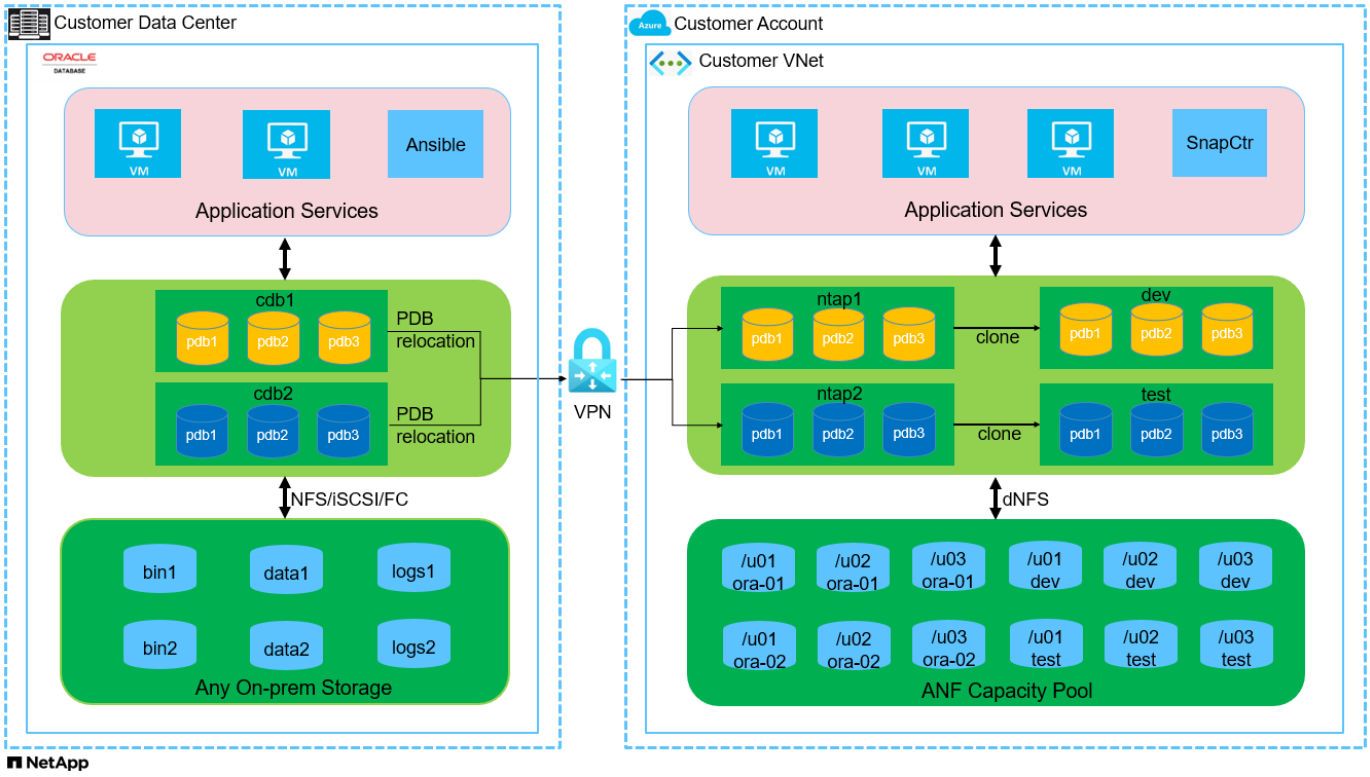
此解决方案 适用于以下人员：

- 希望在Azure NetApp Files上部署Oracle的数据库开发人员。
- 希望在Azure NetApp Files上测试Oracle工作负载的数据库解决方案架构师。
- 希望在Azure NetApp Files上部署和管理Oracle数据库的存储管理员。
- 希望在Azure NetApp Files上建立Oracle数据库的应用程序所有者。

解决方案 测试和验证环境

此解决方案的测试和验证是在实验室环境中执行的、可能与最终部署环境不匹配。请参见一节 [Key Factors for Deployment Consideration] 有关详细信息 …

架构



硬件和软件组件

* 硬件 *		
Azure NetApp Files	Azure中由Microsoft提供的最新产品	一个具有高级服务级别的容量池
适用于数据库服务器的Azure VM	standard_B4ms—4个vCPU、16 GiB	两个Linux虚拟机实例、用于并发部署
适用于SnapCenter的Azure VM	standard_B4ms—4个vCPU、16 GiB	一个Windows虚拟机实例
软件		
RedHat Linux	RHEL Linux 8.6 (LVM)- x64 Gen2	已部署RedHat订阅以进行测试
Windows服务器	2022 DataCenter；Azure Edition HotPatch - x64 Gen2	托管SnapCenter服务器
Oracle 数据库	版本19.18	已应用RU修补程序p34765931_190000_Linux-x86-64.zip
Oracle OPatch	版本12.2.0.1.36	最新修补程序p6880880_190000_Linux-x86-64.zip
SnapCenter 服务器	版本5.0	工作组部署
打开JDK	版本java-11-OpenJDK	DB VM上的SnapCenter插件要求
NFS	版本 3.0	已启用Oracle DNFS
Ansible	核心2.16.2.	Python 3.6.8

实验室环境中的**Oracle**数据库配置

* 服务器 *	* 数据库 *	**DB**存储
ORA-01	NTAP1 (NTAP1_PDB1、NTAP1_PDB2、NTAP_PDB3)	/u01、/u02、/u03 NFS挂载到ANF容量池上
ORA-02.	NTAP2 (NTAP2_PDB1、NTAP2_PDB2、NTAP2_PDB3)	/u01、/u02、/u03 NFS挂载到ANF容量池上

部署注意事项的关键因素

- *Oracle数据库存储布局。*在此自动化Oracle部署中、我们会默认为每个数据库配置三个数据库卷、以托管Oracle二进制文件、数据和日志。卷会通过NFS以/u01 -二进制、/u02 -数据、/u03 -日志的形式挂载在Oracle数据库服务器上。在/u02和/u03挂载点上配置双控制文件、以实现冗余。
- *部署多个数据库服务器。*自动化解决方案可以在一次运行的AnsablePlaybook中将一个Oracle容器数据库部署到多个数据库服务器。无论数据库服务器的数量如何、该操作手册的执行都保持不变。通过使用不同的数据库实例ID (Oracle SID)重复部署、您可以将多个容器数据库部署到一个VM实例。但是、请确保主机上有足够的内存来支持已部署的数据库。
- *DNFS配置。*通过使用DNFS (自Oracle 11g起提供)、在Azure虚拟机上运行的Oracle数据库可以比本机NFS客户端驱动更多的I/O。默认情况下、Oracle自动化部署会在NFSv3上配置DNFS。

- 分配大容量卷以加快部署速度。 ANF文件系统IO吞吐量根据卷大小进行调节。对于初始部署、分配大容量卷可以加快部署速度。之后、可以动态缩减卷的大小、而不会对应用程序造成影响。
- 数据库备份。 NetApp提供了一个SnapCenter软件套件、可通过用户友好的用户界面进行数据库备份、还原和克隆。NetApp建议实施此类管理工具、以实现快速(不到一分钟)的快照备份、快速(几分钟)的数据库还原和数据库克隆。

解决方案 部署

以下各节提供了在Azure NetApp Files上通过NFS直接挂载数据库卷自动部署Oracle 19c以及将数据库迁移到Azure VM的分步过程。

部署的前提条件

部署需要满足以下前提条件。

1. 已设置Azure帐户、并且已在Azure帐户中创建必要的vNet和网段。
2. 在Azure云门户中、将Azure Linux VM部署为Oracle数据库服务器。为Oracle数据库创建Azure NetApp Files容量池和数据库卷。为azureuser到DB服务器启用VM SSH私有/公共密钥身份验证。有关环境设置的详细信息、请参见上一节中的架构图。另见 "Azure VM和Azure NetApp Files 上的Oracle分步部署过程" 了解详细信息。

   对于部署了本地磁盘冗余的Azure VM、请确保在VM根磁盘中至少分配了128 G的空间、以便有足够的空间来暂存Oracle安装文件和添加操作系统交换文件。相应地展开/tmplv和/rootlv OS分区。确保数据库卷命名遵循vmname-u01、vmname-u02和vmname-u03约定。

   ```
   sudo lvresize -r -L +20G /dev/mapper/rootvg-rootlv
   ```

   ```
   sudo lvresize -r -L +10G /dev/mapper/rootvg-tmplv
   ```

3. 从Azure云门户中、配置Windows服务器以使用最新版本运行NetApp SnapCenter UI工具。有关详细信息、请参见以下链接： "安装 SnapCenter 服务器"
4. 将Linux VM配置为安装了最新版本的Ansv近 和Git的Ansv可 控制器节点。有关详细信息、请参见以下链接： "NetApp解决方案 自动化入门" 在第-节中
   `Setup the Ansible Control Node for CLI deployments on RHEL / CentOS` 或
   `Setup the Ansible Control Node for CLI deployments on Ubuntu / Debian`。

   只要通过ssh端口访问Azure数据库VM、则Ansensure控制器节点就可以查找内部未命中或Azure云中的位置。

5. 克隆一份适用于NFS的NetApp Oracle部署自动化工具包副本。

   ```
   git clone https://bitbucket.ngage.netapp.com/scm/ns-
   bb/na_oracle_deploy_nfs.git
   ```

6. Stage Follows Oracle 19c installation files on Azure DB VM /tmp/archive directory with 777 permission。

   ```
   installer_archives:
     - "LINUX.X64_193000_db_home.zip"
     - "p34765931_190000_Linux-x86-64.zip"
     - "p6880880_190000_Linux-x86-64.zip"
   ```

7. 观看以下视频：

借助NFS在Azure NetApp Files上简化并自动化Oracle部署

自动化参数文件

Ans可 通过预定义的参数执行数据库安装和配置任务。对于此Oracle自动化解决方案、有三个用户定义的参数文件需要用户输入才能执行操作手册。

- 主机—定义运行自动化操作手册的目标。
- vars/vars.yml—用于定义应用于所有目标的变量的全局变量文件。
- host_vars/host_name.yml—用于定义仅适用于指定目标的变量的本地变量文件。在我们的使用情形中、这些是Oracle数据库服务器。

除了这些用户定义的变量文件之外、还有多个默认变量文件包含默认参数、除非必要、否则不需要更改这些参数。以下各节介绍如何配置用户定义的变量文件。

参数文件配置

1. 可逆目标 `hosts` 文件配置：

```
# Enter Oracle servers names to be deployed one by one, follow by
each Oracle server public IP address, and ssh private key of
azureuser for the server.
[oracle]
ora-01 ansible_host=10.61.180.21 ansible_ssh_private_key_file=ora-
01.pem
ora-02 ansible_host=10.61.180.23 ansible_ssh_private_key_file=ora-
02.pem
```

2. 全局 `vars/vars.yml` 文件配置

```
#################################################################################
##
###### Oracle 19c deployment user configuration variables
######
###### Consolidate all variables from ANF, linux and oracle
######
#################################################################################
##

############################################
### ANF env specific config variables   ###
############################################

# Prerequisite to create three volumes in NetApp storage pool from
cloud dashboard with following naming convention:
# db_hostname-u01 - Oracle binary
# db_hostname-u02 - Oracle data
# db_hostname-u03 - Oracle redo
# It is important to strictly follow the name convention or the
automation will fail.

# NFS lif ip address to access database volumes in ANF storage pool
(retrievable from cloud dashboard)
nfs_lif: 172.30.136.68

############################################
### Linux env specific config variables ###
############################################

redhat_sub_username: XXXXXXXX
redhat_sub_password: XXXXXXXX


#########################################################
### DB env specific install and config variables ###
#########################################################

# Database domain name
db_domain: solutions.netapp.com

# Set initial password for all required Oracle passwords. Change
them after installation.
initial_pwd_all: XXXXXXXX
```

3. 本地数据库服务器 `host_vars/host_name.yml` 配置，如ora_01.yml、ora_02.yml …

```
# User configurable Oracle host specific parameters

# Enter container database SID. By default, a container DB is
created with 3 PDBs within the CDB
oracle_sid: NTAP1

# Enter database shared memory size or SGA. CDB is created with SGA
at 75% of memory_limit, MB. The grand total of SGA should not exceed
75% available RAM on node.
memory_limit: 8192
```

执行操作手册

自动化工具包中共有五本操作手册。每个任务执行不同的任务块、并用于不同的用途。

```
0-all_playbook.yml - execute playbooks from 1-4 in one playbook run.
1-ansible_requirements.yml - set up Ansible controller with required
libs and collections.
2-linux_config.yml - execute Linux kernel configuration on Oracle DB
servers.
4-oracle_config.yml - install and configure Oracle on DB servers and
create a container database.
5-destroy.yml - optional to undo the environment to dismantle all.
```

使用以下命令可通过三个选项运行这些操作手册。

1. 一次运行即可执行所有部署操作手册。

   ```
   ansible-playbook -i hosts 0-all_playbook.yml -u azureuser -e
   @vars/vars.yml
   ```

2. 使用1-4的数字顺序执行一次一个操作手册。

   ```
   ansible-playbook -i hosts 1-ansible_requirements.yml -u azureuser -e
   @vars/vars.yml
   ```

   ```
   ansible-playbook -i hosts 2-linux_config.yml -u azureuser -e
   @vars/vars.yml
   ```

   ```
   ansible-playbook -i hosts 4-oracle_config.yml -u azureuser -e
   @vars/vars.yml
   ```

3. 使用标记执行0-all_playbook.yml。

   ```
   ansible-playbook -i hosts 0-all_playbook.yml -u azureuser -e
   @vars/vars.yml -t ansible_requirements
   ```

   ```
   ansible-playbook -i hosts 0-all_playbook.yml -u azureuser -e
   @vars/vars.yml -t linux_config
   ```

```
ansible-playbook -i hosts 0-all_playbook.yml -u azureuser -e
@vars/vars.yml -t oracle_config
```

4. 撤消环境

```
ansible-playbook -i hosts 5-destroy.yml -u azureuser -e
@vars/vars.yml
```

执行后验证

运行此操作手册后、登录到Oracle数据库服务器VM、以验证是否已安装和配置Oracle以及是否已成功创建容器数据库。以下是在主机ora-01上验证Oracle数据库的示例。

1. 验证NFS挂载

```
[azureuser@ora-01 ~]$ cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Thu Sep 14 11:04:01 2023
#
# Accessible filesystems, by reference, are maintained under
'/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for
more info.
#
# After editing this file, run 'systemctl daemon-reload' to update
systemd
# units generated from this file.
#
/dev/mapper/rootvg-rootlv /                       xfs     defaults
0 0
UUID=268633bd-f9bb-446d-9a1d-8fca4609a1e1 /boot
xfs     defaults        0 0
UUID=89D8-B037          /boot/efi               vfat
defaults,uid=0,gid=0,umask=077,shortname=winnt 0 2
/dev/mapper/rootvg-homelv /home                   xfs     defaults
0 0
/dev/mapper/rootvg-tmplv /tmp                     xfs     defaults
0 0
/dev/mapper/rootvg-usrlv /usr                     xfs     defaults
0 0
/dev/mapper/rootvg-varlv /var                     xfs     defaults
0 0
/mnt/swapfile swap swap defaults 0 0
172.30.136.68:/ora-01-u01 /u01 nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536 0 0
172.30.136.68:/ora-01-u02 /u02 nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536 0 0
172.30.136.68:/ora-01-u03 /u03 nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536 0 0

[azureuser@ora-01 ~]$ df -h
Filesystem                Size  Used Avail Use% Mounted on
devtmpfs                  7.7G     0  7.7G   0% /dev
```

```
tmpfs                       7.8G     0  7.8G   0% /dev/shm
tmpfs                       7.8G  8.6M  7.7G   1% /run
tmpfs                       7.8G     0  7.8G   0% /sys/fs/cgroup
/dev/mapper/rootvg-rootlv    22G   17G  5.8G  74% /
/dev/mapper/rootvg-usrlv     10G  2.0G  8.1G  20% /usr
/dev/mapper/rootvg-varlv    8.0G  890M  7.2G  11% /var
/dev/sda1                   496M  106M  390M  22% /boot
/dev/mapper/rootvg-homelv 1014M   40M  975M   4% /home
/dev/sda15                  495M  5.9M  489M   2% /boot/efi
/dev/mapper/rootvg-tmplv     12G  8.4G  3.7G  70% /tmp
tmpfs                       1.6G     0  1.6G   0% /run/user/54321
172.30.136.68:/ora-01-u01   500G   11G  490G   3% /u01
172.30.136.68:/ora-01-u03   250G  1.2G  249G   1% /u03
172.30.136.68:/ora-01-u02   250G  7.1G  243G   3% /u02
tmpfs                       1.6G     0  1.6G   0% /run/user/1000
```

2. 验证Oracle侦听器

```
[azureuser@ora-01 ~]$ sudo su
[root@ora-01 azureuser]# su - oracle
Last login: Thu Feb  1 16:13:44 UTC 2024
[oracle@ora-01 ~]$ lsnrctl status listener.ntap1

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 01-FEB-2024
16:25:37

Copyright (c) 1991, 2022, Oracle.  All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=ora-
01.internal.cloudapp.net)(PORT=1521)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER.NTAP1
Version                   TNSLSNR for Linux: Version 19.0.0.0.0 -
Production
Start Date                01-FEB-2024 16:13:49
Uptime                    0 days 0 hr. 11 min. 49 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File
/u01/app/oracle/product/19.0.0/NTAP1/network/admin/listener.ora
Listener Log File         /u01/app/oracle/diag/tnslsnr/ora-
01/listener.ntap1/alert/log.xml
Listening Endpoints Summary...
```

```
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=ora-
01.hr2z2nbmhnqutdsxgscjtuxizd.jx.internal.cloudapp.net)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=ora-
01.hr2z2nbmhnqutdsxgscjtuxizd.jx.internal.cloudapp.net)(PORT=5500))(
Security=(my_wallet_directory=/u01/app/oracle/product/19.0.0/NTAP1/a
dmin/NTAP1/xdb_wallet))(Presentation=HTTP)(Session=RAW))
Services Summary...
Service "104409ac02da6352e063bb891eacf34a.solutions.netapp.com" has
1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "104412c14c2c63cae063bb891eacf64d.solutions.netapp.com" has
1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "1044174670ad63ffe063bb891eac6b34.solutions.netapp.com" has
1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "NTAP1.solutions.netapp.com" has 1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "NTAP1XDB.solutions.netapp.com" has 1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "ntap1_pdb1.solutions.netapp.com" has 1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "ntap1_pdb2.solutions.netapp.com" has 1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "ntap1_pdb3.solutions.netapp.com" has 1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
The command completed successfully
```

3. 验证Oracle数据库和DNFS

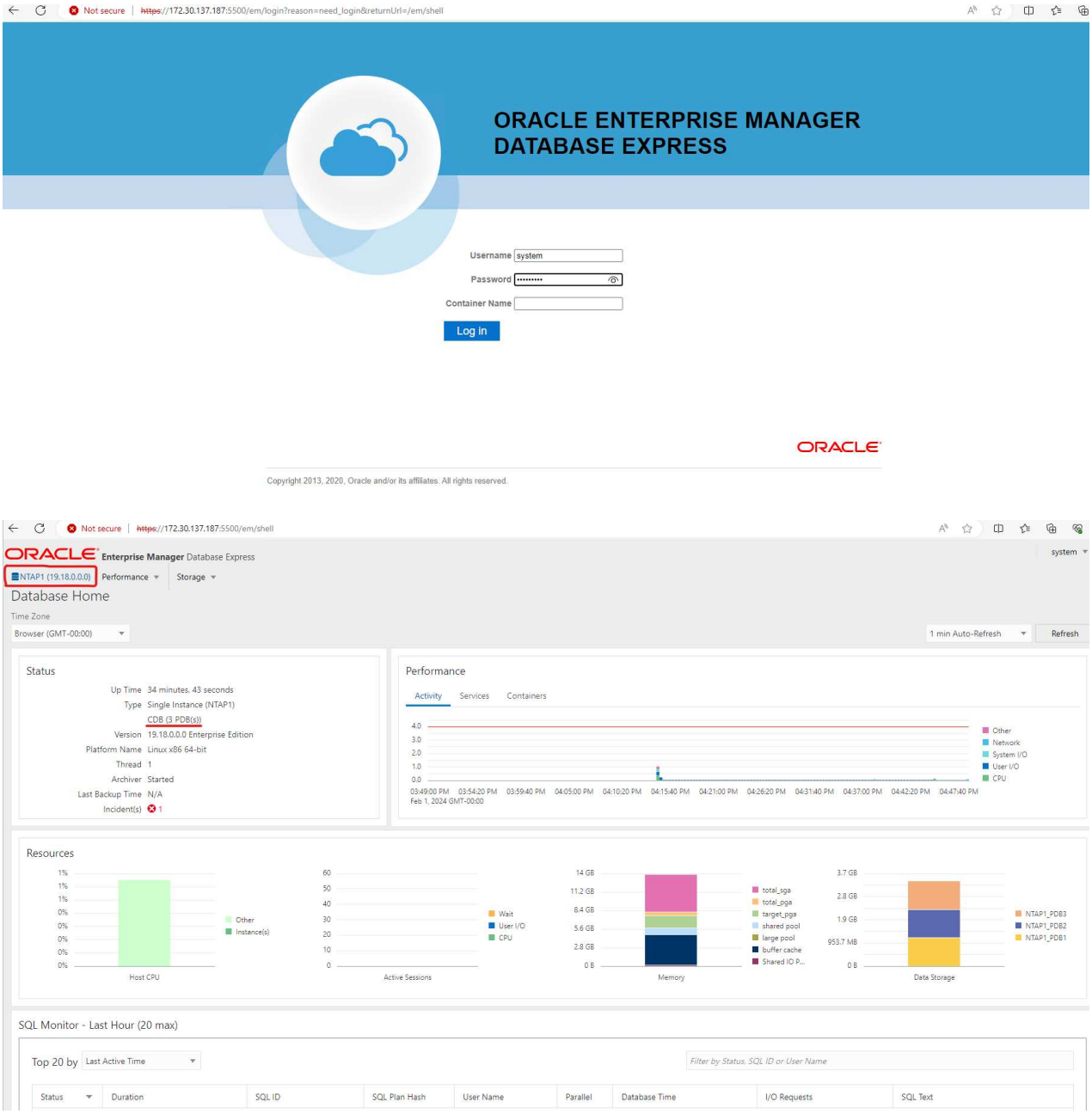
```
[oracle@ora-01 ~]$ cat /etc/oratab
#
# This file is used by ORACLE utilities.  It is created by root.sh
# and updated by either Database Configuration Assistant while
creating
# a database or ASM Configuration Assistant while creating ASM
```

```
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.
#
#
NTAP1:/u01/app/oracle/product/19.0.0/NTAP1:Y


[oracle@ora-01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Feb 1 16:37:51 2024
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
--------- -------------------- ------------
NTAP1     READ WRITE           ARCHIVELOG

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 NTAP1_PDB1                     READ WRITE NO
         4 NTAP1_PDB2                     READ WRITE NO
```

```
         5 NTAP1_PDB3                     READ WRITE NO
SQL> select name from v$datafile;

NAME
------------------------------------------------------------------------
------------
/u02/oradata/NTAP1/system01.dbf
/u02/oradata/NTAP1/sysaux01.dbf
/u02/oradata/NTAP1/undotbs01.dbf
/u02/oradata/NTAP1/pdbseed/system01.dbf
/u02/oradata/NTAP1/pdbseed/sysaux01.dbf
/u02/oradata/NTAP1/users01.dbf
/u02/oradata/NTAP1/pdbseed/undotbs01.dbf
/u02/oradata/NTAP1/NTAP1_pdb1/system01.dbf
/u02/oradata/NTAP1/NTAP1_pdb1/sysaux01.dbf
/u02/oradata/NTAP1/NTAP1_pdb1/undotbs01.dbf
/u02/oradata/NTAP1/NTAP1_pdb1/users01.dbf

NAME
------------------------------------------------------------------------
------------
/u02/oradata/NTAP1/NTAP1_pdb2/system01.dbf
/u02/oradata/NTAP1/NTAP1_pdb2/sysaux01.dbf
/u02/oradata/NTAP1/NTAP1_pdb2/undotbs01.dbf
/u02/oradata/NTAP1/NTAP1_pdb2/users01.dbf
/u02/oradata/NTAP1/NTAP1_pdb3/system01.dbf
/u02/oradata/NTAP1/NTAP1_pdb3/sysaux01.dbf
/u02/oradata/NTAP1/NTAP1_pdb3/undotbs01.dbf
/u02/oradata/NTAP1/NTAP1_pdb3/users01.dbf

19 rows selected.

SQL> select name from v$controlfile;

NAME
------------------------------------------------------------------------
------------
/u02/oradata/NTAP1/control01.ctl
/u03/orareco/NTAP1/control02.ctl

SQL> select member from v$logfile;

MEMBER
------------------------------------------------------------------------
------------
/u03/orareco/NTAP1/onlinelog/redo03.log
```

```
/u03/orareco/NTAP1/onlinelog/redo02.log
/u03/orareco/NTAP1/onlinelog/redo01.log

SQL> select svrname, dirname, nfsversion from v$dnfs_servers;

SVRNAME
------------------------------------------------------------------------
------------
DIRNAME
------------------------------------------------------------------------
------------
NFSVERSION
----------------
172.30.136.68
/ora-01-u02
NFSv3.0

172.30.136.68
/ora-01-u03
NFSv3.0

SVRNAME
------------------------------------------------------------------------
------------
DIRNAME
------------------------------------------------------------------------
------------
NFSVERSION
----------------

172.30.136.68
/ora-01-u01
NFSv3.0
```

4. 登录到Oracle Enterprise Manager Express以验证数据库。



将Oracle数据库迁移到Azure

Oracle数据库从内部迁移到云是一项繁重的工作。使用正确的策略和自动化可以使流程顺畅进行、并最大限度地减少服务中断和停机时间。请遵循此详细说明 "将数据库从内部迁移到Azure云" 指导您的数据库迁移之旅。

使用SnapCenter进行Oracle备份、还原和克隆

NetApp建议使用SnapCenter UI工具来管理部署在Azure云中的Oracle数据库。请参阅TR-4988： "使用SnapCenter在ANF上执行Oracle数据库备份、恢复和克隆" 了解详细信息。

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- 使用SnapCenter在ANF上执行Oracle数据库备份、恢复和克隆

  "使用SnapCenter在ANF上执行Oracle数据库备份、恢复和克隆"

- Azure NetApp Files

  "https://azure.microsoft.com/en-us/products/netapp"

- 部署Oracle Direct NFS

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/deploying-dnfs.html#GUID-D06079DB-8C71-4F68-A1E3-A75D7D96DCE2"

- 使用响应文件安装和配置Oracle数据库

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-and-configuring-oracle-database-using-response-files.html#GUID-D53355E9-E901-4224-9A2A-B882070EDDF7"

TR-4986：《在采用iSCSI的Amazon FSx ONTAP上简化的自动化Oracle部署》

NetApp公司Allen Cao、Niyaz Mohamed

目的

Amazon FSx for NetApp ONTAP是一项存储服务、支持您在AWS云中启动和运行完全托管的NetApp ONTAP文件系统。它提供了NetApp文件系统的熟悉特性、性能、功能和API、并具有完全托管的AWS服务的灵活性、可扩展性和精简性。它支持您在AWS云中运行要求最严苛的数据库工作负载、例如Oracle、让您高枕无忧。

本文档演示了如何使用Ans可 自动化在Amazon FSx ONTAP文件系统中简化Oracle数据库的部署。Oracle数据库部署在独立重新启动配置中、使用iSCSI协议进行数据访问、使用Oracle ASM进行数据库存储磁盘管理。它还提供了有关使用NetApp SnapCenter UI工具在AWS云中执行高效存储数据库操作的Oracle数据库备份、还原和克隆的信息。

此解决方案 可解决以下使用情形：

- 在Amazon FSx ONTAP文件系统上自动部署Oracle数据库
- 使用NetApp SnapCenter工具在Amazon FSx ONTAP文件系统上执行Oracle数据库备份和还原
- 使用NetApp SnapCenter工具在Amazon FSx ONTAP文件系统上为开发/测试或其他使用情形创建Oracle数据库克隆

audience

此解决方案 适用于以下人员：

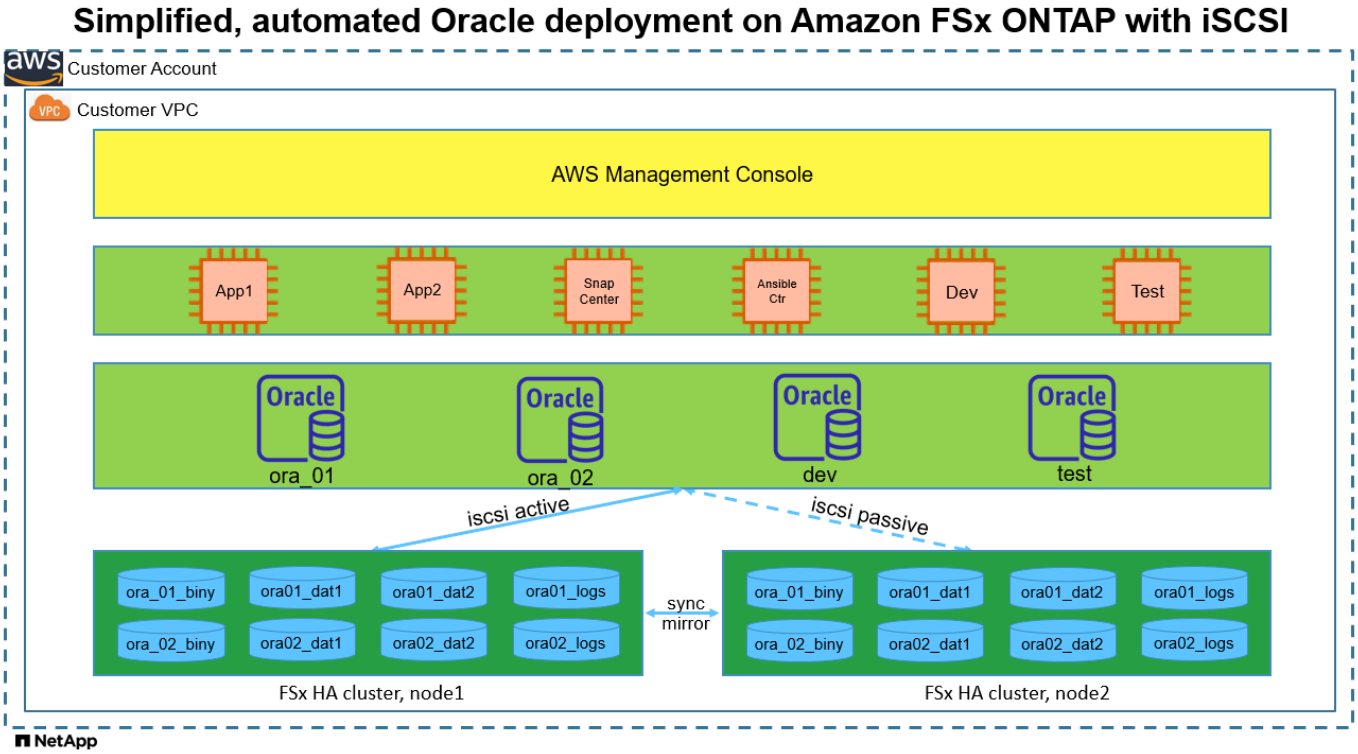
- 希望在Amazon FSx ONTAP文件系统上部署Oracle的数据库开发人员。

- 希望在Amazon FSx ONTAP文件系统上测试Oracle工作负载的数据库解决方案架构师。
- 希望在Amazon FSx ONTAP文件系统上部署和管理Oracle数据库的存储管理员。
- 希望在Amazon FSx ONTAP文件系统上建立Oracle数据库的应用程序所有者。

解决方案 测试和验证环境

此解决方案的测试和验证是在实验室环境中执行的、可能与最终部署环境不匹配。请参见一节 [Key Factors for Deployment Consideration] 有关详细信息 …

架构



硬件和软件组件

* 硬件 *		
Amazon FSx ONTAP存储	AWS提供的当前版本	一个FSX HA集群位于同一VPC和可用性区域中
用于计算的EC2实例	t2.xlarge/4vCPU/16G	两个EC2 T2大型EC2实例、用于并发部署
软件		
RedHat Linux	rell-8.6、4.18.0-372.9.1.el8.x86_64内核	已部署RedHat订阅以进行测试
Windows服务器	2022标准版、10.0.20348内部版本20348	托管SnapCenter服务器

Oracle网格基础架构	版本19.18	已应用RU修补程序p34762026_190000_Linux-x86-64.zip
Oracle 数据库	版本19.18	已应用RU修补程序p34765931_190000_Linux-x86-64.zip
Oracle OPatch	版本12.2.0.1.36	最新修补程序p6880880_190000_Linux-x86-64.zip
SnapCenter 服务器	版本4.9P1	工作组部署
打开JDK	版本java-1.8.0-OpenJDK.x86_64	DB VM上的SnapCenter插件要求

实验室环境中的**Oracle**数据库配置

* 服务器 *	* 数据库 *	**DB**存储
ORA_01	NTAP1 (NTAP1_PDB1、NTAP1_PDB2、NTAP_PDB3)	Amazon FSx ONTAP文件系统上的iSCSI LUN
ORA_02	NTAP2 (NTAP2_PDB1、NTAP2_PDB2、NTAP2_PDB3)	Amazon FSx ONTAP文件系统上的iSCSI LUN

部署注意事项的关键因素

- *Oracle数据库存储布局。*在此自动化Oracle部署中、我们默认配置四个数据库卷来托管Oracle二进制文件、数据和日志。卷中的一个LUN分配给Oracle二进制文件。然后、我们将使用数据和日志LUN创建两个ASM磁盘组。在+data ASM磁盘组中、我们配置两个数据卷、其中一个卷包含两个LUN。在+logs ASM磁盘组中、我们会在一个日志卷中创建两个LUN。一般来说、在一个ONTAP卷中部署多个LUN可提高性能。
- *部署多个数据库服务器。*自动化解决方案可以在一次运行的AnsablePlaybook中将一个Oracle容器数据库部署到多个数据库服务器。无论数据库服务器的数量如何、该操作手册的执行都保持不变。您可以将多个容器数据库部署到具有不同数据库实例ID (Oracle SID)的单个EC2实例。但是、请确保主机上有足够的内存来支持已部署的数据库。
- * iSCSI配置。* EC2实例数据库服务器使用iSCSI协议连接到FSX存储。EC2实例通常使用一个网络接口或ENI进行部署。单个NIC接口可同时传输iSCSI和应用程序流量。请务必通过仔细分析Oracle AWR报告来衡量Oracle数据库峰值I/O吞吐量需求、以便选择既满足应用程序流量吞吐量要求又符合iSCSI流量吞吐量要求的正确EC2计算实例。此外、AWS EC2通常会将每个TCP流量限制为5 Gbps。每个iSCSI路径可提供5 Gbps (625 Mbps)的带宽、要满足更高的吞吐量要求、可能需要多个iSCSI连接。
- *要用于创建的每个Oracle ASM磁盘组的Oracle ASM冗余级别。*由于Amazon FSx ONTAP已启用HA、可在集群磁盘级别保护数据、因此应使用 `External Redundancy`，表示选项不允许Oracle ASM镜像磁盘组的内容。
- 数据库备份。 NetApp提供了一个SnapCenter软件套件、可通过用户友好的用户界面进行数据库备份、还原和克隆。NetApp建议实施此类管理工具、以实现快速(不到一分钟)的快照备份、快速(几分钟)的数据库还原和数据库克隆。

解决方案 部署

以下各节提供了在使用直接挂载的数据库LUN的Amazon FSx ONTAP文件系统上自动部署和保护Oracle 19c的

分步过程、这些LUN通过iSCSI传输到EC2实例VM、采用Oracle ASM作为数据库卷管理器的单节点重新启动配置。

部署的前提条件

部署需要满足以下前提条件。

1. 已设置AWS帐户、并已在您的AWS帐户中创建必要的VPC和网段。
2. 从AWS EC2控制台中、将EC2 Linux实例部署为Oracle数据库服务器。为EC2用户启用SSH专用/公共密钥身份验证。有关环境设置的详细信息、请参见上一节中的架构图。另请查看 "Linux实例用户指南" 有关详细信息 …
3. 从AWS FSx控制台中、配置满足要求的Amazon FSx ONTAP文件系统。查看文档 "为ONTAP 文件系统创建FSX" 了解分步说明。
4. 可以使用以下Terraform自动化工具包执行步骤2和步骤3、该工具包会创建一个名为的EC2实例 `ora_01` 和名为的FSX文件系统 `fsx_01`。执行前、请仔细阅读该说明并根据您的环境更改变量。您可以根据自己的部署要求轻松修改此模板。

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

5. 将EC2 Linux实例配置为安装了最新版本的Ansv近 和Git的Ansv可 控制器节点。有关详细信息、请参见以下链接： "NetApp解决方案 自动化入门" 在第-节中
`Setup the Ansible Control Node for CLI deployments on RHEL / CentOS` 或
`Setup the Ansible Control Node for CLI deployments on Ubuntu / Debian`。
6. 配置Windows服务器以使用最新版本运行NetApp SnapCenter UI工具。有关详细信息、请参见以下链接： "安装 SnapCenter 服务器"
7. 克隆一份适用于iSCSI的NetApp Oracle部署自动化工具包副本。

```
git clone https://bitbucket.ngage.netapp.com/scm/ns-
bb/na_oracle_deploy_iscsi.git
```

8. EC2实例/tmp/archive目录中的Oracle 19c安装文件后的阶段。

```
installer_archives:
  - "LINUX.X64_193000_grid_home.zip"
  - "p34762026_190000_Linux-x86-64.zip"
  - "LINUX.X64_193000_db_home.zip"
  - "p34765931_190000_Linux-x86-64.zip"
  - "p6880880_190000_Linux-x86-64.zip"
```

请确保已在Oracle VM根卷中至少分配50G、以便有足够的空间来暂存Oracle安装文件。

9. 观看以下视频：

在采用iSCSI的Amazon FSx ONTAP上简化和自动化Oracle部署

自动化参数文件

Ans可 通过预定义的参数执行数据库安装和配置任务。对于此Oracle自动化解决方案、有三个用户定义的参数文件需要用户输入才能执行操作手册。

- 主机—定义运行自动化操作手册的目标。
- vars/vars.yml—用于定义应用于所有目标的变量的全局变量文件。
- host_vars/host_name.yml—用于定义仅适用于指定目标的变量的本地变量文件。在我们的使用情形中、这些是Oracle数据库服务器。

除了这些用户定义的变量文件之外、还有多个默认变量文件包含默认参数、除非必要、否则不需要更改这些参数。以下各节介绍如何配置用户定义的变量文件。

参数文件配置

1. 可逆目标 `hosts` 文件配置：

```
# Enter Amazon FSx ONTAP management IP address
[ontap]
172.16.9.32

# Enter name for ec2 instance (not default IP address naming) to be
deployed one by one, follow by ec2 instance IP address, and ssh
private key of ec2-user for the instance.
[oracle]
ora_01 ansible_host=10.61.180.21 ansible_ssh_private_key_file
=ora_01.pem
ora_02 ansible_host=10.61.180.23 ansible_ssh_private_key_file
=ora_02.pem
```

2. 全局 `vars/vars.yml` 文件配置

```
#########################################################################
########################################
######                  Oracle 19c deployment global user
configurable variables                          ######
######                  Consolidate all variables from ONTAP, linux
and oracle                              ######
#########################################################################
########################################

#########################################################################
########################################
######                  ONTAP env specific config variables
######
#########################################################################
########################################

# Enter the supported ONTAP platform: on-prem, aws-fsx.
ontap_platform: aws-fsx

# Enter ONTAP cluster management user credentials
username: "fsxadmin"
password: "xxxxxxxx"

#########################################################################
########################################
###                    Linux env specific config variables
###
```

```
#################################################################################
########################################

# Enter RHEL subscription to enable repo
redhat_sub_username: xxxxxxxx
redhat_sub_password: "xxxxxxxx"


#################################################################################
########################################
###                    Oracle DB env specific config variables
###
#################################################################################
########################################

# Enter Database domain name
db_domain: solutions.netapp.com

# Enter initial password for all required Oracle passwords. Change
them after installation.
initial_pwd_all: xxxxxxxx
```

3. 本地数据库服务器 `host_vars/host_name.yml` 配置，如ora_01.yml、ora_02.yml ...

```
# User configurable Oracle host specific parameters

# Enter container database SID. By default, a container DB is
created with 3 PDBs within the CDB
oracle_sid: NTAP1

# Enter database shared memory size or SGA. CDB is created with SGA
at 75% of memory_limit, MB. The grand total of SGA should not exceed
75% available RAM on node.
memory_limit: 8192
```

执行操作手册

自动化工具包中共有六本操作手册。每个任务执行不同的任务块、并用于不同的用途。

```
0-all_playbook.yml - execute playbooks from 1-4 in one playbook run.
1-ansible_requirements.yml - set up Ansible controller with required
libs and collections.
2-linux_config.yml - execute Linux kernel configuration on Oracle DB
servers.
3-ontap_config.yml - configure ONTAP svm/volumes/luns for Oracle
database and grant DB server access to luns.
4-oracle_config.yml - install and configure Oracle on DB servers for
grid infrastructure and create a container database.
5-destroy.yml - optional to undo the environment to dismantle all.
```

使用以下命令可通过三个选项运行这些操作手册。

1. 一次运行即可执行所有部署操作手册。

```
ansible-playbook -i hosts 0-all_playbook.yml -u ec2-user -e
@vars/vars.yml
```

2. 使用1-4的数字顺序执行一次一个操作手册。

```
ansible-playbook -i hosts 1-ansible_requirements.yml -u ec2-user -e
@vars/vars.yml
```

```
ansible-playbook -i hosts 2-linux_config.yml -u ec2-user -e
@vars/vars.yml
```

```
ansible-playbook -i hosts 3-ontap_config.yml -u ec2-user -e
@vars/vars.yml
```

```
ansible-playbook -i hosts 4-oracle_config.yml -u ec2-user -e
@vars/vars.yml
```

3. 使用标记执行0-all_playbook.yml。

```
ansible-playbook -i hosts 0-all_playbook.yml -u ec2-user -e
@vars/vars.yml -t ansible_requirements
```

```
ansible-playbook -i hosts 0-all_playbook.yml -u ec2-user -e
@vars/vars.yml -t linux_config
```

```
ansible-playbook -i hosts 0-all_playbook.yml -u ec2-user -e
@vars/vars.yml -t ontap_config
```

```
ansible-playbook -i hosts 0-all_playbook.yml -u ec2-user -e
@vars/vars.yml -t oracle_config
```

4. 撤消环境

```
ansible-playbook -i hosts 5-destroy.yml -u ec2-user -e
@vars/vars.yml
```

执行后验证

运行该操作手册后、以Oracle用户身份登录到Oracle数据库服务器、以验证是否已成功创建Oracle网格基础架构和数据库。以下是在主机ora_01上验证Oracle数据库的示例。

1. 验证EC2实例上的Oracle容器数据库

```
[admin@ansiblectl na_oracle_deploy_iscsi]$ ssh -i ora_01.pem ec2-
user@172.30.15.40
Last login: Fri Dec  8 17:14:21 2023 from 10.61.180.18
[ec2-user@ip-172-30-15-40 ~]$ uname -a
Linux ip-172-30-15-40.ec2.internal 4.18.0-372.9.1.el8.x86_64 #1 SMP
Fri Apr 15 22:12:19 EDT 2022 x86_64 x86_64 x86_64 GNU/Linux

[ec2-user@ip-172-30-15-40 ~]$ sudo su
[root@ip-172-30-15-40 ec2-user]# su - oracle
Last login: Fri Dec  8 16:25:52 UTC 2023 on pts/0
[oracle@ip-172-30-15-40 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Fri Dec 8 18:18:20 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
--------- -------------------- ------------
NTAP1     READ WRITE           ARCHIVELOG

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 NTAP1_PDB1                     READ WRITE NO
         4 NTAP1_PDB2                     READ WRITE NO
         5 NTAP1_PDB3                     READ WRITE NO
SQL> select name from v$datafile;

NAME
------------------------------------------------------------------------
```

```
------------
+DATA/NTAP1/DATAFILE/system.257.1155055419
+DATA/NTAP1/DATAFILE/sysaux.258.1155055463
+DATA/NTAP1/DATAFILE/undotbs1.259.1155055489
+DATA/NTAP1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/system.266.115
5056241
+DATA/NTAP1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/sysaux.267.115
5056241
+DATA/NTAP1/DATAFILE/users.260.1155055489
+DATA/NTAP1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/undotbs1.268.1
155056241
+DATA/NTAP1/0C03AAFA7C6FD2E5E063280F1EACFBE0/DATAFILE/system.272.115
5057059
+DATA/NTAP1/0C03AAFA7C6FD2E5E063280F1EACFBE0/DATAFILE/sysaux.273.115
5057059
+DATA/NTAP1/0C03AAFA7C6FD2E5E063280F1EACFBE0/DATAFILE/undotbs1.271.1
155057059
+DATA/NTAP1/0C03AAFA7C6FD2E5E063280F1EACFBE0/DATAFILE/users.275.1155
057075

NAME
--------------------------------------------------------------------
------------
+DATA/NTAP1/0C03AC0089ACD352E063280F1EAC12BD/DATAFILE/system.277.115
5057075
+DATA/NTAP1/0C03AC0089ACD352E063280F1EAC12BD/DATAFILE/sysaux.278.115
5057075
+DATA/NTAP1/0C03AC0089ACD352E063280F1EAC12BD/DATAFILE/undotbs1.276.1
155057075
+DATA/NTAP1/0C03AC0089ACD352E063280F1EAC12BD/DATAFILE/users.280.1155
057091
+DATA/NTAP1/0C03ACEABA54D386E063280F1EACE573/DATAFILE/system.282.115
5057091
+DATA/NTAP1/0C03ACEABA54D386E063280F1EACE573/DATAFILE/sysaux.283.115
5057091
+DATA/NTAP1/0C03ACEABA54D386E063280F1EACE573/DATAFILE/undotbs1.281.1
155057091
+DATA/NTAP1/0C03ACEABA54D386E063280F1EACE573/DATAFILE/users.285.1155
057105

19 rows selected.

SQL> select name from v$controlfile;

NAME
--------------------------------------------------------------------
```

```
------------
+DATA/NTAP1/CONTROLFILE/current.261.1155055529
+LOGS/NTAP1/CONTROLFILE/current.256.1155055529

SQL> select member from v$logfile;

MEMBER
--------------------------------------------------------------------------------
------------
+DATA/NTAP1/ONLINELOG/group_3.264.1155055531
+LOGS/NTAP1/ONLINELOG/group_3.259.1155055539
+DATA/NTAP1/ONLINELOG/group_2.263.1155055531
+LOGS/NTAP1/ONLINELOG/group_2.257.1155055539
+DATA/NTAP1/ONLINELOG/group_1.262.1155055531
+LOGS/NTAP1/ONLINELOG/group_1.258.1155055539

6 rows selected.

SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release
19.0.0.0.0 - Production
Version 19.18.0.0.0
```

2. 验证Oracle侦听器。

```
[oracle@ip-172-30-15-40 ~]$ lsnrctl status listener

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 08-DEC-2023
18:20:24

Copyright (c) 1991, 2022, Oracle.  All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=ip-172-30-
15-40.ec2.internal)(PORT=1521)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 19.0.0.0.0 -
Production
Start Date                08-DEC-2023 16:26:09
Uptime                    0 days 1 hr. 54 min. 14 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File
```

```
/u01/app/oracle/product/19.0.0/grid/network/admin/listener.ora
Listener Log File         /u01/app/oracle/diag/tnslsnr/ip-172-30-15-
40/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=ip-172-30-15-
40.ec2.internal)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=ip-172-30-15-
40.ec2.internal)(PORT=5500))(Security=(my_wallet_directory=/u01/app/
oracle/product/19.0.0/NTAP1/admin/NTAP1/xdb_wallet))(Presentation=HT
TP)(Session=RAW))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM", status READY, has 1 handler(s) for this
service...
Service "+ASM_DATA" has 1 instance(s).
  Instance "+ASM", status READY, has 1 handler(s) for this
service...
Service "+ASM_LOGS" has 1 instance(s).
  Instance "+ASM", status READY, has 1 handler(s) for this
service...
Service "0c03aafa7c6fd2e5e063280f1eacfbe0.solutions.netapp.com" has
1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "0c03ac0089acd352e063280f1eac12bd.solutions.netapp.com" has
1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "0c03aceaba54d386e063280f1eace573.solutions.netapp.com" has
1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "NTAP1.solutions.netapp.com" has 1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "NTAP1XDB.solutions.netapp.com" has 1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "ntap1_pdb1.solutions.netapp.com" has 1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "ntap1_pdb2.solutions.netapp.com" has 1 instance(s).
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
Service "ntap1_pdb3.solutions.netapp.com" has 1 instance(s).
```

```
  Instance "NTAP1", status READY, has 1 handler(s) for this
service...
The command completed successfully
```

3. 验证创建的网格基础架构和资源。

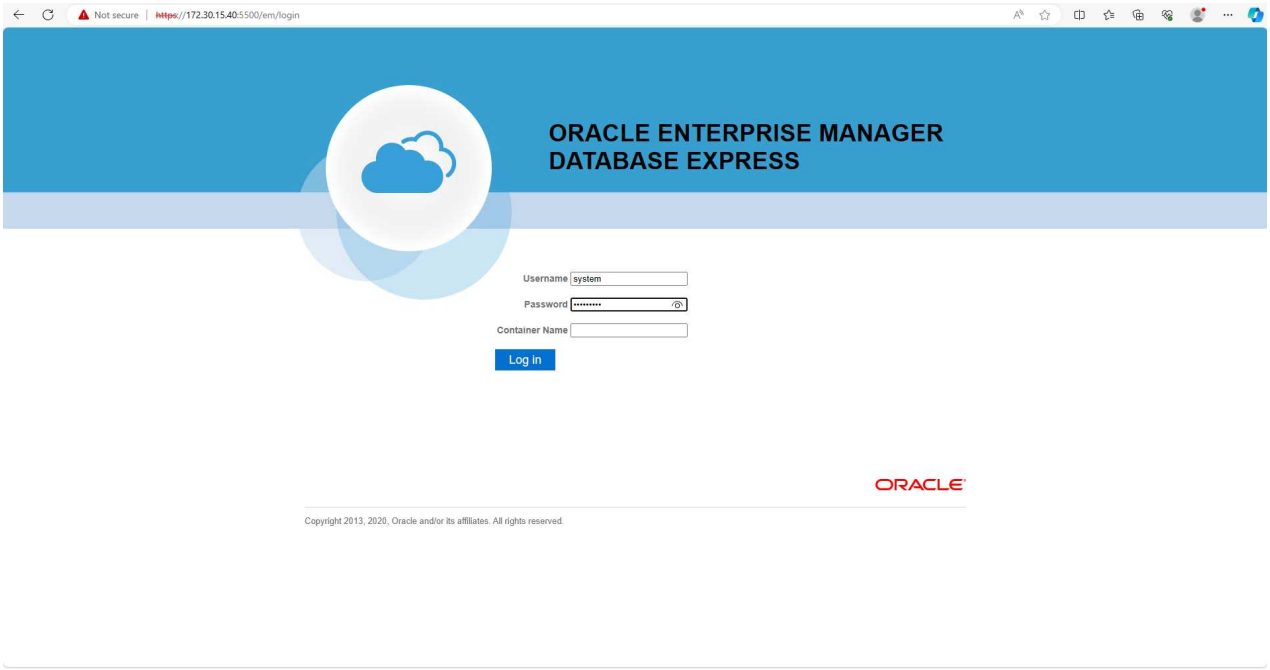
```
[oracle@ip-172-30-15-40 ~]$ asm
[oracle@ip-172-30-15-40 ~]$ crsctl check has
CRS-4638: Oracle High Availability Services is online
[oracle@ip-172-30-15-40 ~]$ crsctl stat res -t
--------------------------------------------------------------------------
------------
Name           Target  State        Server                   State
details
--------------------------------------------------------------------------
------------
Local Resources
--------------------------------------------------------------------------
------------
ora.DATA.dg
               ONLINE  ONLINE       ip-172-30-15-40          STABLE
ora.LISTENER.lsnr
               ONLINE  ONLINE       ip-172-30-15-40          STABLE
ora.LOGS.dg
               ONLINE  ONLINE       ip-172-30-15-40          STABLE
ora.asm
               ONLINE  ONLINE       ip-172-30-15-40
Started,STABLE
ora.ons
               OFFLINE OFFLINE      ip-172-30-15-40          STABLE
--------------------------------------------------------------------------
------------
Cluster Resources
--------------------------------------------------------------------------
------------
ora.cssd
      1        ONLINE  ONLINE       ip-172-30-15-40          STABLE
ora.diskmon
      1        OFFLINE OFFLINE                               STABLE
ora.driver.afd
      1        ONLINE  ONLINE       ip-172-30-15-40          STABLE
ora.evmd
      1        ONLINE  ONLINE       ip-172-30-15-40          STABLE
ora.ntap1.db
      1        ONLINE  ONLINE       ip-172-30-15-40
```

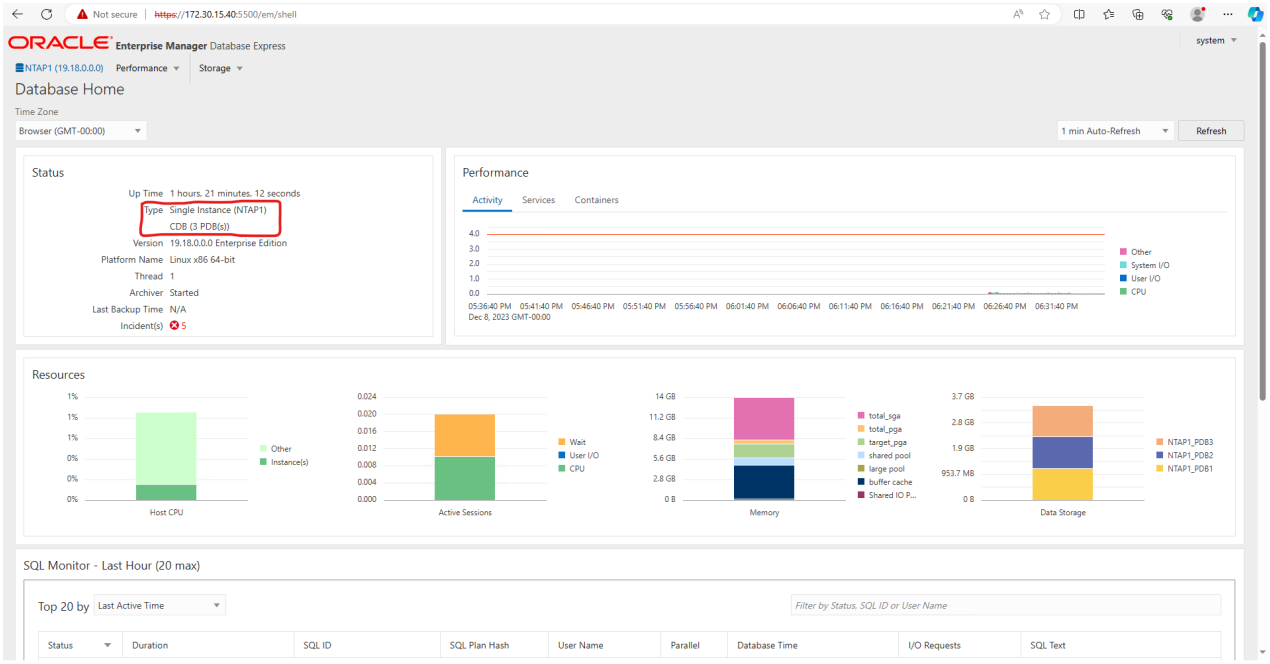
```
Open,HOME=/u01/app/o

racle/product/19.0.0

/NTAP1,STABLE
------------------------------------------------------------------------
------------
```

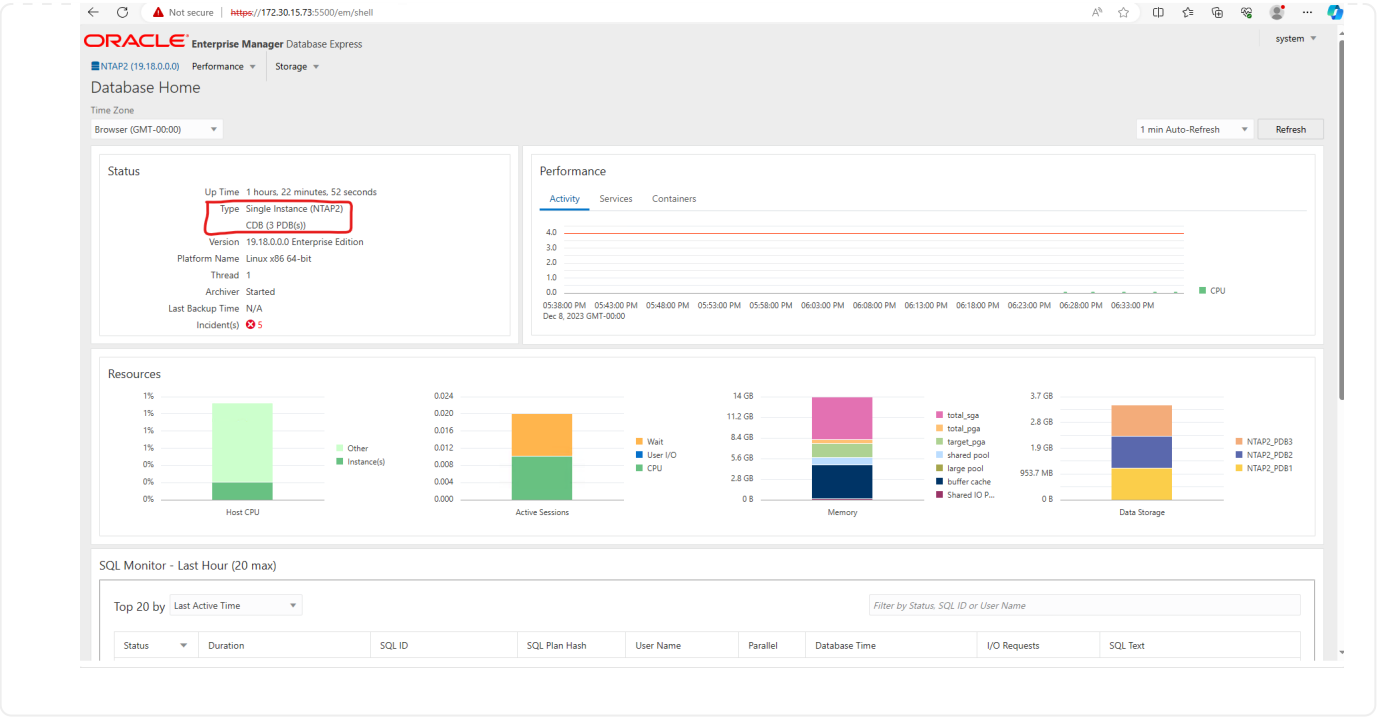
4. 验证Oracle ASM。

```
[oracle@ip-172-30-15-40 ~]$ asmcmd
ASMCMD> lsdg
State    Type    Rebal  Sector  Logical_Sector  Block       AU
Total_MB  Free_MB  Req_mir_free_MB  Usable_file_MB  Offline_disks
Voting_files  Name
MOUNTED  EXTERN  N         512             512   4096  4194304
163840   155376                0          155376              0
N  DATA/
MOUNTED  EXTERN  N         512             512   4096  4194304
81920    80972                0           80972              0
N  LOGS/
ASMCMD> lsdsk
Path
AFD:ORA_01_DAT1_01
AFD:ORA_01_DAT1_03
AFD:ORA_01_DAT2_02
AFD:ORA_01_DAT2_04
AFD:ORA_01_LOGS_01
AFD:ORA_01_LOGS_02
ASMCMD> afd_state
ASMCMD-9526: The AFD state is 'LOADED' and filtering is 'ENABLED' on
host 'ip-172-30-15-40.ec2.internal'
ASMCMD> exit
```

5. 登录到Oracle Enterprise Manager Express以验证数据库。


Not secure | https://172.30.15.40:5500/em/login
ORACLE ENTERPRISE MANAGER
DATABASE EXPRESS
Username system
Password
Container Name
Log in
ORACLE
Copyright 2013, 2020, Oracle and/or its affiliates. All rights reserved.


Not secure | https://172.30.15.40:5500/em/shell
ORACLE Enterprise Manager Database Express
system
NTAP1 (19.18.0.0.0)
Performance
Storage
Database Home
Time Zone
Browser (GMT-00:00)
1 min Auto-Refresh
Refresh
Status
Up Time 1 hours, 21 minutes, 12 seconds
Type Single Instance (NTAP1)
CDB (3 PDB(s))
Version 19.18.0.0.0 Enterprise Edition
Platform Name Linux x86 64-bit
Thread 1
Archiver Started
Last Backup Time N/A
Incident(s) 5
Performance
Activity
Services
Containers
Other
System I/O
User I/O
CPU
Resources
Host CPU
Active Sessions
Memory
Data Storage
NTAP1_PDB3
NTAP1_PDB2
NTAP1_PDB1
SQL Monitor - Last Hour (20 max)
Top 20 by Last Active Time
Filter by Status, SQL ID or User Name
Status
Duration
SQL ID
SQL Plan Hash
User Name
Parallel
Database Time
I/O Requests
SQL Text



使用**SnapCenter**进行**Oracle**备份、还原和克隆

请参阅TR-4979 "借助子系统装载的FSx ONTAP、在基于AWS的VMware Cloud中简化自我管理Oracle" 部分。 `Oracle backup, restore, and clone with SnapCenter` 有关设置SnapCenter以及执行数据库备份、还原和克隆工作流的详细信息。

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- 适用于 NetApp ONTAP 的 Amazon FSX

  "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

- 在安装新数据库的情况下为独立服务器安装Oracle网格基础架构

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-oracle-grid-infrastructure-for-a-standalone-server-with-a-new-database-installation.html#GUID-0B1CEE8C-C893-46AA-8A6A-7B5FAAEC72B3"

- 使用响应文件安装和配置Oracle数据库

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-and-configuring-oracle-

database-using-response-files.html#GUID-D53355E9-E901-4224-9A2A-B882070EDDF7"

- 将Red Hat Enterprise Linux 8.2与ONTAP结合使用

  "https://docs.netapp.com/us-en/ontap-sanhost/hu_rhel_82.html#all-san-array-configurations"

TR-4983：《在采用iSCSI的NetApp ASA上简化的自动化Oracle部署》

NetApp公司Allen Cao、Niyaz Mohamed

目的

NetApp ASA系统为您的SAN基础架构提供现代化的解决方案。它们可以大规模简化并帮助您加快数据库等业务关键型应用程序的运行速度、确保数据始终可用(99.9999%的正常运行时间)、并降低TCO和碳排放量。NetApp ASA系统包括专为要求性能最苛刻的应用程序而设计的A系列型号和针对经济高效的大容量部署而优化的C系列型号。ASA A系列和C系列系统相结合、可提供卓越的性能、以改善客户体验并缩短取得成果的时间、保持业务关键型数据可用、受到保护和安全、并通过行业最有效的担保为任何工作负载提供更有效的容量。

本文档演示了如何在使用ASA系统构建的SAN环境中使用Andsilp自动化简化Oracle数据库的部署。Oracle数据库部署在独立重新启动配置中、使用iSCSI协议在ASA存储阵列上进行数据访问、并使用Oracle ASM进行数据库磁盘管理。此外、还提供了有关使用NetApp SnapCenter UI工具在NetApp ASA系统中执行高效存储数据库操作的Oracle数据库备份、还原和克隆的信息。

此解决方案 可解决以下使用情形：

- 在NetApp ASA系统中自动部署Oracle数据库作为主数据库存储
- 使用NetApp SnapCenter工具在NetApp ASA系统中进行Oracle数据库备份和还原
- 使用NetApp SnapCenter工具在NetApp ASA系统中为开发/测试或其他使用情形创建Oracle数据库克隆

audience

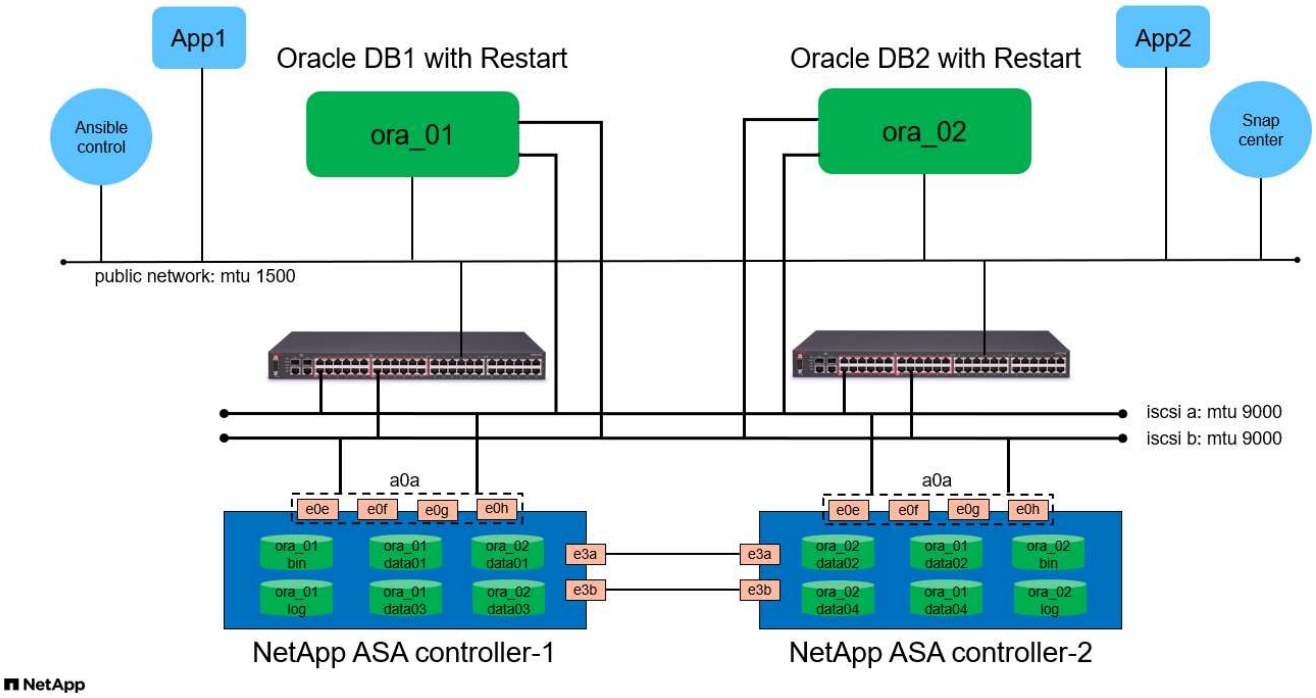
此解决方案 适用于以下人员：

- 希望在NetApp ASA系统中部署Oracle的BA。
- 希望在NetApp ASA系统中测试Oracle工作负载的数据库解决方案架构师。
- 希望在NetApp ASA系统上部署和管理Oracle数据库的存储管理员。
- 希望在NetApp ASA系统中建立Oracle数据库的应用程序所有者。

解决方案 测试和验证环境

此解决方案的测试和验证是在实验室环境中执行的、可能与最终部署环境不匹配。请参见一节 [Key Factors for Deployment Consideration] 有关详细信息 …

架构



硬件和软件组件

* 硬件 *		
NetApp ASA A400	版本9.13.1P1	2个NS224磁盘架、48个NVMe AFF驱动器、总容量为69.3 TiB
UCSB-B200-M4	Intel ® Xeon ® CPU E5-2690 v4 @ 2.60GHz	4节点VMware ESXi集群
软件		
RedHat Linux	rell-8.6、4.18.0-372.9.1.el8.x86_64内核	已部署RedHat订阅以进行测试
Windows服务器	2022标准版、10.0.20348内部版本20348	托管SnapCenter服务器
Oracle网格基础架构	版本19.18	已应用RU修补程序p34762026_190000_Linux-x86-64.zip
Oracle 数据库	版本19.18	已应用RU修补程序p34765931_190000_Linux-x86-64.zip
Oracle OPatch	版本12.2.0.1.36	最新修补程序p6880880_190000_Linux-x86-64.zip
SnapCenter 服务器	版本4.9P1	工作组部署
VMware vSphere 虚拟机管理程序	版本6.5.0.0000	VMware Tools、版本：11365 - Linux、1352 - Windows

打开JDK	版本java-1.8.0-OpenJDK.x86_64	DB VM上的SnapCenter插件要求

实验室环境中的**Oracle**数据库配置

* 服务器 *	* 数据库 *	**DB**存储
ORA_01	NTAP1 (NTAP1_PDB1 、NTAP1_PDB2、NTAP_PDB3)	ASA A400上的iSCSI LUN
ORA_02	NTAP2 (NTAP2_PDB1 、NTAP2_PDB2、NTAP2_PDB3)	ASA A400上的iSCSI LUN

部署注意事项的关键因素

- *Oracle数据库存储布局。*在此自动化Oracle部署中、我们默认配置四个数据库卷来托管Oracle二进制文件、数据和日志。然后、我们将使用数据和日志LUN创建两个ASM磁盘组。在+data ASM磁盘组中、我们会在每个ASA A400集群节点上的一个卷中配置两个数据LUN。在+logs ASM磁盘组中、我们会在一个ASA A400节点的日志卷中创建两个LUN。一般来说、在一个ONTAP卷中部署多个LUN可提高性能。
- *部署多个数据库服务器。*自动化解决方案可以在一次运行的AnsablePlaybook中将一个Oracle容器数据库部署到多个数据库服务器。无论数据库服务器的数量如何、该操作手册的执行都保持不变。在部署多数据库服务器时、该操作手册会使用一种算法来构建数据库LUN、以便以最佳方式将其放置在ASA A400的双控制器上。控制器1上服务器主机索引位置中奇数数据库服务器的二进制和日志LUN。控制器2上服务器主机索引位置中偶数数据库服务器的二进制文件和日志LUN。数据库数据LUN平均分布到两个控制器。Oracle ASM将两个控制器上的数据LUN组合成一个ASM磁盘组、以充分利用这两个控制器的处理能力。
- *iSCSI配置。*数据库VM使用iSCSI协议连接到ASA存储以进行存储访问。您应在每个控制器节点上配置双路径以实现冗余、并在数据库服务器上设置iSCSI多路径以实现多路径存储访问。在存储网络上启用巨型帧、以最大程度地提高性能和吞吐量。
- *Oracle ASM冗余级别用于创建的每个Oracle ASM磁盘组。*由于ASA A400会在RAID DP中配置存储、以便在集群磁盘级别进行数据保护、因此应使用 `External Redundancy`，表示选项不允许Oracle ASM镜像磁盘组的内容。
- 数据库备份。 NetApp提供了一个SnapCenter软件套件、可通过用户友好的用户界面进行数据库备份、还原和克隆。NetApp建议实施此类管理工具、以实现快速(不到一分钟)的快照备份、快速(几分钟)的数据库还原和数据库克隆。

解决方案 部署

以下各节提供了在NetApp ASA A400中使用直接挂载的数据库LUN通过iSCSI自动部署和保护Oracle 19c的分步过程、并将其部署到单节点中的数据库VM使用Oracle ASM作为数据库卷管理器的重新启动配置。

部署的前提条件

部署需要满足以下前提条件。

1. 假定已安装并配置NetApp ASA存储阵列。这包括两个控制器节点上的iSCSI广播域、LACP接口组a0a、两个控制器节点上的iSCSI <iscsi-a-vlan-id>端口(a0a-lacp-<iscsi-b-vlan-id>)。以下链接提供了详细的分步说明(如果需要帮助)。 "详细指南—ASA A400"
2. 将Linux VM配置为安装了最新版本的Ansv近 和Git的Ansv可 控制器节点。有关详细信息、请参见以下链接： "NetApp解决方案 自动化入门" 在第-节中 `Setup the Ansible Control Node for CLI deployments on RHEL / CentOS` 或 `Setup the Ansible Control Node for CLI deployments on Ubuntu / Debian`。
3. 克隆一份适用于iSCSI的NetApp Oracle部署自动化工具包副本。

```
git clone https://bitbucket.ngage.netapp.com/scm/ns-
bb/na_oracle_deploy_iscsi.git
```

4. 配置Windows服务器以使用最新版本运行NetApp SnapCenter UI工具。有关详细信息、请参见以下链接： "安装 SnapCenter 服务器"
5. 构建两个RHEL Oracle数据库服务器、可以是裸机VM、也可以是虚拟化VM。在不具有密码权限的sudo数据库服务器上创建一个管理员用户、并在Andsle主机和Oracle数据库服务器主机之间启用SSH专用/公共密钥身份验证。DB服务器/tmp/archive目录上的Oracle 19c安装文件后的阶段。

```
installer_archives:
  - "LINUX.X64_193000_grid_home.zip"
  - "p34762026_190000_Linux-x86-64.zip"
  - "LINUX.X64_193000_db_home.zip"
  - "p34765931_190000_Linux-x86-64.zip"
  - "p6880880_190000_Linux-x86-64.zip"
```

   请确保已在Oracle VM根卷中至少分配50G、以便有足够的空间来暂存Oracle安装文件。

6. 观看以下视频：

   借助iSCSI在NetApp ASA上简化和自动化Oracle部署

自动化参数文件

Ans可 通过预定义的参数执行数据库安装和配置任务。对于此Oracle自动化解决方案、有三个用户定义的参数文件需要用户输入才能执行操作手册。

- 主机—定义运行自动化操作手册的目标。
- vars/vars.yml—用于定义应用于所有目标的变量的全局变量文件。
- host_vars/host_name.yml—用于定义仅适用于本地目标的变量的本地变量文件。在我们的使用情形中、这些是Oracle数据库服务器。

除了这些用户定义的变量文件之外、还有多个默认变量文件包含默认参数、除非必要、否则不需要更改这些参数。以下各节说明了如何配置用户定义的变量文件。

参数文件配置

1. 可逆目标 `hosts` 文件配置：

```
# Enter NetApp ASA controller management IP address
[ontap]
172.16.9.32

# Enter Oracle servers names to be deployed one by one, follow by
each Oracle server public IP address, and ssh private key of admin
user for the server.
[oracle]
ora_01 ansible_host=10.61.180.21 ansible_ssh_private_key_file
=ora_01.pem
ora_02 ansible_host=10.61.180.23 ansible_ssh_private_key_file
=ora_02.pem
```

2. 全局 `vars/vars.yml` 文件配置

```
#########################################################################
########################################
######                  Oracle 19c deployment global user
configurable variables                          ######
######                  Consolidate all variables from ONTAP, linux
and oracle                              ######
#########################################################################
########################################

#########################################################################
########################################
######                  ONTAP env specific config variables
######
#########################################################################
########################################

# Enter the supported ONTAP platform: on-prem, aws-fsx.
ontap_platform: on-prem

# Enter ONTAP cluster management user credentials
username: "xxxxxxxx"
password: "xxxxxxxx"


###### on-prem platform specific user defined variables ######

# Enter Oracle SVM iSCSI lif addresses. Each controller configures
```

```
with dual paths iscsi_a, iscsi_b for redundancy
ora_iscsi_lif_mgmt:
  - {name: '{{ svm_name }}_mgmt', address: 172.21.253.220, netmask:
255.255.255.0, vlan_name: ora_mgmt, vlan_id: 3509}

ora_iscsi_lifs_node1:
  - {name: '{{ svm_name }}_lif_1a', address: 172.21.234.221,
netmask: 255.255.255.0, vlan_name: ora_iscsi_a, vlan_id: 3490}
  - {name: '{{ svm_name }}_lif_1b', address: 172.21.235.221,
netmask: 255.255.255.0, vlan_name: ora_iscsi_b, vlan_id: 3491}
ora_iscsi_lifs_node2:
  - {name: '{{ svm_name }}_lif_2a', address: 172.21.234.223,
netmask: 255.255.255.0, vlan_name: ora_iscsi_a, vlan_id: 3490}
  - {name: '{{ svm_name }}_lif_2b', address: 172.21.235.223,
netmask: 255.255.255.0, vlan_name: ora_iscsi_b, vlan_id: 3491}


#################################################################################
########################################
###                    Linux env specific config variables
###
#################################################################################
########################################

# Enter RHEL subscription to enable repo
redhat_sub_username: xxxxxxxx
redhat_sub_password: "xxxxxxxx"


#################################################################################
########################################
###                    Oracle DB env specific config variables
###
#################################################################################
########################################

# Enter Database domain name
db_domain: solutions.netapp.com

# Enter initial password for all required Oracle passwords. Change
them after installation.
initial_pwd_all: xxxxxxxx
```

3. 本地数据库服务器 `host_vars/host_name.yml` 配置

```
# User configurable Oracle host specific parameters

# Enter container database SID. By default, a container DB is
created with 3 PDBs within the CDB
oracle_sid: NTAP1

# Enter database shared memory size or SGA. CDB is created with SGA
at 75% of memory_limit, MB. The grand total of SGA should not exceed
75% available RAM on node.
memory_limit: 8192
```

执行操作手册

自动化工具包中共有六本操作手册。每个任务执行不同的任务块、并用于不同的用途。

```
0-all_playbook.yml - execute playbooks from 1-4 in one playbook run.
1-ansible_requirements.yml - set up Ansible controller with required
libs and collections.
2-linux_config.yml - execute Linux kernel configuration on Oracle DB
servers.
3-ontap_config.yml - configure ONTAP svm/volumes/luns for Oracle
database and grant DB server access to luns.
4-oracle_config.yml - install and configure Oracle on DB servers for
grid infrastructure and create a container database.
5-destroy.yml - optional to undo the environment to dismantle all.
```

使用以下命令可通过三个选项运行这些操作手册。

1. 一次运行即可执行所有部署操作手册。

```
ansible-playbook -i hosts 0-all_playbook.yml -u admin -e
@vars/vars.yml
```

2. 使用1-4的数字顺序执行一次一个操作手册。

```
ansible-playbook -i hosts 1-ansible_requirements.yml -u admin -e
@vars/vars.yml
```

```
ansible-playbook -i hosts 2-linux_config.yml -u admin -e
@vars/vars.yml
```

```
ansible-playbook -i hosts 3-ontap_config.yml -u admin -e
@vars/vars.yml
```

```
ansible-playbook -i hosts 4-oracle_config.yml -u admin -e
@vars/vars.yml
```

3. 使用标记执行0-all_playbook.yml。

```
ansible-playbook -i hosts 0-all_playbook.yml -u admin -e
@vars/vars.yml -t ansible_requirements
```

```
ansible-playbook -i hosts 0-all_playbook.yml -u admin -e
@vars/vars.yml -t linux_config
```

```
ansible-playbook -i hosts 0-all_playbook.yml -u admin -e
@vars/vars.yml -t ontap_config
```

```
ansible-playbook -i hosts 0-all_playbook.yml -u admin -e
@vars/vars.yml -t oracle_config
```

4. 撤消环境

```
ansible-playbook -i hosts 5-destroy.yml -u admin -e @vars/vars.yml
```

执行后验证

运行该操作手册后、以Oracle用户身份登录到Oracle数据库服务器、以验证是否已成功创建Oracle网格基础架构和数据库。以下是在主机ora_01上验证Oracle数据库的示例。

1. 验证创建的网格基础架构和资源。

```
[oracle@ora_01 ~]$ df -h
Filesystem                     Size  Used Avail Use% Mounted on
devtmpfs                       7.7G   40K  7.7G   1% /dev
tmpfs                          7.8G  1.1G  6.7G  15% /dev/shm
tmpfs                          7.8G  312M  7.5G   4% /run
tmpfs                          7.8G     0  7.8G   0% /sys/fs/cgroup
/dev/mapper/rhel-root           44G   38G  6.8G  85% /
/dev/sda1                     1014M  258M  757M  26% /boot
tmpfs                          1.6G   12K  1.6G   1% /run/user/42
tmpfs                          1.6G  4.0K  1.6G   1% /run/user/1000
/dev/mapper/ora_01_biny_01p1    40G   21G   20G  52% /u01
[oracle@ora_01 ~]$ asm
[oracle@ora_01 ~]$ crsctl stat res -t
--------------------------------------------------------------------------------
------------
Name           Target  State        Server                   State
details
--------------------------------------------------------------------------------
------------
Local Resources
--------------------------------------------------------------------------------
------------
ora.DATA.dg
               ONLINE  ONLINE       ora_01                   STABLE
ora.LISTENER.lsnr
               ONLINE  INTERMEDIATE ora_01                   Not All
Endpoints Re

gistered,STABLE
ora.LOGS.dg
               ONLINE  ONLINE       ora_01                   STABLE
ora.asm
               ONLINE  ONLINE       ora_01
Started,STABLE
ora.ons
               OFFLINE OFFLINE      ora_01                   STABLE
--------------------------------------------------------------------------------
------------
Cluster Resources
--------------------------------------------------------------------------------
```

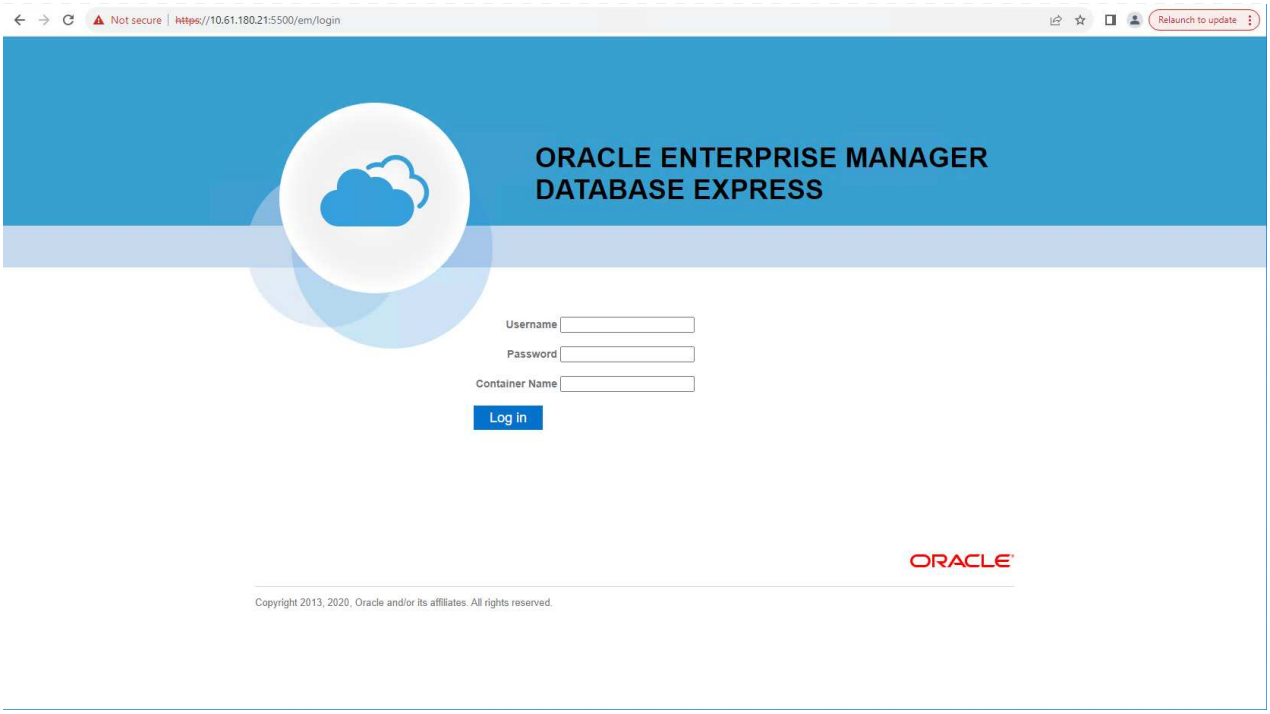
```
------------
ora.cssd
      1        ONLINE  ONLINE       ora_01                   STABLE
ora.diskmon
      1        OFFLINE OFFLINE                               STABLE
ora.driver.afd
      1        ONLINE  ONLINE       ora_01                   STABLE
ora.evmd
      1        ONLINE  ONLINE       ora_01                   STABLE
ora.ntap1.db
      1        ONLINE  ONLINE       ora_01
Open,HOME=/u01/app/o

racle/product/19.0.0

/NTAP1,STABLE
--------------------------------------------------------------------------
------------
[oracle@ora_01 ~]$
```

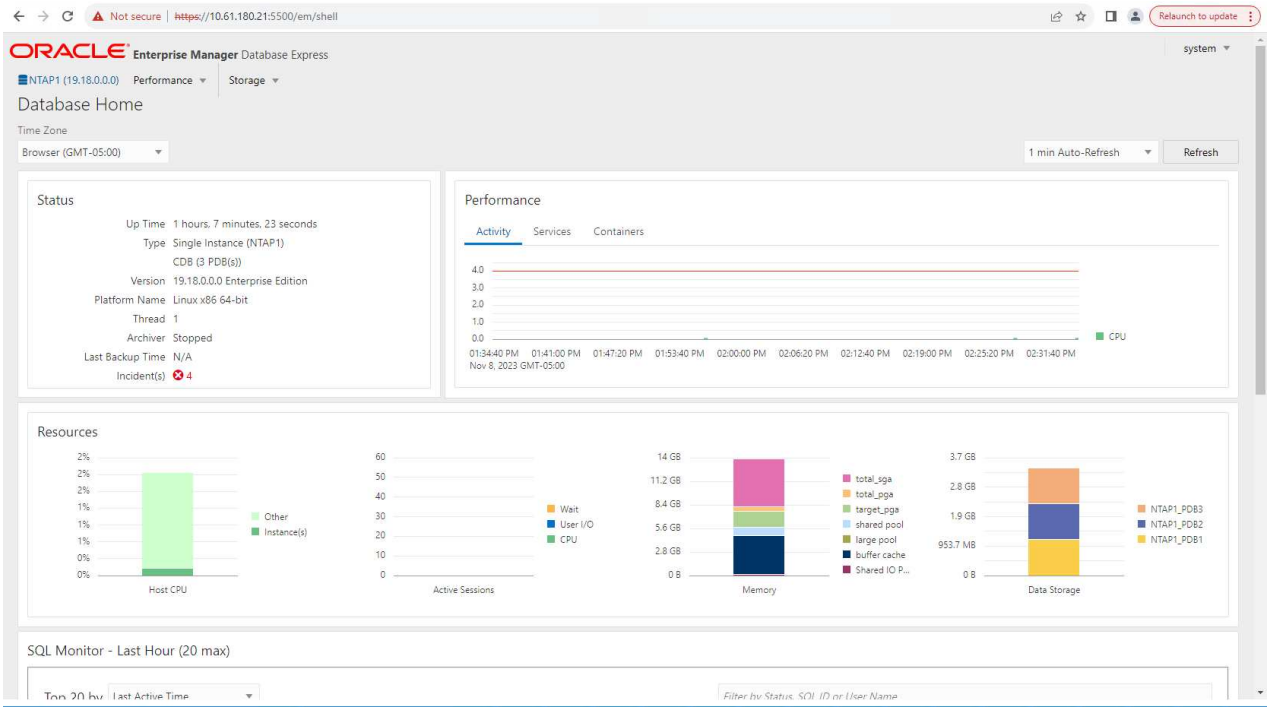
忽略 `Not All Endpoints Registered` 在状态详细信息中。这是由于在侦听器中手动和动态数据库注册发生冲突而导致的、可以放心地忽略。

2. 验证ASM筛选器驱动程序是否按预期工作。

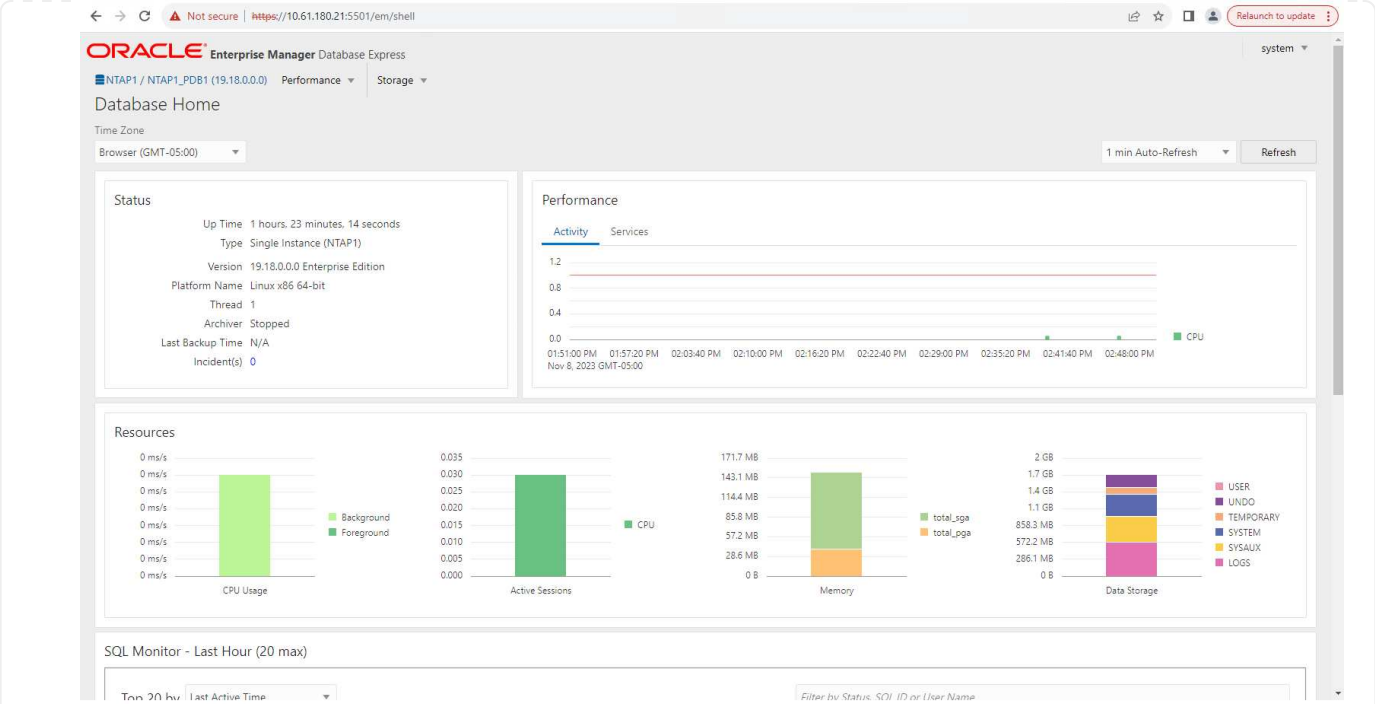
```
[oracle@ora_01 ~]$ asmcmd
ASMCMD> lsdg
State    Type    Rebal  Sector  Logical_Sector  Block       AU
Total_MB  Free_MB  Req_mir_free_MB  Usable_file_MB  Offline_disks
Voting_files  Name
MOUNTED  EXTERN  N         512             512   4096  4194304
327680   318644                0          318644              0
N  DATA/
MOUNTED  EXTERN  N         512             512   4096  4194304
81920    78880                0           78880              0
N  LOGS/
ASMCMD> lsdsk
Path
AFD:ORA_01_DAT1_01
AFD:ORA_01_DAT1_03
AFD:ORA_01_DAT1_05
AFD:ORA_01_DAT1_07
AFD:ORA_01_DAT2_02
AFD:ORA_01_DAT2_04
AFD:ORA_01_DAT2_06
AFD:ORA_01_DAT2_08
AFD:ORA_01_LOGS_01
AFD:ORA_01_LOGS_02
ASMCMD> afd_state
ASMCMD-9526: The AFD state is 'LOADED' and filtering is 'ENABLED' on
host 'ora_01'
ASMCMD>
```

3. 登录到Oracle Enterprise Manager Express以验证数据库。


Not secure | https://10.61.180.21:5500/em/login
Relaunch to update
ORACLE ENTERPRISE MANAGER
DATABASE EXPRESS
Username
Password
Container Name
Log in
ORACLE
Copyright 2013, 2020, Oracle and/or its affiliates. All rights reserved.


Not secure | https://10.61.180.21:5500/em/shell
Relaunch to update
ORACLE Enterprise Manager Database Express
system
NTAP1 (19.18.0.0.0)
Performance
Storage
Database Home
Time Zone
Browser (GMT-05:00)
1 min Auto-Refresh
Refresh
Status
Up Time 1 hours, 7 minutes, 23 seconds
Type Single Instance (NTAP1)
CDB (3 PDB(s))
Version 19.18.0.0.0 Enterprise Edition
Platform Name Linux x86 64-bit
Thread 1
Archiver Stopped
Last Backup Time N/A
Incident(s) 4
Performance
Activity
Services
Containers
CPU
Nov 8, 2023 GMT-05:00
Resources
Host CPU
Other
Instance(s)
Active Sessions
Wait
User I/O
CPU
Memory
total_sga
total_pga
target_pga
shared pool
large pool
buffer cache
Shared IO P...
Data Storage
NTAP1_PDB3
NTAP1_PDB2
NTAP1_PDB1
SQL Monitor - Last Hour (20 max)
Top 20 by
Last Active Time
Filter by Status, SQL ID or User Name

```
Enable additional port from sqlplus for login to individual
container database or PDBs.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 NTAP1_PDB1                     READ WRITE NO
         4 NTAP1_PDB2                     READ WRITE NO
         5 NTAP1_PDB3                     READ WRITE NO
SQL> alter session set container=NTAP1_PDB1;

Session altered.

SQL> select dbms_xdb_config.gethttpsport() from dual;

DBMS_XDB_CONFIG.GETHTTPSPORT()
------------------------------
                             0

SQL> exec DBMS_XDB_CONFIG.SETHTTPSPORT(5501);

PL/SQL procedure successfully completed.

SQL> select dbms_xdb_config.gethttpsport() from dual;

DBMS_XDB_CONFIG.GETHTTPSPORT()
------------------------------
                          5501

login to NTAP1_PDB1 from port 5501.
```



使用**SnapCenter**进行**Oracle**备份、还原和克隆

请参阅TR-4979 "借助子系统装载的FSx ONTAP、在基于AWS的VMware Cloud中简化自我管理Oracle" 部分。 `Oracle backup, restore, and clone with SnapCenter` 有关设置SnapCenter以及执行数据库备份、还原和克隆工作流的详细信息。

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- NetApp ASA：全闪存SAN阵列

  "https://www.netapp.com/data-storage/all-flash-san-storage-array/"

- 在安装新数据库的情况下为独立服务器安装Oracle网格基础架构

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-oracle-grid-infrastructure-for-a-standalone-server-with-a-new-database-installation.html#GUID-0B1CEE8C-C893-46AA-8A6A-7B5FAAEC72B3"

- 使用响应文件安装和配置Oracle数据库

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-and-configuring-oracle-database-using-response-files.html#GUID-D53355E9-E901-4224-9A2A-B882070EDDF7"

- 将Red Hat Enterprise Linux 8.2与ONTAP结合使用

  "https://docs.netapp.com/us-en/ontap-sanhost/hu_rhel_82.html#all-san-array-configurations"

TR-4979：《在AWS上的VMware Cloud中使用子系统装载的FSx ONTAP简化的自行管理Oracle》

NetApp公司Allen Cao、Niyaz Mohamed

目的

几十年来、企业一直在私有数据中心运行基于VMware的Oracle。AWS上的VMware Cloud (VMC)提供了一个按钮式解决方案、可将VMware的企业级软件定义的数据中心(Software-定义 的数据中心、SDDC)软件引入AWS云的专用弹性裸机基础架构。AWS FSx ONTAP为VMC SDDC和Data Fabric提供高级存储、使客户能够在基于vSphere®的私有云、公共云和混合云环境中运行Oracle等业务关键型应用程序、并优化对AWS服务的访问。无论是现有的Oracle工作负载还是新的Oracle工作负载、AWS上的VMC都可以在VMware上提供熟悉、简化且自行管理的Oracle环境、并可享受AWS云的所有优势、同时将所有平台管理和优化工作推迟到VMware。

本文档演示了如何在使用Amazon FSx ONTAP作为主数据库存储的VMC环境中部署和保护Oracle数据库。Oracle数据库可以作为直接VM子系统装载的LUN或NFS装载的VMware VMDK数据存储库磁盘部署到FSx存储上的VMC中。本技术报告重点介绍如何将Oracle数据库部署为使用iSCSI协议和Oracle ASM的VMC集群中的VM的直接子系统装载FSx存储。此外、我们还演示了如何使用NetApp SnapCenter UI工具备份、还原和克隆Oracle数据库以用于开发或测试、或者在AWS上的VMC中使用其他使用情形来实现高效存储数据库操作。

此解决方案 可解决以下使用情形：

- 在AWS上的VMC中部署Oracle数据库、并将Amazon FSx ONTAP用作主数据库存储
- 使用NetApp SnapCenter工具在AWS上的VMC中备份和还原Oracle数据库
- 使用NetApp SnapCenter工具在AWS上的VMC中为开发/测试或其他使用情形创建Oracle数据库克隆

audience

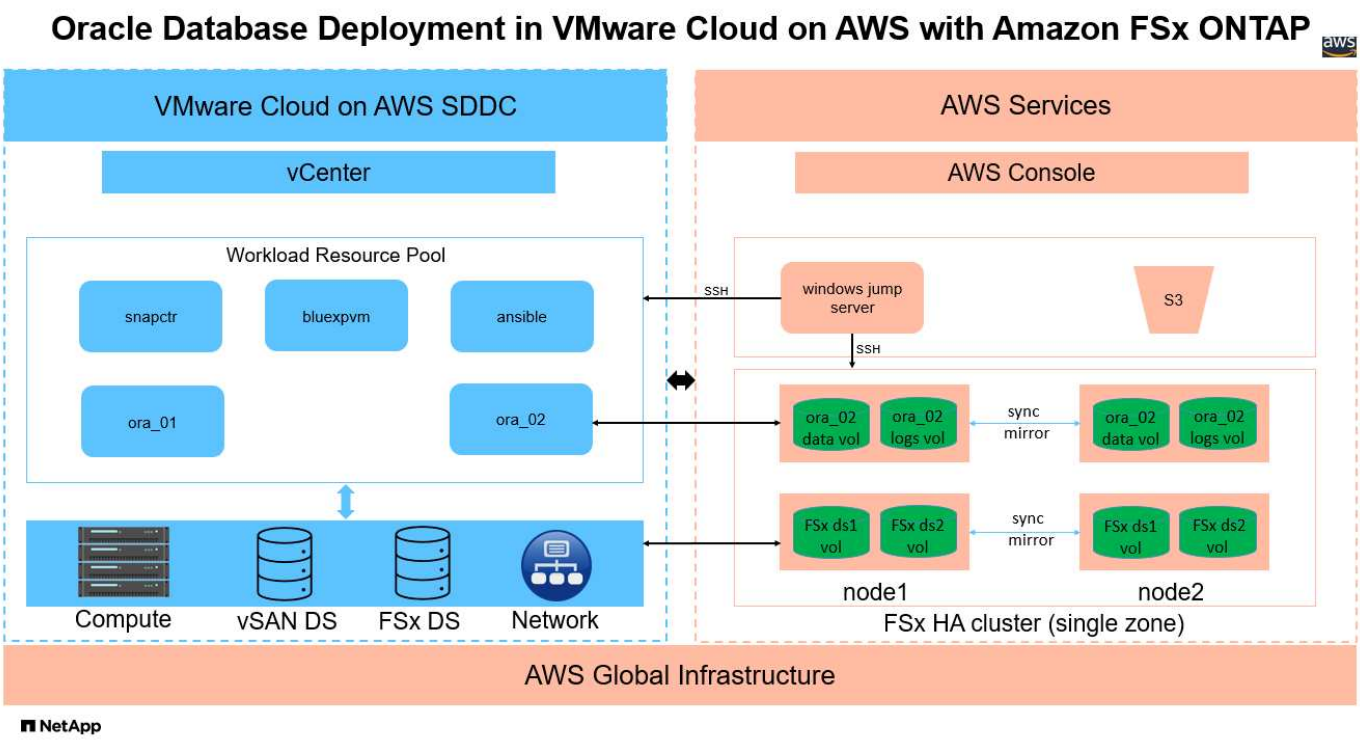
此解决方案 适用于以下人员：

- 希望使用Amazon FSx ONTAP在AWS上的VMC中部署Oracle的数据库开发人员
- 希望在AWS云上的VMC中测试Oracle工作负载的数据库解决方案架构师
- 希望使用Amazon FSx ONTAP部署和管理部署到AWS上VMC的Oracle数据库的存储管理员
- 希望在AWS云上的VMC中建立Oracle数据库的应用程序所有者

解决方案 测试和验证环境

此解决方案的测试和验证是在AWS上使用VMC的实验室环境中执行的、该环境可能与最终部署环境不匹配。有关详细信息，请参见一节 [Key Factors for Deployment Consideration]。

架构



硬件和软件组件

* 硬件 *		
FSX ONTAP 存储	AWS提供的当前版本	一个FSx ONTAP HA集群与VMC位于同一VPC和可用性区域
VMC SDDC集群	Amazon EC2 i3.金属单节点/Intel Xeon E5-2686 CPU、36核/512G RAM	10.37 TB vSAN存储
软件		
RedHat Linux	rell-8.6、4.18.0-372.9.1.el8.x86_64内核	已部署RedHat订阅以进行测试
Windows服务器	2022标准版、10.0.20348内部版本20348	托管SnapCenter服务器
Oracle网格基础架构	版本19.18	已应用RU修补程序p34762026_190000_Linux-x86-64.zip
Oracle 数据库	版本19.18	已应用RU修补程序p34765931_190000_Linux-x86-64.zip
Oracle OPatch	版本12.2.0.1.36	最新修补程序p6880880_190000_Linux-x86-64.zip
SnapCenter 服务器	版本4.9P1	工作组部署
适用于VM的BlueXP备份和恢复	1.0版	部署为一个ova vSphere插件虚拟机

VMware vSphere	8.0.1.00300版	VMware Tools、版本：11365 - Linux、1352 - Windows
打开JDK	版本java-1.8.0-OpenJDK.x86_64	DB VM上的SnapCenter插件要求

在**AWS**上的**VMC**中配置**Oracle**数据库

* 服务器 *	* 数据库 *	**DB**存储
ORA_01	cdb1 (cdb1_pdb1、cdb1_pdb2、cdb1_pdb3)	FSx ONTAP上的VMDK数据存储库
ORA_01	cdb2 (cdb2_pdb)	FSx ONTAP上的VMDK数据存储库
ORA_02	cdb3 (cdb3_pdb1、cdb3_pdb2、cdb3_pdb3)	直接子系统装载FSx ONTAP
ORA_02	cdb4 (cdb4_pdb)	直接子系统装载FSx ONTAP

部署注意事项的关键因素

- *FSx到VMC的连接。*当您在AWS上的VMware Cloud上部署SDDC时、它会在AWS帐户和专用于您的组织的VPC中创建、并由VMware进行管理。您还必须将SDDC连接到属于您的AWS帐户、称为客户AWS帐户。此连接允许SDDC访问属于您的客户帐户的AWS服务。FSx for ONTAP是在您的客户帐户中部署的一项AWS服务。将VMC SDDC连接到您的客户帐户后、VMC SDDC中的VM便可使用FSx存储来直接挂载子系统。
- * FSX存储HA集群单区域或多区域部署。*在这些测试和验证中、我们在一个AWS可用性区域中部署了一个FSX HA集群。NetApp还建议在同一可用性区域中部署适用于NetApp ONTAP的FSx和基于AWS的VMware Cloud、以提高性能并避免可用性区域之间的数据传输费用。
- * FSX存储集群规模估算。*适用于ONTAP 存储文件系统的Amazon FSX可提供高达160、000个原始SSD IOPS、高达4 Gbps吞吐量以及最大192 TiB容量。但是、您可以根据部署时的实际要求、根据已配置的IOPS、吞吐量和存储限制(最小1、024 GiB)来调整集群的大小。可以动态调整容量、而不会影响应用程序可用性。
- * Oracle数据和日志布局。*在测试和验证中、我们分别为数据和日志部署了两个ASM磁盘组。在+data ASM磁盘组中、我们在一个数据卷中配置了四个LUN。在+logs ASM磁盘组中、我们在一个日志卷中配置了两个LUN。通常、在一个Amazon FSx for ONTAP卷中部署多个LUN可提高性能。
- **iSCSI**配置。 VMC SDDC中的数据库VM使用iSCSI协议连接到FSx存储。通过仔细分析Oracle AWR报告来确定应用程序和iSCSI流量吞吐量要求、衡量Oracle数据库峰值I/O吞吐量要求非常重要。NetApp还建议为正确配置多路径的两个FSX iSCSI端点分配四个iSCSI连接。
- *要用于创建的每个Oracle ASM磁盘组的Oracle ASM冗余级别。*由于FSx ONTAP已在FSx集群级别镜像存储、因此应使用外部冗余、这意味着此选项不允许Oracle ASM镜像磁盘组的内容。
- 数据库备份。 NetApp提供了一个SnapCenter软件套件、可通过用户友好的用户界面进行数据库备份、还原和克隆。NetApp建议实施此类管理工具、以实现快速(不到一分钟)的快照备份、快速(几分钟)的数据库还原和数据库克隆。

解决方案 部署

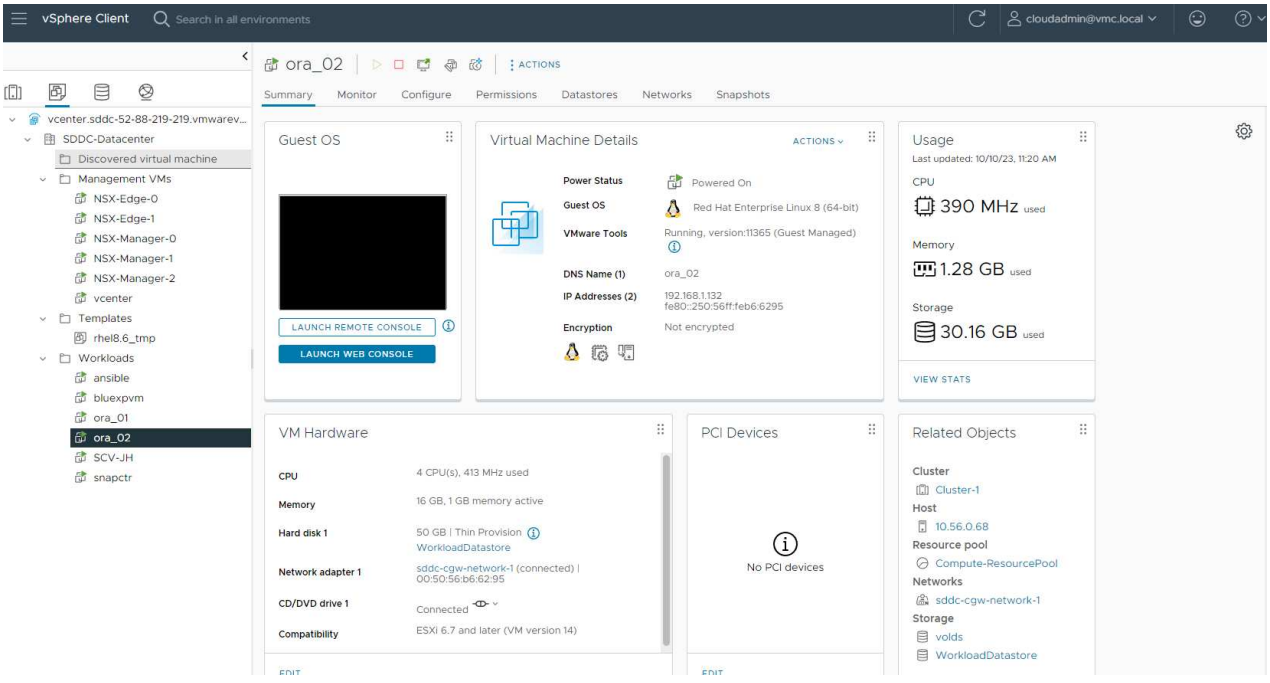
以下各节提供了在AWS上的VMC中将Oracle 19c部署ONTAP到单节点中的数据库VM的分步过程使用Oracle ASM作为数据库卷管理器重新启动配置。

部署需要满足以下前提条件。

1. 我们创建了一个使用VMware Cloud on AWS的软件定义的数据中心(Software-definated Data Center、SDDC)。有关如何在VMC中创建SDDC的详细说明、请参见VMware文档 "在AWS上开始使用VMware Cloud"
2. 已设置AWS帐户、并已在您的AWS帐户中创建必要的VPC和网段。AWS帐户已链接到您的VMC SDDC。
3. 从AWS EC2控制台中、部署适用于ONTAP存储HA集群的Amazon FSx以托管Oracle数据库卷。如果您不熟悉FSX存储的部署、请参见相关文档 "为ONTAP 文件系统创建FSX" 了解分步说明。
4. 可以使用以下Terraform自动化工具包执行上述步骤、该工具包可创建一个EC2实例、作为通过SSH和FSx文件系统进行VMC访问时的SDDC的跳转主机。在执行前、请仔细阅读说明并根据您的环境更改变量。

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

5. 在AWS上的VMware SDDC中构建VM、用于托管要在VMC中部署的Oracle环境。在我们的演示中、我们构建了两个Linux VM作为Oracle数据库服务器、一个Windows服务器用于SnapCenter服务器、一个可选的Linux服务器作为可选的控制器、用于根据需要自动安装或配置Oracle。下面是用于解决方案验证的实验室环境的快照。



6. 此外、NetApp还提供了多个自动化工具包、以便在适用时运行Oracle部署和配置。请参见 "DB自动化工具包" 有关详细信息 …

确保已在Oracle VM根卷中至少分配50G、以便有足够的空间来暂存Oracle安装文件。

DB VM内核配置

在配置了前提条件的情况下、以管理员用户身份通过SSH登录到Oracle VM、并使用sudo向root用户配置Linux内核以进行Oracle安装。Oracle安装文件可以暂存到AWS S3存储分段中、然后传输到VM。

1. 创建暂存目录 `/tmp/archive` 文件夹并设置 `777` 权限。

```
mkdir /tmp/archive
```

```
chmod 777 /tmp/archive
```

2. 将Oracle二进制安装文件和其他所需的rpm文件下载并暂存到 `/tmp/archive` 目录。

   请参见以下要在中说明的安装文件列表 `/tmp/archive` 在DB VM上。

```
[admin@ora_02 ~]$ ls -l /tmp/archive/
total 10539364
-rw-rw-r--. 1 admin  admin         19112 Oct  4 17:04 compat-
libcap1-1.10-7.el7.x86_64.rpm
-rw-rw-r--. 1 admin  admin    3059705302 Oct  4 17:10
LINUX.X64_193000_db_home.zip
-rw-rw-r--. 1 admin  admin    2889184573 Oct  4 17:11
LINUX.X64_193000_grid_home.zip
-rw-rw-r--. 1 admin  admin        589145 Oct  4 17:04
netapp_linux_unified_host_utilities-7-1.x86_64.rpm
-rw-rw-r--. 1 admin  admin         31828 Oct  4 17:04 oracle-
database-preinstall-19c-1.0-2.el8.x86_64.rpm
-rw-rw-r--. 1 admin  admin    2872741741 Oct  4 17:12
p34762026_190000_Linux-x86-64.zip
-rw-rw-r--. 1 admin  admin    1843577895 Oct  4 17:13
p34765931_190000_Linux-x86-64.zip
-rw-rw-r--. 1 admin  admin     124347218 Oct  4 17:13
p6880880_190000_Linux-x86-64.zip
-rw-rw-r--. 1 admin  admin        257136 Oct  4 17:04
policycoreutils-python-utils-2.9-9.el8.noarch.rpm
[admin@ora_02 ~]$
```

3. 安装Oracle 19c预安装RPM、以满足大多数内核配置要求。

```
yum install /tmp/archive/oracle-database-preinstall-19c-1.0-
2.el8.x86_64.rpm
```

4. 下载并安装缺少的 `compat-libcap1` 在Linux 8中。

```
yum install /tmp/archive/compat-libcap1-1.10-7.el7.x86_64.rpm
```

5. 从NetApp下载并安装NetApp主机实用程序。

```
yum install /tmp/archive/netapp_linux_unified_host_utilities-7-
1.x86_64.rpm
```

6. 安装 `policycoreutils-python-utils`。

```
yum install /tmp/archive/policycoreutils-python-utils-2.9-
9.el8.noarch.rpm
```

7. 安装Open JDK 1.8版。

```
yum install java-1.8.0-openjdk.x86_64
```

8. 安装iSCSI启动程序实用程序。

```
yum install iscsi-initiator-utils
```

9. 安装sg3_utils。

```
yum install sg3_utils
```

10. 安装device-maper-Multipath。

```
yum install device-mapper-multipath
```

11. 在当前系统中禁用透明页面。

```
echo never > /sys/kernel/mm/transparent_hugepage/enabled
```

```
echo never > /sys/kernel/mm/transparent_hugepage/defrag
```

12. 在中添加以下行 `/etc/rc.local` 以禁用 `transparent_hugepage` 重新启动后。

```
vi /etc/rc.local
```

```
  # Disable transparent hugepages
          if test -f /sys/kernel/mm/transparent_hugepage/enabled;
then
            echo never > /sys/kernel/mm/transparent_hugepage/enabled
          fi
          if test -f /sys/kernel/mm/transparent_hugepage/defrag;
then
            echo never > /sys/kernel/mm/transparent_hugepage/defrag
          fi
```

13. 通过更改禁用SELinux `SELINUX=enforcing` to `SELINUX=disabled`。要使更改生效、必须重新启动主机。

```
vi /etc/sysconfig/selinux
```

14. 将以下行添加到 `limit.conf` 设置文件描述符限制和堆栈大小。

```
vi /etc/security/limits.conf
```

```
*               hard    nofile          65536
*               soft    stack           10240
```

15. 如果没有按照以下说明配置交换空间、请向DB VM添加交换空间： "如何使用交换文件分配内存以用作Amazon EC2实例中的交换空间？" 要添加的确切空间量取决于RAM大小、最高可达16G。
16. 更改 `node.session.timeo.replacement_timeout` 在中 `iscsi.conf` 配置文件的时间从120秒到5秒不等。

```
vi /etc/iscsi/iscsid.conf
```

17. 在EC2实例上启用并启动iSCSI服务。

```
systemctl enable iscsid
```

```
systemctl start iscsid
```

18. 检索要用于数据库LUN映射的iSCSI启动程序地址。

```
cat /etc/iscsi/initiatorname.iscsi
```

19. 为ASM管理用户(Oracle)添加ASM组。

```
groupadd asmadmin
```

```
groupadd asmdba
```

```
groupadd asmoper
```

20. 修改Oracle用户以将ASM组添加为辅助组(Oracle用户应在安装Oracle预安装RPM后创建)。

```
usermod -a -G asmadmin oracle
```

```
usermod -a -G asmdba oracle
```

```
usermod -a -G asmoper oracle
```

21. 如果Linux防火墙处于活动状态、请停止并禁用它。

```
systemctl stop firewalld
```

```
systemctl disable firewalld
```

22. 通过取消注释为管理员用户启用无密码sudo `# %wheel ALL=(ALL) NOPASSWD: ALL` 行。更改文件权限以进行编辑。

```
chmod 640 /etc/sudoers
```

```
vi /etc/sudoers
```

```
chmod 440 /etc/sudoers
```

23. 重新启动EC2实例。

配置**FSx ONTAP LUN**并将其映射到数据库虚拟机

通过ssh和FSx集群管理IP以fsxadmin用户身份从命令行登录到FSx集群、以配置三个卷。在卷中创建LUN以托管Oracle数据库二进制文件、数据文件和日志文件。

1. 以fsxadmin用户身份通过SSH登录到FSX集群。

```
ssh fsxadmin@10.49.0.74
```

2. 执行以下命令为Oracle二进制文件创建卷。

```
vol create -volume ora_02_biny -aggregate aggr1 -size 50G -state
online  -type RW -snapshot-policy none -tiering-policy snapshot-only
```

3. 执行以下命令为Oracle数据创建卷。

```
vol create -volume ora_02_data -aggregate aggr1 -size 100G -state
online  -type RW -snapshot-policy none -tiering-policy snapshot-only
```

4. 执行以下命令为Oracle日志创建卷。

```
vol create -volume ora_02_logs -aggregate aggr1 -size 100G -state
online  -type RW -snapshot-policy none -tiering-policy snapshot-only
```

5. 验证已创建的卷。

```
vol show ora*
```

命令的输出：

```
FsxId0c00cec8dad373fd1::> vol show ora*
Vserver   Volume       Aggregate    State      Type       Size
Available Used%
--------- ------------ ------------ ---------- ---- ----------
---------- -----
nim       ora_02_biny  aggr1        online     RW         50GB
22.98GB   51%
nim       ora_02_data  aggr1        online     RW        100GB
18.53GB   80%
nim       ora_02_logs  aggr1        online     RW         50GB
7.98GB   83%
```

6. 在数据库二进制卷中创建二进制LUN。

```
lun create -path /vol/ora_02_biny/ora_02_biny_01 -size 40G -ostype
linux
```

7. 在数据库数据卷中创建数据LUN。

```
lun create -path /vol/ora_02_data/ora_02_data_01 -size 20G -ostype
linux
```

```
lun create -path /vol/ora_02_data/ora_02_data_02 -size 20G -ostype
linux
```

```
lun create -path /vol/ora_02_data/ora_02_data_03 -size 20G -ostype
linux
```

```
lun create -path /vol/ora_02_data/ora_02_data_04 -size 20G -ostype
linux
```

8. 在数据库日志卷中创建日志LUN。

```
lun create -path /vol/ora_02_logs/ora_02_logs_01 -size 40G -ostype
linux
```

```
lun create -path /vol/ora_02_logs/ora_02_logs_02 -size 40G -ostype
linux
```

9. 使用从上述EC2内核配置的步骤14中检索到的启动程序为EC2实例创建一个igroup。

```
igroup create -igroup ora_02 -protocol iscsi -ostype linux
-initiator iqn.1994-05.com.redhat:f65fed7641c2
```

10. 将LUN映射到上述创建的igroup。为每个附加LUN依次增加LUN ID。

```
lun map -path /vol/ora_02_biny/ora_02_biny_01 -igroup ora_02
-vserver svm_ora -lun-id 0
lun map -path /vol/ora_02_data/ora_02_data_01 -igroup ora_02
-vserver svm_ora -lun-id 1
lun map -path /vol/ora_02_data/ora_02_data_02 -igroup ora_02
-vserver svm_ora -lun-id 2
lun map -path /vol/ora_02_data/ora_02_data_03 -igroup ora_02
-vserver svm_ora -lun-id 3
lun map -path /vol/ora_02_data/ora_02_data_04 -igroup ora_02
-vserver svm_ora -lun-id 4
lun map -path /vol/ora_02_logs/ora_02_logs_01 -igroup ora_02
-vserver svm_ora -lun-id 5
lun map -path /vol/ora_02_logs/ora_02_logs_02 -igroup ora_02
-vserver svm_ora -lun-id 6
```

11. 验证LUN映射。

```
mapping show
```

这将返回：

```
FsxId0c00cec8dad373fd1::> mapping show
  (lun mapping show)
Vserver    Path                                      Igroup   LUN ID
Protocol
---------- ----------------------------------------  -------  ------
--------
nim        /vol/ora_02_biny/ora_02_u01_01            ora_02        0
iscsi
nim        /vol/ora_02_data/ora_02_u02_01            ora_02        1
iscsi
nim        /vol/ora_02_data/ora_02_u02_02            ora_02        2
iscsi
nim        /vol/ora_02_data/ora_02_u02_03            ora_02        3
iscsi
nim        /vol/ora_02_data/ora_02_u02_04            ora_02        4
iscsi
nim        /vol/ora_02_logs/ora_02_u03_01            ora_02        5
iscsi
nim        /vol/ora_02_logs/ora_02_u03_02            ora_02        6
iscsi
```

现在、导入并设置FSx ONTAP存储、用于在VMC数据库VM上安装Oracle网格基础架构和数据库。

1. 从Windows跳转服务器使用Putty以管理员用户身份通过SSH登录到数据库VM。
2. 使用任一SVM iSCSI IP地址发现FSX iSCSI端点。更改特定于环境的门户地址。

```
sudo iscsiadm iscsiadm --mode discovery --op update --type
sendtargets --portal 10.49.0.12
```

3. 登录到每个目标以建立iSCSI会话。

```
sudo iscsiadm --mode node -l all
```

命令的预期输出为：

```
[ec2-user@ip-172-30-15-58 ~]$ sudo iscsiadm --mode node -l all
Logging in to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
10.49.0.12,3260]
Logging in to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
10.49.0.186,3260]
Login to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
10.49.0.12,3260] successful.
Login to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
10.49.0.186,3260] successful.
```

4. 查看并验证活动iSCSI会话的列表。

```
sudo iscsiadm --mode session
```

返回iSCSI会话。

```
[ec2-user@ip-172-30-15-58 ~]$ sudo iscsiadm --mode session
tcp: [1] 10.49.0.186:3260,1028 iqn.1992-
08.com.netapp:sn.545a38bf06ac11ee8503e395ab90d704:vs.3 (non-flash)
tcp: [2] 10.49.0.12:3260,1029 iqn.1992-
08.com.netapp:sn.545a38bf06ac11ee8503e395ab90d704:vs.3 (non-flash)
```

5. 验证LUN是否已导入到主机中。

```
sudo sanlun lun show
```

此操作将从FSX返回Oracle LUN的列表。

```
[admin@ora_02 ~]$ sudo sanlun lun show
controller(7mode/E-Series)/
device          host                  lun
vserver(cDOT/FlashRay)        lun-pathname
filename        adapter    protocol   size    product
------------------------------------------------------------------------
------------------------------------------------------------
nim                            /vol/ora_02_logs/ora_02_u03_02
/dev/sdo        host34     iSCSI      20g     cDOT
nim                            /vol/ora_02_logs/ora_02_u03_01
/dev/sdn        host34     iSCSI      20g     cDOT
nim                            /vol/ora_02_data/ora_02_u02_04
/dev/sdm        host34     iSCSI      20g     cDOT
nim                            /vol/ora_02_data/ora_02_u02_03
/dev/sdl        host34     iSCSI      20g     cDOT
nim                            /vol/ora_02_data/ora_02_u02_02
/dev/sdk        host34     iSCSI      20g     cDOT
nim                            /vol/ora_02_data/ora_02_u02_01
/dev/sdj        host34     iSCSI      20g     cDOT
nim                            /vol/ora_02_biny/ora_02_u01_01
/dev/sdi        host34     iSCSI      40g     cDOT
nim                            /vol/ora_02_logs/ora_02_u03_02
/dev/sdh        host33     iSCSI      20g     cDOT
nim                            /vol/ora_02_logs/ora_02_u03_01
/dev/sdg        host33     iSCSI      20g     cDOT
nim                            /vol/ora_02_data/ora_02_u02_04
/dev/sdf        host33     iSCSI      20g     cDOT
nim                            /vol/ora_02_data/ora_02_u02_03
/dev/sde        host33     iSCSI      20g     cDOT
nim                            /vol/ora_02_data/ora_02_u02_02
/dev/sdd        host33     iSCSI      20g     cDOT
nim                            /vol/ora_02_data/ora_02_u02_01
/dev/sdc        host33     iSCSI      20g     cDOT
nim                            /vol/ora_02_biny/ora_02_u01_01
/dev/sdb        host33     iSCSI      40g     cDOT
```

6. 配置 `multipath.conf` 包含以下默认条目和黑名单条目的文件。

```
sudo vi /etc/multipath.conf
```

添加以下条目：

```
defaults {
    find_multipaths yes
    user_friendly_names yes
}

blacklist {
    devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
    devnode "^hd[a-z]"
    devnode "^cciss.*"
}
```

7. 启动多路径服务。

```
sudo systemctl start multipathd
```

现在、多路径设备将显示在中 `/dev/mapper` 目录。

```
[ec2-user@ip-172-30-15-58 ~]$ ls -l /dev/mapper
total 0
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e68512d -> ../dm-0
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685141 -> ../dm-1
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685142 -> ../dm-2
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685143 -> ../dm-3
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685144 -> ../dm-4
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685145 -> ../dm-5
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685146 -> ../dm-6
crw------- 1 root root 10, 236 Mar 21 18:19 control
```

8. 以fsxadmin用户身份通过SSH登录到FSx ONTAP集群、以检索以6c574xxx…开头的每个LUN的串行十六进制编号、十六进制编号以3600a0980开头、即AWS供应商ID。

```
lun show -fields serial-hex
```

并返回如下内容：

```
FsxId02ad7bf3476b741df::> lun show -fields serial-hex
vserver path                            serial-hex
------- ------------------------------- ------------------------
svm_ora /vol/ora_02_biny/ora_02_biny_01 6c574235472455534e68512d
svm_ora /vol/ora_02_data/ora_02_data_01 6c574235472455534e685141
svm_ora /vol/ora_02_data/ora_02_data_02 6c574235472455534e685142
svm_ora /vol/ora_02_data/ora_02_data_03 6c574235472455534e685143
svm_ora /vol/ora_02_data/ora_02_data_04 6c574235472455534e685144
svm_ora /vol/ora_02_logs/ora_02_logs_01 6c574235472455534e685145
svm_ora /vol/ora_02_logs/ora_02_logs_02 6c574235472455534e685146
7 entries were displayed.
```

9. 更新 `/dev/multipath.conf` 文件、用于为多路径设备添加用户友好名称。

```
sudo vi /etc/multipath.conf
```

包含以下条目：

```
multipaths {
        multipath {
                wwid            3600a09806c574235472455534e68512d
                alias           ora_02_biny_01
        }
        multipath {
                wwid            3600a09806c574235472455534e685141
                alias           ora_02_data_01
        }
        multipath {
                wwid            3600a09806c574235472455534e685142
                alias           ora_02_data_02
        }
        multipath {
                wwid            3600a09806c574235472455534e685143
                alias           ora_02_data_03
        }
        multipath {
                wwid            3600a09806c574235472455534e685144
                alias           ora_02_data_04
        }
        multipath {
                wwid            3600a09806c574235472455534e685145
                alias           ora_02_logs_01
        }
        multipath {
                wwid            3600a09806c574235472455534e685146
                alias           ora_02_logs_02
        }
}
```

10. 重新启动多路径服务以验证下的设备 `/dev/mapper` 已更改为LUN名称与串行十六进制ID。

```
sudo systemctl restart multipathd
```

检查 `/dev/mapper` 返回如下内容：

```
[ec2-user@ip-172-30-15-58 ~]$ ls -l /dev/mapper
total 0
crw------- 1 root root 10, 236 Mar 21 18:19 control
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_02_biny_01 -> ../dm-
0
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_02_data_01 -> ../dm-
1
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_02_data_02 -> ../dm-
2
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_02_data_03 -> ../dm-
3
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_02_data_04 -> ../dm-
4
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_02_logs_01 -> ../dm-
5
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_02_logs_02 -> ../dm-
6
```

11. 使用一个主分区对二进制LUN进行分区。

```
sudo fdisk /dev/mapper/ora_02_biny_01
```

12. 使用XFS文件系统格式化分区的二进制LUN。

```
sudo mkfs.xfs /dev/mapper/ora_02_biny_01p1
```

13. 将二进制LUN挂载到 `/u01`。

```
sudo mkdir /u01
```

```
sudo mount -t xfs /dev/mapper/ora_02_biny_01p1 /u01
```

14. 更改 `/u01` Oracle用户及其关联主组的挂载点所有权。

```
sudo chown oracle:oinstall /u01
```

15. 查找二进制LUN的UUI。

```
sudo blkid /dev/mapper/ora_02_biny_01p1
```

16. 将挂载点添加到 `/etc/fstab`。

```
sudo vi /etc/fstab
```

添加以下行。

```
UUID=d89fb1c9-4f89-4de4-b4d9-17754036d11d       /u01    xfs
defaults,nofail 0       2
```

17. 以root用户身份为Oracle设备添加udev规则。

```
vi /etc/udev/rules.d/99-oracle-asmdevices.rules
```

包括以下条目：

```
ENV{DM_NAME}=="ora*", GROUP:="oinstall", OWNER:="oracle",
MODE:="660"
```

18. 以root用户身份重新加载udev规则。

```
udevadm control --reload-rules
```

19. 以root用户身份触发udev规则。

```
udevadm trigger
```

20. 以root用户身份重新加载multipathd。

```
systemctl restart multipathd
```

21. 重新启动EC2实例主机。

Oracle网格基础架构安装

1. 以管理员用户身份通过SSH登录到DB VM、并通过取消注释启用密码身份验证 `PasswordAuthentication yes` 然后进行注释 `PasswordAuthentication no`。

```
sudo vi /etc/ssh/sshd_config
```

2. 重新启动sshd服务。

```
sudo systemctl restart sshd
```

3. 重置Oracle用户密码。

```
sudo passwd oracle
```

4. 以Oracle Restart软件所有者用户(Oracle)身份登录。按如下所示创建Oracle目录：

```
mkdir -p /u01/app/oracle
```

```
mkdir -p /u01/app/oraInventory
```

5. 更改目录权限设置。

```
chmod -R 775 /u01/app
```

6. 创建网格主目录并进行更改。

```
mkdir -p /u01/app/oracle/product/19.0.0/grid
```

```
cd /u01/app/oracle/product/19.0.0/grid
```

7. 解压缩网格安装文件。

```
unzip -q /tmp/archive/LINUX.X64_193000_grid_home.zip
```

8. 从网格主页中、删除 `OPatch` 目录。

```
rm -rf OPatch
```

9. 从网格主页解压缩 `p6880880_190000_Linux-x86-64.zip`。

```
unzip -q /tmp/archive/p6880880_190000_Linux-x86-64.zip
```

10. 从网格主页修改 `cv/admin/cvu_config`、取消注释并替换 `CV_ASSUME_DISTID=OEL5` 使用 `CV_ASSUME_DISTID=OL7`。

```
vi cv/admin/cvu_config
```

11. 准备 `gridsetup.rsp` 文件以进行静默安装、并将rsp文件置于中 `/tmp/archive` 目录。rsp文件应使用以下信息涵盖A、B和G部分：

```
INVENTORY_LOCATION=/u01/app/oraInventory
oracle.install.option=HA_CONFIG
ORACLE_BASE=/u01/app/oracle
oracle.install.asm.OSDBA=asmdba
oracle.install.asm.OSOPER=asmoper
oracle.install.asm.OSASM=asmadmin
oracle.install.asm.SYSASMPassword="SetPWD"
oracle.install.asm.diskGroup.name=DATA
oracle.install.asm.diskGroup.redundancy=EXTERNAL
oracle.install.asm.diskGroup.AUSize=4
oracle.install.asm.diskGroup.disks=/dev/mapper/ora_02_data_01,/dev/m
apper/ora_02_data_02,/dev/mapper/ora_02_data_03,/dev/mapper/ora_02_d
ata_04
oracle.install.asm.diskGroup.diskDiscoveryString=/dev/mapper/*
oracle.install.asm.monitorPassword="SetPWD"
oracle.install.asm.configureAFD=true
```

12. 以root用户身份登录到EC2实例并进行设置 `ORACLE_HOME` 和 `ORACLE_BASE`。

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

```
export ORACLE_BASE=/tmp
```

```
cd /u01/app/oracle/product/19.0.0/grid/bin
```

13. 初始化磁盘设备以与Oracle ASM筛选器驱动程序结合使用。

```
./asmcmd afd_label DATA01 /dev/mapper/ora_02_data_01 --init
```

```
./asmcmd afd_label DATA02 /dev/mapper/ora_02_data_02 --init
```

```
./asmcmd afd_label DATA03 /dev/mapper/ora_02_data_03 --init
```

```
./asmcmd afd_label DATA04 /dev/mapper/ora_02_data_04 --init
```

```
./asmcmd afd_label LOGS01 /dev/mapper/ora_02_logs_01 --init
```

```
./asmcmd afd_label LOGS02 /dev/mapper/ora_02_logs_02 --init
```

14. 安装 `cvuqdisk-1.0.10-1.rpm`。

```
rpm -ivh /u01/app/oracle/product/19.0.0/grid/cv/rpm/cvuqdisk-1.0.10-
1.rpm
```

15. 未设置 `$ORACLE_BASE`。

```
unset ORACLE_BASE
```

16. 以Oracle用户身份登录到EC2实例、然后在中提取修补程序 `/tmp/archive` 文件夹。

```
unzip -q /tmp/archive/p34762026_190000_Linux-x86-64.zip -d
/tmp/archive
```

17. 从Grid home /u01/app/oracle/product/19.0.0/grid中、以Oracle用户身份启动 `gridSetup.sh` 用于网格基础架构安装。

```
 ./gridSetup.sh -applyRU /tmp/archive/34762026/ -silent
-responseFile /tmp/archive/gridsetup.rsp
```

18. 以root用户身份执行以下脚本：

```
/u01/app/oraInventory/orainstRoot.sh
```

```
/u01/app/oracle/product/19.0.0/grid/root.sh
```

19. 以root用户身份重新加载multipathd。

```
systemctl restart multipathd
```

20. 以Oracle用户身份执行以下命令以完成配置：

```
/u01/app/oracle/product/19.0.0/grid/gridSetup.sh -executeConfigTools
-responseFile /tmp/archive/gridsetup.rsp -silent
```

21. 以Oracle用户身份创建日志磁盘组。

```
bin/asmca -silent -sysAsmPassword 'yourPWD' -asmsnmpPassword
'yourPWD' -createDiskGroup -diskGroupName LOGS -disk 'AFD:LOGS*'
-redundancy EXTERNAL -au_size 4
```

22. 以Oracle用户身份、在安装配置后验证网格服务。

```
bin/crsctl stat res -t
```

```
[oracle@ora_02 grid]$ bin/crsctl stat res -t
--------------------------------------------------------------------------------
------------
Name           Target  State        Server                   State
details
--------------------------------------------------------------------------------
------------
Local Resources
--------------------------------------------------------------------------------
------------
ora.DATA.dg
               ONLINE  ONLINE       ora_02                   STABLE
ora.LISTENER.lsnr
               ONLINE  INTERMEDIATE ora_02                   Not All
Endpoints Re

gistered,STABLE
ora.LOGS.dg
               ONLINE  ONLINE       ora_02                   STABLE
ora.asm
               ONLINE  ONLINE       ora_02
Started,STABLE
ora.ons
               OFFLINE OFFLINE      ora_02                   STABLE
--------------------------------------------------------------------------------
------------
Cluster Resources
--------------------------------------------------------------------------------
------------
ora.cssd
      1        ONLINE  ONLINE       ora_02                   STABLE
ora.diskmon
      1        OFFLINE OFFLINE                               STABLE
ora.driver.afd
      1        ONLINE  ONLINE       ora_02                   STABLE
ora.evmd
      1        ONLINE  ONLINE       ora_02                   STABLE
--------------------------------------------------------------------------------
------------
```

23. Valiate ASM筛选器驱动程序状态。

```
[oracle@ora_02 grid]$ export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/grid
[oracle@ora_02 grid]$ export ORACLE_SID=+ASM
[oracle@ora_02 grid]$ export PATH=$PATH:$ORACLE_HOME/bin
[oracle@ora_02 grid]$ asmcmd
ASMCMD> lsdg
State    Type    Rebal  Sector  Logical_Sector  Block       AU
Total_MB  Free_MB  Req_mir_free_MB  Usable_file_MB  Offline_disks
Voting_files  Name
MOUNTED  EXTERN  N         512             512   4096  4194304
81920    81780                0           81780              0
N  DATA/
MOUNTED  EXTERN  N         512             512   4096  4194304
40960    40852                0           40852              0
N  LOGS/
ASMCMD> afd_state
ASMCMD-9526: The AFD state is 'LOADED' and filtering is 'ENABLED' on
host 'ora_02'
ASMCMD> exit
[oracle@ora_02 grid]$
```

24. 验证HA服务状态。

```
[oracle@ora_02 bin]$ ./crsctl check has
CRS-4638: Oracle High Availability Services is online
```

Oracle数据库安装

1. 以Oracle用户身份登录并取消设置 `$ORACLE_HOME` 和 `$ORACLE_SID` 如果已设置。

```
unset ORACLE_HOME
```

```
unset ORACLE_SID
```

2. 创建Oracle DB主目录并将其更改为该目录。

```
mkdir /u01/app/oracle/product/19.0.0/cdb3
```

```
cd /u01/app/oracle/product/19.0.0/cdb3
```

3. 解压缩Oracle数据库安装文件。

```
unzip -q /tmp/archive/LINUX.X64_193000_db_home.zip
```

4. 从数据库主目录中、删除 `OPatch` 目录。

```
rm -rf OPatch
```

5. 从DB主目录中、解压缩 `p6880880_190000_Linux-x86-64.zip`。

```
unzip -q /tmp/archive/p6880880_190000_Linux-x86-64.zip
```

6. 在数据库主页中、修改 `cv/admin/cvu_config` 并取消注释并替换 `CV_ASSUME_DISTID=OEL5` 使用 `CV_ASSUME_DISTID=OL7`。

```
vi cv/admin/cvu_config
```

7. 从 `/tmp/archive` 目录中、解压缩DB 19.18 RU修补程序。

```
unzip -q /tmp/archive/p34765931_190000_Linux-x86-64.zip -d
/tmp/archive
```

8. 在中准备DB静默安装rsp文件 `/tmp/archive/dbinstall.rsp` 具有以下值的目录：

```
oracle.install.option=INSTALL_DB_SWONLY
UNIX_GROUP_NAME=oinstall
INVENTORY_LOCATION=/u01/app/oraInventory
ORACLE_HOME=/u01/app/oracle/product/19.0.0/cdb3
ORACLE_BASE=/u01/app/oracle
oracle.install.db.InstallEdition=EE
oracle.install.db.OSDBA_GROUP=dba
oracle.install.db.OSOPER_GROUP=oper
oracle.install.db.OSBACKUPDBA_GROUP=oper
oracle.install.db.OSDGDBA_GROUP=dba
oracle.install.db.OSKMDBA_GROUP=dba
oracle.install.db.OSRACDBA_GROUP=dba
oracle.install.db.rootconfig.executeRootScript=false
```

9. 从cdb3 home /u01/app/oracle/product/19.0.0/cdb3中、执行无提示纯软件DB安装。

```
 ./runInstaller -applyRU /tmp/archive/34765931/ -silent
-ignorePrereqFailure -responseFile /tmp/archive/dbinstall.rsp
```

10. 以root用户身份运行 `root.sh` 在纯软件安装后执行脚本。

```
/u01/app/oracle/product/19.0.0/db1/root.sh
```

11. 以Oracle用户身份创建 `dbca.rsp` 包含以下条目的文件：

```
gdbName=cdb3.demo.netapp.com
sid=cdb3
createAsContainerDatabase=true
numberOfPDBs=3
pdbName=cdb3_pdb
useLocalUndoForPDBs=true
pdbAdminPassword="yourPWD"
templateName=General_Purpose.dbc
sysPassword="yourPWD"
systemPassword="yourPWD"
dbsnmpPassword="yourPWD"
datafileDestination=+DATA
recoveryAreaDestination=+LOGS
storageType=ASM
diskGroupName=DATA
characterSet=AL32UTF8
nationalCharacterSet=AL16UTF16
listeners=LISTENER
databaseType=MULTIPURPOSE
automaticMemoryManagement=false
totalMemory=8192
```

12. 以Oracle用户身份、使用dbca启动数据库创建。

```
bin/dbca -silent -createDatabase -responseFile /tmp/archive/dbca.rsp
```

输出：

```
Prepare for db operation
7% complete
Registering database with Oracle Restart
11% complete
Copying database files
33% complete
Creating and starting Oracle instance
35% complete
38% complete
42% complete
45% complete
48% complete
Completing Database Creation
53% complete
55% complete
56% complete
Creating Pluggable Databases
60% complete
64% complete
69% complete
78% complete
Executing Post Configuration Actions
100% complete
Database creation complete. For details check the logfiles at:
 /u01/app/oracle/cfgtoollogs/dbca/cdb3.
Database Information:
Global Database Name:cdb3.vmc.netapp.com
System Identifier(SID):cdb3
Look at the log file "/u01/app/oracle/cfgtoollogs/dbca/cdb3/cdb3.log"
for further details.
```

1. 重复步骤2中的相同过程、使用一个PDB在单独的oracle_home /u01/app/oracle/product/19.0.0/cdb4中创建容器数据库cdb4。
2. 作为Oracle用户、请在创建数据库后验证Oracle Restart HA服务、确认所有数据库(cdb3、cdb4)均已向HA服务注册。

   ```
   /u01/app/oracle/product/19.0.0/grid/crsctl stat res -t
   ```

   输出：

   ```
   [oracle@ora_02 bin]$ ./crsctl stat res -t
   --------------------------------------------------------------------------
   ------------
   ```

```
Name           Target  State        Server                   State
details
--------------------------------------------------------------------
------------
Local Resources
--------------------------------------------------------------------
------------
ora.DATA.dg
               ONLINE  ONLINE       ora_02                   STABLE
ora.LISTENER.lsnr
               ONLINE  INTERMEDIATE ora_02                   Not All
Endpoints Re

gistered,STABLE
ora.LOGS.dg
               ONLINE  ONLINE       ora_02                   STABLE
ora.asm
               ONLINE  ONLINE       ora_02
Started,STABLE
ora.ons
               OFFLINE OFFLINE      ora_02                   STABLE
--------------------------------------------------------------------
------------
Cluster Resources
--------------------------------------------------------------------
------------
ora.cdb3.db
      1        ONLINE  ONLINE       ora_02
Open,HOME=/u01/app/o

racle/product/19.0.0

/cdb3,STABLE
ora.cdb4.db
      1        ONLINE  ONLINE       ora_02
Open,HOME=/u01/app/o

racle/product/19.0.0

/cdb4,STABLE
ora.cssd
      1        ONLINE  ONLINE       ora_02                   STABLE
ora.diskmon
      1        OFFLINE OFFLINE                               STABLE
ora.driver.afd
      1        ONLINE  ONLINE       ora_02                   STABLE
```

```
ora.evmd
      1        ONLINE  ONLINE       ora_02                   STABLE
--------------------------------------------------------------------------------
------------
```

3. 设置Oracle用户 `.bash_profile`。

```
vi ~/.bash_profile
```

添加以下条目：

```
export ORACLE_HOME=/u01/app/oracle/product/19.0.0/db3
export ORACLE_SID=db3
export PATH=$PATH:$ORACLE_HOME/bin
alias asm='export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/grid;export
ORACLE_SID=+ASM;export PATH=$PATH:$ORACLE_HOME/bin'
alias cdb3='export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/cdb3;export
ORACLE_SID=cdb3;export PATH=$PATH:$ORACLE_HOME/bin'
alias cdb4='export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/cdb4;export
ORACLE_SID=cdb4;export PATH=$PATH:$ORACLE_HOME/bin'
```

4. 验证为cdb3创建的CDB/PDB。

```
cdb3
```

```
[oracle@ora_02 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 9 08:19:20 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
--------- --------------------
CDB3      READ WRITE

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 CDB3_PDB1                      READ WRITE NO
         4 CDB3_PDB2                      READ WRITE NO
         5 CDB3_PDB3                      READ WRITE NO
SQL>

SQL> select name from v$datafile;

NAME
------------------------------------------------------------------------
------------
+DATA/CDB3/DATAFILE/system.257.1149420273
+DATA/CDB3/DATAFILE/sysaux.258.1149420317
+DATA/CDB3/DATAFILE/undotbs1.259.1149420343
+DATA/CDB3/86B637B62FE07A65E053F706E80A27CA/DATAFILE/system.266.1149
421085
+DATA/CDB3/86B637B62FE07A65E053F706E80A27CA/DATAFILE/sysaux.267.1149
421085
+DATA/CDB3/DATAFILE/users.260.1149420343
+DATA/CDB3/86B637B62FE07A65E053F706E80A27CA/DATAFILE/undotbs1.268.11
49421085
+DATA/CDB3/06FB206DF15ADEE8E065025056B66295/DATAFILE/system.272.1149
422017
+DATA/CDB3/06FB206DF15ADEE8E065025056B66295/DATAFILE/sysaux.273.1149
422017
+DATA/CDB3/06FB206DF15ADEE8E065025056B66295/DATAFILE/undotbs1.271.11
49422017
+DATA/CDB3/06FB206DF15ADEE8E065025056B66295/DATAFILE/users.275.11494
22033

NAME
------------------------------------------------------------------------
------------
+DATA/CDB3/06FB21766256DF9AE065025056B66295/DATAFILE/system.277.1149
422033
+DATA/CDB3/06FB21766256DF9AE065025056B66295/DATAFILE/sysaux.278.1149
422033
```

```
+DATA/CDB3/06FB21766256DF9AE065025056B66295/DATAFILE/undotbs1.276.11
49422033
+DATA/CDB3/06FB21766256DF9AE065025056B66295/DATAFILE/users.280.11494
22049
+DATA/CDB3/06FB22629AC1DFD7E065025056B66295/DATAFILE/system.282.1149
422049
+DATA/CDB3/06FB22629AC1DFD7E065025056B66295/DATAFILE/sysaux.283.1149
422049
+DATA/CDB3/06FB22629AC1DFD7E065025056B66295/DATAFILE/undotbs1.281.11
49422049
+DATA/CDB3/06FB22629AC1DFD7E065025056B66295/DATAFILE/users.285.11494
22063

19 rows selected.

SQL>
```

5. 验证为cdb4创建的CDB/PDB。

```
cdb4
```

```
[oracle@ora_02 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Oct 9 08:20:26 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
--------- --------------------
CDB4      READ WRITE

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
```

```
         2 PDB$SEED                       READ ONLY  NO
         3 CDB4_PDB                       READ WRITE NO
SQL>

SQL> select name from v$datafile;

NAME
------------------------------------------------------------------------
------------
+DATA/CDB4/DATAFILE/system.286.1149424943
+DATA/CDB4/DATAFILE/sysaux.287.1149424989
+DATA/CDB4/DATAFILE/undotbs1.288.1149425015
+DATA/CDB4/86B637B62FE07A65E053F706E80A27CA/DATAFILE/system.295.1149
425765
+DATA/CDB4/86B637B62FE07A65E053F706E80A27CA/DATAFILE/sysaux.296.1149
425765
+DATA/CDB4/DATAFILE/users.289.1149425015
+DATA/CDB4/86B637B62FE07A65E053F706E80A27CA/DATAFILE/undotbs1.297.11
49425765
+DATA/CDB4/06FC3070D5E12C23E065025056B66295/DATAFILE/system.301.1149
426581
+DATA/CDB4/06FC3070D5E12C23E065025056B66295/DATAFILE/sysaux.302.1149
426581
+DATA/CDB4/06FC3070D5E12C23E065025056B66295/DATAFILE/undotbs1.300.11
49426581
+DATA/CDB4/06FC3070D5E12C23E065025056B66295/DATAFILE/users.304.11494
26597

11 rows selected.
```

6. 使用sqlplus以sysdba身份登录到每个cdb、并将两个CDBS的数据库恢复目标大小设置为+logs磁盘组大小。

```
alter system set db_recovery_file_dest_size = 40G scope=both;
```

7. 使用sqlplus以sysdba身份登录到每个cdb、并使用以下命令集按顺序启用归档日志模式。

```
sqlplus /as sysdba
```

```
shutdown immediate;
```

```
startup mount;
```

```
alter database archivelog;
```

```
alter database open;
```

至此、在适用于ONTAP存储的Amazon FSx和VMC DB VM上完成了Oracle 19c 19.18版重新启动部署。如果需要、NetApp建议将Oracle控制文件和联机日志文件重新定位到+logs磁盘组。

使用SnapCenter进行Oracle备份、还原和克隆

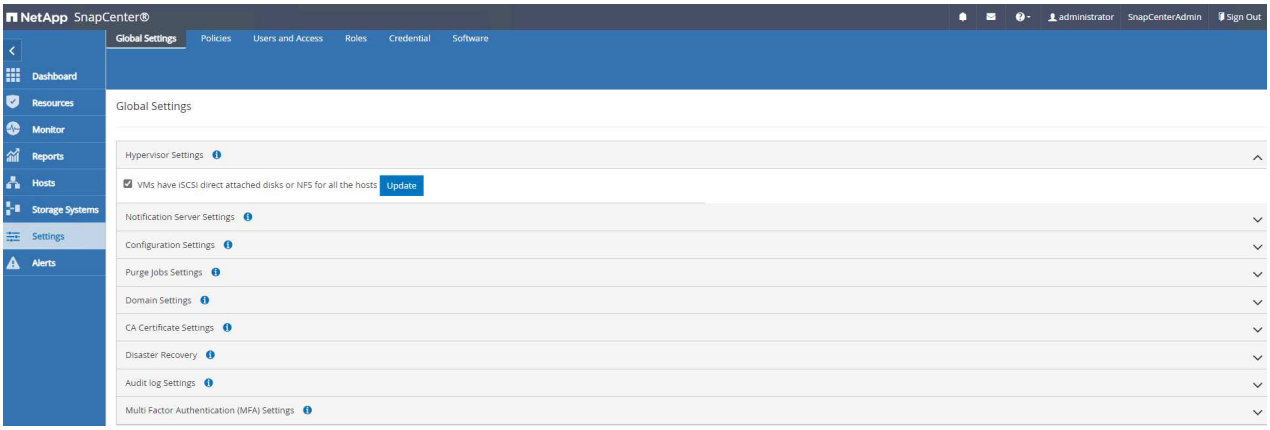
SnapCenter设置

SnapCenter依靠数据库VM上的主机端插件来执行应用程序感知型数据保护管理活动。有关适用于Oracle的NetApp SnapCenter插件的详细信息、请参见此文档 "您可以使用适用于 Oracle 数据库的插件执行什么操作"。下面简要介绍了为Oracle数据库备份、恢复和克隆设置SnapCenter的步骤。

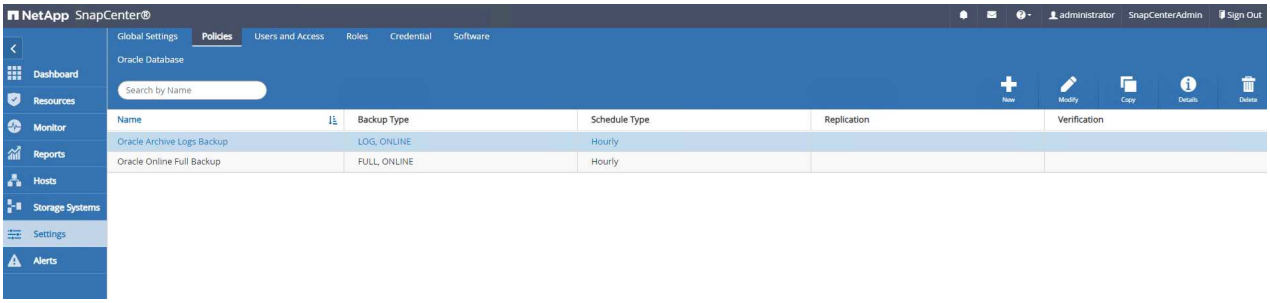
1. 从NetApp 支持站点 下载最新版本的SnapCenter软件： "NetApp 支持下载"。
2. 以管理员身份从安装最新的Java JDK "获取适用于桌面应用程序的Java" 在SnapCenter服务器Windows主机上。

> 如果Windows服务器部署在域环境中、请将域用户添加到SnapCenter服务器本地管理员组、然后使用域用户运行SnapCenter安装。

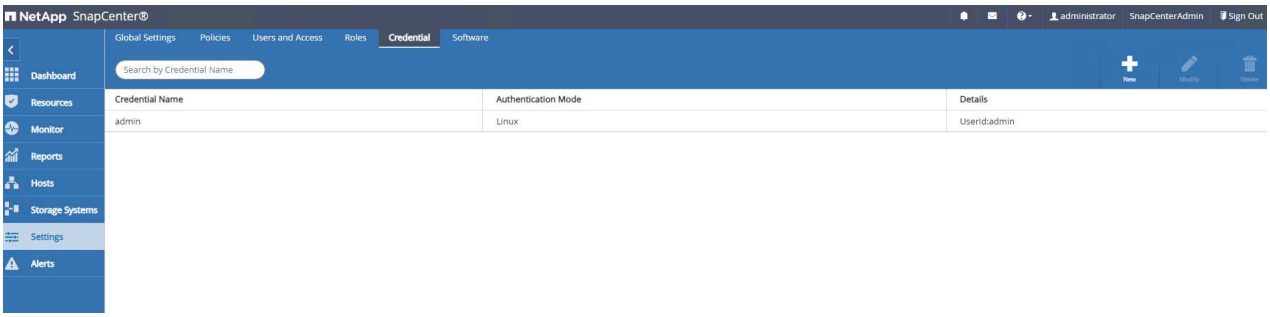
3. 以安装用户身份通过HTTPS端口8846登录到SnapCenter UI、以配置适用于Oracle的SnapCenter。
4. 更新 `Hypervisor Settings` 在全局设置中。



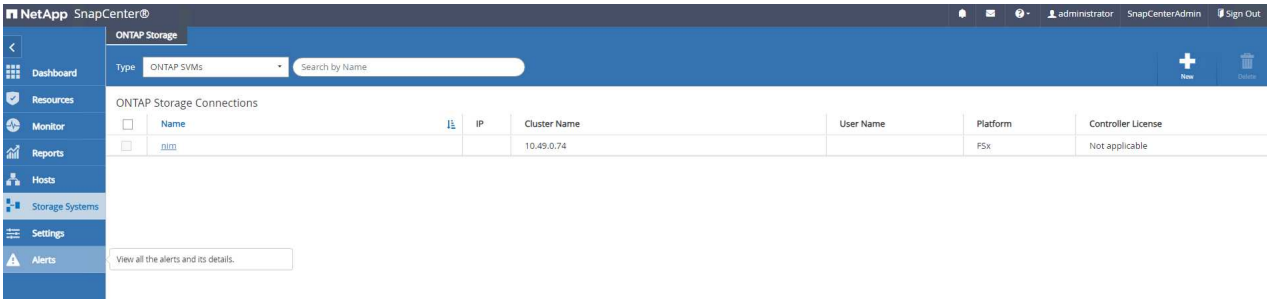
5. 创建Oracle数据库备份策略。理想情况下、请创建一个单独的归档日志备份策略、以便更频繁地进行备份、从而最大限度地减少发生故障时的数据丢失。



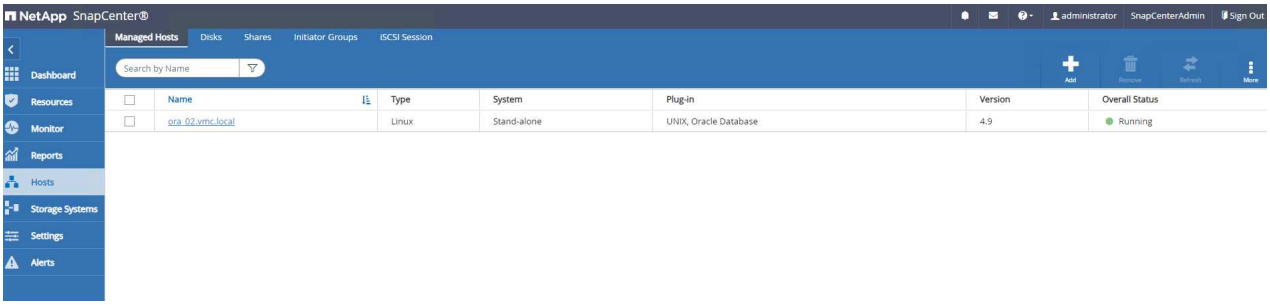
6. 添加数据库服务器 `Credential` 用于通过SnapCenter访问DB VM。此凭据在Linux VM上应具有sudo权限、在Windows VM上应具有管理员权限。



7. 将FSx ONTAP存储集群添加到 `Storage Systems` 使用集群管理IP并通过fsxadmin用户ID进行身份验证。



8. 将VMC中的Oracle数据库VM添加到 `Hosts` 使用上一步中创建的服务器凭据6.

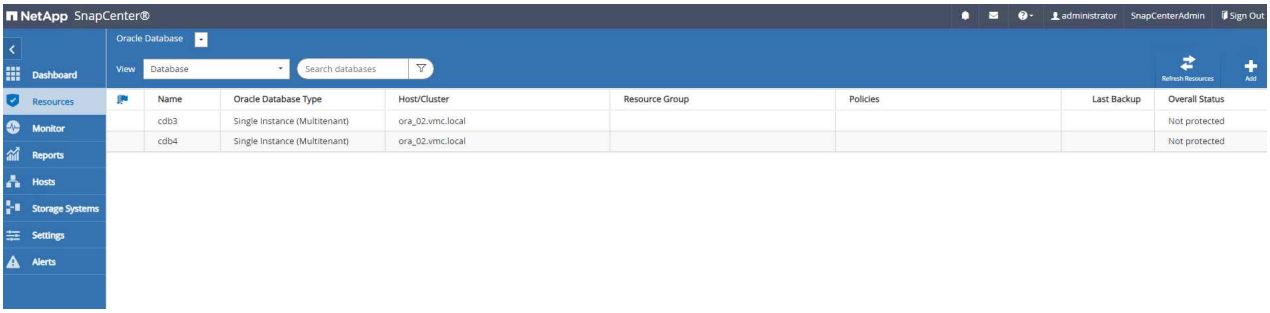


确保SnapCenter服务器名称可解析为数据库VM中的IP地址、而DB VM名称可解析为SnapCenter服务器中的IP地址。

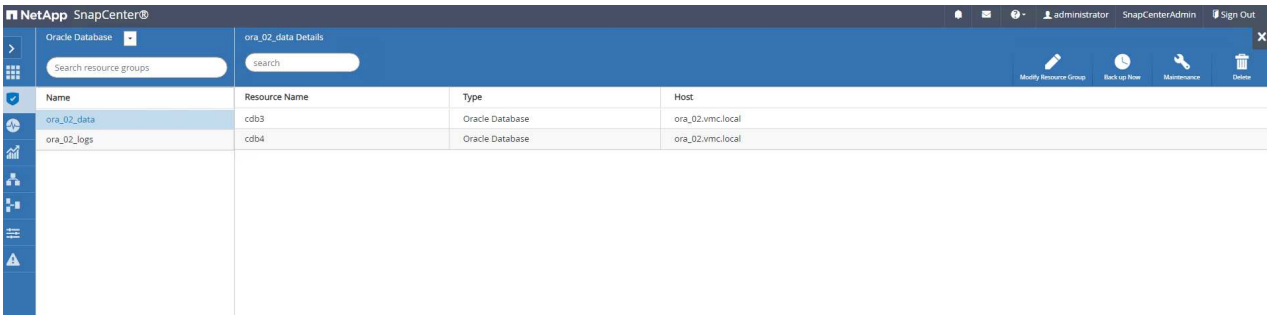
数据库备份

与基于RMAN的传统方法相比、SnapCenter利用FSx ONTAP卷快照加快数据库备份、还原或克隆速度。由于数据库在创建快照之前处于Oracle备份模式、因此这些快照是应用程序一致的。

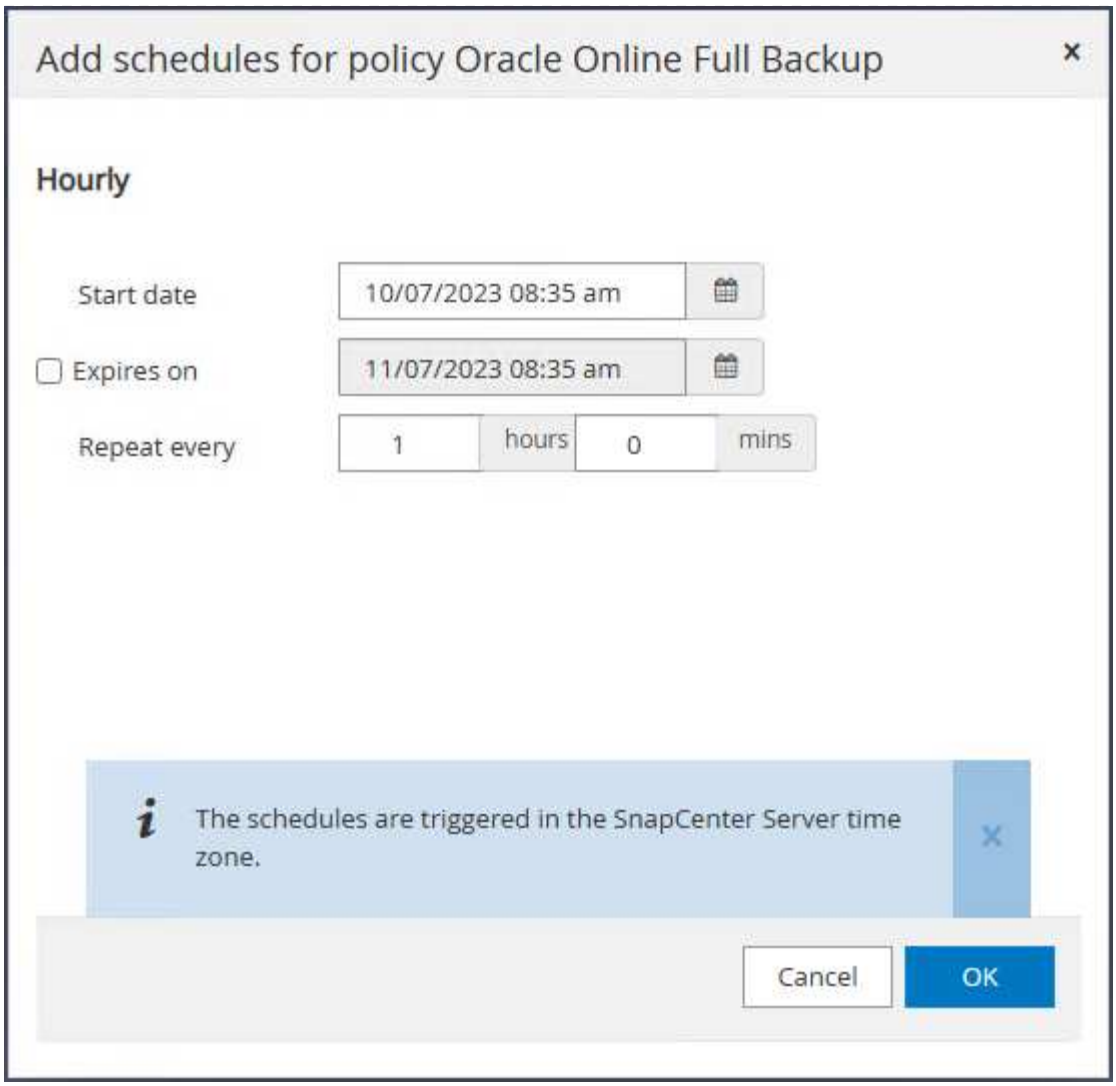
1. 从 `Resources` 选项卡中、将虚拟机添加到SnapCenter后、系统会自动发现虚拟机上的任何数据库。最初、数据库状态显示为 `Not protected`。



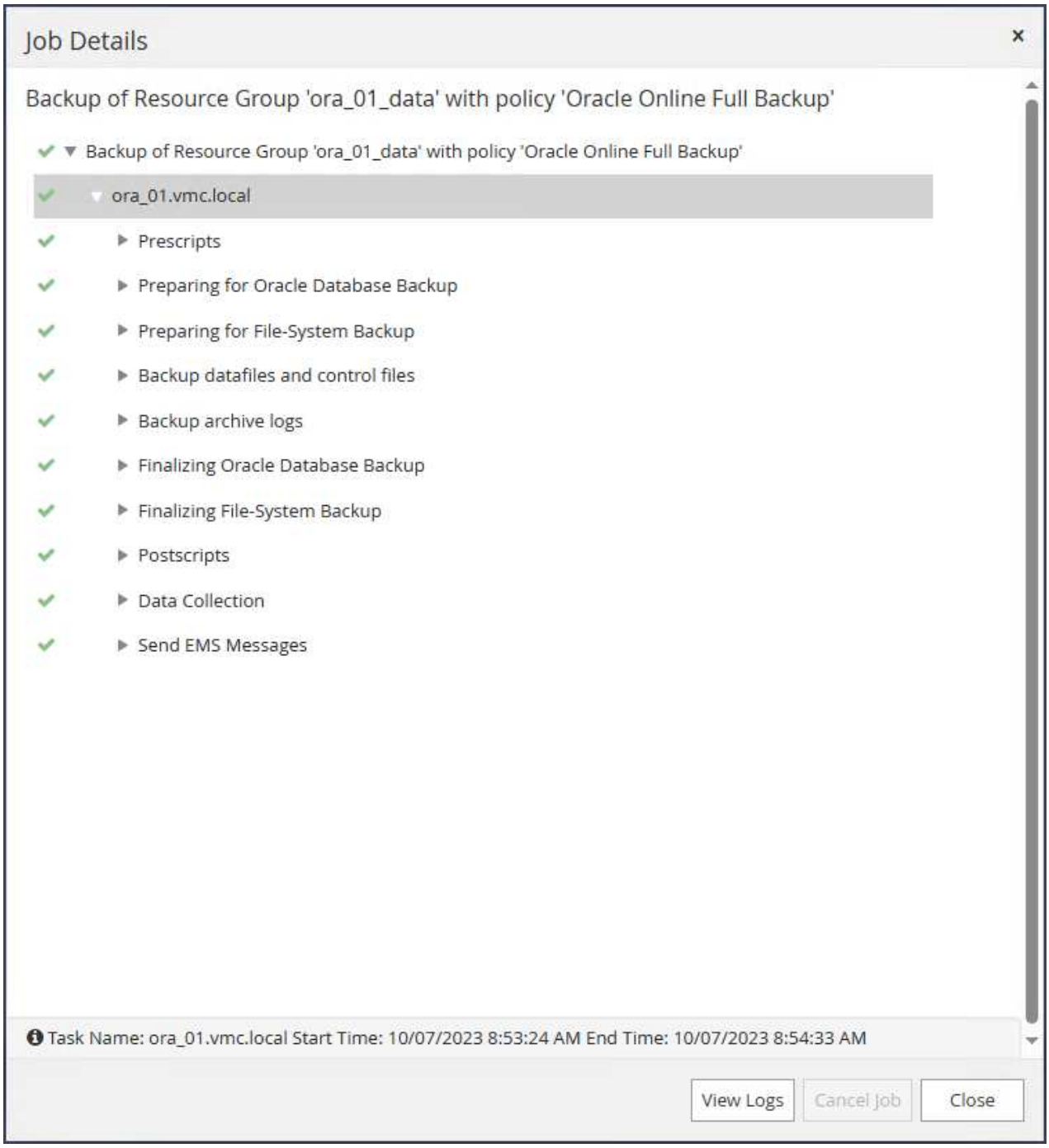
2. 创建一个资源组、以按逻辑分组(如DB VM等)备份数据库 在此示例中、我们创建了一个ora_02_data组、用于对VM ora_02上的所有数据库执行完整的联机数据库备份。资源组ora_02_log仅在VM上执行归档日志备份。创建资源组还会定义执行备份的计划。



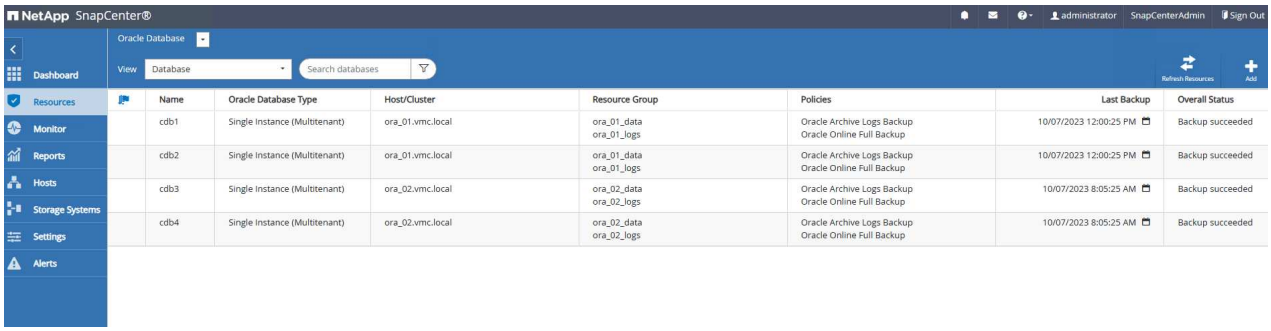
3. 单击也可以手动触发资源组备份 `Back up Now` 并使用资源组中定义的策略执行备份。



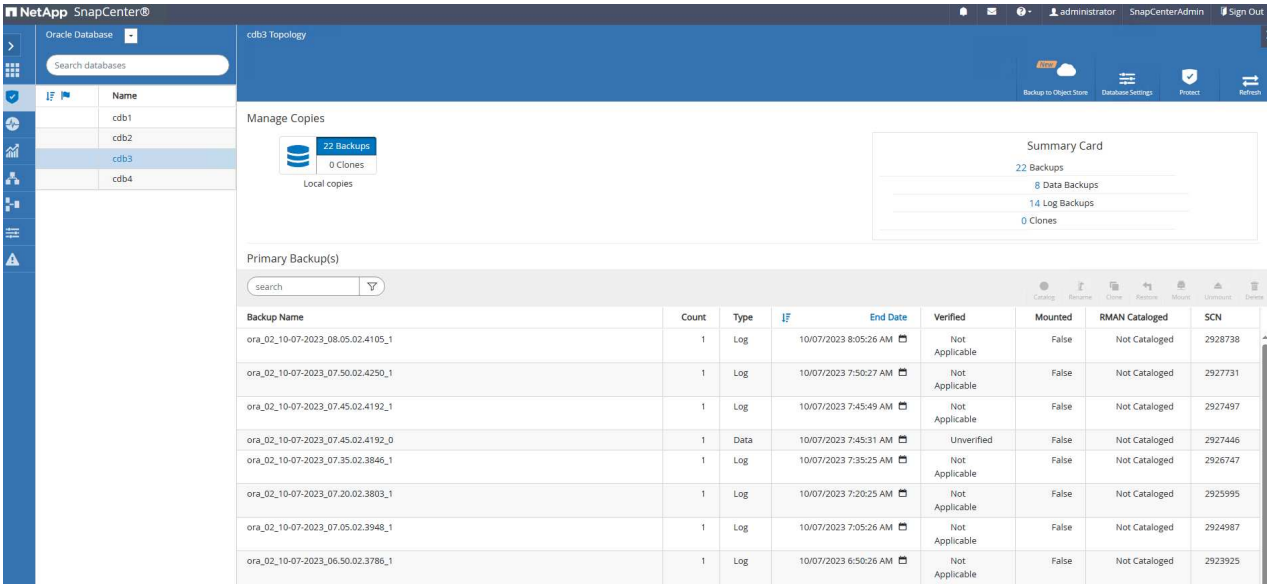
4. 可通过监控备份作业 `Monitor` 选项卡、单击正在运行的作业。



5. 成功备份后、数据库状态将显示作业状态和最近的备份时间。



6. 单击数据库以查看每个数据库的备份集。



数据库恢复

SnapCenter为Oracle数据库提供了许多从快照备份还原和恢复选项。在本示例中、我们展示了一个时间点还原、用于恢复因错误而丢弃的表。在VM ora_02上、两个数据库cdb3和cdb4共享相同的+data和+logs.磁盘组。一个数据库的数据库还原不会影响另一个数据库的可用性。

1. 首先、创建一个测试表并在表中插入一行、以验证时间点恢复。

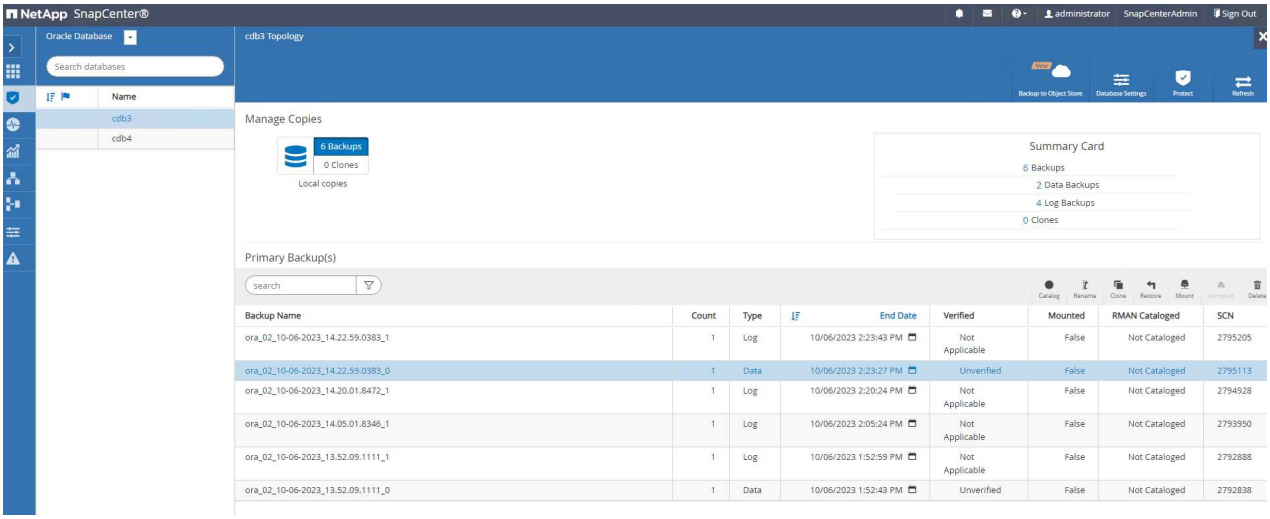
```
[oracle@ora_02 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Fri Oct 6 14:15:21 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
--------- --------------------
CDB3      READ WRITE

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 CDB3_PDB1                      READ WRITE NO
         4 CDB3_PDB2                      READ WRITE NO
         5 CDB3_PDB3                      READ WRITE NO
SQL>


SQL> alter session set container=cdb3_pdb1;

Session altered.

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

Table created.

SQL> insert into test values(1, sysdate, 'test oracle recovery on
```

```
guest mounted fsx storage to VMC guest vm ora_02');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from test;

        ID
----------
DT
---------------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------------
------------
         1
06-OCT-23 03.18.24.000000 PM
test oracle recovery on guest mounted fsx storage to VMC guest vm
ora_02


SQL> select current_timestamp from dual;

CURRENT_TIMESTAMP
---------------------------------------------------------------------------
-------
06-OCT-23 03.18.53.996678 PM -07:00
```

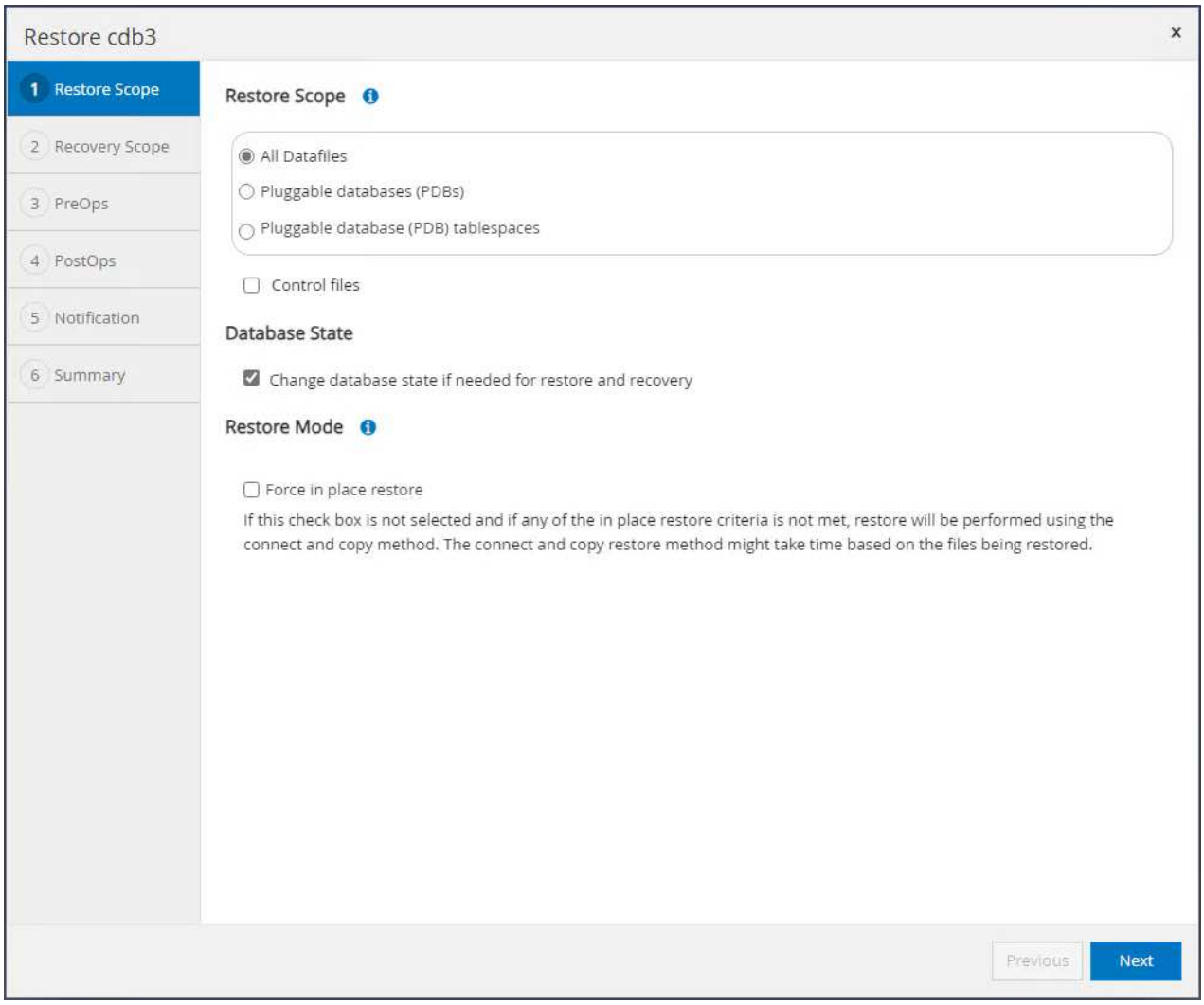
2. 我们从SnapCenter运行手动快照备份。然后丢弃该表。

```
SQL> drop table test;

Table dropped.

SQL> commit;

Commit complete.

SQL> select current_timestamp from dual;

CURRENT_TIMESTAMP
---------------------------------------------------------------------------
-------
06-OCT-23 03.26.30.169456 PM -07:00

SQL> select * from test;
select * from test
              *
ERROR at line 1:
ORA-00942: table or view does not exist
```

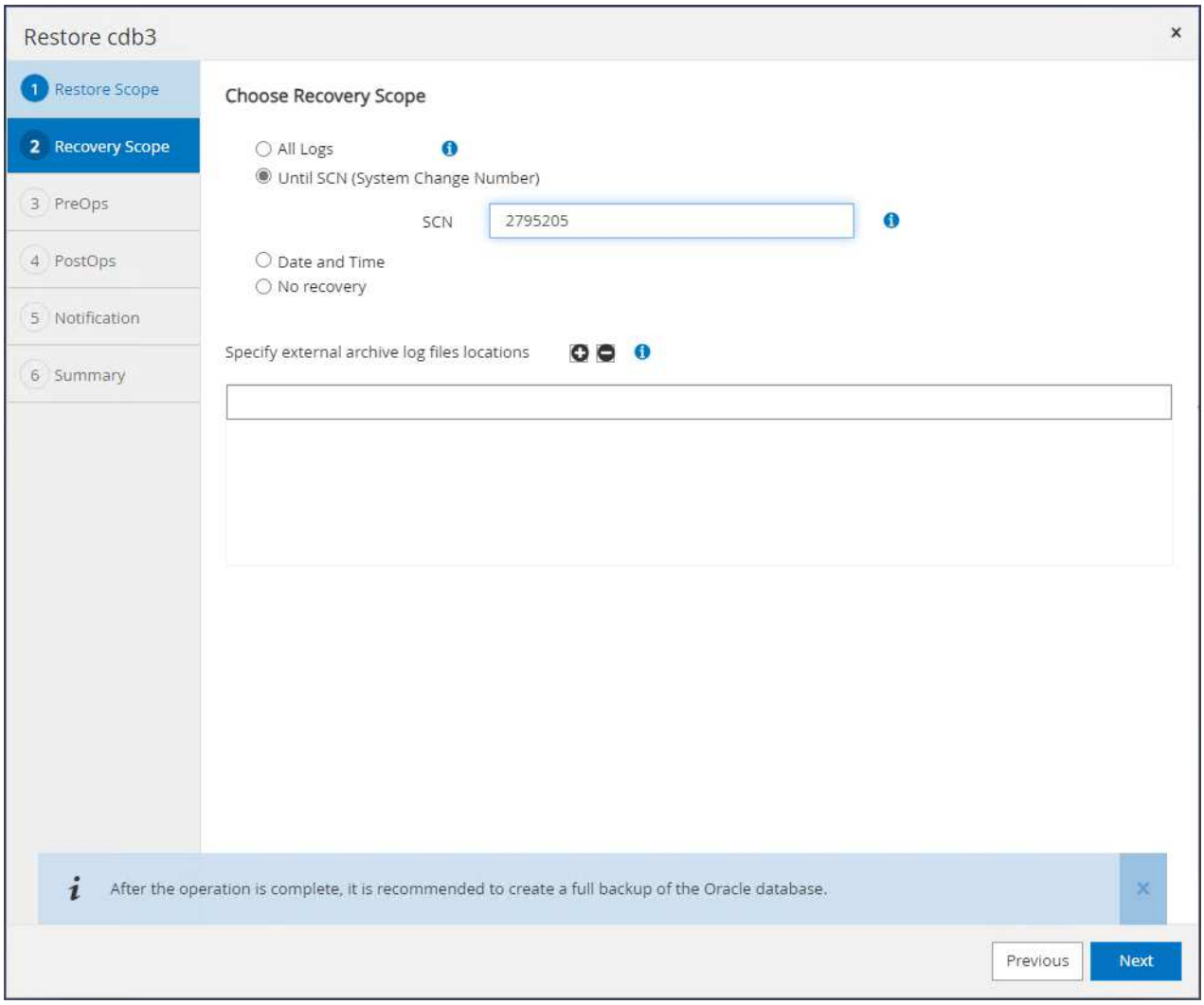
3. 从上一步创建的备份集中、记下日志备份的scn编号。单击 `Restore` 启动还原-恢复工作流。



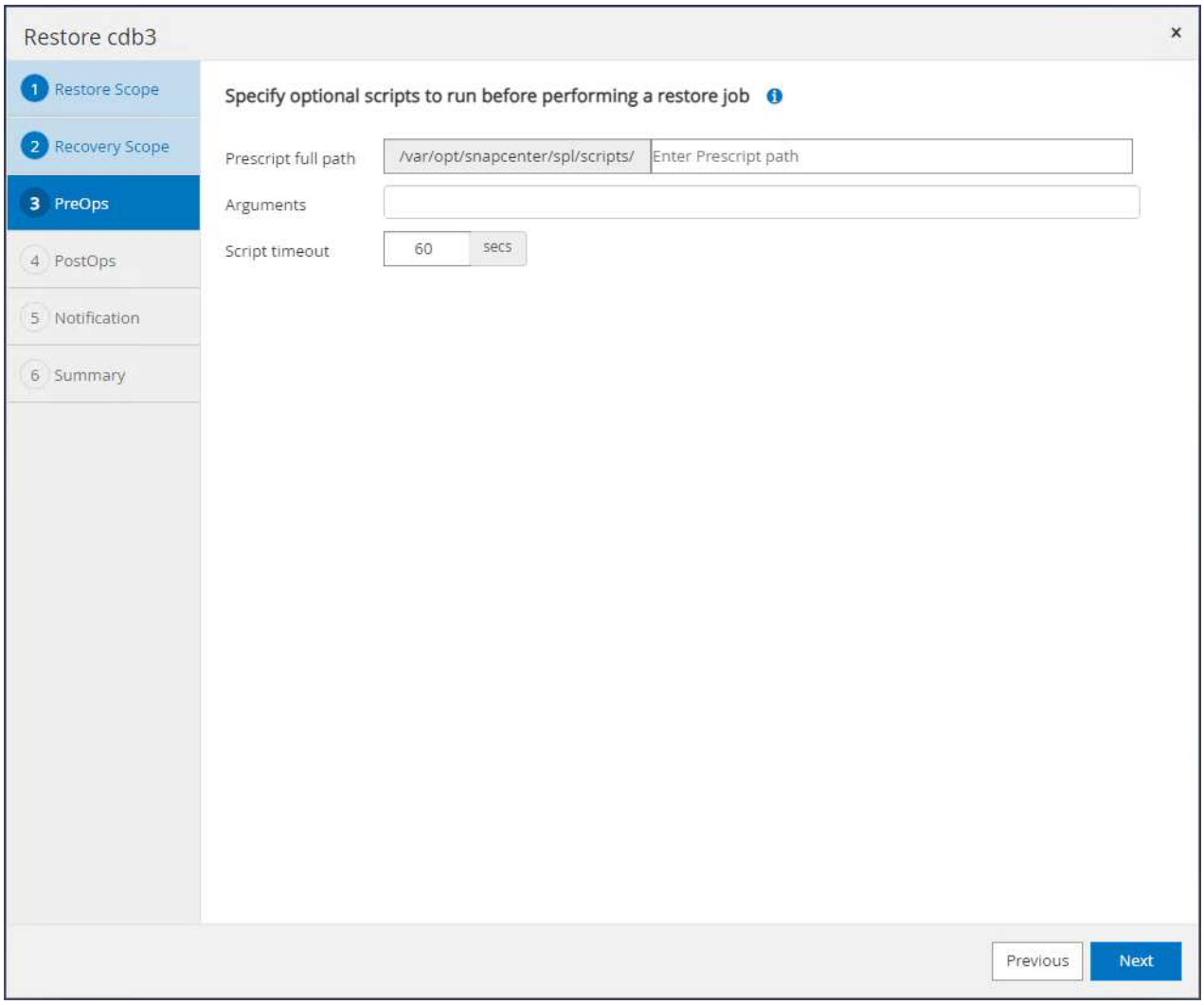
4. 选择还原范围。



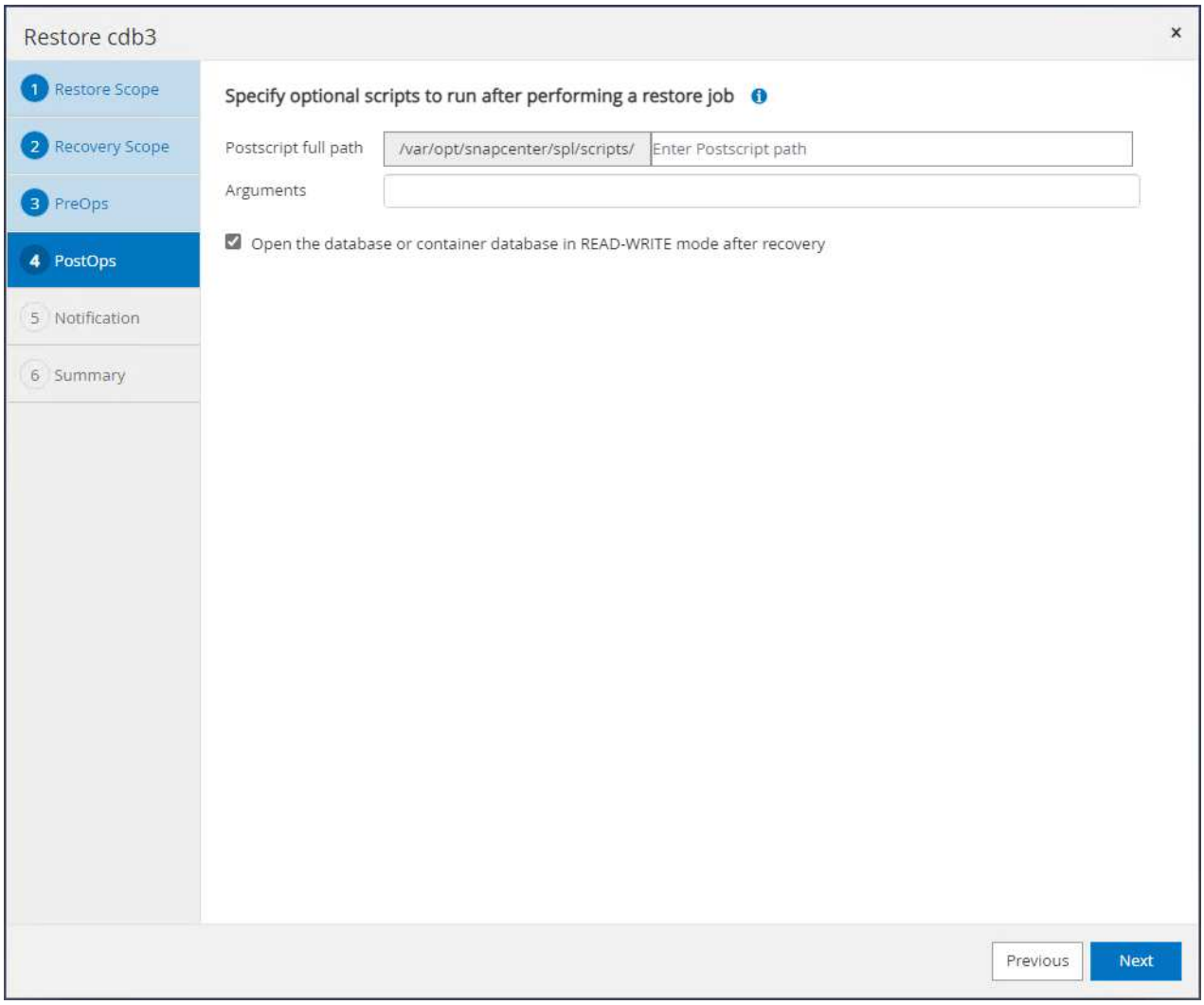
5. 选择从上次完整数据库备份到日志scn的恢复范围。



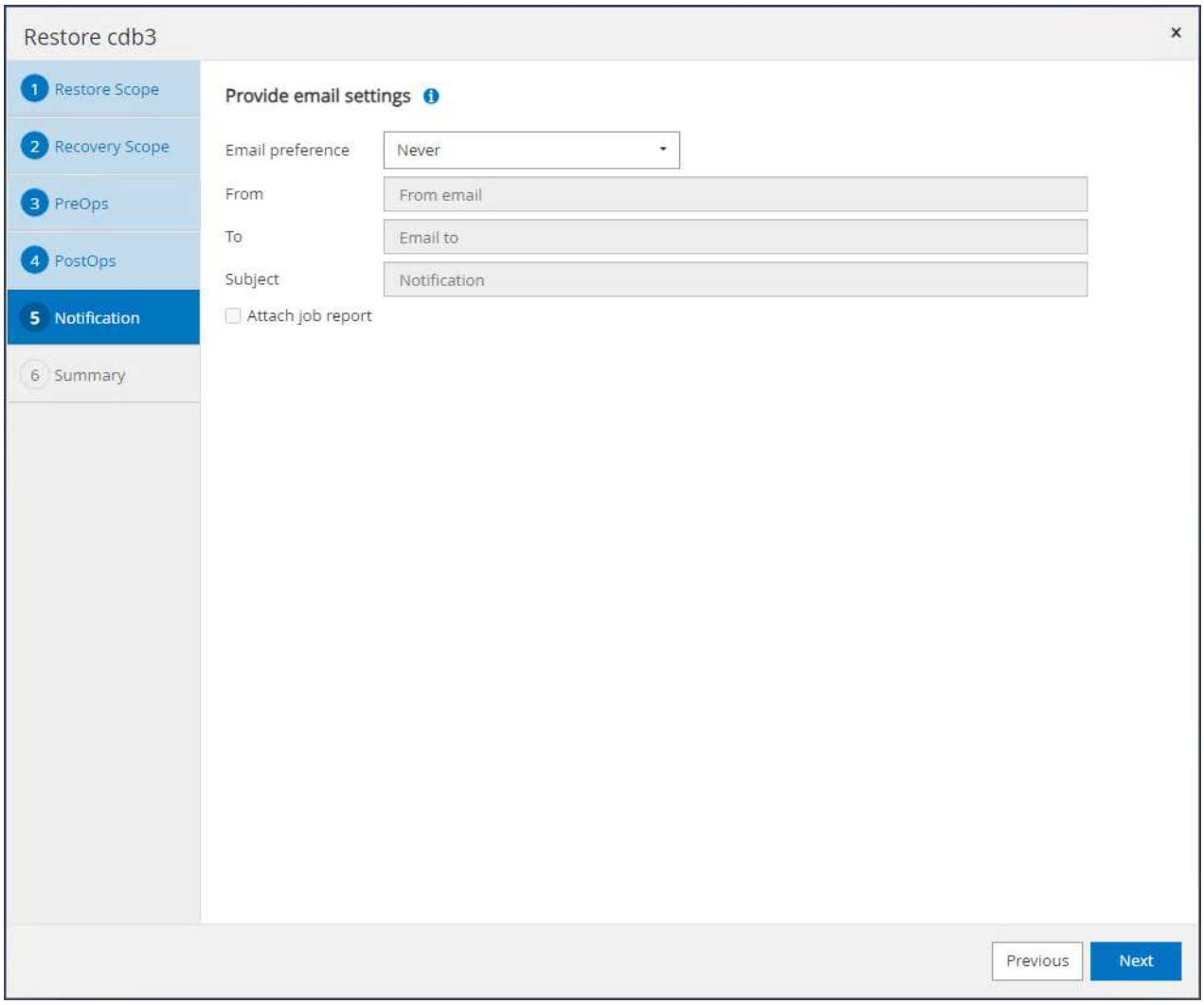
6. 指定要运行的任何可选预处理脚本。



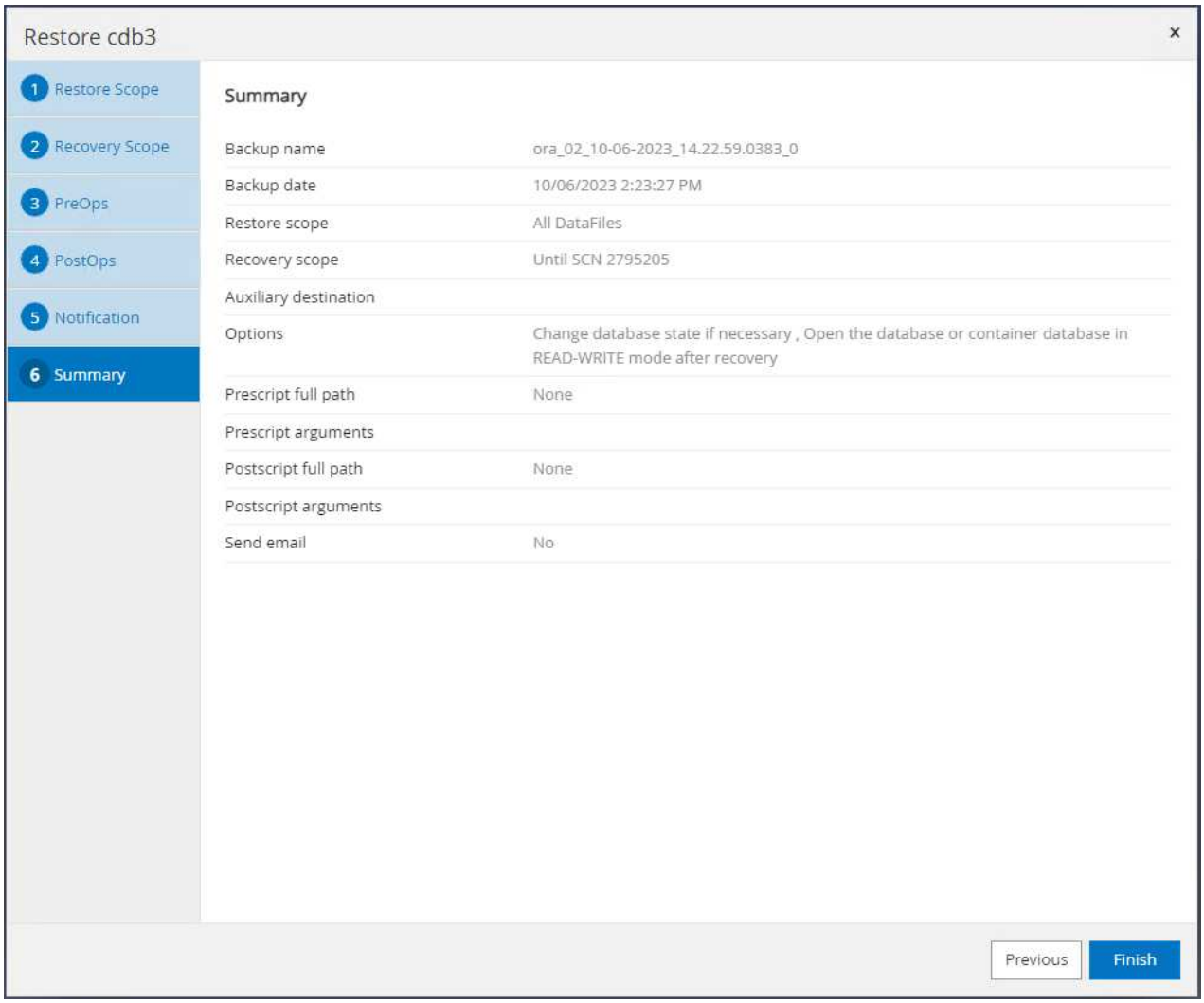
7. 指定要运行的任何可选后处理脚本。



8. 根据需要发送作业报告。



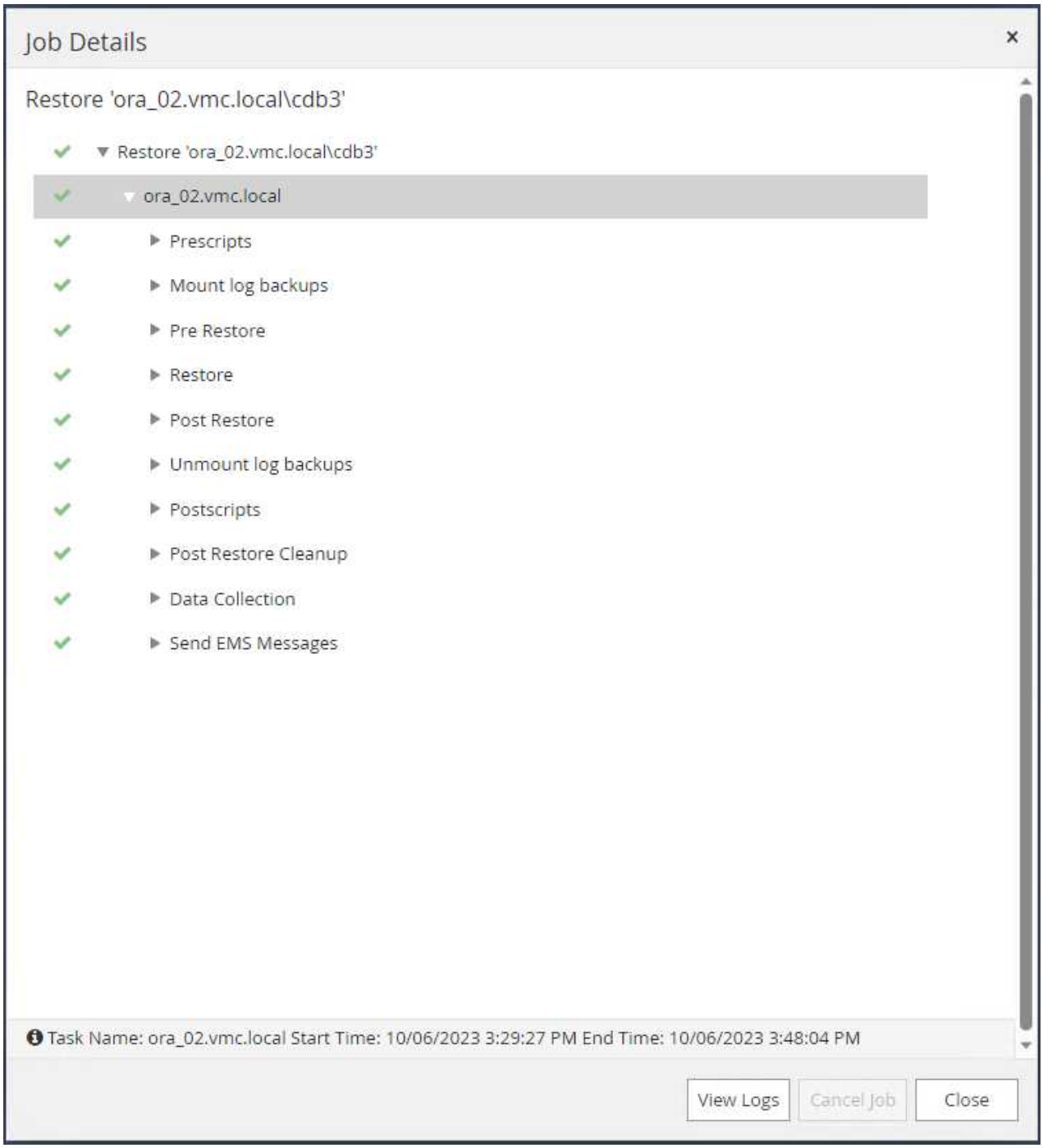
9. 查看摘要、然后单击 `Finish` 启动还原和恢复。



10. 通过Oracle Restart网格控件、我们可以观察到、在恢复cdb3时、cdb4处于联机和可用状态。

```
[oracle@ora_02 bin]$ ./crsctl stat res -t
--------------------------------------------------------------------------------
Name           Target  State        Server                   State details
--------------------------------------------------------------------------------
Local Resources
--------------------------------------------------------------------------------
ora.DATA.dg
               ONLINE  ONLINE       ora_02                   STABLE
ora.LISTENER.lsnr
               ONLINE  INTERMEDIATE ora_02                   Not All Endpoints Re
                                                             gistered,STABLE
ora.LOGS.dg
               ONLINE  ONLINE       ora_02                   STABLE
ora.LOGS_CDB3_22.dg
               ONLINE  ONLINE       ora_02                   STABLE
ora.asm
               ONLINE  ONLINE       ora_02                   Started,STABLE
ora.ons
               OFFLINE OFFLINE      ora_02                   STABLE
--------------------------------------------------------------------------------
Cluster Resources
--------------------------------------------------------------------------------
ora.cdb3.db
      1        ONLINE  INTERMEDIATE ora_02                   Dismounted,Mount Ini
                                                             tiated,HOME=/u01/app
                                                             /oracle/product/19.0
                                                             .0/cdb3,STABLE
ora.cdb4.db
      1        ONLINE  ONLINE       ora_02                   Open,HOME=/u01/app/o
                                                             racle/product/19.0.0
                                                             /cdb4,STABLE
ora.cssd
      1        ONLINE  ONLINE       ora_02                   STABLE
ora.diskmon
      1        OFFLINE OFFLINE                               STABLE
ora.driver.afd
      1        ONLINE  ONLINE       ora_02                   STABLE
ora.evmd
      1        ONLINE  ONLINE       ora_02                   STABLE
--------------------------------------------------------------------------------
[oracle@ora_02 bin]$
```

11. from `Monitor` 选项卡中、打开作业以查看详细信息。



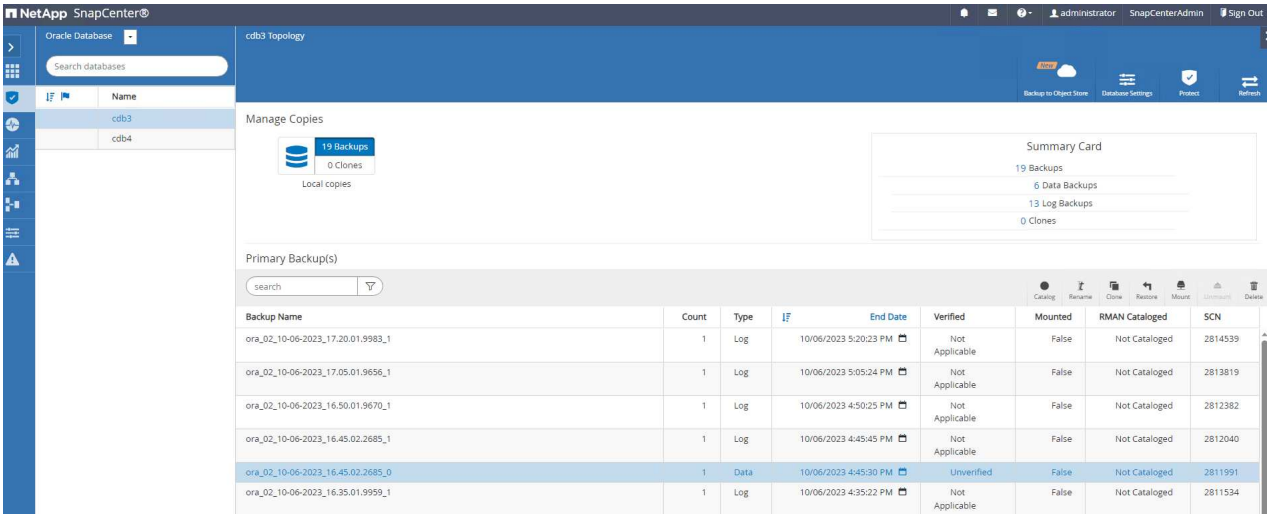
12. 从DB VM ora_02中、验证在成功恢复后是否已恢复丢弃的表。

```
[oracle@ora_02 bin]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Fri Oct 6 17:01:28 2023
Version 19.18.0.0.0

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


Connected to:
```

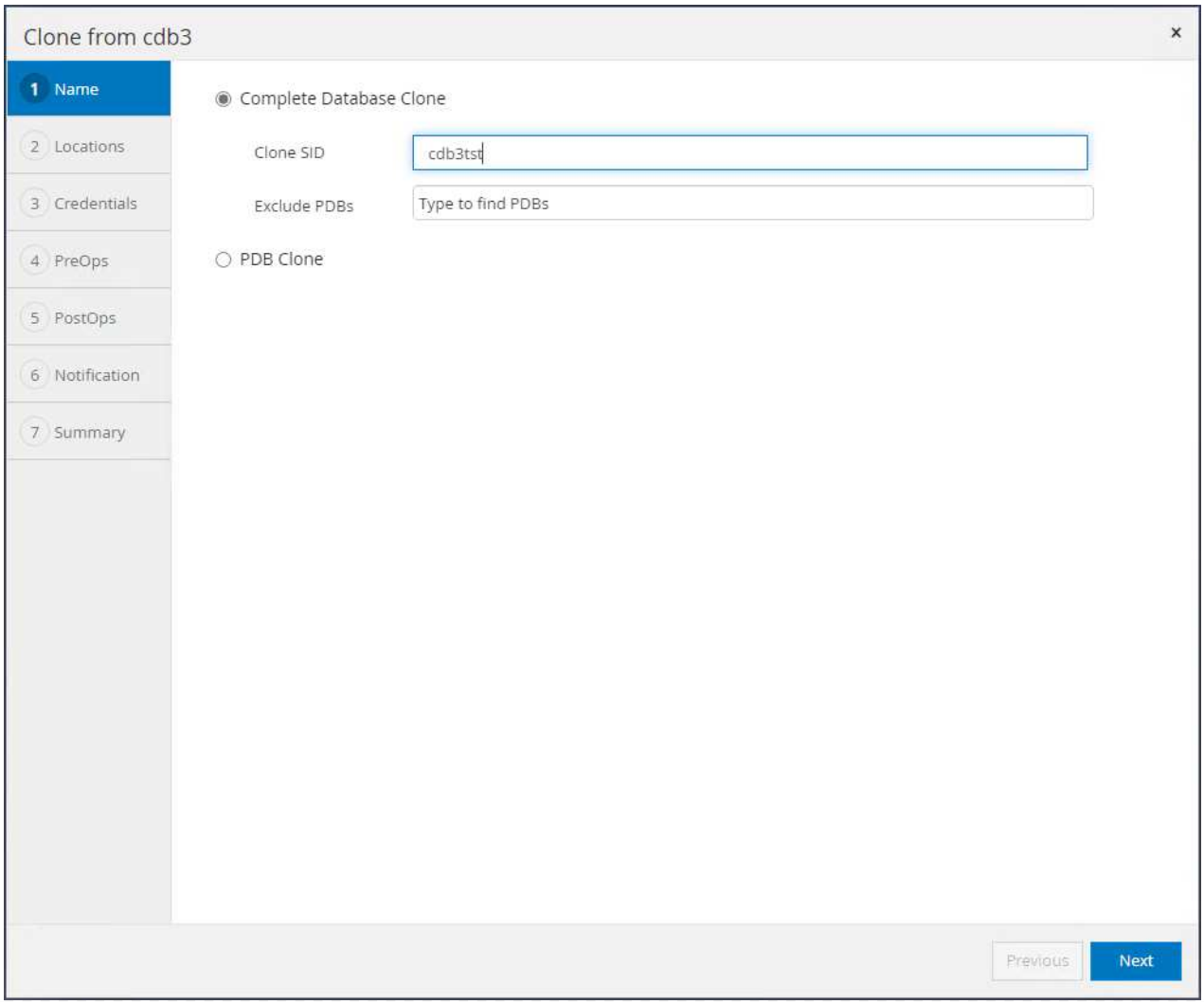
```
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 -
Production
Version 19.18.0.0.0

SQL> select name, open_mode from v$database;

NAME      OPEN_MODE
--------- --------------------
CDB3      READ WRITE

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 CDB3_PDB1                      READ WRITE NO
         4 CDB3_PDB2                      READ WRITE NO
         5 CDB3_PDB3                      READ WRITE NO
SQL> alter session set container=CDB3_PDB1;

Session altered.

SQL> select * from test;

        ID
----------
DT
---------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------
------------
         1
06-OCT-23 03.18.24.000000 PM
test oracle recovery on guest mounted fsx storage to VMC guest vm
ora_02


SQL> select current_timestamp from dual;

CURRENT_TIMESTAMP
---------------------------------------------------------------------
-------
06-OCT-23 05.02.20.382702 PM -07:00

SQL>
```

在此示例中、使用相同的备份集克隆不同oracle_home中同一VM上的数据库。如果需要、这些过程同样适用于将数据库从备份克隆到VMC中的单独虚拟机。

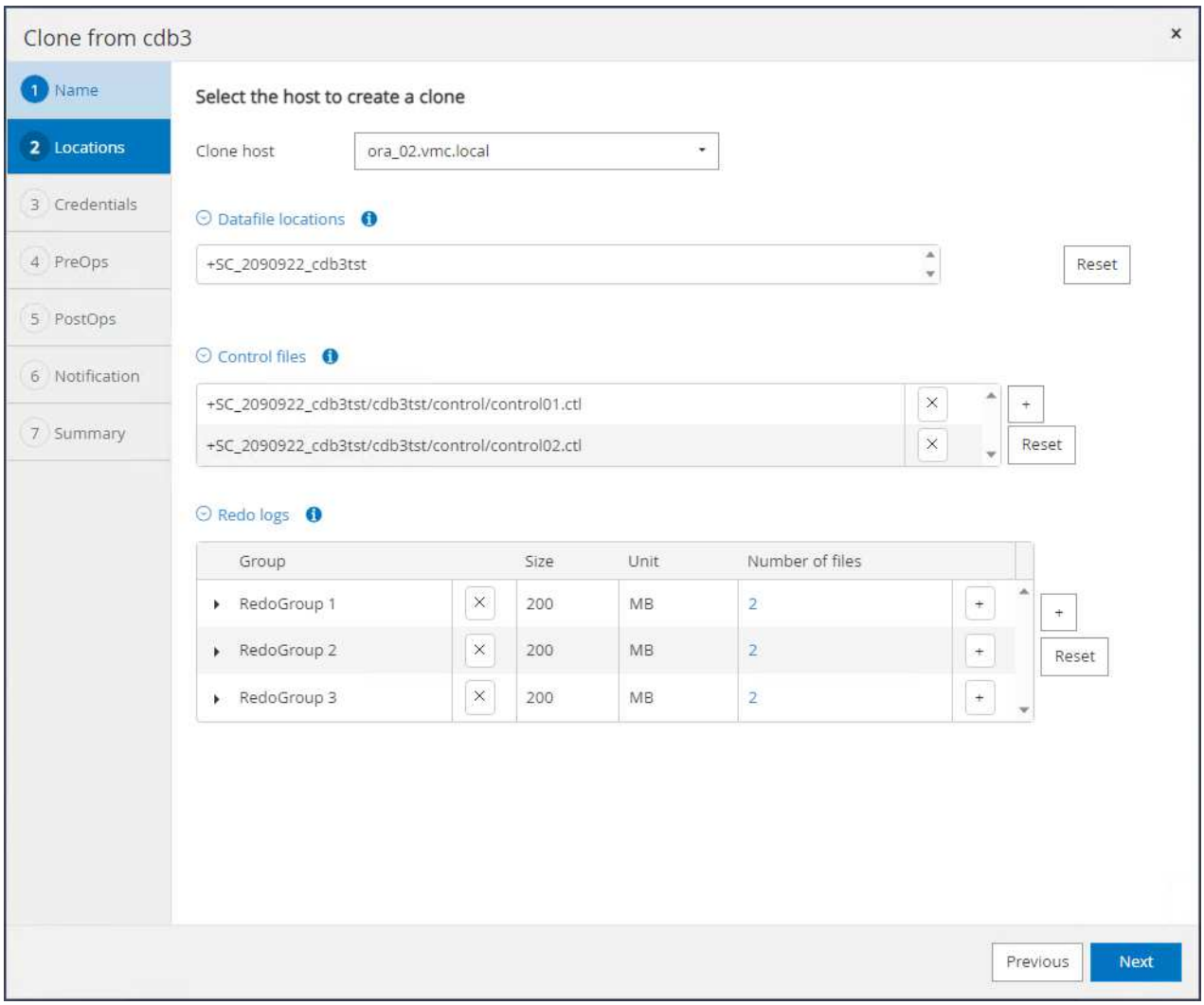
1. 打开数据库cdb3备份列表。从所选的数据备份中、单击 `Clone` 用于启动数据库克隆工作流的按钮。



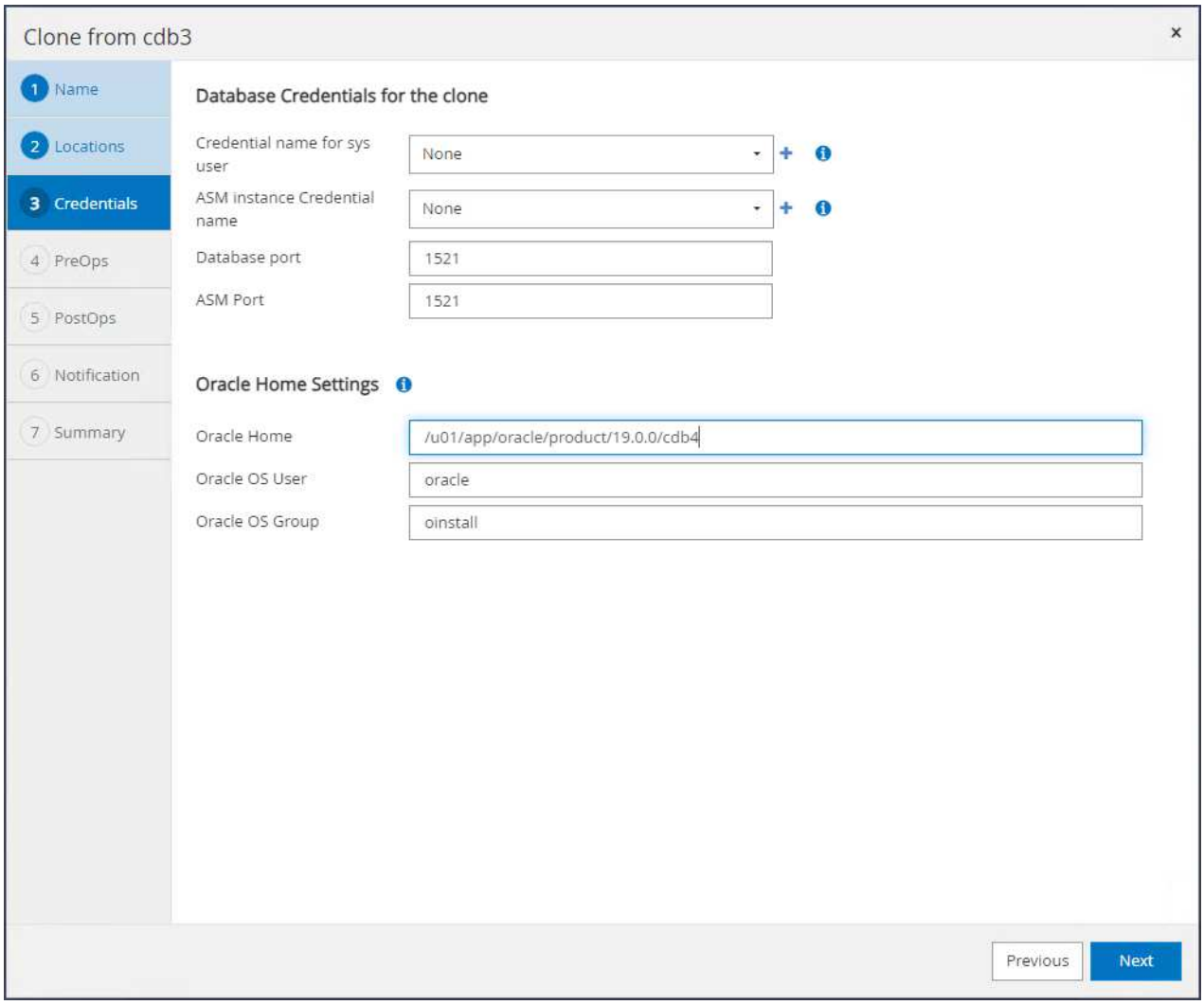
2. 将克隆数据库命名为SID。



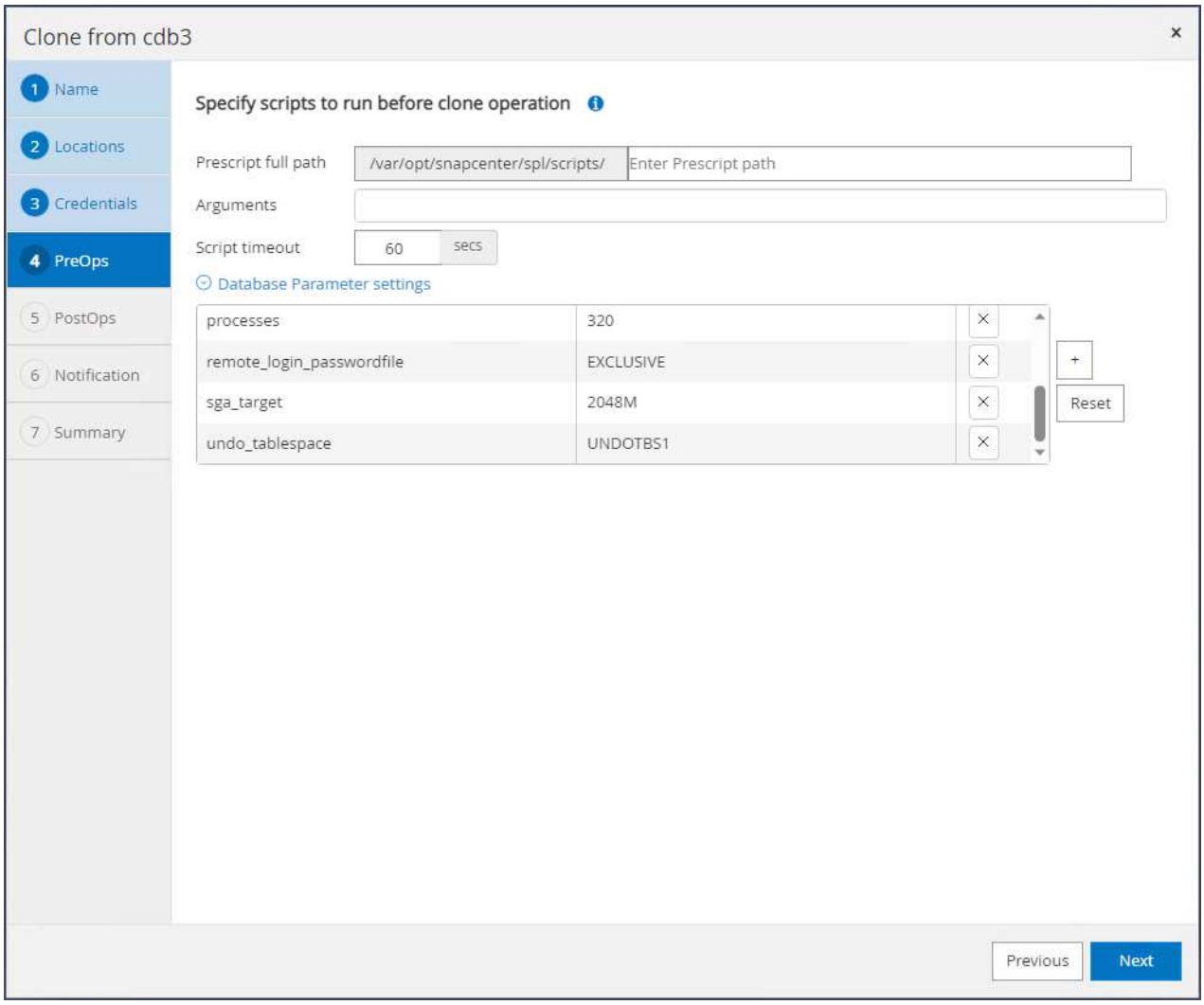
3. 在VMC中选择一个VM作为目标数据库主机。主机上应已安装和配置相同版本的Oracle。



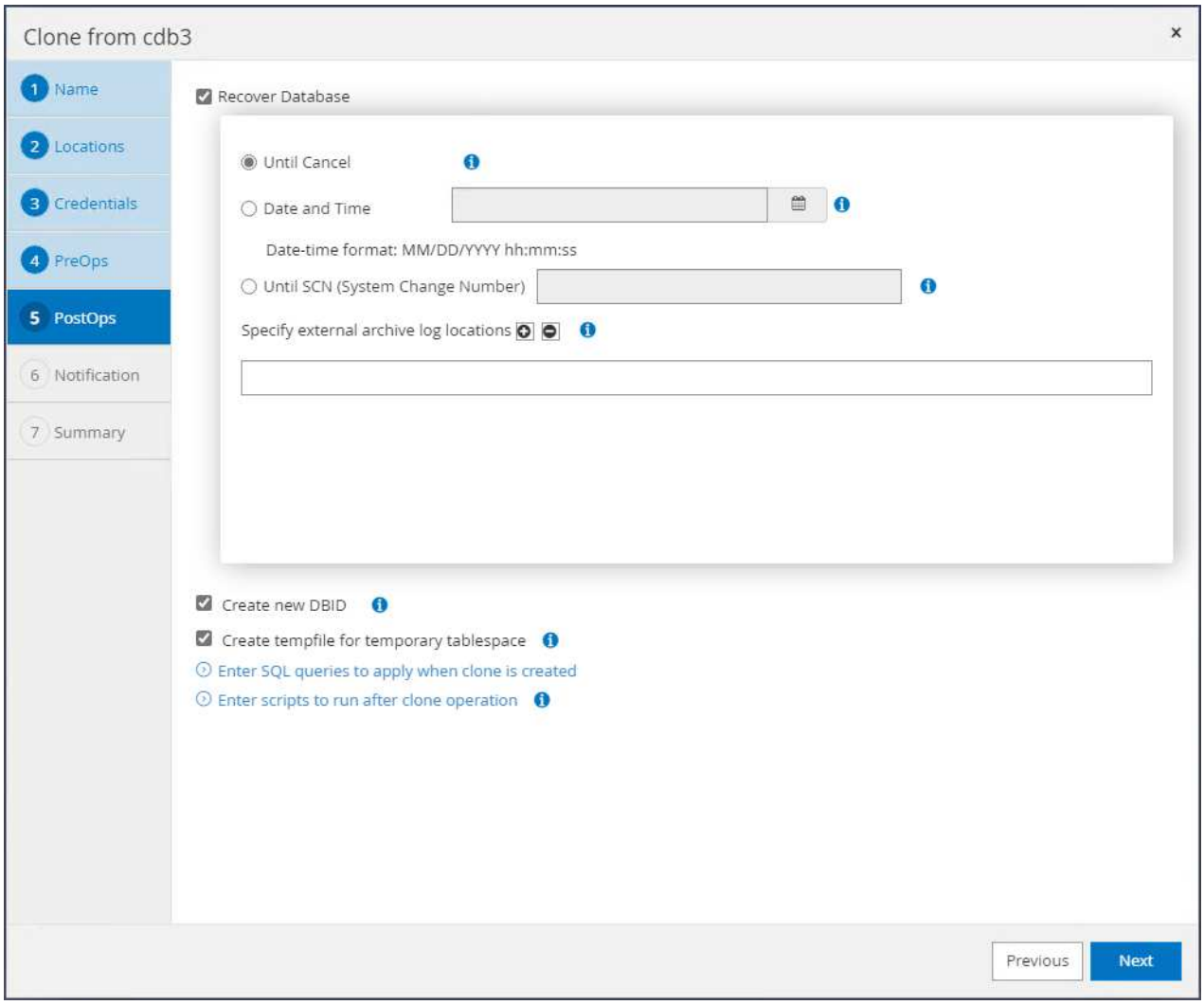
4. 在目标主机上选择正确的oracle_home、用户和组。保留默认凭据。



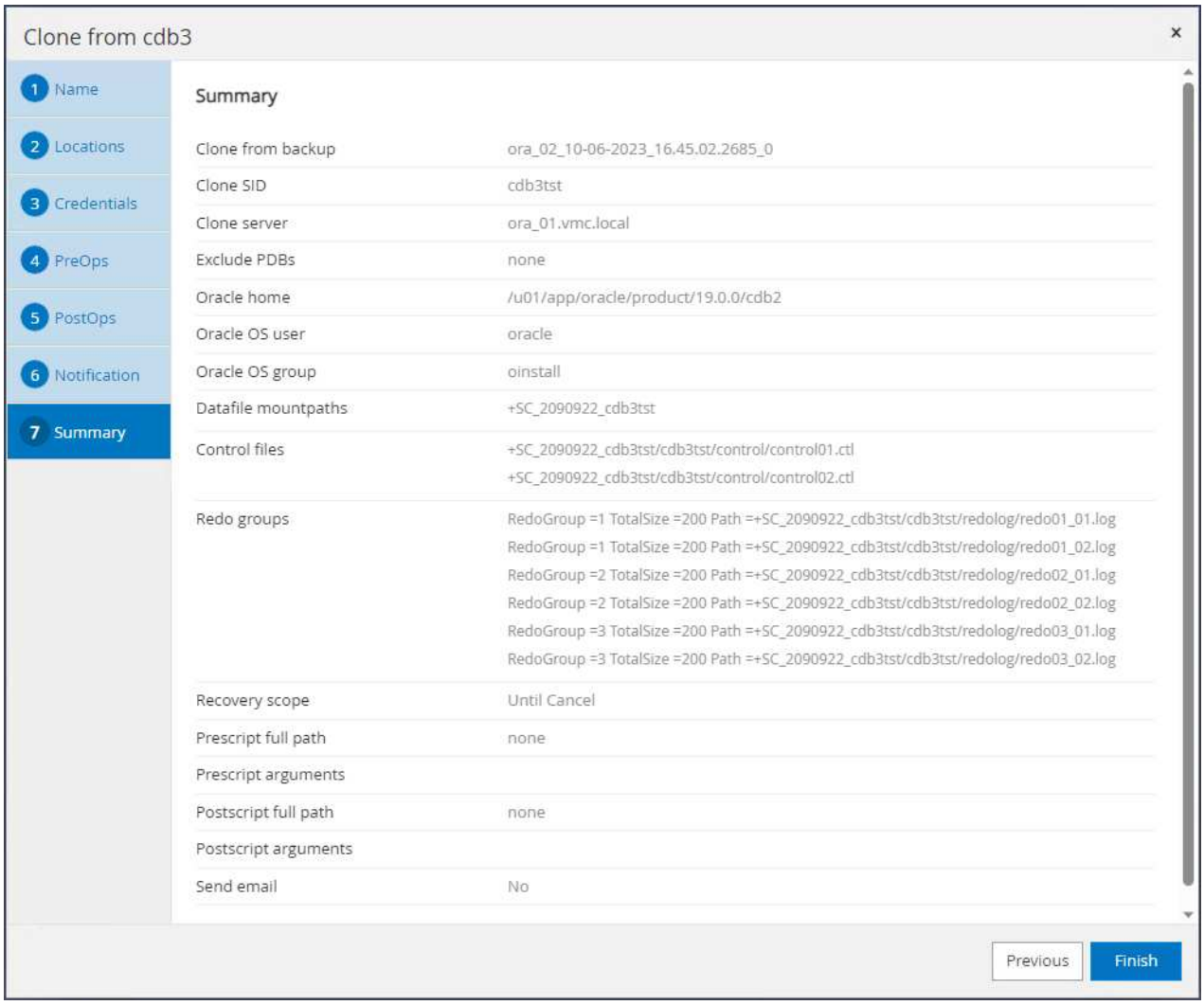
5. 更改克隆数据库参数以满足克隆数据库的配置或资源要求。



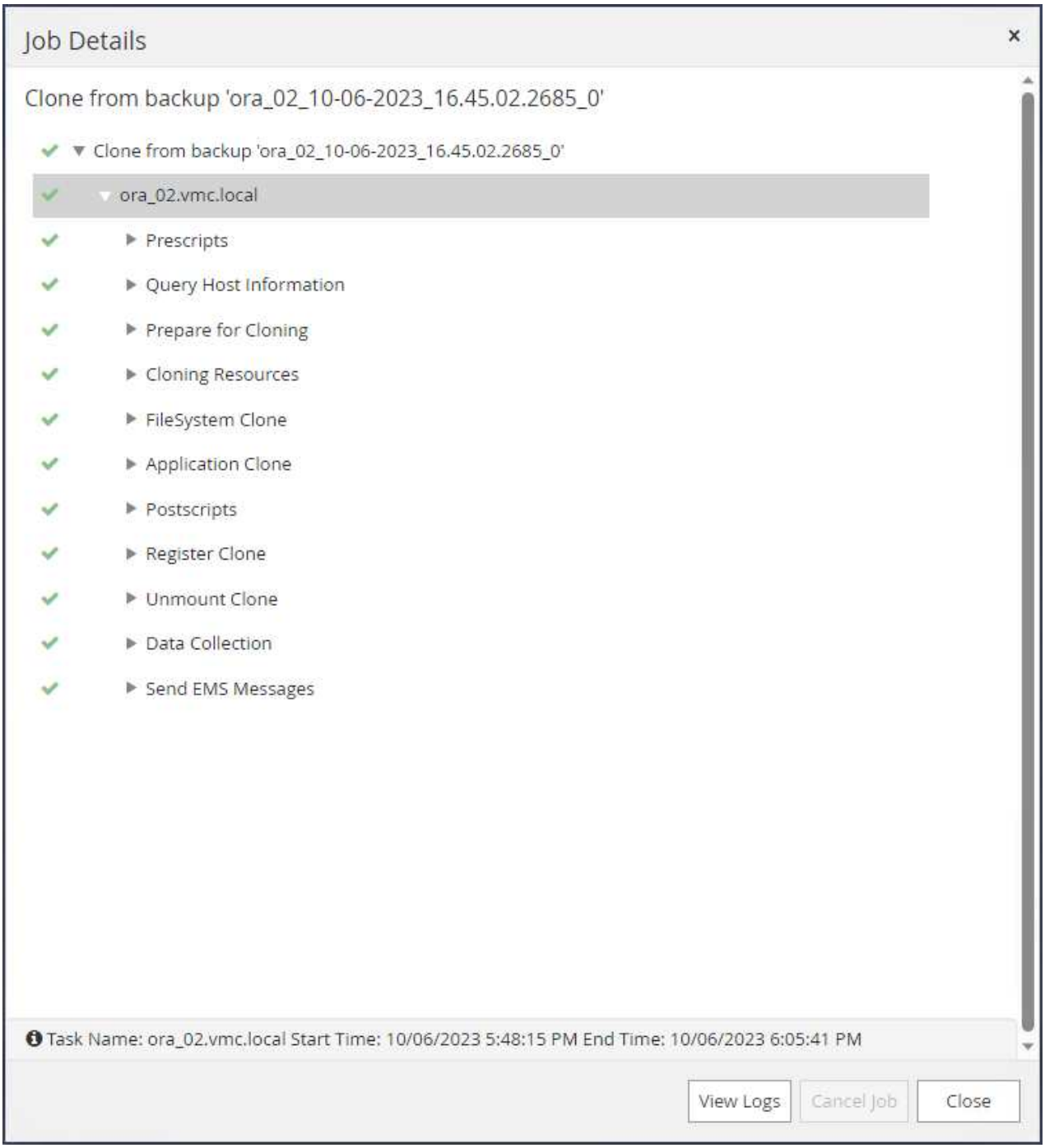
6. 选择恢复范围。 `Until Cancel` 将克隆恢复到备份集中最后一个可用日志文件。



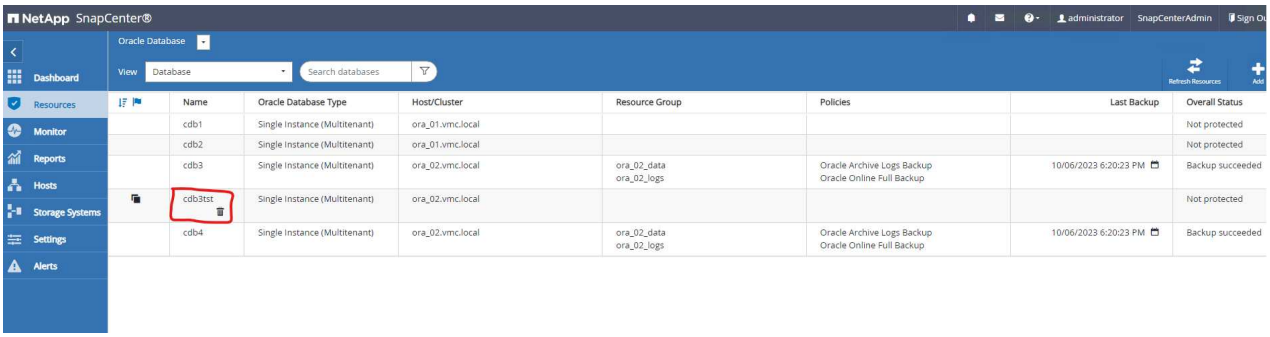
7. 查看摘要并启动克隆作业。



8. 通过监控克隆作业的执行情况 `Monitor` 选项卡。



9. 克隆的数据库会立即注册到SnapCenter中。



10. 从DB VM ora_02中、克隆的数据库也会注册到Oracle Restart网格控件中、而丢弃的测试表会恢复到克隆的数据库cdb3tst中、如下所示。

```
[oracle@ora_02 ~]$ /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       ora_02                   STABLE
ora.LISTENER.lsnr
               ONLINE  INTERMEDIATE ora_02                   Not All
Endpoints Re

gistered,STABLE
ora.LOGS.dg
               ONLINE  ONLINE       ora_02                   STABLE
ora.SC_2090922_CDB3TST.dg
               ONLINE  ONLINE       ora_02                   STABLE
ora.asm
               ONLINE  ONLINE       ora_02
Started,STABLE
ora.ons
               OFFLINE OFFLINE      ora_02                   STABLE
--------------------------------------------------------------------------
------------
Cluster Resources
--------------------------------------------------------------------------
------------
ora.cdb3.db
      1        ONLINE  ONLINE       ora_02
Open,HOME=/u01/app/o

racle/product/19.0.0

/cdb3,STABLE
ora.cdb3tst.db
      1        ONLINE  ONLINE       ora_02
Open,HOME=/u01/app/o
```

```
racle/product/19.0.0

/cdb4,STABLE
ora.cdb4.db
      1        ONLINE  ONLINE       ora_02
Open,HOME=/u01/app/o

racle/product/19.0.0

/cdb4,STABLE
ora.cssd
      1        ONLINE  ONLINE       ora_02                   STABLE
ora.diskmon
      1        OFFLINE OFFLINE                               STABLE
ora.driver.afd
      1        ONLINE  ONLINE       ora_02                   STABLE
ora.evmd
      1        ONLINE  ONLINE       ora_02                   STABLE
--------------------------------------------------------------------
------------

[oracle@ora_02 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/cdb4
[oracle@ora_02 ~]$ export ORACLE_SID=cdb3tst
[oracle@ora_02 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Sat Oct 7 08:04:51 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
--------- --------------------
CDB3TST   READ WRITE

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
```

```
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 CDB3_PDB1                      READ WRITE NO
         4 CDB3_PDB2                      READ WRITE NO
         5 CDB3_PDB3                      READ WRITE NO
SQL> alter session set container=CDB3_PDB1;

Session altered.

SQL> select * from test;

        ID
----------
DT
---------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------
------------
         1
06-OCT-23 03.18.24.000000 PM
test oracle recovery on guest mounted fsx storage to VMC guest vm
ora_02


SQL>
```

至此、我们完成了在AWS上的VMC SDDC中对Oracle数据库进行SnapCenter备份、还原和克隆的演示。

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- VMware Cloud on AWS文档

  "https://docs.vmware.com/en/VMware-Cloud-on-AWS/index.html"

- 在安装新数据库的情况下为独立服务器安装Oracle网格基础架构

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-oracle-grid-infrastructure-for-a-standalone-server-with-a-new-database-installation.html#GUID-0B1CEE8C-C893-46AA-8A6A-7B5FAAEC72B3"

- 使用响应文件安装和配置Oracle数据库

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-and-configuring-oracle-database-using-response-files.html#GUID-D53355E9-E901-4224-9A2A-B882070EDDF7"

- 适用于 NetApp ONTAP 的 Amazon FSX

  "https://aws.amazon.com/fsx/netapp-ontap/"

TR-4981：《借助Amazon FSx ONTAP降低Oracle Active Data Guard成本》

NetApp公司Allen Cao、Niyaz Mohamed

目的

Oracle Data Guard可确保主数据库和备用数据库复制配置中企业数据的高可用性、数据保护和灾难恢复。Oracle Active Data Guard使用户能够在从主数据库到备用数据库的数据复制处于活动状态时访问备用数据库。Data Guard是Oracle数据库企业版的一项功能。它不需要单独的许可。另一方面、Active Data Guard是Oracle数据库企业版选件、因此需要单独的许可。在Active Data Guard设置中、多个备用数据库可以从主数据库接收数据复制。但是、每个附加备用数据库都需要Active Data Guard许可证以及与主数据库大小相同的额外存储。运营成本会迅速增加。

如果您希望降低Oracle数据库运营成本、并计划在AWS中设置Active Data Guard、则应考虑另一种选择。使用Data Guard将数据从主数据库复制到Amazon FSx ONTAP存储上的单个物理备用数据库、而不是Active Data Guard。随后、可以克隆此备用数据库的多个副本并打开以进行读/写访问、以满足许多其他使用情形的需要、例如报告、开发、测试等 最终结果有效地提供了Active Data Guard的功能、同时消除了Active Data Guard许可证、并为每个额外的备用数据库节省了额外的存储成本。在本文档中、我们将演示如何在AWS中使用现有主数据库设置Oracle Data Guard、并将物理备用数据库放置在Amazon FSx ONTAP存储上。备用数据库通过Snapshot进行备份、并根据需要进行克隆、以便进行读/写访问。

此解决方案 可解决以下使用情形：

- 在AWS中任何存储上的主数据库与Amazon FSx ONTAP存储上的备用数据库之间建立Oracle Data Guard。
- 在关闭以进行数据复制的情况下克隆备用数据库、以满足报告、开发、测试等使用情形的要求

audience

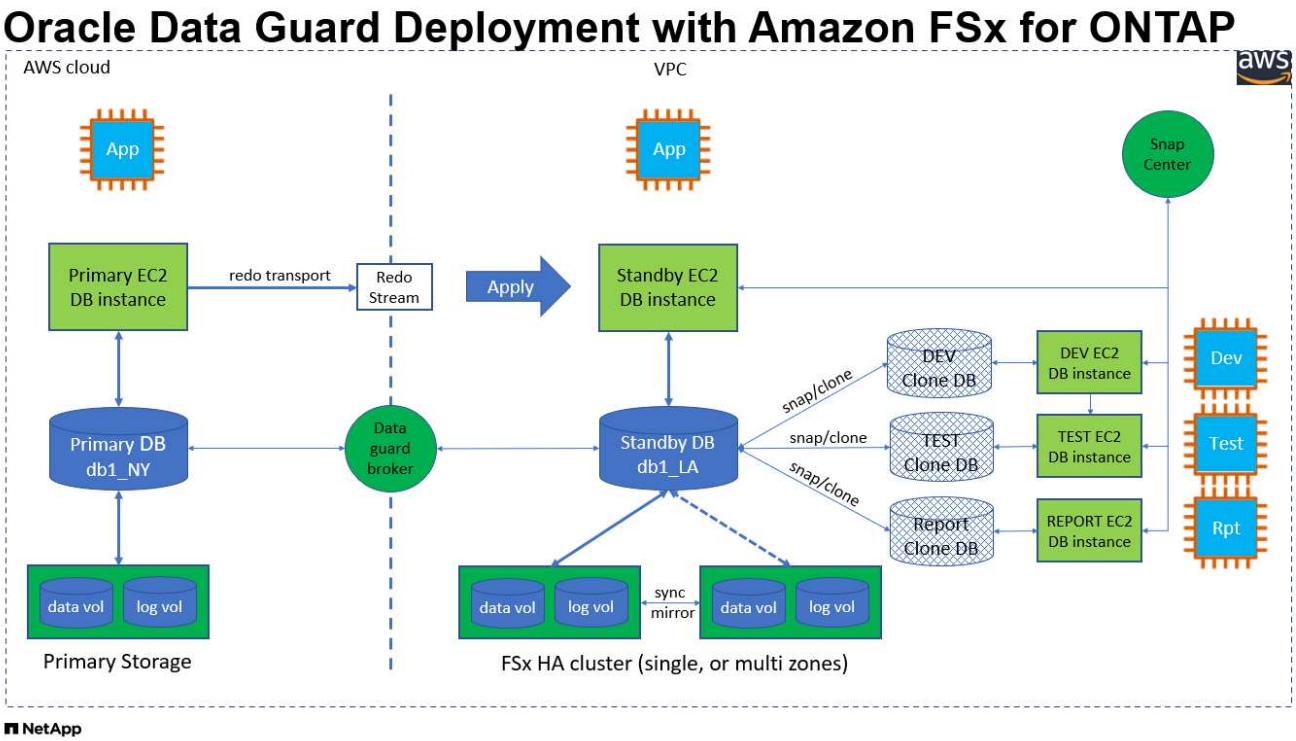
此解决方案 适用于以下人员：

- 在AWS中设置Oracle Active Data Guard以实现高可用性、数据保护和灾难恢复的数据库管理人员。
- 对AWS云中的Oracle Active Data Guard配置感兴趣的数据库解决方案架构师。
- 负责管理支持Oracle Data Guard的AWS FSx ONTAP存储的存储管理员。
- 希望在AWS FSX/EC2环境中部署Oracle Data Guard的应用程序所有者。

解决方案 测试和验证环境

此解决方案的测试和验证是在AWS FSx ONTAP和EC2实验室环境中执行的、该环境可能与最终部署环境不匹配。有关详细信息，请参见一节 [Key Factors for Deployment Consideration]。

架构



硬件和软件组件

* 硬件 *		
FSX ONTAP 存储	AWS提供的当前版本	一个FSX HA集群位于同一VPC和可用性区域中
用于计算的EC2实例	t2.xlarge/4vCPU/16G	三个EC2 T2大型EC2实例、一个用作主数据库服务器、一个用作备用数据库服务器、第三个用作克隆数据库服务器
软件		
RedHat Linux	rhel-8.6.0_hvm-20220503-x86_64-2-Hourly2-gp2	已部署RedHat订阅以进行测试
Oracle网格基础架构	版本19.18	已应用RU修补程序p34762026_190000_Linux-x86-64.zip
Oracle 数据库	版本19.18	已应用RU修补程序p34765931_190000_Linux-x86-64.zip
Oracle OPatch	版本12.2.0.1.36	最新修补程序p6880880_190000_Linux-x86-64.zip

采用从纽约到洛杉矶灾难恢复的假设设置的**Oracle Data Guard**配置

* 数据库 *	**DB_UNIQUE_NAME**	**Oracle Net Service Name**
主卷	DB1_NY	db1_NY.demo.netapp.com
物理备用	DB1_LA	db1_LA.demo.netapp.com

部署注意事项的关键因素

- **Oracle**备用数据库**FlexClone**的工作原理。 AWS FSx ONTAP FlexClone为可写的同一备用数据库卷提供共享副本。卷的副本实际上是指向原始数据块的指针、直到克隆开始新的写入为止。然后、ONTAP会为新写入分配新的存储块。所有读取IO都由活动复制下的原始数据块提供服务。因此、克隆的存储效率非常高、可用于许多其他使用情形、只需为新写入IO分配最少的增量新存储即可。这样可以大幅减少Active Data Guard存储占用空间、从而显著节省存储成本。NetApp建议在数据库从主存储切换到备用FSx存储时尽量减少FlexClone活动、以便将Oracle性能保持在较高水平。
- * Oracle软件要求。*一般来说、物理备用数据库必须与主数据库具有相同的Database Home版本、包括修补程序集例外(Patch Set例外、PSE)、关键修补程序更新(Critical Patch Update、CPU)、 和补丁集更新(PSU)、除非正在执行Oracle Data Guard Standby-First Patch Apply进程(如上的My Oracle Support note 1265700.1中所述) "support.oracle.com"
- *备用数据库目录结构注意事项。*如果可能、主系统和备用系统上的数据文件、日志文件和控制文件应具有相同的名称和路径名称、并使用最佳灵活架构(OFA)命名约定。备用数据库上的归档目录也应在站点之间完全相同、包括大小和结构。此策略允许备份、切换和故障转移等其他操作执行相同的步骤集、从而降低维护复杂性。
- *强制日志记录模式。*要防止主数据库中未记录的无法传播到备用数据库的直接写入、请在执行数据文件备份以创建备用数据库之前、在主数据库中启用强制日志记录。
- *数据库存储管理。*为简化操作、Oracle建议在Oracle Data Guard配置中设置Oracle自动存储管理(Oracle ASM)和Oracle托管文件(Oracle Managed Files、OMF)时、在主数据库和备用数据库上对称设置。
- *EC2计算实例。*在这些测试和验证中，我们使用AWS EC2 T2.xlea占用 空间实例作为Oracle数据库计算实例。NetApp建议在生产部署中使用M5类型的EC2实例作为Oracle的计算实例、因为它已针对数据库工作负载进行了优化。您需要根据实际工作负载要求根据vCPU数量和RAM量适当调整EC2实例的大小。
- * FSX存储HA集群单区域或多区域部署。*在这些测试和验证中、我们在一个AWS可用性区域中部署了一个FSX HA集群。对于生产部署、NetApp建议在两个不同的可用性区域中部署一个FSX HA对。FSx集群始终配置在HA对中、该HA对会在一对主动-被动文件系统中同步镜像、以提供存储级别冗余。多区域部署可在单个AWS区域发生故障时进一步提高高可用性。
- * FSX存储集群规模估算。*适用于ONTAP 存储文件系统的Amazon FSX可提供高达160、000个原始SSD IOPS、高达4 Gbps吞吐量以及最大192 TiB容量。但是、您可以根据部署时的实际要求、根据已配置的IOPS、吞吐量和存储限制(最小1、024 GiB)来调整集群的大小。可以动态调整容量、而不会影响应用程序可用性。

解决方案 部署

我们假定您已将主Oracle数据库部署在VPC中的AWS EC2环境中、并以此作为设置Data Guard的起点。主数据库使用Oracle ASM进行部署以进行存储管理。 为Oracle数据文件、日志文件和控制文件等创建了两个ASM磁盘组-+data和+logs 有关使用ASM在AWS中部署Oracle的详细信息、请参阅以下技术报告以获得帮助。

- "基于EC2和FSx的Oracle数据库部署最佳实践"
- "使用iSCSI/ASM在AWS FSX/EC2中部署和保护Oracle数据库"
- "Oracle 19c在使用NFS/ASM的AWS FSX/EC2上独立重新启动"

主Oracle数据库可以运行在FSx ONTAP上、也可以运行在AWS EC2生态系统中的任何其他可选存储上。下一节介绍了在使用ASM存储的主EC2数据库实例与使用ASM存储的备用EC2数据库实例之间设置Oracle Data Guard的分步部署过程。

部署的前提条件

部署需要满足以下前提条件。

1. 已设置AWS帐户、并已在您的AWS帐户中创建必要的VPC和网段。
2. 在AWS EC2控制台中、您至少需要部署三个EC2 Linux实例、一个作为主Oracle数据库实例、一个作为备用Oracle数据库实例、一个克隆目标数据库实例用于报告、开发和测试等 有关环境设置的详细信息、请参见上一节中的架构图。另请查看AWS "Linux实例用户指南" 有关详细信息 …
3. 从AWS EC2控制台中、部署Amazon FSx for ONTAP存储HA集群以托管存储Oracle备用数据库的Oracle卷。如果您不熟悉FSX存储的部署、请参见相关文档 "为ONTAP 文件系统创建FSX" 了解分步说明。
4. 可以使用以下Terraform自动化工具包执行步骤2和步骤3、该工具包会创建一个名为的EC2实例 `ora_01` 和名为的FSX文件系统 `fsx_01`。执行前、请仔细阅读该说明并根据您的环境更改变量。您可以根据自己的部署要求轻松修改此模板。

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

确保您已在EC2实例根卷中至少分配50G、以便有足够的空间来暂存Oracle安装文件。

为**Data Guard**准备主数据库

在此演示中、我们已在主EC2数据库实例上设置了一个名为db1的主Oracle数据库、其中两个ASM磁盘组采用独立的Restart配置、数据文件位于ASM磁盘组+data中、闪存恢复区域位于ASM磁盘组+logs中。下面说明了为Data Guard设置主数据库的详细过程。所有步骤均应以数据库所有者Oracle用户身份执行。

1. 主EC2数据库实例IP-172-30-15-45上的主数据库db1配置。ASM磁盘组可以位于EC2生态系统中的任何类型的存储上。

```
[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. 从sqlplus中、在主系统上启用强制日志记录。

```
alter database force logging;
```

3. 从sqlplus中、在主系统上启用回闪。通过回闪、可以在故障转移后轻松地将主数据库恢复为备用数据库。

```
alter database flashback on;
```

4. 使用Oracle密码文件配置重做传输身份验证—如果未设置、请使用orapwd实用程序在主系统上创建一个pwd文件、然后复制到备用数据库$oracle_HOME/dbs目录。

5. 在主数据库上创建与当前联机日志文件大小相同的备用重做日志。日志组比联机日志文件组多一个。然后、主数据库可以根据需要快速过渡到备用角色并开始接收重做数据。

```
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. 从sqlplus中、从spfile创建一个要编辑的pfile。

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

7. 修改pfile并添加以下参数。

```
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. 从sqlplus中、从/HOME/oracle目录中经过修订的pfile在ASM +data目录中创建spfile。

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

9. 在+data disk group下找到新创建的spfile (如有必要、请使用asmcmd实用程序)。使用srvCTL)修改网格，以便从新的spfile启动数据库，如下所示。

```
[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. 修改tnsnames.ora以添加db_UNIQUE_NAME进行名称解析。

```
# 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. 将主数据库的数据防护服务名称db1_NY_DGMGRL.demo.netapp添加到listener.ora文件中。

```
#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. 使用srvCTL关闭 并重新启动数据库，并验证数据保护参数现在是否处于活动状态。

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

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

至此、Data Guard的主数据库设置完成。

准备备用数据库并激活Data Guard

Oracle Data Guard要求操作系统内核配置和Oracle软件堆栈(包括备用EC2数据库实例上的修补程序集)与主EC2数据库实例匹配。为了便于管理和简化、备用EC2数据库实例数据库存储配置也应与主EC2数据库实例(例如ASM磁盘组的名称、数量和大小)完美匹配。下面是为Data Guard设置备用EC2数据库实例的详细过程。所有命令都应以Oracle所有者用户id的身份执行。

1. 首先、查看主EC2实例上的主数据库配置。在此演示中、我们已在主EC2数据库实例上设置了一个名为db1的主Oracle数据库、其中两个ASM磁盘组+data和+logs采用独立的Restart配置。主ASM磁盘组可以位于EC2生态系统中的任何类型的存储上。
2. 请按照文档中的步骤进行操作 "TR-4965：《使用iSCSI/ASM在AWS FSX/EC2中部署和保护Oracle数据库》" 在备用EC2数据库实例上安装和配置GRID和Oracle以与主数据库匹配。应配置数据库存储、并将其分配给FSx ONTAP中的备用EC2数据库实例、其存储容量应与主EC2数据库实例相同。

   在中的步骤10处停止 `Oracle database installation` 部分。备用数据库将使用dbca数据库复制功能从主数据库中进行初始化。

3. 安装并配置Oracle软件后、从standby $oracle_home DBS目录中、从主数据库复制Oracle密码。

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

4. 使用以下条目创建tnsnames.ora文件。

```
# 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. 将数据库数据防护服务名称添加到listener.ora文件。

```
#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. 设置Oracle主目录和路径。

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

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

7. 使用dbca从主数据库db1中对备用数据库进行初始化。

```
[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. 验证重复的备用数据库。新复制的备用数据库最初以只读模式打开。

```
[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. 在中重新启动备用数据库 `mount` 暂存并执行以下命令以激活备用数据库受管恢复。

```
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. 验证备用数据库恢复状态。请注意 `recovery logmerger` 在中 `APPLYING_LOG` 操作。

```
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>
```

这样就完成了在启用受管备用恢复的情况下、将db1从主存储到备用存储的Data Guard保护设置。

设置Data Guard代理

Oracle Data Guard代理是一个分布式管理框架、可自动集中创建、维护和监控Oracle Data Guard配置。以下部分演示如何设置Data Guard Broker以管理Data Guard环境。

1. 通过sqlplus使用以下命令在主数据库和备用数据库上启动数据防护代理。

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

2. 从主数据库中、作为SYSDBA连接到Data Guard Borker。

```
[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. 创建并启用Data Guard Broker配置。

```
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. 在Data Guard Broker管理框架内验证数据库状态。

```
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>
```

发生故障时、可以使用Data Guard Broker将主数据库瞬时故障转移到备用数据库。

克隆备用数据库以用于其他使用情形

在Data Guard中的AWS FSx ONTAP上暂存备用数据库的主要优势在于、可以通过FlexCloned以最少的额外存储投资来处理许多其他用例。在下一节中、我们将演示如何在FSx ONTAP上为已挂载和正在恢复的备用数据库卷创建快照和克隆以用于其他目的、例如开发、测试、报告等。 使用NetApp SnapCenter工具。

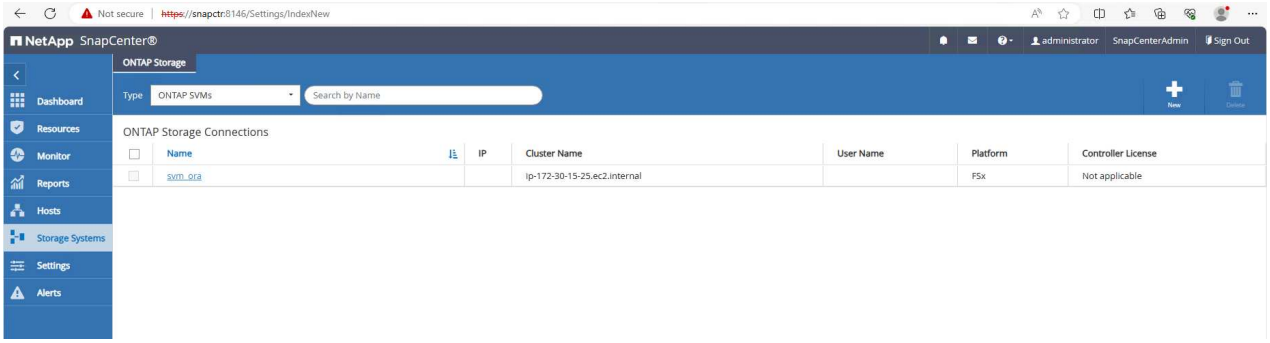
下面简要介绍了使用SnapCenter从Data Guard中托管的物理备用数据库克隆读/写数据库的过程。有关如何设置和配置SnapCenter的详细说明、请参阅 "采用 SnapCenter 的混合云数据库解决方案" Relavant Oracle (重新初始Oracle)部分。

1. 我们首先创建一个测试表、然后在主数据库的测试表中插入一行。然后、我们将验证事务是否向下遍历到备用、最后遍历克隆。

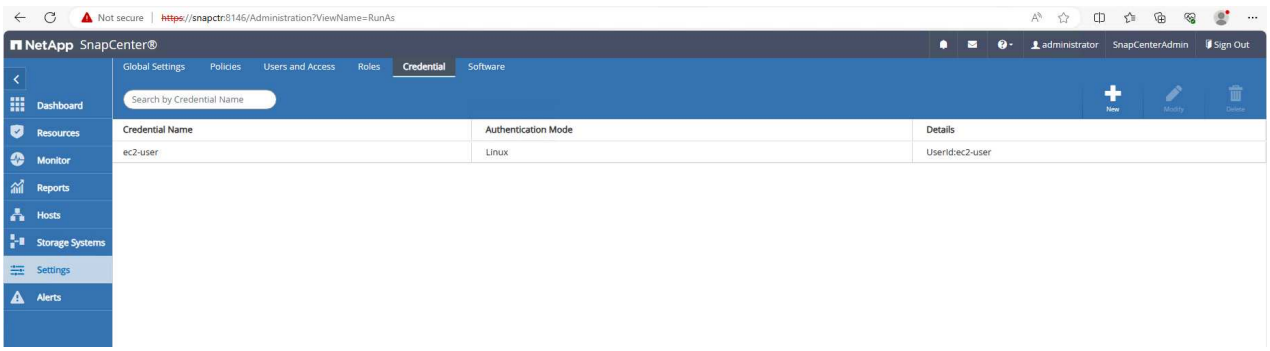
```
[oracle@ip-172-30-15-45 db1]$ sqlplus / as sysdba

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
```

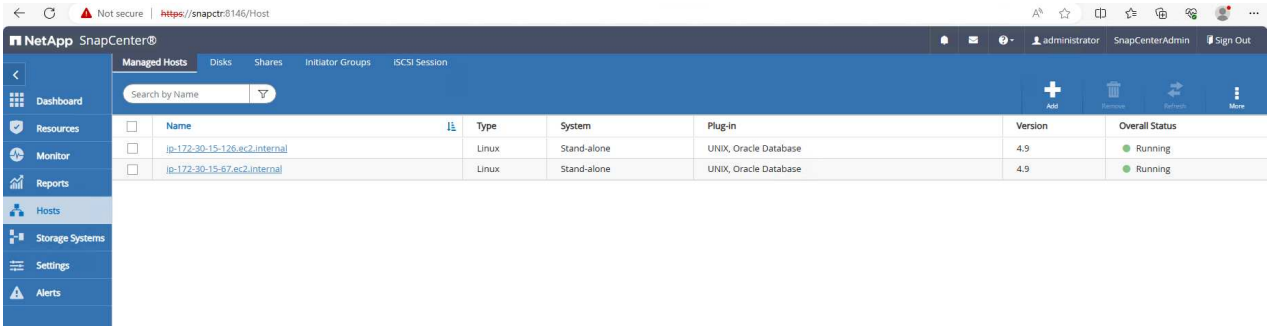
2. 将FSx存储集群添加到 `Storage Systems` 在具有FSx集群管理IP和fsxadmin凭据的SnapCenter中。



3. 将AWS EC2-user添加到 `Credential` 在中 `Settings`。

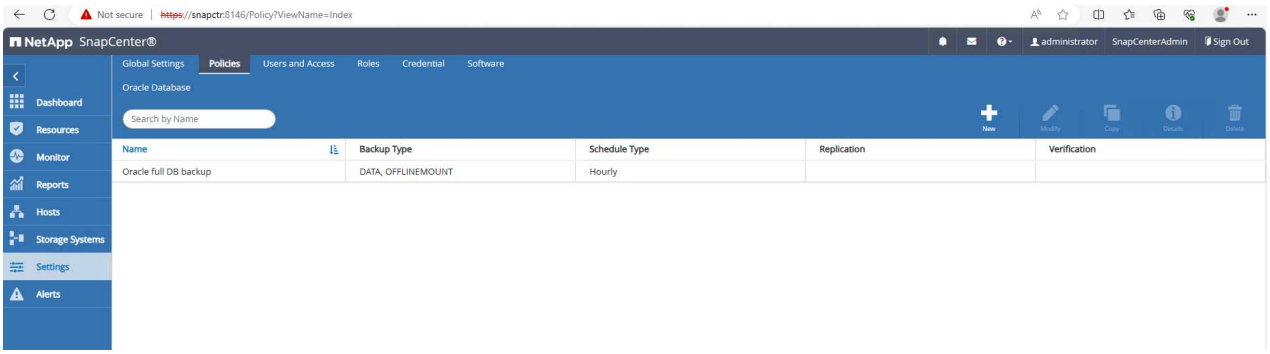


4. 添加备用EC2数据库实例并将EC2数据库实例克隆到 `Hosts`。

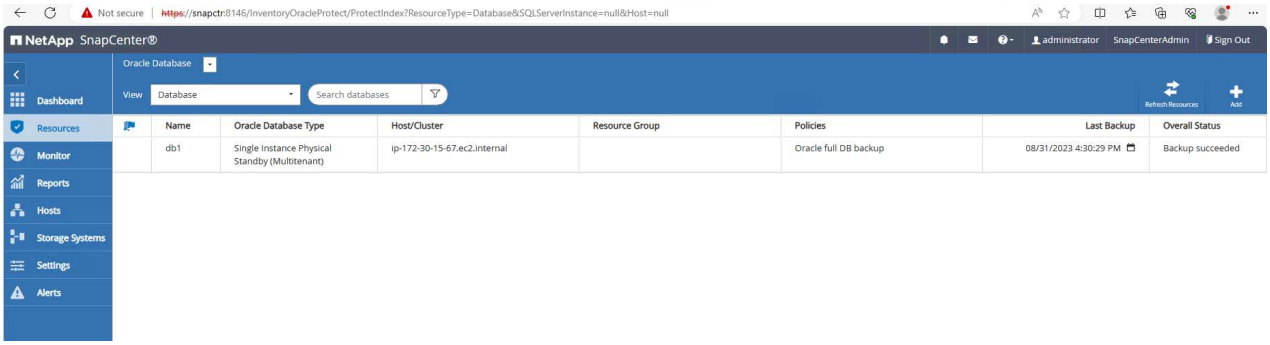


克隆EC2数据库实例应安装和配置类似的Oracle软件堆栈。在我们的测试案例中、安装并配置了网格基础架构和Oracle 19C、但未创建数据库。

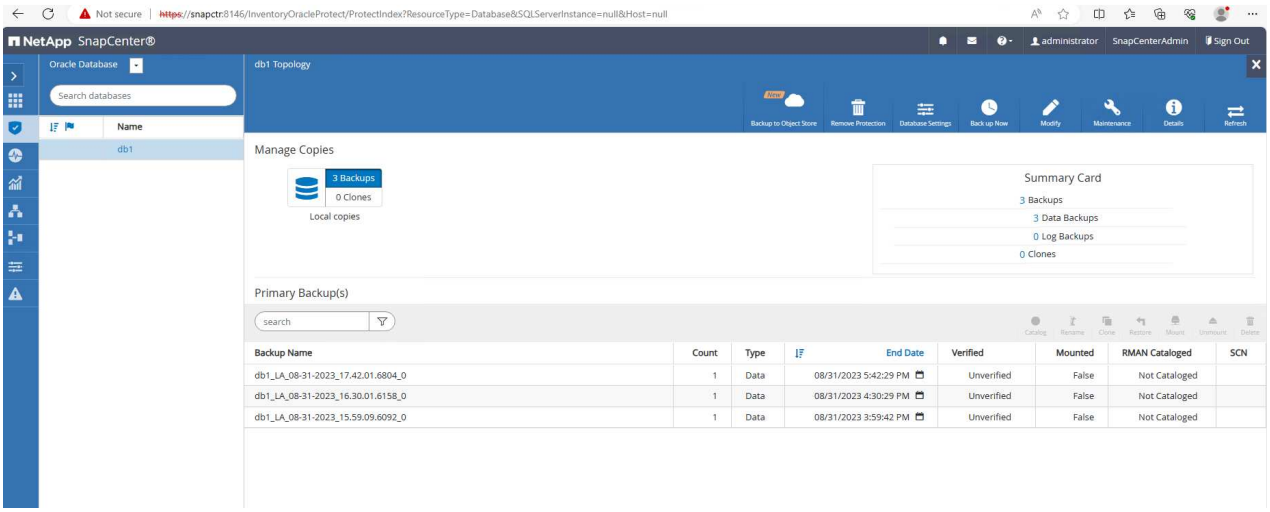
5. 创建为脱机/挂载完整数据库备份而定制的备份策略。



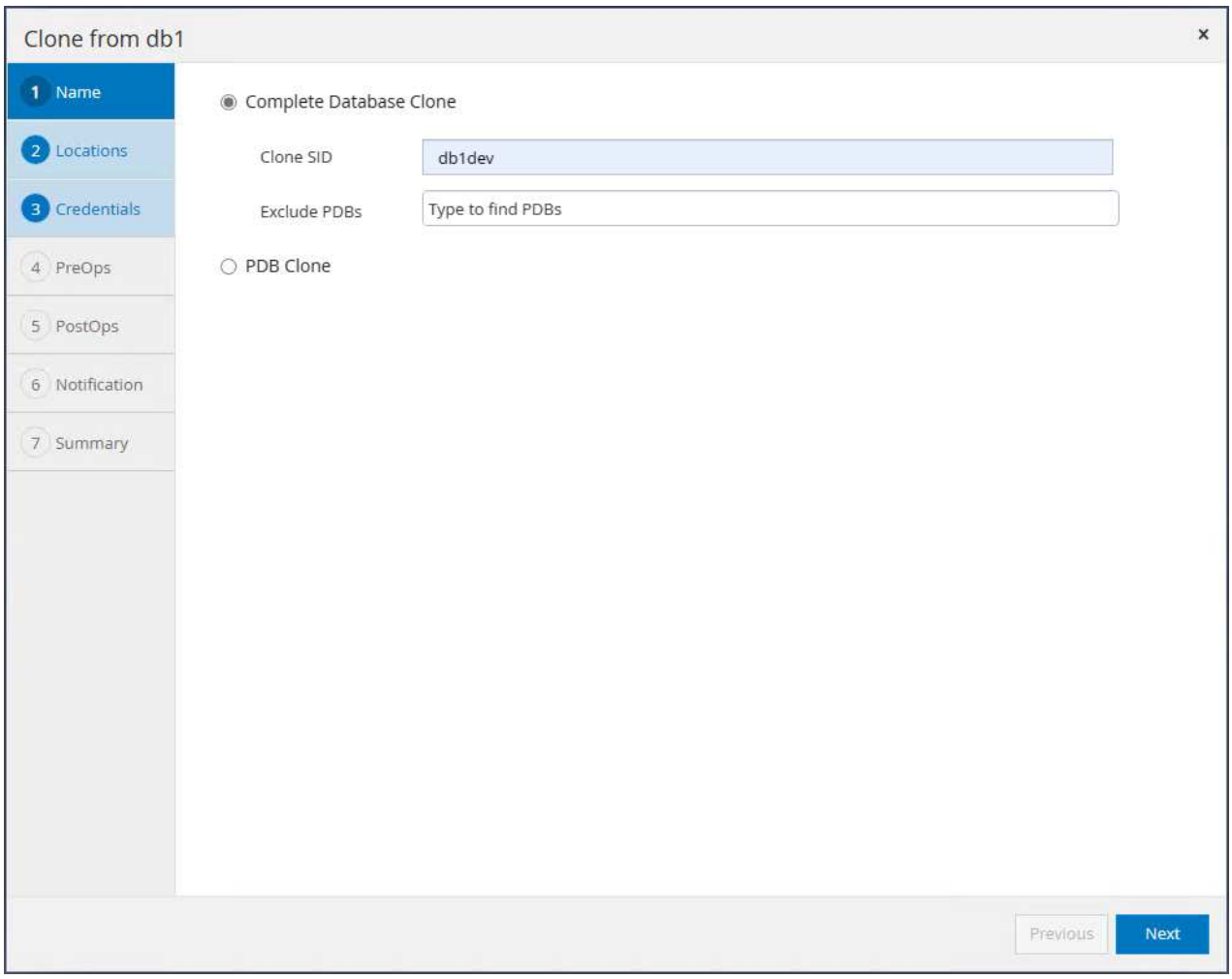
6. 在中应用备份策略以保护备用数据库 `Resources` 选项卡。



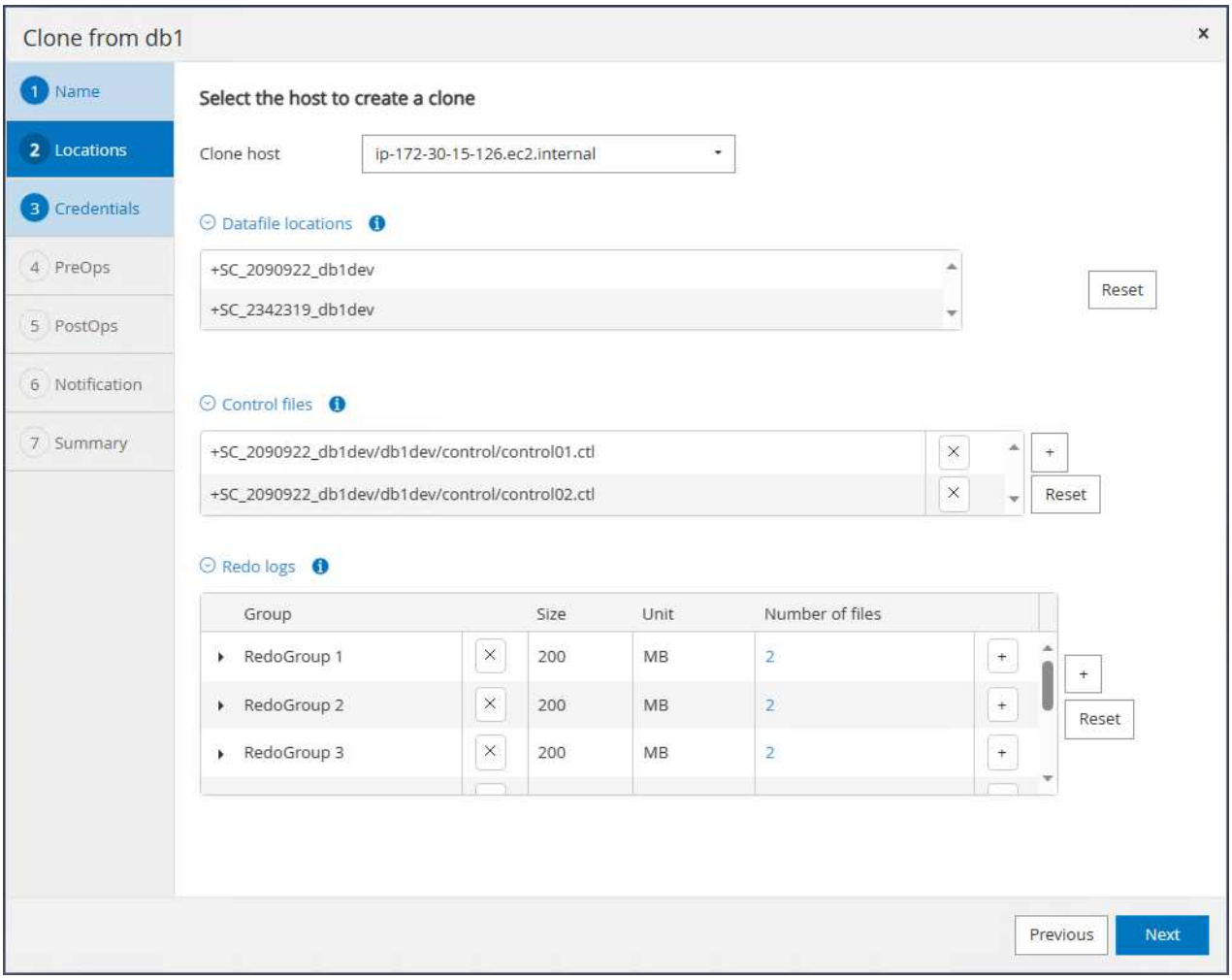
7. 单击数据库名称以打开数据库备份页面。选择要用于数据库克隆的备份、然后单击 `Clone` 用于启动克隆工作流的按钮。



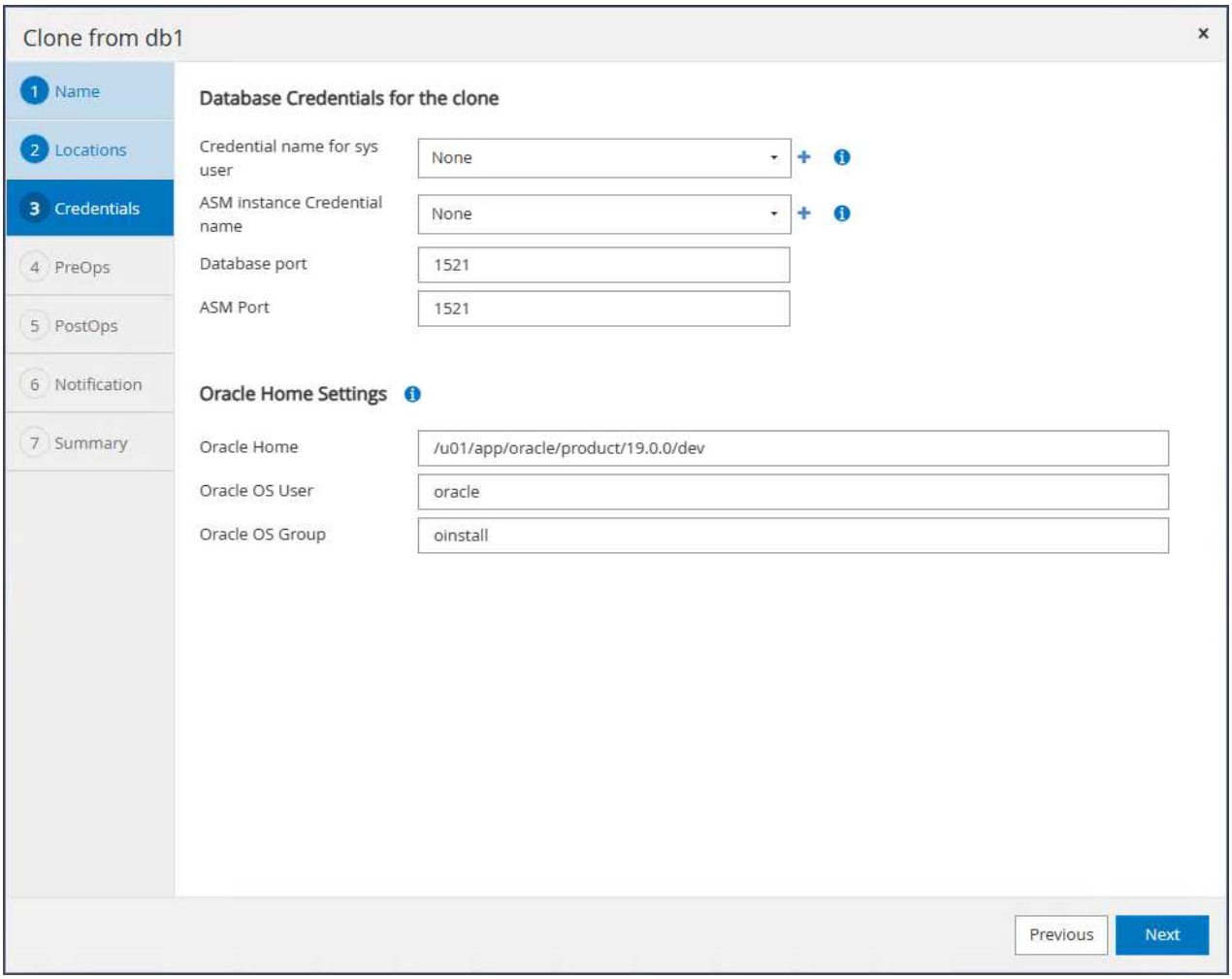
8. 选择 ... `Complete Database Clone` 并将克隆实例命名为SID。



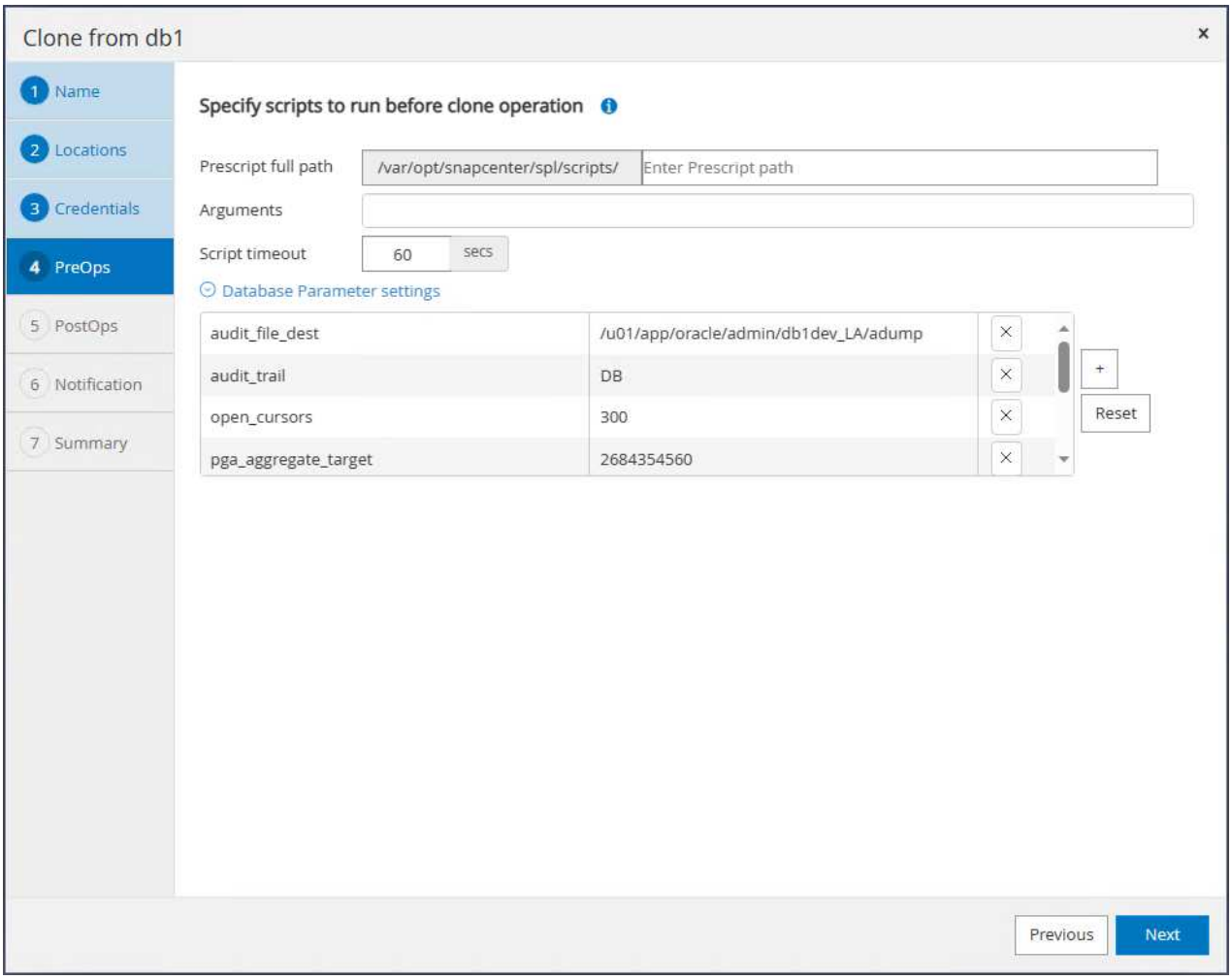
9. 选择克隆主机、此主机用于托管备用数据库中的克隆数据库。接受数据文件、控制文件和重做日志的默认设置。将在克隆主机上创建两个ASM磁盘组、它们与备用数据库上的磁盘组对应。



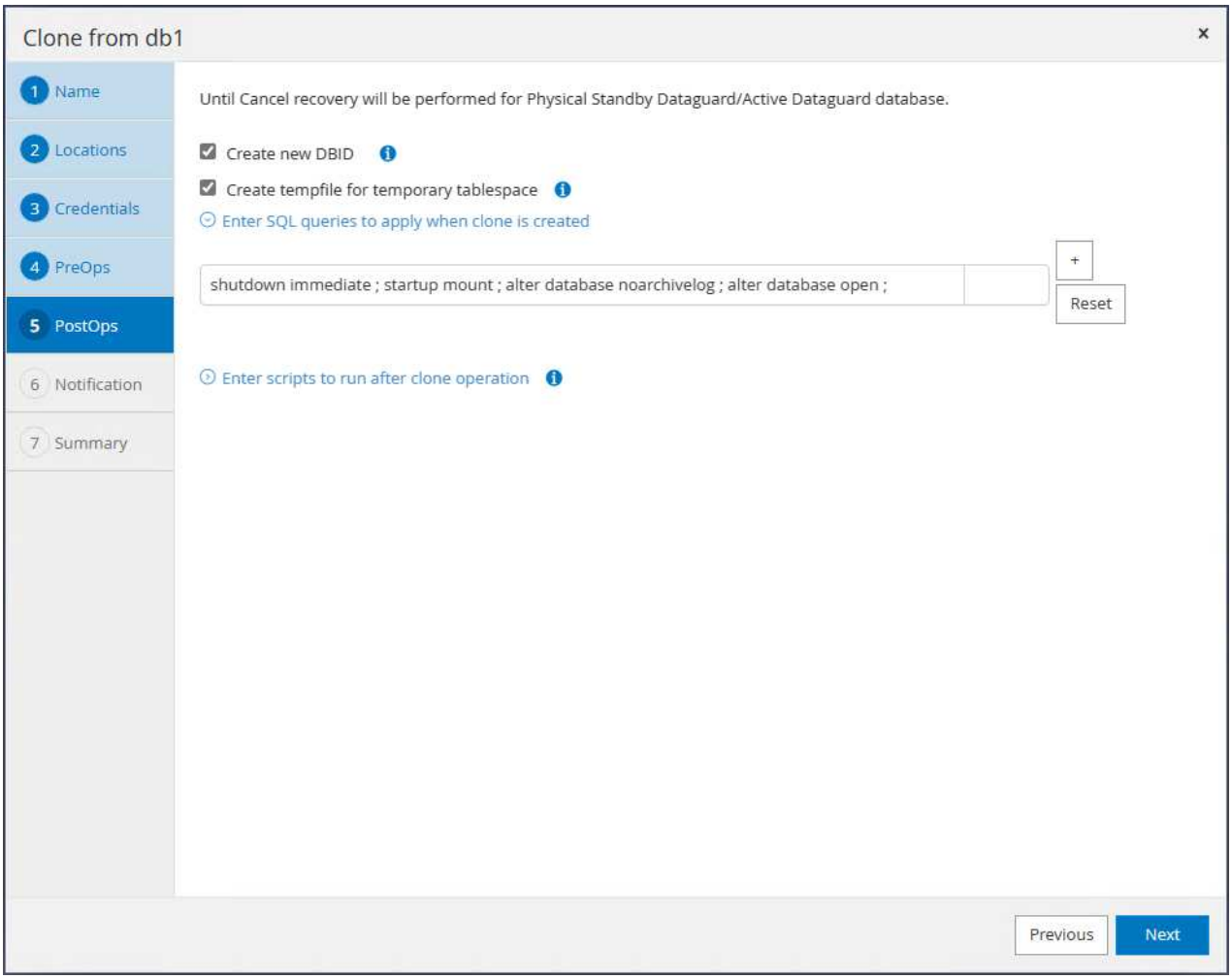
10. 基于操作系统的身份验证不需要数据库凭据。将Oracle主目录设置与克隆EC2数据库实例上配置的设置进行匹配。



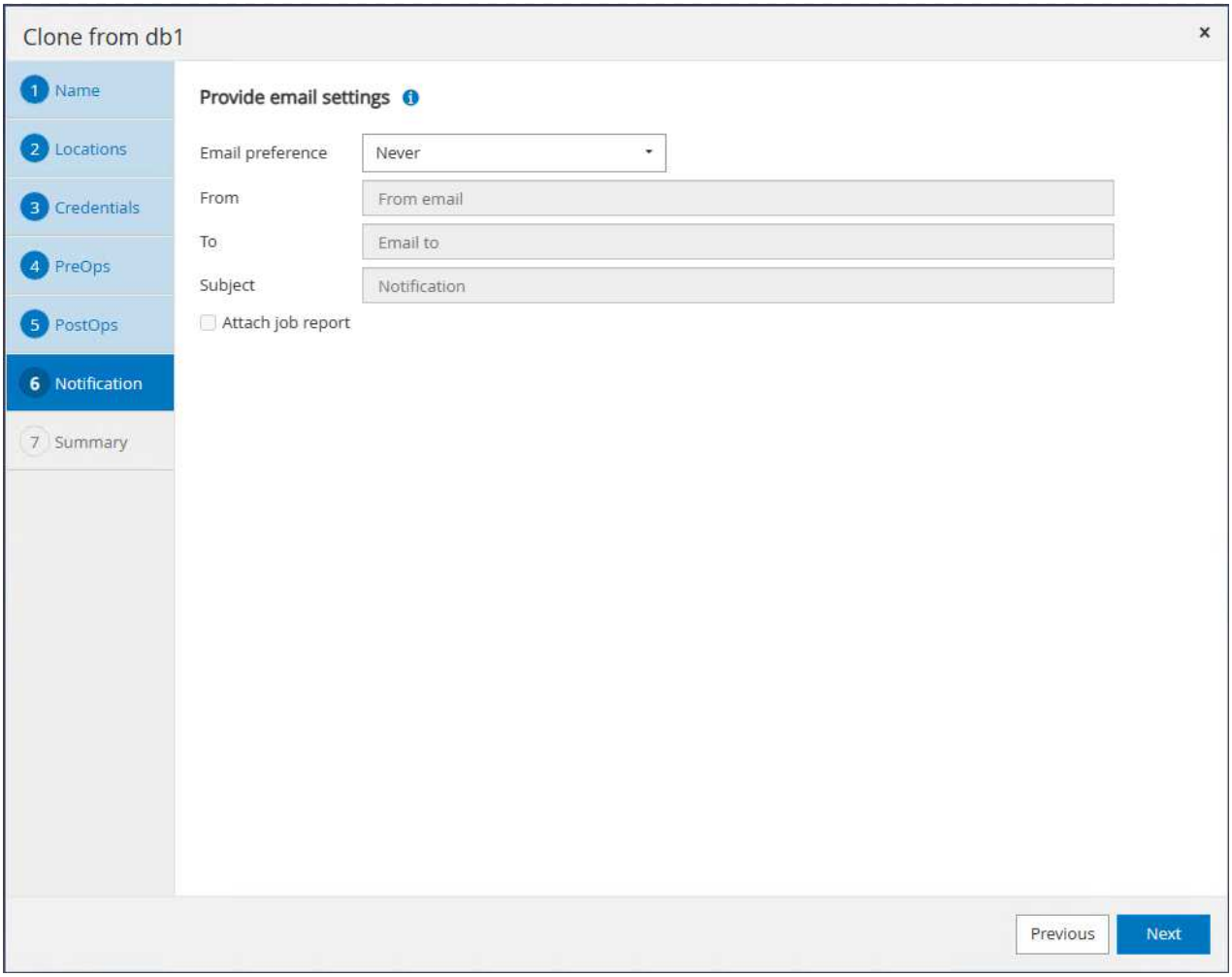
11. 根据需要更改克隆数据库参数、并指定要在回放之前运行的脚本(如果有)。



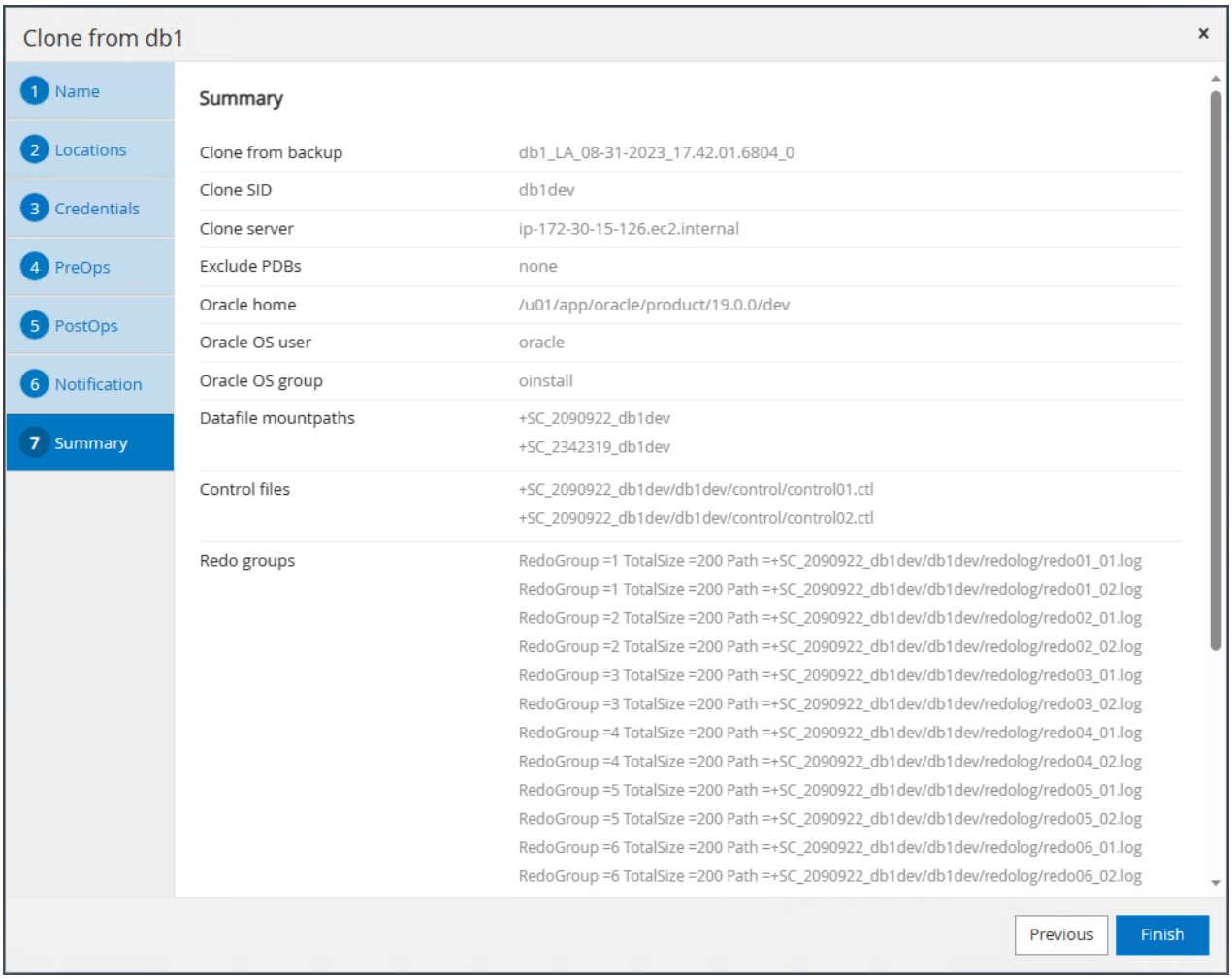
12. 输入要在克隆后运行的SQL。在演示中、我们执行了一些命令来关闭开发/测试/报告数据库的数据库归档模式。



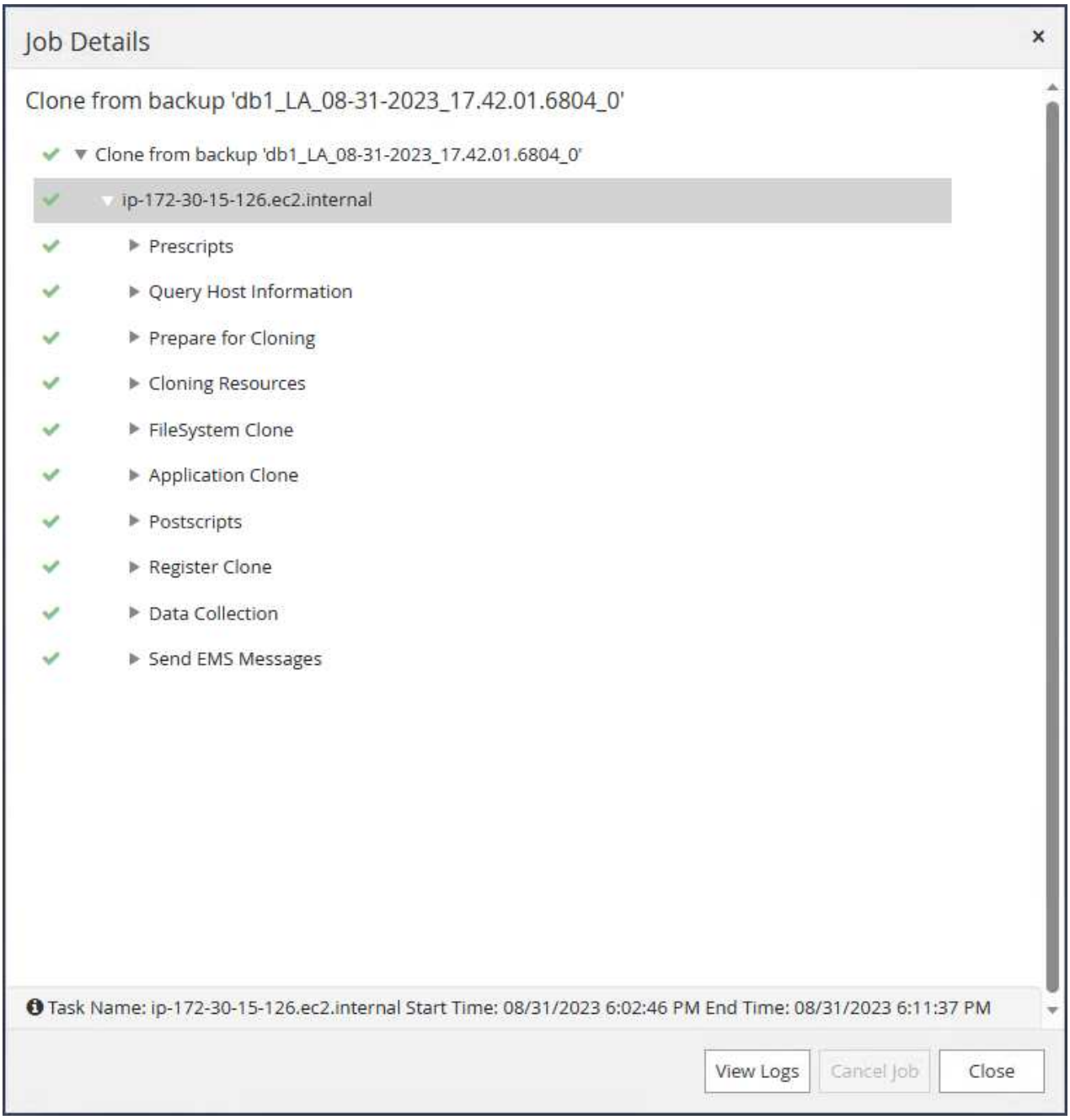
13. 根据需要配置电子邮件通知。



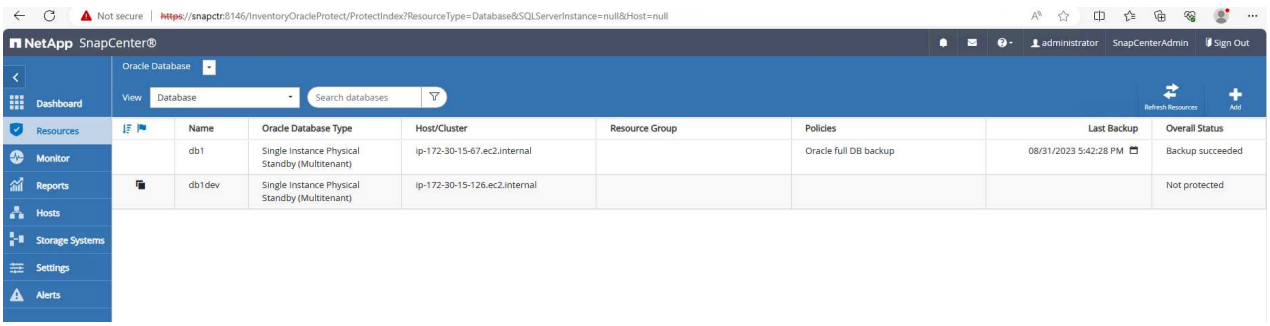
14. 查看摘要、单击 `Finish` 以启动克隆。



15. 在中监控克隆作业 `Monitor` 选项卡。我们发现、克隆数据库卷大小约为300 GB的数据库大约需要8分钟。



16. 验证SnapCenter中的克隆数据库、该数据库会立即注册到中 `Resources` 克隆操作后立即单击选项卡。



17. 从克隆EC2实例查询克隆数据库。我们验证了主数据库中发生的测试事务已向下遍历到克隆数据库。

```
[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>
```

这样就可以从FSx存储上的Data Guard中的备用数据库克隆和验证新的Oracle数据库、以供开发、测试、报告或任何其他使用情形使用。在Data Guard中、可以从同一备用数据库克隆多个Oracle数据库。

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- Data Guard概念和管理

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

- WP-7357：《基于EC2和FSx的Oracle数据库部署最佳实践》

  "https://docs.netapp.com/us-en/netapp-solutions/databases/aws_ora_fsx_ec2_deploy_intro.html"

- 适用于 NetApp ONTAP 的 Amazon FSX

  "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

TR-4973：《在AWS FSx ONTAP上使用增量合并快速恢复和克隆Oracle VLDB》

NetApp公司Allen Cao、Niyaz Mohamed

目的

使用Oracle Recovery Manager (RMAN)备份工具在Oracle中恢复超大型数据库(VLDB)可能是一项极具挑战性的任务。发生故障时从备份介质还原数据库的过程可能会非常耗时、从而会延迟数据库恢复、并可能显著影响服务级别协议(Service Level Agreement、SLA)。但是、从10g开始、Oracle引入了RMAN功能、允许用户在DB服务器主机上的其他磁盘存储上创建Oracle数据库数据文件的暂存映像副本。这些映像副本可以每天使用RMAN进行增量更新。如果发生故障、数据库管理员(Database Administrator、DBA)可以快速将Oracle数据库从故障介质切换到映像副本、而无需进行完整的数据库介质恢复。结果是SLA得到了大幅改进、尽管成本是所需数据库存储

的两倍。

如果您热衷于VLDB的SLA、并考虑将Oracle数据库迁移到AWS等公共云、则可以使用AWS FSx ONTAP等资源设置类似的数据库保护结构来暂存备用数据库映像副本。在本文档中、我们将演示如何从AWS FSx ONTAP配置和导出NFS文件系统、以挂载到Oracle数据库服务器上、从而暂存备用数据库副本、以便在主存储发生故障时快速恢复。

更好的是、我们还会介绍如何利用NetApp FlexClone为同一暂存NFS文件系统创建一份副本、以用于其他使用情形、例如使用此备用数据库映像副本建立开发或测试Oracle环境、而无需额外的存储投资。

此解决方案 可解决以下使用情形：

- 通过AWS FSx ONTAP存储上的NFS挂载点上的RMAN执行Oracle VLDB映像副本增量合并。
- 发生故障时、通过切换到FSx ONTAP存储上的数据库映像副本快速恢复Oracle VLDB。
- 克隆FSx ONTAP NFS文件系统卷、用于存储Oracle VLDB映像副本、以便为其他使用情形建立另一个数据库实例。

audience

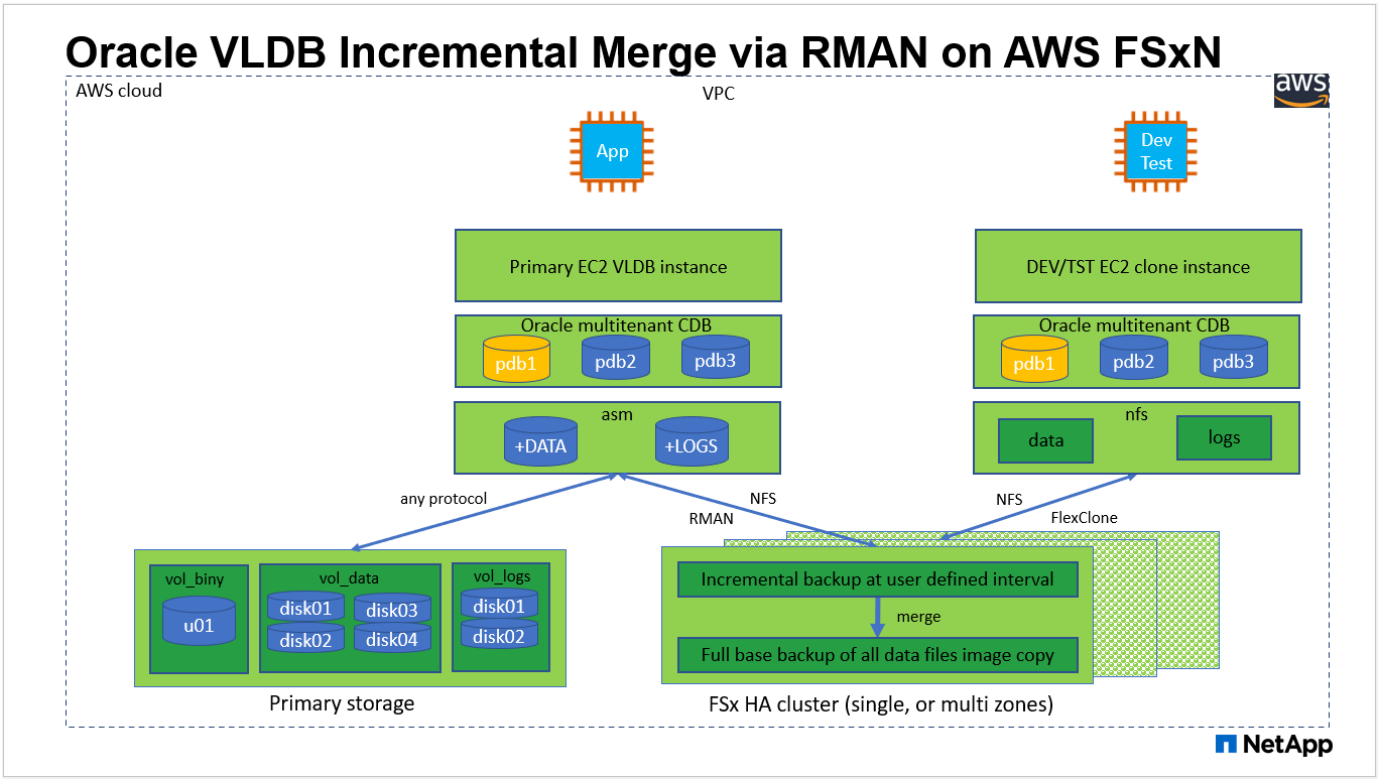
此解决方案 适用于以下人员：

- 在AWS中通过RMAN设置Oracle VLDB映像副本增量合并以加快数据库恢复的数据库提供商。
- 一名数据库解决方案架构师、负责在AWS公共云中测试Oracle工作负载。
- 负责管理部署到AWS FSx ONTAP存储的Oracle数据库的存储管理员。
- 希望在AWS FSX/EC2环境中设置Oracle数据库的应用程序所有者。

解决方案 测试和验证环境

此解决方案的测试和验证是在AWS FSx ONTAP和EC2环境中执行的、此环境可能与最终部署环境不匹配。有关详细信息，请参见一节 [Key Factors for Deployment Consideration]。

架构



硬件和软件组件

* 硬件 *		
FSX ONTAP 存储	AWS提供的当前版本	一个FSX HA集群位于同一VPC和可用性区域中
用于计算的EC2实例	t2.xlarge/4vCPU/16G	两个EC2 t2 xlarge EC2实例、一个用作主数据库服务器、另一个用作克隆数据库服务器
软件		
RedHat Linux	rhel-8.6.0_hvm-20220503-x86_64-2-Hourly2-gp2	已部署RedHat订阅以进行测试
Oracle网格基础架构	版本19.18	已应用RU修补程序p34762026_190000_Linux-x86-64.zip
Oracle 数据库	版本19.18	已应用RU修补程序p34765931_190000_Linux-x86-64.zip
Oracle OPatch	版本12.2.0.1.36	最新修补程序p6880880_190000_Linux-x86-64.zip

部署注意事项的关键因素

- *用于RMAN增量合并的Oracle VLDB存储布局。*在我们的测试和验证中、用于Oracle增量备份和合并的NFS卷是从一个FSx文件系统中分配的、该文件系统的吞吐量为4Gbps、原始SSD IOPS为160000、容量

限制为192 TiB。对于超过阈值的部署、可以将多个FSx文件系统与多个NFS挂载点并行连接、以提供更高的容量。

- 使用**RMAN**增量合并的**Oracle**可恢复性。 RMAN增量备份和合并通常根据RTO和RPO目标以用户定义的频率执行。如果主数据存储和/或归档日志完全丢失、则可能会发生数据丢失。Oracle数据库可以恢复到FSx数据库备份映像副本中提供的最后一次增量备份。为了最大限度地减少数据丢失、可以在FSx NFS挂载点上设置Oracle闪存恢复区域、并将归档日志与数据库映像副本一起备份到FSx NFS挂载。
- *在FSx NFS文件系统之外运行Oracle VLDB。*与用于数据库备份的其他批量存储不同、AWS FSx ONTAP是一款支持云的生产级存储、可提供高级别的性能和存储效率。一旦Oracle VLDB从主存储切换到FSx ONTAP NFS文件系统上的映像副本、在解决主存储故障的同时、数据库性能可以保持较高水平。您可以放心地知道、主存储故障不会影响用户应用程序体验。
- 适用于其他使用情形的**FlexClone NFS**卷的**Oracle VLDB**映像副本。 AWS FSx ONTAP FlexClone可为同一个NFS数据卷提供可写的共享副本。因此、它们可用于许多其他使用情形、同时仍可保持暂存Oracle VLDB映像副本的完整性、即使Oracle数据库已切换也是如此。这样可以大幅减少VLDB存储占用空间、从而显著节省存储成本。NetApp建议在数据库从主存储切换到数据库映像副本时尽量减少FlexClone活动、以便将Oracle性能保持在较高水平。
- *EC2计算实例。*在这些测试和验证中，我们使用AWS EC2 T2.xlea占用 空间实例作为Oracle数据库计算实例。NetApp建议在生产部署中使用M5类型的EC2实例作为Oracle的计算实例、因为它已针对数据库工作负载进行了优化。您需要根据实际工作负载要求根据vCPU数量和RAM量适当调整EC2实例的大小。
- * FSX存储HA集群单区域或多区域部署。*在这些测试和验证中、我们在一个AWS可用性区域中部署了一个FSX HA集群。对于生产部署、NetApp建议在两个不同的可用性区域中部署一个FSX HA对。FSX HA集群始终配置在一个HA对中、该HA对在一对主动-被动文件系统中进行同步镜像、以提供存储级别的冗余。多区域部署可在单个AWS区域发生故障时进一步提高高可用性。
- * FSX存储集群规模估算。*适用于ONTAP 存储文件系统的Amazon FSX可提供高达160、000个原始SSD IOPS、高达4 Gbps吞吐量以及最大192 TiB容量。但是、您可以根据部署时的实际要求、根据已配置的IOPS、吞吐量和存储限制(最小1、024 GiB)来调整集群的大小。可以动态调整容量、而不会影响应用程序可用性。
- **DNFS**配置。 DNFS内置在Oracle内核中、众所周知、在将Oracle部署到NFS存储时、它可以显著提高Oracle数据库性能。DNFS打包到Oracle二进制文件中、但默认情况下不启用。对于NFS上的任何Oracle数据库部署、都应启用此功能。对于VLDB的多FSx文件系统部署、应正确配置指向不同FSx NFS文件系统的DNFS多路径。

解决方案 部署

我们假定您已在VPC中的AWS EC2环境中部署Oracle VLDB。如果您需要有关在AWS中部署Oracle的帮助、请参阅以下技术报告以获取帮助。

- "基于EC2和FSx的Oracle数据库部署最佳实践"
- "使用iSCSI/ASM在AWS FSX/EC2中部署和保护Oracle数据库"
- "Oracle 19c在使用NFS/ASM的AWS FSX/EC2上独立重新启动"

您的Oracle VLDB可以在FSx ONTAP上运行、也可以在AWS EC2生态系统中的任何其他可选存储上运行。下一节将分步介绍如何设置RMAN增量合并到Oracle VLDB的映像副本、该副本暂存在AWS FSx ONTAP存储的NFS挂载中。

部署的前提条件

部署需要满足以下前提条件。

1. 已设置AWS帐户、并已在您的AWS帐户中创建必要的VPC和网段。
2. 在AWS EC2控制台中、您必须部署两个EC2 Linux实例、一个用作主Oracle数据库服务器、另一个用作可选的克隆目标数据库服务器。有关环境设置的详细信息、请参见上一节中的架构图。另请查看 "Linux实例用户指南" 有关详细信息 …
3. 从AWS EC2控制台中、部署Amazon FSx for ONTAP存储HA集群、以托管用于存储Oracle数据库备用映像副本的NFS卷。如果您不熟悉FSX存储的部署、请参见相关文档 "为ONTAP 文件系统创建FSX" 了解分步说明。
4. 可以使用以下Terraform自动化工具包执行步骤2和步骤3、该工具包会创建一个名为的EC2实例 `ora_01` 和名为的FSX文件系统 `fsx_01`。执行前、请仔细阅读该说明并根据您的环境更改变量。您可以根据自己的部署要求轻松修改此模板。

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

确保您已在EC2实例根卷中至少分配50G、以便有足够的空间来暂存Oracle安装文件。

配置和导出要挂载到**EC2**数据库实例主机的**NFS**卷

在此演示中、我们将展示如何通过命令行配置NFS卷、方法是以fsxadmin用户身份通过FSx集群管理IP以ssh登录到FSx集群。或者、也可以使用AWS FSx控制台分配卷。如果设置了多个FSx文件系统以适应数据库大小、请在其他FSx文件系统上重复上述过程。

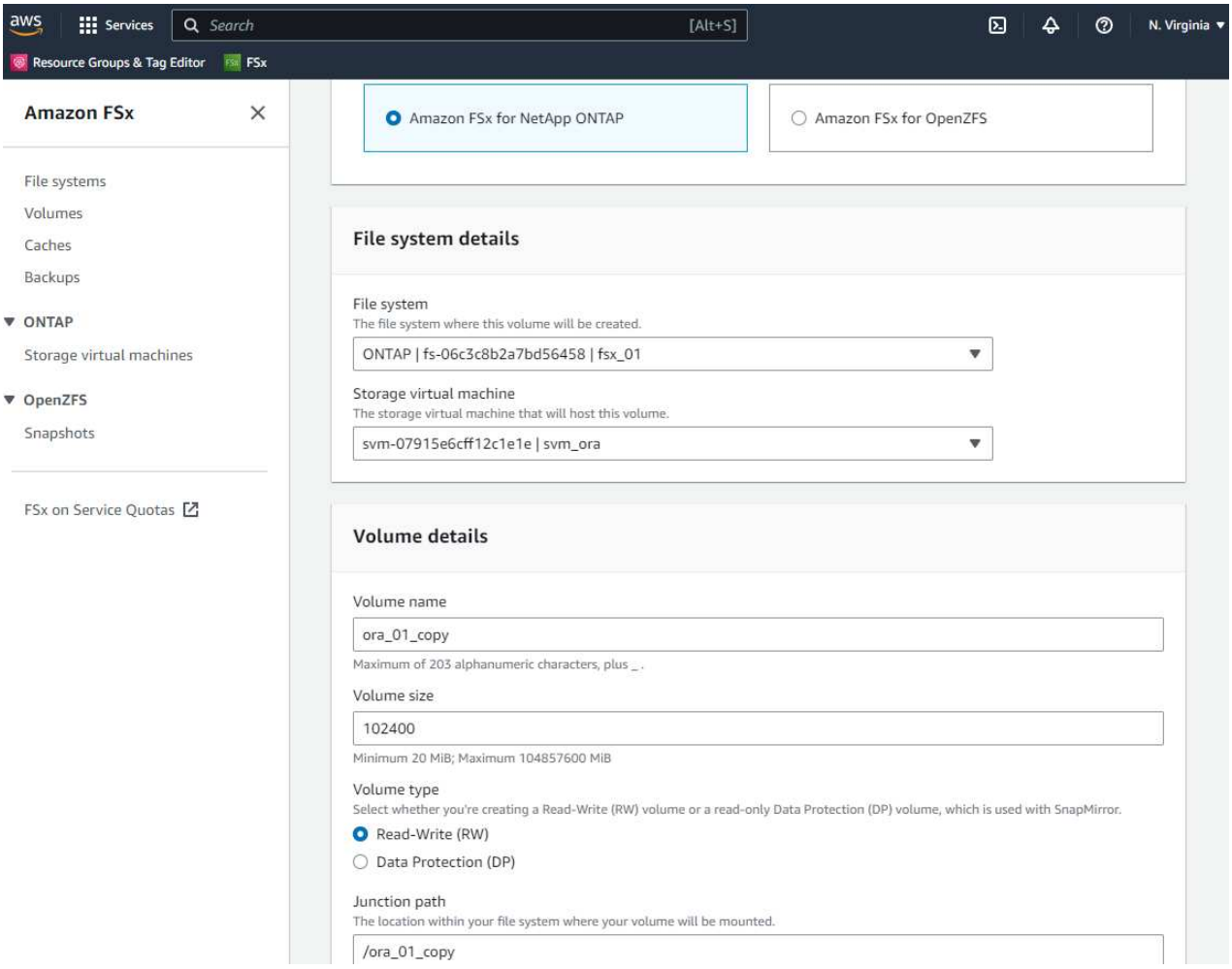
1. 首先、以fsxadmin用户身份通过SSH登录到FSx集群、通过命令行界面配置NFS卷。更改为FSx集群管理IP地址、此地址可从AWS FSx ONTAP UI控制台检索。

```
ssh fsxadmin@172.30.15.53
```

2. 创建与主存储大小相同的NFS卷、用于存储主Oracle VLDB数据库数据文件映像副本。

```
vol create -volume ora_01_copy -aggregate aggr1 -size 100G -state
online -type RW -junction-path /ora_01_copy -snapshot-policy none
-tiering-policy snapshot-only
```

3. 或者、也可以从AWS FSx控制台UI中使用以下选项配置此卷：存储效率 `Enabled`，安全模式 `Unix`、Snapshot策略 `None`和存储层 `Snapshot Only` 如下所示。



4. 为Oracle数据库创建一个具有每日计划和30天保留期限的自定义快照策略。您应根据快照频率和保留时

间窗口的具体需求调整策略。

```
snapshot policy create -policy oracle -enabled true -schedule1 daily
-count1 30
```

将策略应用于配置的NFS卷以进行RMAN增量备份和合并。

```
vol modify -volume ora_01_copy -snapshot-policy oracle
```

5. 以EC2-user身份登录到EC2实例并创建目录/nfsfsxn。为其他FSx文件系统创建其他挂载点目录。

```
sudo mkdir /nfsfsxn
```

6. 将FSx ONTAP NFS卷挂载到EC2数据库实例主机。更改为FSx虚拟服务器NFS lf地址。可以从FSx ONTAP UI控制台检索NFS lf地址。

```
sudo mount 172.30.15.19:/ora_01_copy /nfsfsxn -o
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=262144,wsize=262144,noin
tr
```

7. 将挂载点所有权更改为oracle：oisntall、根据需要更改为Oracle用户名和主组。

```
sudo chown oracle:oinstall /nfsfsxn
```

将**Oracle RMAN**增量合并设置为**FSx**上的映像副本

RMAN增量合并会在每个增量备份/合并间隔持续更新暂存数据库数据文件映像副本。数据库备份的映像副本将与您执行增量备份/合并的频率相同。因此、在确定RMAN增量备份和合并的频率时、应考虑数据库性能、RTO和RPO目标。

1. 以Oracle用户身份登录到主数据库服务器EC2实例
2. 在挂载点/nfsfsxn下创建oracopy目录、用于存储Oracle闪存恢复区域的Oracle数据文件映像副本和归档日志目录。

```
mkdir /nfsfsxn/oracopy
```

```
mkdir /nfsfsxn/archlog
```

3. 通过sqlplus登录到Oracle数据库、启用块更改跟踪以加快增量备份、如果Oracle闪存恢复区域当前位于主存储上、则将其更改为FSxN挂载。这样、RMAN默认控制文件/spfile自动备份和归档日志便可备份到FSxN NFS挂载以进行恢复。

```
sqlplus / as sysdba
```

从sqlplus提示符处、执行以下命令。

```
alter database enable block change tracking using file
'/nfsfsxn/oracopy/bct_db1.ctf'
```

```
alter system set db_recovery_file_dest='/nfsfsxn/archlog/'
scope=both;
```

4. 创建RMAN备份和增量合并脚本。该脚本会为并行RMAN备份和合并分配多个通道。首次执行将生成初始完整基线映像副本。在完整运行中、它会首先清除保留窗口之外的过时备份、以保持暂存区域干净。然后、它会在合并和备份之前切换当前日志文件。增量备份会在合并后进行、以便数据库映像副本会在当前数据库状态后经过一个备份/合并周期。可以反转合并和备份顺序、以便根据用户的偏好加快恢复速度。RMAN脚本可以集成到一个简单的shell脚本中、以便从主数据库服务器上的crontab执行。确保在RMAN设置中打开控制文件自动备份。

```
vi /home/oracle/rman_bkup_merge.cmd

Add following lines:

RUN
{
  allocate channel c1 device type disk format '/nfsfsxn/oracopy/%U';
  allocate channel c2 device type disk format '/nfsfsxn/oracopy/%U';
  allocate channel c3 device type disk format '/nfsfsxn/oracopy/%U';
  allocate channel c4 device type disk format '/nfsfsxn/oracopy/%U';
  delete obsolete;
  sql 'alter system archive log current';
  recover copy of database with tag 'OraCopyBKUPonFSxN_level_0';
  backup incremental level 1 copies=1 for recover of copy with tag
'OraCopyBKUPonFSxN_level_0' database;
}
```

5. 在EC2数据库服务器上、以Oracle用户身份本地登录到RMAN、无论是否具有RMAN目录。在此演示中、我们不会连接到RMAN目录。

```
rman target / nocatalog;

output:

[oracle@ip-172-30-15-99 ~]$ rman target / nocatalog;

Recovery Manager: Release 19.0.0.0.0 - Production on Wed May 24
17:44:49 2023
Version 19.18.0.0.0

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

connected to target database: DB1 (DBID=1730530050)
using target database control file instead of recovery catalog

RMAN>
```

6. 从RMAN提示符处、执行该脚本。首次执行时创建基线数据库映像副本、后续执行时合并并增量更新基线映像副本。下面介绍了如何执行该脚本以及典型输出。设置通道数、使其与主机上的CPU核匹配。

```
RMAN> @/home/oracle/rman_bkup_merge.cmd

RMAN> RUN
```

```
2> {
3>   allocate channel c1 device type disk format
'/nfsfsxn/oracopy/%U';
4>   allocate channel c2 device type disk format
'/nfsfsxn/oracopy/%U';
5>   allocate channel c3 device type disk format
'/nfsfsxn/oracopy/%U';
6>   allocate channel c4 device type disk format
'/nfsfsxn/oracopy/%U';
7>   delete obsolete;
8>   sql 'alter system archive log current';
9>   recover copy of database with tag 'OraCopyBKUPonFSxN_level_0';
10>  backup incremental level 1 copies=1 for recover of copy with
tag 'OraCopyBKUPonFSxN_level_0' database;
11> }

allocated channel: c1
channel c1: SID=411 device type=DISK

allocated channel: c2
channel c2: SID=146 device type=DISK

allocated channel: c3
channel c3: SID=402 device type=DISK

allocated channel: c4
channel c4: SID=37 device type=DISK

Starting recover at 17-MAY-23
no copy of datafile 1 found to recover
no copy of datafile 3 found to recover
no copy of datafile 4 found to recover
no copy of datafile 5 found to recover
no copy of datafile 6 found to recover
no copy of datafile 7 found to recover
.
.
Finished recover at 17-MAY-23

Starting backup at 17-MAY-23
channel c1: starting incremental level 1 datafile backup set
channel c1: specifying datafile(s) in backup set
input datafile file number=00022
name=+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.287.113
7018311
input datafile file number=00026
```

```
name=+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.291.113
7018481
input datafile file number=00030
name=+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.295.113
7018787
input datafile file number=00011
name=+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/undotbs1.27
1.1136668041
input datafile file number=00035
name=+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.300.113
7019181
channel c1: starting piece 1 at 17-MAY-23
channel c2: starting incremental level 1 datafile backup set
channel c2: specifying datafile(s) in backup set
input datafile file number=00023
name=+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.288.113
7018359
input datafile file number=00027
name=+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.292.113
7018523
input datafile file number=00031
name=+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.296.113
7018837
input datafile file number=00009
name=+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/system.272.
1136668041
input datafile file number=00034
name=+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.299.113
7019117
.
.
Finished backup at 17-MAY-23

Starting Control File and SPFILE Autobackup at 17-MAY-23
piece
handle=+LOGS/DB1/AUTOBACKUP/2023_05_17/s_1137095435.367.1137095435
comment=NONE
Finished Control File and SPFILE Autobackup at 17-MAY-23
released channel: c1
released channel: c2
released channel: c3
released channel: c4
```

```
RMAN> **end-of-file**
```

7. 在备份后列出数据库映像副本、以观察是否已在FSx ONTAP NFS挂载点中创建数据库映像副本。

```
RMAN> list copy of database tag 'OraCopyBKUPonFSxN_level_0';

List of Datafile Copies
=======================

Key     File S Completion Time Ckp SCN    Ckp Time        Sparse
------- ---- - --------------- ---------- --------------- ------
19      1    A 17-MAY-23       3009819    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-1_0h1sd7ae
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

20      3    A 17-MAY-23       3009826    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-3_0i1sd7at
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

21      4    A 17-MAY-23       3009830    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-4_0j1sd7b4
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

27      5    A 17-MAY-23       2383520    12-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-5_0p1sd7cf
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

26      6    A 17-MAY-23       2383520    12-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-6_0o1sd7c8
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

34      7    A 17-MAY-23       3009907    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
7_101sd7dl
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

33      8    A 17-MAY-23       2383520    12-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-8_0v1sd7di
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

28      9    A 17-MAY-23       3009871    17-MAY-23       NO
```

```
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-9_0q1sd7cm
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

22      10   A 17-MAY-23       3009849    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-10_0k1sd7bb
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

25      11   A 17-MAY-23       3009862    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-11_0n1sd7c1
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

35      12   A 17-MAY-23       3009909    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
12_111sd7dm
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

29      13   A 17-MAY-23       3009876    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-13_0r1sd7ct
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

23      14   A 17-MAY-23       3009854    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-14_0l1sd7bi
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

31      15   A 17-MAY-23       3009900    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-15_0t1sd7db
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

36      16   A 17-MAY-23       3009911    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
16_121sd7dn
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2
```

```
30      17   A 17-MAY-23       3009895    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-17_0s1sd7d4
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

24      18   A 17-MAY-23       3009858    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-18_0m1sd7bq
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

32      19   A 17-MAY-23       3009903    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-19_0u1sd7de
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

37      20   A 17-MAY-23       3009914    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
20_131sd7do
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

4       21   A 17-MAY-23       3009019    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
21_021sd6pv
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

5       22   A 17-MAY-23       3009419    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
22_031sd6r2
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

6       23   A 17-MAY-23       3009460    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
23_041sd6s5
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

7       24   A 17-MAY-23       3009473    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
24_051sd6t9
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
```

```
        Container ID: 3, PDB Name: DB1_PDB1

8       25   A 17-MAY-23       3009502    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
25_061sd6uc
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

9       26   A 17-MAY-23       3009548    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
26_071sd6vf
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

10      27   A 17-MAY-23       3009576    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
27_081sd70i
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

11      28   A 17-MAY-23       3009590    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
28_091sd71l
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

12      29   A 17-MAY-23       3009619    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
29_0a1sd72o
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

13      30   A 17-MAY-23       3009648    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
30_0b1sd73r
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

14      31   A 17-MAY-23       3009671    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
31_0c1sd74u
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

15      32   A 17-MAY-23       3009729    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
```

```
32_0d1sd762
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

16      33   A 17-MAY-23       3009743    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
33_0e1sd775
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

17      34   A 17-MAY-23       3009771    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
34_0f1sd788
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

18      35   A 17-MAY-23       3009805    17-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
35_0g1sd79b
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1


RMAN>
```

8. 通过Oracle RMAN命令提示符报告架构、以观察当前活动数据库数据文件是否位于主存储ASM +数据磁盘组中。

```
RMAN> report schema;

Report of database schema for database with db_unique_name DB1

List of Permanent Datafiles
===========================
File Size(MB) Tablespace           RB segs Datafile Name
---- -------- -------------------- ------- ------------------------
1    1060     SYSTEM               YES
+DATA/DB1/DATAFILE/system.257.1136666315
3    810      SYSAUX               NO
+DATA/DB1/DATAFILE/sysaux.258.1136666361
4    675      UNDOTBS1             YES
+DATA/DB1/DATAFILE/undotbs1.259.1136666385
5    400      PDB$SEED:SYSTEM      NO
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/system.266.11366
67165
6    460      PDB$SEED:SYSAUX      NO
```

```
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/sysaux.267.11366
67165
7    5        USERS                NO
+DATA/DB1/DATAFILE/users.260.1136666387
8    230      PDB$SEED:UNDOTBS1    NO
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/undotbs1.268.113
6667165
9    400      DB1_PDB1:SYSTEM      YES
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/system.272.11366
68041
10   490      DB1_PDB1:SYSAUX      NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/sysaux.273.11366
68041
11   465      DB1_PDB1:UNDOTBS1    YES
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/undotbs1.271.113
6668041
12   5        DB1_PDB1:USERS       NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/users.275.113666
8057
13   400      DB1_PDB2:SYSTEM      YES
+DATA/DB1/FB867EA89ECF81C0E053630F1EACB901/DATAFILE/system.277.11366
68057
14   470      DB1_PDB2:SYSAUX      NO
+DATA/DB1/FB867EA89ECF81C0E053630F1EACB901/DATAFILE/sysaux.278.11366
68057
15   235      DB1_PDB2:UNDOTBS1    YES
+DATA/DB1/FB867EA89ECF81C0E053630F1EACB901/DATAFILE/undotbs1.276.113
6668057
16   5        DB1_PDB2:USERS       NO
+DATA/DB1/FB867EA89ECF81C0E053630F1EACB901/DATAFILE/users.280.113666
8071
17   400      DB1_PDB3:SYSTEM      YES
+DATA/DB1/FB867F8A4D4F821CE053630F1EAC69CC/DATAFILE/system.282.11366
68073
18   470      DB1_PDB3:SYSAUX      NO
+DATA/DB1/FB867F8A4D4F821CE053630F1EAC69CC/DATAFILE/sysaux.283.11366
68073
19   235      DB1_PDB3:UNDOTBS1    YES
+DATA/DB1/FB867F8A4D4F821CE053630F1EAC69CC/DATAFILE/undotbs1.281.113
6668073
20   5        DB1_PDB3:USERS       NO
+DATA/DB1/FB867F8A4D4F821CE053630F1EAC69CC/DATAFILE/users.285.113666
8087
21   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.286.11370182
39
```

```
22   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.287.11370183
11
23   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.288.11370183
59
24   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.289.11370184
05
25   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.290.11370184
43
26   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.291.11370184
81
27   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.292.11370185
23
28   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.293.11370187
07
29   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.294.11370187
45
30   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.295.11370187
87
31   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.296.11370188
37
32   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.297.11370189
35
33   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.298.11370190
77
34   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.299.11370191
17
35   4096     DB1_PDB1:SOE         NO
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/DATAFILE/soe.300.11370191
81

List of Temporary Files
=======================
File Size(MB) Tablespace           Maxsize(MB) Tempfile Name
```

```
---- -------- -------------------- ----------- --------------------
1    123      TEMP                 32767
+DATA/DB1/TEMPFILE/temp.265.1136666447
2    123      PDB$SEED:TEMP        32767
+DATA/DB1/FB864A929AEB79B9E053630F1EAC7046/TEMPFILE/temp.269.1136667
185
3    10240    DB1_PDB1:TEMP        32767
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/TEMPFILE/temp.274.1136668
051
4    123      DB1_PDB2:TEMP        32767
+DATA/DB1/FB867EA89ECF81C0E053630F1EACB901/TEMPFILE/temp.279.1136668
067
5    123      DB1_PDB3:TEMP        32767
+DATA/DB1/FB867F8A4D4F821CE053630F1EAC69CC/TEMPFILE/temp.284.1136668
081

RMAN>
```

9. 验证从操作系统NFS挂载点复制的数据库映像。

```
[oracle@ip-172-30-15-99 ~]$ ls -l /nfsfsxn/oracopy/
total 70585148
-rw-r----- 1 oracle asm 4294975488 May 17 18:09 data_D-DB1_I-
1730530050_TS-SOE_FNO-21_021sd6pv
-rw-r----- 1 oracle asm 4294975488 May 17 18:10 data_D-DB1_I-
1730530050_TS-SOE_FNO-22_031sd6r2
-rw-r----- 1 oracle asm 4294975488 May 17 18:10 data_D-DB1_I-
1730530050_TS-SOE_FNO-23_041sd6s5
-rw-r----- 1 oracle asm 4294975488 May 17 18:11 data_D-DB1_I-
1730530050_TS-SOE_FNO-24_051sd6t9
-rw-r----- 1 oracle asm 4294975488 May 17 18:11 data_D-DB1_I-
1730530050_TS-SOE_FNO-25_061sd6uc
-rw-r----- 1 oracle asm 4294975488 May 17 18:12 data_D-DB1_I-
1730530050_TS-SOE_FNO-26_071sd6vf
-rw-r----- 1 oracle asm 4294975488 May 17 18:13 data_D-DB1_I-
1730530050_TS-SOE_FNO-27_081sd70i
-rw-r----- 1 oracle asm 4294975488 May 17 18:13 data_D-DB1_I-
1730530050_TS-SOE_FNO-28_091sd71l
-rw-r----- 1 oracle asm 4294975488 May 17 18:14 data_D-DB1_I-
1730530050_TS-SOE_FNO-29_0a1sd72o
-rw-r----- 1 oracle asm 4294975488 May 17 18:14 data_D-DB1_I-
1730530050_TS-SOE_FNO-30_0b1sd73r
-rw-r----- 1 oracle asm 4294975488 May 17 18:15 data_D-DB1_I-
1730530050_TS-SOE_FNO-31_0c1sd74u
-rw-r----- 1 oracle asm 4294975488 May 17 18:16 data_D-DB1_I-
```

```
1730530050_TS-SOE_FNO-32_0d1sd762
-rw-r----- 1 oracle asm 4294975488 May 17 18:16 data_D-DB1_I-
1730530050_TS-SOE_FNO-33_0e1sd775
-rw-r----- 1 oracle asm 4294975488 May 17 18:17 data_D-DB1_I-
1730530050_TS-SOE_FNO-34_0f1sd788
-rw-r----- 1 oracle asm 4294975488 May 17 18:17 data_D-DB1_I-
1730530050_TS-SOE_FNO-35_0g1sd79b
-rw-r----- 1 oracle asm  513810432 May 17 18:18 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-10_0k1sd7bb
-rw-r----- 1 oracle asm  492838912 May 17 18:18 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-14_0l1sd7bi
-rw-r----- 1 oracle asm  492838912 May 17 18:18 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-18_0m1sd7bq
-rw-r----- 1 oracle asm  849354752 May 17 18:18 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-3_0i1sd7at
-rw-r----- 1 oracle asm  482353152 May 17 18:18 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-6_0o1sd7c8
-rw-r----- 1 oracle asm 1111498752 May 17 18:18 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-1_0h1sd7ae
-rw-r----- 1 oracle asm  419438592 May 17 18:19 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-13_0r1sd7ct
-rw-r----- 1 oracle asm  419438592 May 17 18:19 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-17_0s1sd7d4
-rw-r----- 1 oracle asm  419438592 May 17 18:19 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-5_0p1sd7cf
-rw-r----- 1 oracle asm  419438592 May 17 18:19 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-9_0q1sd7cm
-rw-r----- 1 oracle asm  487596032 May 17 18:18 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-11_0n1sd7c1
-rw-r----- 1 oracle asm  246423552 May 17 18:19 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-15_0t1sd7db
-rw-r----- 1 oracle asm  246423552 May 17 18:19 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-19_0u1sd7de
-rw-r----- 1 oracle asm  707796992 May 17 18:18 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-4_0j1sd7b4
-rw-r----- 1 oracle asm  241180672 May 17 18:19 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-8_0v1sd7di
-rw-r----- 1 oracle asm    5251072 May 17 18:19 data_D-DB1_I-
1730530050_TS-USERS_FNO-12_111sd7dm
-rw-r----- 1 oracle asm    5251072 May 17 18:19 data_D-DB1_I-
1730530050_TS-USERS_FNO-16_121sd7dn
-rw-r----- 1 oracle asm    5251072 May 17 18:19 data_D-DB1_I-
1730530050_TS-USERS_FNO-20_131sd7do
-rw-r----- 1 oracle asm    5251072 May 17 18:19 data_D-DB1_I-
1730530050_TS-USERS_FNO-7_101sd7dl
```

至此、Oracle数据库备用映像副本备份和合并的设置完成。

将**Oracle**数据库切换到映像副本、以便快速恢复

如果因主存储问题描述发生故障(例如数据丢失或损坏)、则可以快速将数据库切换到FSx ONTAP NFS挂载上的映像副本、并将其恢复到当前状态、而无需还原数据库。消除介质还原可显著加快VLDB的数据库恢复速度。此使用情形假定数据库主机实例完好无损、并且数据库控制文件、归档日志和当前日志均可用于恢复。

1. 在切换之前、以Oracle用户身份登录到EC2数据库服务器主机并创建测试表。

```
[ec2-user@ip-172-30-15-99 ~]$ sudo su
[root@ip-172-30-15-99 ec2-user]# su - oracle
Last login: Thu May 18 14:22:34 UTC 2023
[oracle@ip-172-30-15-99 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 18 14:30:36
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> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 DB1_PDB1                       READ WRITE NO
         4 DB1_PDB2                       READ WRITE NO
         5 DB1_PDB3                       READ WRITE NO
SQL> alter session set container=db1_pdb1;

Session altered.

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

Table created.

SQL> insert into test values(1, sysdate, 'test oracle incremental
merge switch to copy');

1 row created.
```

```
SQL> commit;

Commit complete.

SQL> select * from test;

        ID
----------
DT
---------------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------------
------------
         1
18-MAY-23 02.35.37.000000 PM
test oracle incremental merge switch to copy


SQL>
```

2. 通过关闭中止数据库、然后在挂载阶段启动Oracle来模拟故障。

```
SQL> shutdown abort;
ORACLE instance shut down.
SQL> startup mount;
ORACLE instance started.

Total System Global Area 1.2885E+10 bytes
Fixed Size                  9177880 bytes
Variable Size            1778384896 bytes
Database Buffers         1.1073E+10 bytes
Redo Buffers               24375296 bytes
Database mounted.
SQL>
```

3. 作为Oracle用户、通过RMAN连接到Oracle数据库、以切换要复制的数据库。

```
RMAN> switch database to copy;

datafile 1 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-1_0h1sd7ae"
datafile 3 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-3_0i1sd7at"
```

```
datafile 4 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-4_0j1sd7b4"
datafile 5 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-5_0p1sd7cf"
datafile 6 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-6_0o1sd7c8"
datafile 7 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-USERS_FNO-7_101sd7dl"
datafile 8 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-8_0v1sd7di"
datafile 9 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-9_0q1sd7cm"
datafile 10 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSAUX_FNO-10_0k1sd7bb"
datafile 11 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-UNDOTBS1_FNO-11_0n1sd7c1"
datafile 12 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-12_111sd7dm"
datafile 13 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSTEM_FNO-13_0r1sd7ct"
datafile 14 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSAUX_FNO-14_0l1sd7bi"
datafile 15 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-UNDOTBS1_FNO-15_0t1sd7db"
datafile 16 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-16_121sd7dn"
datafile 17 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSTEM_FNO-17_0s1sd7d4"
datafile 18 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSAUX_FNO-18_0m1sd7bq"
datafile 19 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-UNDOTBS1_FNO-19_0u1sd7de"
datafile 20 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-20_131sd7do"
datafile 21 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-21_021sd6pv"
datafile 22 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-22_031sd6r2"
datafile 23 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-23_041sd6s5"
datafile 24 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-24_051sd6t9"
datafile 25 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-25_061sd6uc"
datafile 26 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-26_071sd6vf"
```

```
datafile 27 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-27_081sd70i"
datafile 28 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-28_091sd71l"
datafile 29 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-29_0a1sd72o"
datafile 30 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-30_0b1sd73r"
datafile 31 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-31_0c1sd74u"
datafile 32 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-32_0d1sd762"
datafile 33 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-33_0e1sd775"
datafile 34 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-34_0f1sd788"
datafile 35 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-35_0g1sd79b"
```

4. 恢复并打开数据库、使其从上次增量备份恢复到最新状态。

```
RMAN> recover database;

Starting recover at 18-MAY-23
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=392 device type=DISK
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup
set
destination for restore of datafile 00009: /nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSTEM_FNO-9_0q1sd7cm
destination for restore of datafile 00023: /nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-23_041sd6s5
destination for restore of datafile 00027: /nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-27_081sd70i
destination for restore of datafile 00031: /nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-31_0c1sd74u
destination for restore of datafile 00034: /nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-34_0f1sd788
channel ORA_DISK_1: reading from backup piece
/nfsfsxn/oracopy/321sfous_98_1_1
channel ORA_DISK_1: piece handle=/nfsfsxn/oracopy/321sfous_98_1_1
tag=ORACOPYBKUPONFSXN_LEVEL_0
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:01
```

```
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup
set
destination for restore of datafile 00010: /nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSAUX_FNO-10_0k1sd7bb
destination for restore of datafile 00021: /nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-21_021sd6pv
destination for restore of datafile 00025: /nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-25_061sd6uc
.
.
.
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup
set
destination for restore of datafile 00016: /nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-16_121sd7dn
channel ORA_DISK_1: reading from backup piece
/nfsfsxn/oracopy/3i1sfov0_114_1_1
channel ORA_DISK_1: piece handle=/nfsfsxn/oracopy/3i1sfov0_114_1_1
tag=ORACOPYBKUPONFSXN_LEVEL_0
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:01
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup
set
destination for restore of datafile 00020: /nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-20_131sd7do
channel ORA_DISK_1: reading from backup piece
/nfsfsxn/oracopy/3j1sfov0_115_1_1
channel ORA_DISK_1: piece handle=/nfsfsxn/oracopy/3j1sfov0_115_1_1
tag=ORACOPYBKUPONFSXN_LEVEL_0
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:01

starting media recovery
media recovery complete, elapsed time: 00:00:01

Finished recover at 18-MAY-23

RMAN> alter database open;

Statement processed

RMAN>
```

5. 在恢复后从sqlplus检查数据库结构、观察所有数据库数据文件(控制、临时和当前日志文件除外)现在都已切换到FSx ONTAP NFS文件系统上的副本。

```
SQL> select name from v$datafile
  2  union
  3  select name from v$tempfile
  4  union
  5  select name from v$controlfile
  6  union
  7  select member from v$logfile;

NAME
--------------------------------------------------------------------------------
------------
+DATA/DB1/CONTROLFILE/current.261.1136666435
+DATA/DB1/FB864A929AEB79B9E053630F1EAC7046/TEMPFILE/temp.269.1136667
185
+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/TEMPFILE/temp.274.1136668
051
+DATA/DB1/FB867EA89ECF81C0E053630F1EACB901/TEMPFILE/temp.279.1136668
067
+DATA/DB1/FB867F8A4D4F821CE053630F1EAC69CC/TEMPFILE/temp.284.1136668
081
+DATA/DB1/ONLINELOG/group_1.262.1136666437
+DATA/DB1/ONLINELOG/group_2.263.1136666437
+DATA/DB1/ONLINELOG/group_3.264.1136666437
+DATA/DB1/TEMPFILE/temp.265.1136666447
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-21_021sd6pv
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-22_031sd6r2

NAME
--------------------------------------------------------------------------------
------------
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-23_041sd6s5
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-24_051sd6t9
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-25_061sd6uc
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-26_071sd6vf
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-27_081sd70i
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-28_091sd71l
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-29_0a1sd72o
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-30_0b1sd73r
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-31_0c1sd74u
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-32_0d1sd762
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-33_0e1sd775

NAME
```

```
------------------------------------------------------------------
------------
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-34_0f1sd788
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-35_0g1sd79b
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-10_0k1sd7bb
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-14_0l1sd7bi
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-18_0m1sd7bq
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-3_0i1sd7at
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-6_0o1sd7c8
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-13_0r1sd7ct
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-17_0s1sd7d4
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-1_0h1sd7ae
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-5_0p1sd7cf

NAME
------------------------------------------------------------------
------------
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-9_0q1sd7cm
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-UNDOTBS1_FNO-11_0n1sd7c1
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-UNDOTBS1_FNO-15_0t1sd7db
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-UNDOTBS1_FNO-19_0u1sd7de
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-UNDOTBS1_FNO-4_0j1sd7b4
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-UNDOTBS1_FNO-8_0v1sd7di
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-12_111sd7dm
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-16_121sd7dn
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-20_131sd7do
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-7_101sd7dl

43 rows selected.

SQL>
```

6. 从SQL plus中、检查切换到复制之前插入的测试表的内容

```
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 DB1_PDB1                       READ WRITE NO
         4 DB1_PDB2                       READ WRITE NO
         5 DB1_PDB3                       READ WRITE NO
SQL> alter session set container=db1_pdb1;

Session altered.

SQL> select * from test;

        ID
----------
DT
---------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------
------------
         1
18-MAY-23 02.35.37.000000 PM
test oracle incremental merge switch to copy


SQL>
```

7. 您可以在FSx NFS挂载中长时间运行Oracle数据库、而不会影响性能、因为FSx ONTAP是可提供高性能的冗余生产级存储。修复主存储问题描述后、您可以通过反转增量备份合并过程并将停机时间降至最低来回滚到该主存储LUN。

从映像副本到不同**EC2**数据库实例主机的**Oracle**数据库恢复

如果主存储和EC2数据库实例主机均丢失、则无法从原始服务器执行恢复。幸运的是、冗余FSxN NFS文件系统上仍有Oracle数据库备份映像副本。您可以快速配置另一个相同的EC2数据库实例、并通过NFS轻松地将VLDB的映像副本挂载到新的EC2数据库主机中以运行恢复。在本节中、我们将演示执行此操作的分步过程。

1. 插入一行以测试我们之前为Oracle数据库还原到备用主机验证创建的表。

```
[oracle@ip-172-30-15-99 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Tue May 30 17:21:05
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> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 DB1_PDB1                       READ WRITE NO
         4 DB1_PDB2                       READ WRITE NO
         5 DB1_PDB3                       READ WRITE NO
SQL> alter session set container=db1_pdb1;

Session altered.


SQL> insert into test values(2, sysdate, 'test recovery on a new EC2
instance host with image copy on FSxN');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from test;

        ID
```

```
----------
DT
---------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------
------------
         1
18-MAY-23 02.35.37.000000 PM
test oracle incremental merge switch to copy

         2
30-MAY-23 05.23.11.000000 PM
test recovery on a new EC2 instance host with image copy on FSxN


SQL>
```

2. 以Oracle用户身份运行RMAN增量备份并合并、以将事务转储到FSxN NFS挂载上的备份集。

```
[oracle@ip-172-30-15-99 ~]$ rman target / nocatalog

Recovery Manager: Release 19.0.0.0.0 - Production on Tue May 30
17:26:03 2023
Version 19.18.0.0.0

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

connected to target database: DB1 (DBID=1730530050)
using target database control file instead of recovery catalog

RMAN> @rman_bkup_merge.cmd
```

3. 关闭主EC2数据库实例主机、以模拟存储和数据库服务器主机的完全故障。
4. 通过AWS EC2控制台访问操作系统和版本相同的新EC2数据库实例主机ora_02。使用与主EC2数据库服务器主机相同的修补程序和Oracle预安装RPM配置操作系统内核、并向主机添加交换空间。使用纯软件选项在主EC2数据库服务器主机中安装相同版本和修补程序的Oracle。这些任务可通过以下链接中提供的NetApp自动化工具包自动执行。

   工具包： "na_oracle19c_dDeploy"
   文档： "在 NFS 上自动部署适用于 ONTAP 的 Oracle19c"

5. 配置Oracle环境的方式与主EC2数据库实例主机ora_01类似、例如oratab、oraInst.loc和Oracle用户.bash_profile。最好将这些文件备份到FSxN NFS挂载点。

6. FSxN NFS挂载上的Oracle数据库备份映像副本存储在跨越AWS可用性区域的FSx集群上、以实现冗余、高可用性和高性能。只要网络连接可访问、NFS文件系统就可以轻松挂载到新服务器上。以下过程会将Oracle VLDB备份的映像副本挂载到新配置的EC2数据库实例主机以进行恢复。

   以EC2用户身份创建挂载点。

```
sudo mkdir /nfsfsxn
```

   以EC2用户身份挂载用于存储Oracle VLDB备份映像副本的NFS卷。

```
sudo mount 172.30.15.19:/ora_01_copy /nfsfsxn -o
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=262144,wsize=262144,noin
tr
```

7. 验证FSxN NFS挂载点上的Oracle数据库备份映像副本。

```
[ec2-user@ip-172-30-15-124 ~]$ ls -ltr /nfsfsxn/oracopy
total 78940700
-rw-r-----. 1 oracle 54331  482353152 May 26 18:45 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-6_4m1t508t
-rw-r-----. 1 oracle 54331  419438592 May 26 18:45 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-5_4q1t509n
-rw-r-----. 1 oracle 54331  241180672 May 26 18:45 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-8_4t1t50a6
-rw-r-----. 1 oracle 54331     450560 May 30 15:29 6b1tf6b8_203_1_1
-rw-r-----. 1 oracle 54331     663552 May 30 15:29 6c1tf6b8_204_1_1
-rw-r-----. 1 oracle 54331     122880 May 30 15:29 6d1tf6b8_205_1_1
-rw-r-----. 1 oracle 54331     507904 May 30 15:29 6e1tf6b8_206_1_1
-rw-r-----. 1 oracle 54331    4259840 May 30 15:29 6f1tf6b9_207_1_1
-rw-r-----. 1 oracle 54331    9060352 May 30 15:29 6h1tf6b9_209_1_1
-rw-r-----. 1 oracle 54331     442368 May 30 15:29 6i1tf6b9_210_1_1
-rw-r-----. 1 oracle 54331     475136 May 30 15:29 6j1tf6bb_211_1_1
-rw-r-----. 1 oracle 54331   48660480 May 30 15:29 6g1tf6b9_208_1_1
-rw-r-----. 1 oracle 54331     589824 May 30 15:29 6l1tf6bb_213_1_1
-rw-r-----. 1 oracle 54331     606208 May 30 15:29 6m1tf6bb_214_1_1
-rw-r-----. 1 oracle 54331     368640 May 30 15:29 6o1tf6bb_216_1_1
-rw-r-----. 1 oracle 54331     368640 May 30 15:29 6p1tf6bc_217_1_1
-rw-r-----. 1 oracle 54331      57344 May 30 15:29 6r1tf6bc_219_1_1
-rw-r-----. 1 oracle 54331      57344 May 30 15:29 6s1tf6bc_220_1_1
-rw-r-----. 1 oracle 54331      57344 May 30 15:29 6t1tf6bc_221_1_1
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-23_3q1t4ut3
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-21_3o1t4ut2
```

```
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-27_461t4vt7
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-25_3s1t4v1a
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-22_3p1t4ut3
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-31_4a1t5015
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-29_481t4vt7
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-34_4d1t5058
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-26_451t4vt7
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-24_3r1t4ut3
-rw-r-----. 1 oracle 54331  555753472 May 30 17:26 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-10_4i1t5083
-rw-r-----. 1 oracle 54331  429924352 May 30 17:26 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-9_4n1t509m
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-30_491t5014
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-28_471t4vt7
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-35_4e1t5059
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-32_4b1t501u
-rw-r-----. 1 oracle 54331  487596032 May 30 17:26 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-11_4l1t508t
-rw-r-----. 1 oracle 54331 4294975488 May 30 17:26 data_D-DB1_I-
1730530050_TS-SOE_FNO-33_4c1t501v
-rw-r-----. 1 oracle 54331    5251072 May 30 17:26 data_D-DB1_I-
1730530050_TS-USERS_FNO-12_4v1t50aa
-rw-r-----. 1 oracle 54331 1121984512 May 30 17:26 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-1_4f1t506m
-rw-r-----. 1 oracle 54331  707796992 May 30 17:26 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-4_4h1t5083
-rw-r-----. 1 oracle 54331  534781952 May 30 17:26 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-14_4j1t508s
-rw-r-----. 1 oracle 54331  429924352 May 30 17:26 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-13_4o1t509m
-rw-r-----. 1 oracle 54331  429924352 May 30 17:26 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-17_4p1t509m
-rw-r-----. 1 oracle 54331  534781952 May 30 17:26 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-18_4k1t508t
```

```
-rw-r-----. 1 oracle 54331 1027612672 May 30 17:26 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-3_4g1t506m
-rw-r-----. 1 oracle 54331    5251072 May 30 17:26 data_D-DB1_I-
1730530050_TS-USERS_FNO-7_4u1t50a6
-rw-r-----. 1 oracle 54331  246423552 May 30 17:26 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-15_4r1t50a6
-rw-r-----. 1 oracle 54331    5251072 May 30 17:26 data_D-DB1_I-
1730530050_TS-USERS_FNO-16_501t50ad
-rw-r-----. 1 oracle 54331  246423552 May 30 17:26 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-19_4s1t50a6
-rw-r-----. 1 oracle 54331    5251072 May 30 17:26 data_D-DB1_I-
1730530050_TS-USERS_FNO-20_511t50ad
-rw-r-----. 1 oracle 54331 2318712832 May 30 17:32 721tfd6b_226_1_1
-rw-r-----. 1 oracle 54331 1813143552 May 30 17:33 701tfd6a_224_1_1
-rw-r-----. 1 oracle 54331     966656 May 30 17:33 731tfdic_227_1_1
-rw-r-----. 1 oracle 54331    5980160 May 30 17:33 751tfdij_229_1_1
-rw-r-----. 1 oracle 54331     458752 May 30 17:33 761tfdin_230_1_1
-rw-r-----. 1 oracle 54331     458752 May 30 17:33 771tfdiq_231_1_1
-rw-r-----. 1 oracle 54331   11091968 May 30 17:33 741tfdij_228_1_1
-rw-r-----. 1 oracle 54331     401408 May 30 17:33 791tfdit_233_1_1
-rw-r-----. 1 oracle 54331 2070708224 May 30 17:33 6v1tfd6a_223_1_1
-rw-r-----. 1 oracle 54331     376832 May 30 17:33 7a1tfdit_234_1_1
-rw-r-----. 1 oracle 54331 1874903040 May 30 17:33 711tfd6b_225_1_1
-rw-r-----. 1 oracle 54331     303104 May 30 17:33 7c1tfdiu_236_1_1
-rw-r-----. 1 oracle 54331     319488 May 30 17:33 7d1tfdiv_237_1_1
-rw-r-----. 1 oracle 54331      57344 May 30 17:33 7f1tfdiv_239_1_1
-rw-r-----. 1 oracle 54331      57344 May 30 17:33 7g1tfdiv_240_1_1
-rw-r-----. 1 oracle 54331      57344 May 30 17:33 7h1tfdiv_241_1_1
-rw-r--r--. 1 oracle 54331      12720 May 30 17:33 db1_ctl.sql
-rw-r-----. 1 oracle 54331   11600384 May 30 17:54 bct_db1.ctf
```

8. 验证FSxN NFS挂载上可用于恢复的Oracle归档日志、并记下最后一个日志文件日志顺序编号。在本例中、此值为175。我们的恢复点最高为日志顺序编号176。

```
 [ec2-user@ip-172-30-15-124 ~]$ ls -ltr
/nfsfsxn/archlog/DB1/archivelog/2023_05_30
total 5714400
-r--r-----. 1 oracle 54331    321024 May 30 14:59
o1_mf_1_140__003t9mvn_.arc
-r--r-----. 1 oracle 54331  48996352 May 30 15:29
o1_mf_1_141__01t9qf6r_.arc
-r--r-----. 1 oracle 54331 167477248 May 30 15:44
o1_mf_1_142__02n3x2qb_.arc
-r--r-----. 1 oracle 54331 165684736 May 30 15:46
o1_mf_1_143__02rotwyb_.arc
```

```
-r--r-----. 1 oracle 54331 165636608 May 30 15:49
o1_mf_1_144__02x563wh_.arc
-r--r-----. 1 oracle 54331 168408064 May 30 15:51
o1_mf_1_145__031kg2co_.arc
-r--r-----. 1 oracle 54331 169446400 May 30 15:54
o1_mf_1_146__035xpcdt_.arc
-r--r-----. 1 oracle 54331 167595520 May 30 15:56
o1_mf_1_147__03bds8qf_.arc
-r--r-----. 1 oracle 54331 169270272 May 30 15:59
o1_mf_1_148__03gyt7rx_.arc
-r--r-----. 1 oracle 54331 170712576 May 30 16:01
o1_mf_1_149__03mfxl7v_.arc
-r--r-----. 1 oracle 54331 170744832 May 30 16:04
o1_mf_1_150__03qzz0ty_.arc
-r--r-----. 1 oracle 54331 169380864 May 30 16:06
o1_mf_1_151__03wgxdry_.arc
-r--r-----. 1 oracle 54331 169833984 May 30 16:09
o1_mf_1_152__040y85v3_.arc
-r--r-----. 1 oracle 54331 165134336 May 30 16:20
o1_mf_1_153__04ox946w_.arc
-r--r-----. 1 oracle 54331 169929216 May 30 16:22
o1_mf_1_154__04rbv7n8_.arc
-r--r-----. 1 oracle 54331 171903488 May 30 16:23
o1_mf_1_155__04tv1yvn_.arc
-r--r-----. 1 oracle 54331 179061248 May 30 16:25
o1_mf_1_156__04xgfjtl_.arc
-r--r-----. 1 oracle 54331 173593088 May 30 16:26
o1_mf_1_157__04zyg8hw_.arc
-r--r-----. 1 oracle 54331 175999488 May 30 16:27
o1_mf_1_158__052gp9mt_.arc
-r--r-----. 1 oracle 54331 179092992 May 30 16:29
o1_mf_1_159__0551wk7s_.arc
-r--r-----. 1 oracle 54331 175524352 May 30 16:30
o1_mf_1_160__057l46my_.arc
-r--r-----. 1 oracle 54331 173949440 May 30 16:32
o1_mf_1_161__05b2dmwp_.arc
-r--r-----. 1 oracle 54331 184166912 May 30 16:33
o1_mf_1_162__05drbj8n_.arc
-r--r-----. 1 oracle 54331 173026816 May 30 16:35
o1_mf_1_163__05h8lm1h_.arc
-r--r-----. 1 oracle 54331 174286336 May 30 16:36
o1_mf_1_164__05krsqmh_.arc
-r--r-----. 1 oracle 54331 166092288 May 30 16:37
o1_mf_1_165__05n378pw_.arc
-r--r-----. 1 oracle 54331 177640960 May 30 16:39
o1_mf_1_166__05pmg74l_.arc
```

```
-r--r-----. 1 oracle 54331 173972992 May 30 16:40
o1_mf_1_167__05s3o01r_.arc
-r--r-----. 1 oracle 54331 178474496 May 30 16:41
o1_mf_1_168__05vmwt34_.arc
-r--r-----. 1 oracle 54331 177694208 May 30 16:43
o1_mf_1_169__05y45qdd_.arc
-r--r-----. 1 oracle 54331 170814976 May 30 16:44
o1_mf_1_170__060kgh33_.arc
-r--r-----. 1 oracle 54331 177325056 May 30 16:46
o1_mf_1_171__0631tvgv_.arc
-r--r-----. 1 oracle 54331 164455424 May 30 16:47
o1_mf_1_172__065d94fq_.arc
-r--r-----. 1 oracle 54331 178252288 May 30 16:48
o1_mf_1_173__067wnwy8_.arc
-r--r-----. 1 oracle 54331 170579456 May 30 16:50
o1_mf_1_174__06b9zdh8_.arc
-r--r-----. 1 oracle 54331  93928960 May 30 17:26
o1_mf_1_175__08c7jc2b_.arc
[ec2-user@ip-172-30-15-124 ~]$
```

9. 作为Oracle用户、将oracle_home变量设置为新EC2实例数据库主机ora_02上的当前Oracle安装、将oracle_sid设置为主Oracle实例SID。在此示例中、此值为db1。
10. 以Oracle用户身份、在$oracle_HOME/dbs目录中创建一个通用Oracle init文件、并配置适当的管理目录。最重要的是、拥有Oracle `flash recovery area` 指向主Oracle VLDB实例中定义的FSxN NFS挂载路径。 `flash recovery area` 第节介绍了配置 `Setup Oracle RMAN incremental merge to image copy on FSx`。将Oracle控制文件设置为FSx ONTAP NFS文件系统。

```
vi $ORACLE_HOME/dbs/initdb1.ora
```

包含以下示例条目：

```
*.audit_file_dest='/u01/app/oracle/admin/db1/adump'
*.audit_trail='db'
*.compatible='19.0.0'
*.control_files=('/nfsfsxn/oracopy/db1.ctl')
*.db_block_size=8192
*.db_create_file_dest='/nfsfsxn/oracopy/'
*.db_domain='demo.netapp.com'
*.db_name='db1'
*.db_recovery_file_dest_size=85899345920
*.db_recovery_file_dest='/nfsfsxn/archlog/'
*.diagnostic_dest='/u01/app/oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=db1XDB)'
*.enable_pluggable_database=true
*.local_listener='LISTENER'
*.nls_language='AMERICAN'
*.nls_territory='AMERICA'
*.open_cursors=300
*.pga_aggregate_target=1024m
*.processes=320
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_target=10240m
*.undo_tablespace='UNDOTBS1'
```

如果出现差异、应将上述init文件替换为从主Oracle数据库服务器还原的备份init文件。

11. 以Oracle用户身份启动RMAN、以便在新的EC2数据库实例主机上运行Oracle恢复。

```
[oracle@ip-172-30-15-124 dbs]$ rman target / nocatalog;

Recovery Manager: Release 19.0.0.0.0 - Production on Wed May 31
00:56:07 2023
Version 19.18.0.0.0

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

connected to target database (not started)

RMAN> startup nomount;

Oracle instance started

Total System Global Area   12884900632 bytes

Fixed Size                     9177880 bytes
Variable Size               1778384896 bytes
Database Buffers           11072962560 bytes
Redo Buffers                  24375296 bytes
```

12. 设置数据库ID。数据库ID可从FSx NFS挂载点上映像副本的Oracle文件名中检索。

```
RMAN> set dbid = 1730530050;

executing command: SET DBID
```

13. 从自动备份还原控制文件。如果启用了Oracle控制文件和spfile自动备份、则它们会在每个增量备份和合并周期中进行备份。如果有多个副本可用、则会还原最新备份。

```
RMAN> restore controlfile from autobackup;

Starting restore at 31-MAY-23
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=2 device type=DISK

recovery area destination: /nfsfsxn/archlog
database name (or database unique name) used for search: DB1
channel ORA_DISK_1: AUTOBACKUP
/nfsfsxn/archlog/DB1/autobackup/2023_05_30/o1_mf_s_1138210401__08qlx
rrr_.bkp found in the recovery area
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20230531
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20230530
channel ORA_DISK_1: restoring control file from AUTOBACKUP
/nfsfsxn/archlog/DB1/autobackup/2023_05_30/o1_mf_s_1138210401__08qlx
rrr_.bkp
channel ORA_DISK_1: control file restore from AUTOBACKUP complete
output file name=/nfsfsxn/oracopy/db1.ctl
Finished restore at 31-MAY-23
```

14. 将init文件从spfile还原到/tmp文件夹、以便稍后更新参数文件以与主数据库实例匹配。

```
RMAN> restore spfile to pfile '/tmp/archive/initdb1.ora' from
autobackup;

Starting restore at 31-MAY-23
using channel ORA_DISK_1

recovery area destination: /nfsfsxn/archlog
database name (or database unique name) used for search: DB1
channel ORA_DISK_1: AUTOBACKUP
/nfsfsxn/archlog/DB1/autobackup/2023_05_30/o1_mf_s_1138210401__08qlx
rrr_.bkp found in the recovery area
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20230531
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20230530
channel ORA_DISK_1: restoring spfile from AUTOBACKUP
/nfsfsxn/archlog/DB1/autobackup/2023_05_30/o1_mf_s_1138210401__08qlx
rrr_.bkp
channel ORA_DISK_1: SPFILE restore from AUTOBACKUP complete
Finished restore at 31-MAY-23
```

15. 挂载控制文件并验证数据库备份映像副本。

```
RMAN> alter database mount;
```

```
released channel: ORA_DISK_1
Statement processed

RMAN> list copy of database tag 'OraCopyBKUPonFSxN_level_0';

List of Datafile Copies
=======================

Key     File S Completion Time Ckp SCN    Ckp Time        Sparse
------- ---- - --------------- ---------- --------------- ------
316     1    A 30-MAY-23       4120170    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-1_4f1t506m
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

322     3    A 30-MAY-23       4120175    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-3_4g1t506m
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

317     4    A 30-MAY-23       4120179    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-4_4h1t5083
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

221     5    A 26-MAY-23       2383520    12-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-5_4q1t509n
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

216     6    A 26-MAY-23       2383520    12-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-6_4m1t508t
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

323     7    A 30-MAY-23       4120207    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
7_4u1t50a6
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

227     8    A 26-MAY-23       2383520    12-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-8_4t1t50a6
```

```
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

308     9    A 30-MAY-23       4120158    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-9_4n1t509m
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

307     10   A 30-MAY-23       4120166    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-10_4i1t5083
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

313     11   A 30-MAY-23       4120154    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-11_4l1t508t
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

315     12   A 30-MAY-23       4120162    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
12_4v1t50aa
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

319     13   A 30-MAY-23       4120191    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-13_4o1t509m
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

318     14   A 30-MAY-23       4120183    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-14_4j1t508s
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

324     15   A 30-MAY-23       4120199    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-15_4r1t50a6
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

325     16   A 30-MAY-23       4120211    30-MAY-23       NO
```

```
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
16_501t50ad
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

320     17   A 30-MAY-23       4120195    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-17_4p1t509m
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

321     18   A 30-MAY-23       4120187    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-18_4k1t508t
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

326     19   A 30-MAY-23       4120203    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-19_4s1t50a6
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

327     20   A 30-MAY-23       4120216    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
20_511t50ad
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

298     21   A 30-MAY-23       4120166    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
21_3o1t4ut2
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

302     22   A 30-MAY-23       4120154    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
22_3p1t4ut3
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

297     23   A 30-MAY-23       4120158    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
23_3q1t4ut3
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1
```

```
306     24   A 30-MAY-23       4120162    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
24_3r1t4ut3
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

300     25   A 30-MAY-23       4120166    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
25_3s1t4v1a
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

305     26   A 30-MAY-23       4120154    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
26_451t4vt7
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

299     27   A 30-MAY-23       4120158    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
27_461t4vt7
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

310     28   A 30-MAY-23       4120162    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
28_471t4vt7
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

303     29   A 30-MAY-23       4120166    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
29_481t4vt7
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

309     30   A 30-MAY-23       4120154    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
30_491t5014
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

301     31   A 30-MAY-23       4120158    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
31_4a1t5015
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
```

```
        Container ID: 3, PDB Name: DB1_PDB1

312     32   A 30-MAY-23       4120162    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
32_4b1t501u
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

314     33   A 30-MAY-23       4120162    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
33_4c1t501v
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

304     34   A 30-MAY-23       4120158    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
34_4d1t5058
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

311     35   A 30-MAY-23       4120154    30-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
35_4e1t5059
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1
```

16. 切换要复制的数据库、以便在不还原数据库的情况下运行恢复。

```
RMAN> switch database to copy;

Starting implicit crosscheck backup at 31-MAY-23
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=11 device type=DISK
Crosschecked 33 objects
Finished implicit crosscheck backup at 31-MAY-23

Starting implicit crosscheck copy at 31-MAY-23
using channel ORA_DISK_1
Crosschecked 68 objects
Finished implicit crosscheck copy at 31-MAY-23

searching for all files in the recovery area
cataloging files...
cataloging done
```

```
List of Cataloged Files
=======================
File Name:
/nfsfsxn/archlog/DB1/autobackup/2023_05_30/o1_mf_s_1138210401__08qlx
rrr_.bkp

datafile 1 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-1_4f1t506m"
datafile 3 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-3_4g1t506m"
datafile 4 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-4_4h1t5083"
datafile 5 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-5_4q1t509n"
datafile 6 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-6_4m1t508t"
datafile 7 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-USERS_FNO-7_4u1t50a6"
datafile 8 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-8_4t1t50a6"
datafile 9 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-9_4n1t509m"
datafile 10 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSAUX_FNO-10_4i1t5083"
datafile 11 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-UNDOTBS1_FNO-11_4l1t508t"
datafile 12 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-12_4v1t50aa"
datafile 13 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSTEM_FNO-13_4o1t509m"
datafile 14 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSAUX_FNO-14_4j1t508s"
datafile 15 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-UNDOTBS1_FNO-15_4r1t50a6"
datafile 16 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-16_501t50ad"
datafile 17 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSTEM_FNO-17_4p1t509m"
datafile 18 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSAUX_FNO-18_4k1t508t"
datafile 19 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-UNDOTBS1_FNO-19_4s1t50a6"
datafile 20 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-20_511t50ad"
datafile 21 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-21_3o1t4ut2"
```

```
datafile 22 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-22_3p1t4ut3"
datafile 23 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-23_3q1t4ut3"
datafile 24 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-24_3r1t4ut3"
datafile 25 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-25_3s1t4v1a"
datafile 26 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-26_451t4vt7"
datafile 27 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-27_461t4vt7"
datafile 28 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-28_471t4vt7"
datafile 29 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-29_481t4vt7"
datafile 30 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-30_491t5014"
datafile 31 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-31_4a1t5015"
datafile 32 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-32_4b1t501u"
datafile 33 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-33_4c1t501v"
datafile 34 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-34_4d1t5058"
datafile 35 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-35_4e1t5059"
```

17. 在闪存恢复区域运行Oracle恢复、直到最后一个可用归档日志为止。

```
RMAN> run {
2> set until sequence=176;
3> recover database;
4> }

executing command: SET until clause

Starting recover at 31-MAY-23
using channel ORA_DISK_1

starting media recovery

archived log for thread 1 with sequence 142 is already on disk as
file
```

```
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_142__02n3x2qb_.ar
c
archived log for thread 1 with sequence 143 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_143__02rotwyb_.ar
c
archived log for thread 1 with sequence 144 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_144__02x563wh_.ar
c
archived log for thread 1 with sequence 145 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_145__031kg2co_.ar
c
archived log for thread 1 with sequence 146 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_146__035xpcdt_.ar
c
archived log for thread 1 with sequence 147 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_147__03bds8qf_.ar
c
archived log for thread 1 with sequence 148 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_148__03gyt7rx_.ar
c
archived log for thread 1 with sequence 149 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_149__03mfxl7v_.ar
c
archived log for thread 1 with sequence 150 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_150__03qzz0ty_.ar
c
archived log for thread 1 with sequence 151 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_151__03wgxdry_.ar
c
archived log for thread 1 with sequence 152 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_152__040y85v3_.ar
c
archived log for thread 1 with sequence 153 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_153__04ox946w_.ar
c
```

```
archived log for thread 1 with sequence 154 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_154__04rbv7n8_.ar
c
archived log for thread 1 with sequence 155 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_155__04tv1yvn_.ar
c
archived log for thread 1 with sequence 156 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_156__04xgfjtl_.ar
c
archived log for thread 1 with sequence 157 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_157__04zyg8hw_.ar
c
archived log for thread 1 with sequence 158 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_158__052gp9mt_.ar
c
archived log for thread 1 with sequence 159 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_159__0551wk7s_.ar
c
archived log for thread 1 with sequence 160 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_160__057l46my_.ar
c
archived log for thread 1 with sequence 161 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_161__05b2dmwp_.ar
c
archived log for thread 1 with sequence 162 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_162__05drbj8n_.ar
c
archived log for thread 1 with sequence 163 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_163__05h8lm1h_.ar
c
archived log for thread 1 with sequence 164 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_164__05krsqmh_.ar
c
archived log for thread 1 with sequence 165 is already on disk as
file
```

```
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_165__05n378pw_.ar
c
archived log for thread 1 with sequence 166 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_166__05pmg74l_.ar
c
archived log for thread 1 with sequence 167 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_167__05s3o01r_.ar
c
archived log for thread 1 with sequence 168 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_168__05vmwt34_.ar
c
archived log for thread 1 with sequence 169 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_169__05y45qdd_.ar
c
archived log for thread 1 with sequence 170 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_170__060kgh33_.ar
c
archived log for thread 1 with sequence 171 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_171__0631tvgv_.ar
c
archived log for thread 1 with sequence 172 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_172__065d94fq_.ar
c
archived log for thread 1 with sequence 173 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_173__067wnwy8_.ar
c
archived log for thread 1 with sequence 174 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_174__06b9zdh8_.ar
c
archived log for thread 1 with sequence 175 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_175__08c7jc2b_.ar
c
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_142__02n3x2q
b_.arc thread=1 sequence=142
archived log file
```

```
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_143__02rotwy
b_.arc thread=1 sequence=143
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_144__02x563w
h_.arc thread=1 sequence=144
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_145__031kg2c
o_.arc thread=1 sequence=145
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_146__035xpcd
t_.arc thread=1 sequence=146
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_147__03bds8q
f_.arc thread=1 sequence=147
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_148__03gyt7r
x_.arc thread=1 sequence=148
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_149__03mfxl7
v_.arc thread=1 sequence=149
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_150__03qzz0t
y_.arc thread=1 sequence=150
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_151__03wgxdr
y_.arc thread=1 sequence=151
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_152__040y85v
3_.arc thread=1 sequence=152
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_153__04ox946
w_.arc thread=1 sequence=153
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_154__04rbv7n
8_.arc thread=1 sequence=154
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_155__04tv1yv
n_.arc thread=1 sequence=155
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_156__04xgfjt
l_.arc thread=1 sequence=156
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_157__04zyg8h
w_.arc thread=1 sequence=157
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_158__052gp9m
```

```
t_.arc thread=1 sequence=158
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_159__0551wk7
s_.arc thread=1 sequence=159
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_160__057l46m
y_.arc thread=1 sequence=160
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_161__05b2dmw
p_.arc thread=1 sequence=161
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_162__05drbj8
n_.arc thread=1 sequence=162
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_163__05h8lm1
h_.arc thread=1 sequence=163
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_164__05krsqm
h_.arc thread=1 sequence=164
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_165__05n378p
w_.arc thread=1 sequence=165
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_166__05pmg74
l_.arc thread=1 sequence=166
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_167__05s3o01
r_.arc thread=1 sequence=167
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_168__05vmwt3
4_.arc thread=1 sequence=168
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_169__05y45qd
d_.arc thread=1 sequence=169
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_170__060kgh3
3_.arc thread=1 sequence=170
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_171__0631tvg
v_.arc thread=1 sequence=171
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_172__065d94f
q_.arc thread=1 sequence=172
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_173__067wnwy
8_.arc thread=1 sequence=173
```

```
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_174__06b9zdh
8_.arc thread=1 sequence=174
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_05_30/o1_mf_1_175__08c7jc2
b_.arc thread=1 sequence=175
media recovery complete, elapsed time: 00:48:34
Finished recover at 31-MAY-23
```

> 要加快恢复速度、请使用recovery _parlism参数启用并行会话、或者在恢复命令中指定并行程度以进行数据库恢复： `RECOVER DATABASE PARALLEL (DEGREE d INSTANCES DEFAULT);`。通常、并行度应等于主机上的CPU核数。

18. 退出RMAN、以Oracle用户身份通过sqlplus登录到Oracle、以便在恢复不完整后打开数据库并重置日志。

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

NAME      OPEN_MODE
--------- --------------------
DB1       MOUNTED

SQL> select member from v$logfile;

MEMBER
-------------------------------------------------------------------------
------------
+DATA/DB1/ONLINELOG/group_3.264.1136666437
+DATA/DB1/ONLINELOG/group_2.263.1136666437
+DATA/DB1/ONLINELOG/group_1.262.1136666437

SQL> alter database rename file
'+DATA/DB1/ONLINELOG/group_1.262.1136666437' to
'/nfsfsxn/oracopy/redo01.log';

Database altered.

SQL> alter database rename file
'+DATA/DB1/ONLINELOG/group_2.263.1136666437' to
'/nfsfsxn/oracopy/redo02.log';

Database altered.

SQL> alter database rename file
'+DATA/DB1/ONLINELOG/group_3.264.1136666437' to
'/nfsfsxn/oracopy/redo03.log';

Database altered.

SQL> alter database open resetlogs;

Database altered.
```

19. 验证是否已将数据库还原到主数据库出现故障之前已插入行的新主机。

```
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 DB1_PDB1                       READ WRITE NO
         4 DB1_PDB2                       READ WRITE NO
         5 DB1_PDB3                       READ WRITE NO
SQL> alter session set container=db1_pdb1;

Session altered.

SQL> select * from test;

        ID DT
EVENT
----------
--------------------------------------------------------------------
-------
--------------------------------------------------------------------
-------------------------------
         1 18-MAY-23 02.35.37.000000 PM
test oracle incremental merge switch to copy
         2 30-MAY-23 05.23.11.000000 PM
test recovery on a new EC2 instance host with image copy on FSxN
```

20. 其他恢复后任务

```
Add FSxN NFS mount to fstab so that the NFS file system will be
mounted when EC2 instance host rebooted.

As EC2 user, vi /etc/fstab and add following entry:

172.30.15.19:/ora_01_copy       /nfsfsxn        nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=262144,wsize=262144,noin
tr  0       0

Update the Oracle init file from primary databse init file backup
that is restored to /tmp/archive and create spfile as needed.
```

这样、Oracle VLDB数据库便可从FSxN NFS文件系统上的备份映像副本恢复到新的EC2数据库实例主机。

克隆**Oracle**备用映像副本、以供其他使用情形使用

使用AWS FSx ONTAP暂存Oracle VLDB映像副本的另一个优势是、可以通过FlexCloned将其用于其他许多用途、而额外的存储投资极少。在以下用例中、我们将演示如何在FSx ONTAP上为其他Oracle用例(如开发、UAT等)创建暂存NFS卷的快照和克隆

1. 首先、我们会在之前创建的同一测试表中插入一行。

```
 SQL> insert into test values (3, sysdate, 'test clone on a new EC2
instance host with image copy on FSxN');

1 row created.

SQL> select * from test;

        ID
----------
DT
---------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------
------------
         1
18-MAY-23 02.35.37.000000 PM
test oracle incremental merge switch to copy

         2
30-MAY-23 05.23.11.000000 PM
test recovery on a new EC2 instance host with image copy on FSxN

        ID
----------
DT
---------------------------------------------------------------------
-------
EVENT
---------------------------------------------------------------------
------------

         3
05-JUN-23 03.19.46.000000 PM
test clone on a new EC2 instance host with image copy on FSxN


SQL>
```

2. 创建RMAN备份并合并到FSx ONTAP数据库映像副本、以便事务将捕获到FSx NFS挂载上的备份集中、但在恢复克隆的数据库之前不会合并到副本中。

```
RMAN> @/home/oracle/rman_bkup_merge.cmd
```

3. 以fsxadmin用户身份通过ssh登录到FSx集群、以观察由计划备份策略Oracle创建的快照、并创建一次性快照、以使其包含我们在步骤1中执行的事务。

```
FsxId06c3c8b2a7bd56458::> vol snapshot create -vserver svm_ora
-volume ora_01_copy -snapshot one-off.2023-06-05-1137 -foreground
true

FsxId06c3c8b2a7bd56458::> snapshot show

---Blocks---
Vserver  Volume   Snapshot                                  Size
Total% Used%
-------- -------- ------------------------------------ --------
------ -----
svm_ora  ora_01_copy
                  daily.2023-06-02_0010                   3.59GB
2%    5%
                  daily.2023-06-03_0010                   1.10GB
1%    1%
                  daily.2023-06-04_0010                    608KB
0%    0%
                  daily.2023-06-05_0010                   3.81GB
2%    5%
                  one-off.2023-06-05-1137                  168KB
0%    0%
         svm_ora_root
                  weekly.2023-05-28_0015                  1.86MB
0%   78%
                  daily.2023-06-04_0010                    152KB
0%   22%
                  weekly.2023-06-04_0015                  1.24MB
0%   70%
                  daily.2023-06-05_0010                    196KB
0%   27%
                  hourly.2023-06-05_1005                   156KB
0%   22%
                  hourly.2023-06-05_1105                   156KB
0%   22%
                  hourly.2023-06-05_1205                   156KB
0%   22%
                  hourly.2023-06-05_1305                   156KB
0%   22%
                  hourly.2023-06-05_1405                  1.87MB
0%   78%
                  hourly.2023-06-05_1505                   148KB
0%   22%
15 entries were displayed.
```

4. 从一次性快照克隆、用于在备用EC2 Oracle主机上建立新的DB1克隆实例。您可以选择从卷ora_01_copy的任何可用每日快照克隆。

```
FsxId06c3c8b2a7bd56458::> vol clone create -flexclone db1_20230605of
-type RW -parent-vserver svm_ora -parent-volume ora_01_copy
-junction-path /db1_20230605of -junction-active true -parent
-snapshot one-off.2023-06-05-1137
[Job 464] Job succeeded: Successful

FsxId06c3c8b2a7bd56458::>

FsxId06c3c8b2a7bd56458::> vol show db1*
Vserver   Volume       Aggregate    State      Type       Size
Available Used%
--------- ------------ ------------ ---------- ---- ----------
---------- -----
svm_ora   db1_20230605of
                       aggr1        online     RW        200GB
116.6GB   38%

FsxId06c3c8b2a7bd56458::>
```

5. 关闭克隆卷的Snapshot策略、因为它会继承父卷的Snapshot策略、除非您要保护克隆的卷、否则请将其保留为独立卷。

```
FsxId06c3c8b2a7bd56458::> vol modify -volume db1_20230605of
-snapshot-policy none

Warning: You are changing the Snapshot policy on volume
"db1_20230605of" to "none". Snapshot copies on this volume that do
not match any of the prefixes of the new Snapshot policy will not be
deleted. However, when the new Snapshot policy
         takes effect, depending on the new retention count, any
existing Snapshot copies that continue to use the same prefixes
might be deleted. See the 'volume modify' man page for more
information.
Do you want to continue? {y|n}: y
Volume modify successful on volume db1_20230605of of Vserver
svm_ora.

FsxId06c3c8b2a7bd56458::>
```

6. 登录到一个新的EC2 Linux实例、此实例已预安装Oracle软件、并且版本和修补程序级别与主Oracle EC2实例相同、然后挂载克隆的卷。

```
[ec2-user@ip-172-30-15-124 ~]$ sudo mkdir /nfsfsxn
[ec2-user@ip-172-30-15-124 ~]$ sudo mount -t nfs
172.30.15.19:/db1_20230605of /nfsfsxn -o
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=262144,wsize=262144,noin
tr
```

7. 验证FSx NFS挂载上的数据库增量备份集、映像副本和可用归档日志。

```
[ec2-user@ip-172-30-15-124 ~]$ ls -ltr /nfsfsxn/oracopy
total 79450332
-rw-r----- 1 oracle 54331  482353152 Jun  1 19:02 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-6_891tkrhr
-rw-r----- 1 oracle 54331  419438592 Jun  1 19:03 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-5_8d1tkril
-rw-r----- 1 oracle 54331  241180672 Jun  1 19:03 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-8_8g1tkrj7
-rw-r----- 1 oracle 54331  912506880 Jun  1 20:21 8n1tkvv2_279_1_1
-rw-r----- 1 oracle 54331     925696 Jun  1 20:21 8q1tl05i_282_1_1
-rw-r----- 1 oracle 54331 1169014784 Jun  1 20:21 8p1tkvv2_281_1_1
-rw-r----- 1 oracle 54331    6455296 Jun  1 20:21 8r1tl05m_283_1_1
-rw-r----- 1 oracle 54331     139264 Jun  1 20:21 8t1tl05t_285_1_1
-rw-r----- 1 oracle 54331    3514368 Jun  1 20:21 8s1tl05t_284_1_1
-rw-r----- 1 oracle 54331     139264 Jun  1 20:21 8u1tl060_286_1_1
-rw-r----- 1 oracle 54331     425984 Jun  1 20:21 901tl062_288_1_1
-rw-r----- 1 oracle 54331     344064 Jun  1 20:21 911tl062_289_1_1
-rw-r----- 1 oracle 54331     245760 Jun  1 20:21 931tl063_291_1_1
-rw-r----- 1 oracle 54331     237568 Jun  1 20:21 941tl064_292_1_1
-rw-r----- 1 oracle 54331      57344 Jun  1 20:21 961tl065_294_1_1
-rw-r----- 1 oracle 54331      57344 Jun  1 20:21 971tl066_295_1_1
-rw-r----- 1 oracle 54331      57344 Jun  1 20:21 981tl067_296_1_1
-rw-r----- 1 oracle 54331 1040760832 Jun  1 20:23 8m1tkvv2_278_1_1
-rw-r----- 1 oracle 54331  932847616 Jun  1 20:24 8o1tkvv2_280_1_1
-rw-r----- 1 oracle 54331 1121984512 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-1_821tkrb8
-rw-r----- 1 oracle 54331 1027612672 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-3_831tkrd9
-rw-r----- 1 oracle 54331  429924352 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-9_8a1tkrhr
-rw-r----- 1 oracle 54331  707796992 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-4_851tkrgf
-rw-r----- 1 oracle 54331  534781952 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-14_871tkrhr
-rw-r----- 1 oracle 54331  534781952 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-18_881tkrhr
```

```
-rw-r----- 1 oracle 54331  429924352 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-13_8b1tkril
-rw-r----- 1 oracle 54331  429924352 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-17_8c1tkril
-rw-r----- 1 oracle 54331  246423552 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-15_8e1tkril
-rw-r----- 1 oracle 54331  246423552 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-19_8f1tkrj4
-rw-r----- 1 oracle 54331    5251072 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-USERS_FNO-7_8h1tkrj9
-rw-r----- 1 oracle 54331    5251072 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-USERS_FNO-16_8j1tkrja
-rw-r----- 1 oracle 54331    5251072 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-USERS_FNO-20_8k1tkrjb
-rw-r----- 1 oracle 54331    5251072 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-USERS_FNO-12_8i1tkrj9
-rw-r----- 1 oracle 54331  555753472 Jun  5 15:21 data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-10_861tkrgo
-rw-r----- 1 oracle 54331  796925952 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-11_841tkrf2
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-21_7j1tkqk6
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-34_801tkram
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-29_7r1tkr32
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-25_7n1tkqrh
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-31_7t1tkr3i
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-33_7v1tkra6
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-23_7l1tkqk6
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-27_7p1tkqrq
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-35_811tkrap
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-32_7u1tkr42
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-22_7k1tkqk6
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-24_7m1tkqk6
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-28_7q1tkqs1
```

```
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-30_7s1tkr3a
-rw-r----- 1 oracle 54331 4294975488 Jun  5 15:22 data_D-DB1_I-
1730530050_TS-SOE_FNO-26_7o1tkqrj
-rw-r----- 1 oracle 54331 1241432064 Jun  5 15:30 9d1tv06n_301_1_1
-rw-r----- 1 oracle 54331 1019805696 Jun  5 15:31 9a1tv06m_298_1_1
-rw-r----- 1 oracle 54331    4612096 Jun  5 15:31 9e1tv0ld_302_1_1
-rw-r----- 1 oracle 54331  967163904 Jun  5 15:31 9b1tv06n_299_1_1
-rw-r----- 1 oracle 54331   31563776 Jun  5 15:31 9g1tv0lt_304_1_1
-rw-r----- 1 oracle 54331     319488 Jun  5 15:31 9h1tv0lt_305_1_1
-rw-r----- 1 oracle 54331     335872 Jun  5 15:31 9i1tv0m0_306_1_1
-rw-r----- 1 oracle 54331     565248 Jun  5 15:31 9k1tv0m1_308_1_1
-rw-r----- 1 oracle 54331     581632 Jun  5 15:31 9l1tv0m5_309_1_1
-rw-r----- 1 oracle 54331   54345728 Jun  5 15:31 9f1tv0lt_303_1_1
-rw-r----- 1 oracle 54331     368640 Jun  5 15:31 9n1tv0m5_311_1_1
-rw-r----- 1 oracle 54331     385024 Jun  5 15:31 9o1tv0m6_312_1_1
-rw-r----- 1 oracle 54331  985858048 Jun  5 15:31 9c1tv06n_300_1_1
-rw-r----- 1 oracle 54331      57344 Jun  5 15:31 9q1tv0m7_314_1_1
-rw-r----- 1 oracle 54331      57344 Jun  5 15:31 9r1tv0m8_315_1_1
-rw-r----- 1 oracle 54331      57344 Jun  5 15:31 9s1tv0m9_316_1_1
-rw-r--r-- 1 oracle 54331      12720 Jun  5 15:31 db1_ctl.sql
-rw-r----- 1 oracle 54331   11600384 Jun  5 15:48 bct_db1.ctf
[ec2-user@ip-172-30-15-124 ~]$

[oracle@ip-172-30-15-124 ~]$ ls -l
/nfsfsxn/archlog/DB1/archivelog/2023_06_05
total 2008864
-rw-r----- 1 oracle 54331    729088 Jun  5 14:38
o1_mf_1_190_l7vwvvt9_.arc
-rw-r----- 1 oracle 54331 166651904 Jun  5 14:44
o1_mf_1_191_l7vx6vmg_.arc
-rw-r----- 1 oracle 54331 167406080 Jun  5 14:47
o1_mf_1_192_l7vxctms_.arc
-rw-r----- 1 oracle 54331 166868992 Jun  5 14:49
o1_mf_1_193_l7vxjjps_.arc
-rw-r----- 1 oracle 54331 166087168 Jun  5 14:52
o1_mf_1_194_l7vxnxrh_.arc
-rw-r----- 1 oracle 54331 175210496 Jun  5 14:54
o1_mf_1_195_l7vxswv5_.arc
-rw-r----- 1 oracle 54331 167078400 Jun  5 14:57
o1_mf_1_196_l7vxylwp_.arc
-rw-r----- 1 oracle 54331 169701888 Jun  5 14:59
o1_mf_1_197_l7vy3cyw_.arc
-rw-r----- 1 oracle 54331 167845376 Jun  5 15:02
o1_mf_1_198_l7vy8245_.arc
-rw-r----- 1 oracle 54331 170763776 Jun  5 15:05
```

```
o1_mf_1_199_l7vydv4c_.arc
-rw-r----- 1 oracle 54331 193853440 Jun  5 15:07
o1_mf_1_200_l7vykf23_.arc
-rw-r----- 1 oracle 54331 165523968 Jun  5 15:09
o1_mf_1_201_l7vyp1dh_.arc
-rw-r----- 1 oracle 54331 161117184 Jun  5 15:12
o1_mf_1_202_l7vyvrm5_.arc
-rw-r----- 1 oracle 54331  10098176 Jun  5 15:21
o1_mf_1_203_l7vzdfwm_.arc
```

8. 现在、恢复过程类似于以前在发生故障后恢复到新EC2数据库实例的用例—将Oracle环境(oratab、$oracle_home、$oracle_sid)设置为与主生产实例匹配、 创建一个init文件、其中包含db_recovery文件_dest和db_recovery文件_dest、该文件指向FSx NFS挂载上的闪存恢复目录。然后、通过lanuch RMAN运行恢复。以下是命令步骤和输出。

```
[oracle@ip-172-30-15-124 dbs]$ rman target / nocatalog

Recovery Manager: Release 19.0.0.0.0 - Production on Wed Jun 7
14:44:33 2023
Version 19.18.0.0.0

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

connected to target database (not started)

RMAN> startup nomount;

Oracle instance started

Total System Global Area   10737418000 bytes

Fixed Size                     9174800 bytes
Variable Size               1577058304 bytes
Database Buffers            9126805504 bytes
Redo Buffers                  24379392 bytes

RMAN> set dbid = 1730530050;

executing command: SET DBID

RMAN> restore controlfile from autobackup;

Starting restore at 07-JUN-23
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=2 device type=DISK
```

```
recovery area destination: /nfsfsxn/archlog/
database name (or database unique name) used for search: DB1
channel ORA_DISK_1: AUTOBACKUP
/nfsfsxn/archlog/DB1/autobackup/2023_06_05/o1_mf_s_1138721482_l7vzyb
vq_.bkp found in the recovery area
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20230607
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20230606
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20230605
channel ORA_DISK_1: restoring control file from AUTOBACKUP
/nfsfsxn/archlog/DB1/autobackup/2023_06_05/o1_mf_s_1138721482_l7vzyb
vq_.bkp
channel ORA_DISK_1: control file restore from AUTOBACKUP complete
output file name=/nfsfsxn/oracopy/db1.ctl
Finished restore at 07-JUN-23

RMAN> alter database mount;

released channel: ORA_DISK_1
Statement processed

RMAN> list incarnation;


List of Database Incarnations
DB Key  Inc Key DB Name  DB ID            STATUS  Reset SCN  Reset
Time
------- ------- -------- ---------------- --- ---------- ----------
1       1       DB1      1730530050       PARENT  1          17-APR-
19
2       2       DB1      1730530050       CURRENT 1920977    12-MAY-
23

RMAN> list copy of database tag 'OraCopyBKUPonFSxN_level_0';

List of Datafile Copies
========================

Key     File S Completion Time Ckp SCN    Ckp Time        Sparse
------- ---- - -------------- ---------- --------------- ------
362     1    A 05-JUN-23       8319160    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-1_821tkrb8
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

363     3    A 05-JUN-23       8319165    01-JUN-23       NO
```

```
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-3_831tkrd9
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

365     4    A 05-JUN-23       8319171    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-4_851tkrgf
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

355     5    A 01-JUN-23       2383520    12-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-5_8d1tkril
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

349     6    A 01-JUN-23       2383520    12-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-6_891tkrhr
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

372     7    A 05-JUN-23       8319201    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
7_8h1tkrj9
        Tag: ORACOPYBKUPONFSXN_LEVEL_0

361     8    A 01-JUN-23       2383520    12-MAY-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-8_8g1tkrj7
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 2, PDB Name: PDB$SEED

364     9    A 05-JUN-23       8318717    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-9_8a1tkrhr
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

376     10   A 05-JUN-23       8318714    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-10_861tkrgo
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

377     11   A 05-JUN-23       8318720    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
```

```
UNDOTBS1_FNO-11_841tkrf2
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

375     12   A 05-JUN-23       8318719    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
12_8i1tkrj9
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

368     13   A 05-JUN-23       8319184    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-13_8b1tkril
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

366     14   A 05-JUN-23       8319175    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-14_871tkrhr
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

370     15   A 05-JUN-23       8319193    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-15_8e1tkril
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

373     16   A 05-JUN-23       8319206    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
16_8j1tkrja
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 4, PDB Name: DB1_PDB2

369     17   A 05-JUN-23       8319188    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSTEM_FNO-17_8c1tkril
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

367     18   A 05-JUN-23       8319180    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
SYSAUX_FNO-18_881tkrhr
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3
```

```
371     19   A 05-JUN-23       8319197    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-19_8f1tkrj4
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

374     20   A 05-JUN-23       8319210    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
20_8k1tkrjb
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 5, PDB Name: DB1_PDB3

378     21   A 05-JUN-23       8318720    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
21_7j1tkqk6
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

388     22   A 05-JUN-23       8318714    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
22_7k1tkqk6
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

384     23   A 05-JUN-23       8318717    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
23_7l1tkqk6
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

389     24   A 05-JUN-23       8318719    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
24_7m1tkqk6
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

381     25   A 05-JUN-23       8318720    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
25_7n1tkqrh
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

392     26   A 05-JUN-23       8318714    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
26_7o1tkqrj
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
```

```
        Container ID: 3, PDB Name: DB1_PDB1

385     27   A 05-JUN-23       8318717    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
27_7p1tkqrq
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

390     28   A 05-JUN-23       8318719    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
28_7q1tkqs1
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

380     29   A 05-JUN-23       8318720    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
29_7r1tkr32
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

391     30   A 05-JUN-23       8318714    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
30_7s1tkr3a
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

382     31   A 05-JUN-23       8318717    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
31_7t1tkr3i
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

387     32   A 05-JUN-23       8318719    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
32_7u1tkr42
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

383     33   A 05-JUN-23       8318719    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
33_7v1tkra6
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

379     34   A 05-JUN-23       8318717    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
```

```
34_801tkram
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

386     35   A 05-JUN-23       8318714    01-JUN-23       NO
        Name: /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
35_811tkrap
        Tag: ORACOPYBKUPONFSXN_LEVEL_0
        Container ID: 3, PDB Name: DB1_PDB1

RMAN> switch database to copy;

datafile 1 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-1_821tkrb8"
datafile 3 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-3_831tkrd9"
datafile 4 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-4_851tkrgf"
datafile 5 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-5_8d1tkril"
datafile 6 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSAUX_FNO-6_891tkrhr"
datafile 7 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-USERS_FNO-7_8h1tkrj9"
datafile 8 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-UNDOTBS1_FNO-8_8g1tkrj7"
datafile 9 switched to datafile copy "/nfsfsxn/oracopy/data_D-DB1_I-
1730530050_TS-SYSTEM_FNO-9_8a1tkrhr"
datafile 10 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSAUX_FNO-10_861tkrgo"
datafile 11 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-UNDOTBS1_FNO-11_841tkrf2"
datafile 12 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-12_8i1tkrj9"
datafile 13 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSTEM_FNO-13_8b1tkril"
datafile 14 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSAUX_FNO-14_871tkrhr"
datafile 15 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-UNDOTBS1_FNO-15_8e1tkril"
datafile 16 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-16_8j1tkrja"
datafile 17 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSTEM_FNO-17_8c1tkril"
datafile 18 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SYSAUX_FNO-18_881tkrhr"
```

```
datafile 19 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-UNDOTBS1_FNO-19_8f1tkrj4"
datafile 20 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-USERS_FNO-20_8k1tkrjb"
datafile 21 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-21_7j1tkqk6"
datafile 22 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-22_7k1tkqk6"
datafile 23 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-23_7l1tkqk6"
datafile 24 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-24_7m1tkqk6"
datafile 25 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-25_7n1tkqrh"
datafile 26 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-26_7o1tkqrj"
datafile 27 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-27_7p1tkqrq"
datafile 28 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-28_7q1tkqs1"
datafile 29 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-29_7r1tkr32"
datafile 30 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-30_7s1tkr3a"
datafile 31 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-31_7t1tkr3i"
datafile 32 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-32_7u1tkr42"
datafile 33 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-33_7v1tkra6"
datafile 34 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-34_801tkram"
datafile 35 switched to datafile copy "/nfsfsxn/oracopy/data_D-
DB1_I-1730530050_TS-SOE_FNO-35_811tkrap"

RMAN> run {
2> set until sequence 204;
3> recover database;
4> }

executing command: SET until clause

Starting recover at 07-JUN-23
using channel ORA_DISK_1

starting media recovery
```

```
archived log for thread 1 with sequence 190 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_190_l7vwvvt9_.arc
archived log for thread 1 with sequence 191 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_191_l7vx6vmg_.arc
archived log for thread 1 with sequence 192 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_192_l7vxctms_.arc
archived log for thread 1 with sequence 193 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_193_l7vxjjps_.arc
archived log for thread 1 with sequence 194 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_194_l7vxnxrh_.arc
archived log for thread 1 with sequence 195 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_195_l7vxswv5_.arc
archived log for thread 1 with sequence 196 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_196_l7vxylwp_.arc
archived log for thread 1 with sequence 197 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_197_l7vy3cyw_.arc
archived log for thread 1 with sequence 198 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_198_l7vy8245_.arc
archived log for thread 1 with sequence 199 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_199_l7vydv4c_.arc
archived log for thread 1 with sequence 200 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_200_l7vykf23_.arc
archived log for thread 1 with sequence 201 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_201_l7vyp1dh_.arc
archived log for thread 1 with sequence 202 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_202_l7vyvrm5_.arc
archived log for thread 1 with sequence 203 is already on disk as
file
/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_203_l7vzdfwm_.arc
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_190_l7vwvvt9
_.arc thread=1 sequence=190
archived log file
```

```
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_191_l7vx6vmg
_.arc thread=1 sequence=191
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_192_l7vxctms
_.arc thread=1 sequence=192
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_193_l7vxjjps
_.arc thread=1 sequence=193
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_194_l7vxnxrh
_.arc thread=1 sequence=194
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_195_l7vxswv5
_.arc thread=1 sequence=195
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_196_l7vxylwp
_.arc thread=1 sequence=196
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_197_l7vy3cyw
_.arc thread=1 sequence=197
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_198_l7vy8245
_.arc thread=1 sequence=198
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_199_l7vydv4c
_.arc thread=1 sequence=199
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_200_l7vykf23
_.arc thread=1 sequence=200
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_201_l7vyp1dh
_.arc thread=1 sequence=201
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_202_l7vyvrm5
_.arc thread=1 sequence=202
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_203_l7vzdfwm
_.arc thread=1 sequence=203
media recovery complete, elapsed time: 00:19:30
Finished recover at 07-JUN-23

RMAN> exit

Recovery Manager complete.
[oracle@ip-172-30-15-124 dbs]$ sqlplus / as sysdba
```

```
SQL*Plus: Release 19.0.0.0.0 - Production on Wed Jun 7 15:58:12 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 member from v$logfile;

MEMBER
--------------------------------------------------------------------------
------------
+DATA/DB1/ONLINELOG/group_3.264.1136666437
+DATA/DB1/ONLINELOG/group_2.263.1136666437
+DATA/DB1/ONLINELOG/group_1.262.1136666437

SQL> alter database rename file
'+DATA/DB1/ONLINELOG/group_1.262.1136666437' to
'/nfsfsxn/oracopy/redo01.log';

Database altered.

SQL> alter database rename file
'+DATA/DB1/ONLINELOG/group_2.263.1136666437' to
'/nfsfsxn/oracopy/redo02.log';

Database altered.

SQL> alter database rename file
'+DATA/DB1/ONLINELOG/group_3.264.1136666437' to
'/nfsfsxn/oracopy/redo03.log';

Database altered.

SQL> alter database noarchivelog;

Database altered.

SQL> alter database open resetlogs;

Database altered.

SQL> set lin 200;
```

```
SQL> select name from v$datafile
  2  union
  3  select name from v$controlfile
  4  union
  5  select name from v$tempfile
  6  union
  7  select member from v$logfile;

NAME
--------------------------------------------------------------------------------
--------------------
/nfsfsxn/oracopy/DB1/FB864A929AEB79B9E053630F1EAC7046/datafile/o1_mf
_temp_l81bhz6g_.tmp
/nfsfsxn/oracopy/DB1/FB867DA8C68C816EE053630F1EAC2BCF/datafile/o1_mf
_temp_l81bj16t_.tmp
/nfsfsxn/oracopy/DB1/FB867EA89ECF81C0E053630F1EACB901/datafile/o1_mf
_temp_l81bj135_.tmp
/nfsfsxn/oracopy/DB1/FB867F8A4D4F821CE053630F1EAC69CC/datafile/o1_mf
_temp_l81bj13g_.tmp
/nfsfsxn/oracopy/DB1/datafile/o1_mf_temp_l81bhwjg_.tmp
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-21_7j1tkqk6
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-22_7k1tkqk6
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-23_7l1tkqk6
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-24_7m1tkqk6
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-25_7n1tkqrh
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-26_7o1tkqrj

NAME
--------------------------------------------------------------------------------
--------------------
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-27_7p1tkqrq
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-28_7q1tkqs1
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-29_7r1tkr32
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-30_7s1tkr3a
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-31_7t1tkr3i
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-32_7u1tkr42
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-33_7v1tkra6
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-34_801tkram
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-35_811tkrap
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-10_861tkrgo
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-14_871tkrhr

NAME
--------------------------------------------------------------------------------
--------------------
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-18_881tkrhr
```

```
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-3_831tkrd9
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-6_891tkrhr
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-13_8b1tkril
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-17_8c1tkril
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-1_821tkrb8
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-5_8d1tkril
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-9_8a1tkrhr
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-UNDOTBS1_FNO-11_841tkrf2
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-UNDOTBS1_FNO-15_8e1tkril
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-UNDOTBS1_FNO-19_8f1tkrj4

NAME
--------------------------------------------------------------------------
--------------------
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-UNDOTBS1_FNO-4_851tkrgf
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-UNDOTBS1_FNO-8_8g1tkrj7
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-12_8i1tkrj9
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-16_8j1tkrja
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-20_8k1tkrjb
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-7_8h1tkrj9
/nfsfsxn/oracopy/db1.ctl
/nfsfsxn/oracopy/redo01.log
/nfsfsxn/oracopy/redo02.log
/nfsfsxn/oracopy/redo03.log

43 rows selected.

SQL> show pdbs;

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 DB1_PDB1                       READ WRITE NO
         4 DB1_PDB2                       READ WRITE NO
         5 DB1_PDB3                       READ WRITE NO
SQL> alter session set container=db1_pdb1;

Session altered.

SQL> select * from test;

        ID DT
EVENT
----------
--------------------------------------------------------------------------
-------
```

```
-------------------------------------------------------------------
         1 18-MAY-23 02.35.37.000000 PM
test oracle incremental merge switch to copy
         2 30-MAY-23 05.23.11.000000 PM
test recovery on a new EC2 instance host with image copy on FSxN
         3 05-JUN-23 03.19.46.000000 PM
test clone on a new EC2 instance host with image copy on FSxN

SQL>
```

9. 使用Oracle nid实用程序重命名克隆的数据库实例并更改数据库ID。数据库实例需要处于状态 `mount` 以执行命令。

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

NAME      OPEN_MODE            LOG_MODE
--------- -------------------- ------------
DB1       READ WRITE           NOARCHIVELOG

SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.

SQL> startup mount;
ORACLE instance started.

Total System Global Area 1.0737E+10 bytes
Fixed Size                  9174800 bytes
Variable Size            1577058304 bytes
Database Buffers         9126805504 bytes
Redo Buffers               24379392 bytes
Database mounted.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release
19.0.0.0.0 - Production
Version 19.18.0.0.0
[oracle@ip-172-30-15-124 dbs]$ nid target=/ dbname=db1tst

DBNEWID: Release 19.0.0.0.0 - Production on Wed Jun 7 16:15:14 2023

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

Connected to database DB1 (DBID=1730530050)
```

```
Connected to server version 19.18.0

Control Files in database:
    /nfsfsxn/oracopy/db1.ctl

Change database ID and database name DB1 to DB1TST? (Y/[N]) => Y

Proceeding with operation
Changing database ID from 1730530050 to 3054879890
Changing database name from DB1 to DB1TST
    Control File /nfsfsxn/oracopy/db1.ctl - modified
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-
1_821tkrb - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-
3_831tkrd - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-4_851tkrg - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-
5_8d1tkri - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-
6_891tkrh - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
7_8h1tkrj - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-8_8g1tkrj - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-
9_8a1tkrh - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-
10_861tkrg - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-11_841tkrf - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
12_8i1tkrj - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-
13_8b1tkri - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-
14_871tkrh - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
UNDOTBS1_FNO-15_8e1tkri - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
16_8j1tkrj - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSTEM_FNO-
17_8c1tkri - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SYSAUX_FNO-
18_881tkrh - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-
```

```
UNDOTBS1_FNO-19_8f1tkrj - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-USERS_FNO-
20_8k1tkrj - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
21_7j1tkqk - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
22_7k1tkqk - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
23_7l1tkqk - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
24_7m1tkqk - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
25_7n1tkqr - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
26_7o1tkqr - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
27_7p1tkqr - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
28_7q1tkqs - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
29_7r1tkr3 - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
30_7s1tkr3 - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
31_7t1tkr3 - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
32_7u1tkr4 - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
33_7v1tkra - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
34_801tkra - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-
35_811tkra - dbid changed, wrote new name
    Datafile /nfsfsxn/oracopy/DB1/datafile/o1_mf_temp_l81bhwjg_.tm -
dbid changed, wrote new name
    Datafile
/nfsfsxn/oracopy/DB1/FB864A929AEB79B9E053630F1EAC7046/datafile/o1_mf
_temp_l81bhz6g_.tm - dbid changed, wrote new name
    Datafile
/nfsfsxn/oracopy/DB1/FB867DA8C68C816EE053630F1EAC2BCF/datafile/o1_mf
_temp_l81bj16t_.tm - dbid changed, wrote new name
    Datafile
/nfsfsxn/oracopy/DB1/FB867EA89ECF81C0E053630F1EACB901/datafile/o1_mf
_temp_l81bj135_.tm - dbid changed, wrote new name
    Datafile
/nfsfsxn/oracopy/DB1/FB867F8A4D4F821CE053630F1EAC69CC/datafile/o1_mf
```

```
_temp_l81bj13g_.tm - dbid changed, wrote new name
    Control File /nfsfsxn/oracopy/db1.ctl - dbid changed, wrote new
name
    Instance shut down

Database name changed to DB1TST.
Modify parameter file and generate a new password file before
restarting.
Database ID for database DB1TST changed to 3054879890.
All previous backups and archived redo logs for this database are
unusable.
Database is not aware of previous backups and archived logs in
Recovery Area.
Database has been shutdown, open database with RESETLOGS option.
Succesfully changed database name and ID.
DBNEWID - Completed succesfully.
```

10. 在oratab、init文件中将Oracle数据库环境配置更改为新的数据库名称或实例ID、并创建与新实例ID匹配的必要管理目录。然后，使用resetlogs"选项启动实例。

```
SQL> startup mount;
ORACLE instance started.

Total System Global Area 1.0737E+10 bytes
Fixed Size                  9174800 bytes
Variable Size            1577058304 bytes
Database Buffers         9126805504 bytes
Redo Buffers               24379392 bytes
Database mounted.
SQL> alter database open resetlogs;

Database altered.

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

NAME      OPEN_MODE            LOG_MODE
--------- -------------------- ------------
DB1TST    READ WRITE           NOARCHIVELOG

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 DB1_PDB1                       MOUNTED
         4 DB1_PDB2                       MOUNTED
         5 DB1_PDB3                       MOUNTED
SQL> alter pluggable database all open;

Pluggable database altered.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 DB1_PDB1                       READ WRITE NO
         4 DB1_PDB2                       READ WRITE NO
         5 DB1_PDB3                       READ WRITE NO
SQL>
```

这样、就可以在开发、UAT或任何其他使用情形下、通过FSx NFS挂载上的暂存数据库副本克隆新的Oracle实例。可以从同一暂存映像副本克隆多个Oracle实例。

遇到错误时 `RMAN-06571: datafile 1 does not have recoverable copy` 在将数据库切换为副本时、请检查与主生产数据库匹配的数据库配置。如果需要、请使用RMAN命令重置此转存方式、使其与主服务器匹配 `reset database to incarnation n;`。

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- RMAN：合并增量备份策略(文档ID 745798.1)

  "https://support.oracle.com/knowledge/Oracle%20Database%20Products/745798_1.html"

- RMAN备份和恢复用户指南

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/bradv/getting-started-rman.html"

- 适用于 NetApp ONTAP 的 Amazon FSX

  "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

TR-4974：《Oracle 19c在使用NFS/ASM的AWS FSX/EC2上独立重新启动》

NetApp公司Allen Cao、Niyaz Mohamed

目的

ASM (自动存储管理)是一种受欢迎的Oracle存储卷管理器、广泛应用于许多Oracle安装。这也是Oracle建议的存储管理解决方案。它提供了传统卷管理器和文件系统的替代方案。自Oracle 11g以来、ASM已随网格基础架构打包、而不是数据库打包在一起。因此、要在不使用RAC的情况下使用Oracle ASM进行存储管理、您必须在独立服务器中安装Oracle网格基础架构、也称为Oracle Restart。这样做确实会在以其他方式简化Oracle数据库部署过程中增加更多的复杂性。但是、顾名思义、在重新启动模式下部署Oracle时、任何出现故障的Oracle服务都会在主机重新启动后重新启动、而无需用户干预、从而提供一定程度的高可用性或HA功能。

Oracle ASM通常部署在FC、iSCSI存储协议和LUN中作为原始存储设备。但是、Oracle也支持基于NFS协议和NFS文件系统的ASM配置。在本文档中、我们将演示如何在使用EC2计算实例的Amazon FSx for ONTAP 存储环境中部署采用NFS协议和Oracle ASM的Oracle 19c数据库。我们还演示了如何通过NetApp BlueXP控制台使用NetApp SnapCenter 服务来备份、还原和克隆Oracle数据库、以便在AWS公共云中进行开发/测试或其他存储效率数据库操作用例。

此解决方案 可解决以下使用情形：

- 在Amazon FSx中部署Oracle数据库、用于使用NFS/ASM的ONTAP 存储和EC2计算实例
- 使用NFS/ASM在公共AWS云中测试和验证Oracle工作负载
- 测试和验证AWS中部署的Oracle数据库重新启动功能

audience

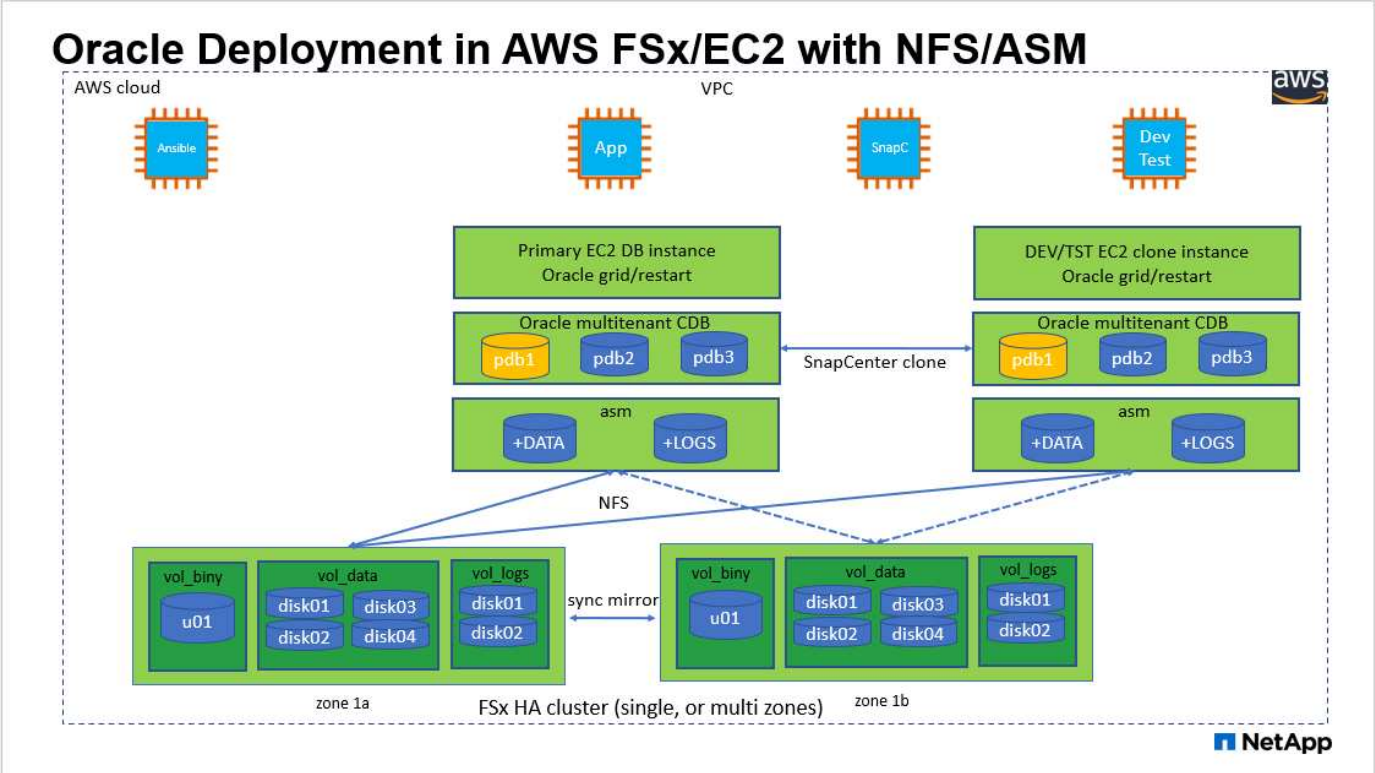
此解决方案 适用于以下人员：

- 希望使用NFS/ASM在AWS公共云中部署Oracle的数据库助理。
- 一名数据库解决方案 架构师、希望在AWS公共云中测试Oracle工作负载。
- 希望部署和管理部署到AWS FSX存储的Oracle数据库的存储管理员。
- 希望在AWS FSX/EC2中建立Oracle数据库的应用程序所有者。

解决方案 测试和验证环境

此解决方案 的测试和验证是在AWS FSX和EC2环境中执行的、该环境可能与最终部署环境不匹配。有关详细信息，请参见一节 [Key Factors for Deployment Consideration]。

架构



硬件和软件组件

* 硬件 *		
FSX ONTAP 存储	AWS提供的当前版本	一个FSX HA集群位于同一VPC和可用性区域中

用于计算的EC2实例	t2.xlarge/4vCPU/16G	两个EC2 t2 xlarge EC2实例、一个用作主数据库服务器、另一个用作克隆数据库服务器
软件		
RedHat Linux	rhel-8.6.0_hvm-20220503-x86_64-2-Hourly2-gp2	已部署RedHat订阅以进行测试
Oracle网格基础架构	版本19.18	已应用RU修补程序p34762026_190000_Linux-x86-64.zip
Oracle 数据库	版本19.18	已应用RU修补程序p34765931_190000_Linux-x86-64.zip
Oracle OPatch	版本12.2.0.1.36	最新修补程序p6880880_190000_Linux-x86-64.zip
SnapCenter 服务	version	v2.3.1.2324

部署注意事项的关键因素

- * EC2计算实例。*在这些测试和验证中、我们对Oracle数据库计算实例使用了AWS EC2 T2.xlarge实例类型。NetApp建议在生产部署中使用M5类型的EC2实例作为Oracle的计算实例、因为它针对数据库工作负载进行了优化。您需要根据实际工作负载要求根据vCPU数量和RAM量适当调整EC2实例的大小。
- * FSX存储HA集群单区域或多区域部署。*在这些测试和验证中、我们在一个AWS可用性区域中部署了一个FSX HA集群。对于生产部署、NetApp建议在两个不同的可用性区域中部署一个FSX HA对。FSX HA集群始终配置在一个HA对中、该HA对在一对主动-被动文件系统中进行同步镜像、以提供存储级别的冗余。多区域部署可在单个AWS区域发生故障时进一步提高高可用性。
- * FSX存储集群规模估算。*适用于ONTAP 存储文件系统的Amazon FSX可提供高达160、000个原始SSD IOPS、高达4 Gbps吞吐量以及最大192 TiB容量。但是、您可以根据部署时的实际要求、根据已配置的IOPS、吞吐量和存储限制(最小1、024 GiB)来调整集群的大小。可以动态调整容量、而不会影响应用程序可用性。
- * Oracle数据和日志布局。*在测试和验证中、我们分别为数据和日志部署了两个ASM磁盘组。在+data ASM磁盘组中、我们在一个数据NFS文件系统挂载点配置了四个磁盘。在+logs ASM磁盘组中、我们在日志NFS文件系统挂载点配置了两个磁盘。对于大型数据库部署、可以构建ASM磁盘组、使其跨越多个FSx文件系统、并通过固定在FSx文件系统上的多个NFS挂载点分布ASM NFS磁盘。此特定设置旨在满足超过4 Gbps吞吐量和16万次原始SSD IOPS的数据库吞吐量要求。
- **DNFS**配置。 DNFS内置在Oracle内核中、众所周知、在将Oracle部署到NFS存储时、它可以显著提高Oracle数据库性能。DNFS打包到Oracle二进制文件中、但默认情况下不启用。对于NFS上的任何Oracle数据库部署、都应启用此功能。对于大型数据库的多FSx文件系统部署、应正确配置DNFS多路径。
- *要用于创建的每个Oracle ASM磁盘组的Oracle ASM冗余级别。*由于FSx已在FSx集群级别镜像存储、因此您应执行此操作 `ONLY` 使用外部冗余、这意味着该选项不允许Oracle ASM镜像磁盘组的内容。这一点尤其重要、因为用于Oracle数据库数据存储的NFS需要硬NFS挂载选项、而在Oracle级别镜像ASM内容时、这种方法并不理想。
- 数据库备份。 NetApp提供SaaS版本的SnapCenter 软件服务、用于在云中备份、还原和克隆数据库、该服务可通过NetApp BlueXP控制台UI访问。NetApp建议实施此类服务、以实现快速(不到一分钟)的快照备份、快速(几分钟)的数据库还原和数据库克隆。

解决方案 部署

下一节介绍了分步部署过程。

部署的前提条件

部署需要满足以下前提条件。

1. 已设置AWS帐户、并已在您的AWS帐户中创建必要的VPC和网段。
2. 在AWS EC2控制台中、您必须部署两个EC2 Linux实例、一个用作主Oracle数据库服务器、另一个用作可选的克隆目标数据库服务器。有关环境设置的详细信息、请参见上一节中的架构图。另请查看 "Linux实例用户指南" 有关详细信息 …
3. 在AWS EC2控制台中、部署适用于ONTAP 存储HA集群的Amazon FSX以托管Oracle数据库卷。如果您不熟悉FSX存储的部署、请参见相关文档 "为ONTAP 文件系统创建FSX" 了解分步说明。
4. 可以使用以下Terraform自动化工具包执行步骤2和步骤3、该工具包会创建一个名为的EC2实例 `ora_01` 和名为的FSX文件系统 `fsx_01`。执行前、请仔细阅读该说明并根据您的环境更改变量。

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

确保您已在EC2实例根卷中至少分配50G、以便有足够的空间来暂存Oracle安装文件。

EC2实例内核配置

在配置了前提条件的情况下、以EC2-user身份登录到EC2实例、并使用sudo to root user配置Linux内核以进行Oracle安装。

1. 创建暂存目录 `/tmp/archive` 文件夹并设置 `777` 权限。

```
mkdir /tmp/archive

chmod 777 /tmp/archive
```

2. 将Oracle二进制安装文件和其他所需的rpm文件下载并暂存到 `/tmp/archive` 目录。

   请参见以下要在中说明的安装文件列表 `/tmp/archive` 在EC2实例上。

```
[ec2-user@ip-172-30-15-58 ~]$ ls -l /tmp/archive
total 10537316
-rw-rw-r--. 1 ec2-user ec2-user      19112 Mar 21 15:57 compat-
libcap1-1.10-7.el7.x86_64.rpm
-rw-rw-r--  1 ec2-user ec2-user 3059705302 Mar 21 22:01
LINUX.X64_193000_db_home.zip
-rw-rw-r--  1 ec2-user ec2-user 2889184573 Mar 21 21:09
LINUX.X64_193000_grid_home.zip
-rw-rw-r--. 1 ec2-user ec2-user     589145 Mar 21 15:56
netapp_linux_unified_host_utilities-7-1.x86_64.rpm
-rw-rw-r--. 1 ec2-user ec2-user      31828 Mar 21 15:55 oracle-
database-preinstall-19c-1.0-2.el8.x86_64.rpm
-rw-rw-r--  1 ec2-user ec2-user 2872741741 Mar 21 22:31
p34762026_190000_Linux-x86-64.zip
-rw-rw-r--  1 ec2-user ec2-user 1843577895 Mar 21 22:32
p34765931_190000_Linux-x86-64.zip
-rw-rw-r--  1 ec2-user ec2-user  124347218 Mar 21 22:33
p6880880_190000_Linux-x86-64.zip
-rw-r--r--  1 ec2-user ec2-user     257136 Mar 22 16:25
policycoreutils-python-utils-2.9-9.el8.noarch.rpm
```

3. 安装Oracle 19c预安装RPM、以满足大多数内核配置要求。

```
yum install /tmp/archive/oracle-database-preinstall-19c-1.0-
2.el8.x86_64.rpm
```

4. 下载并安装缺少的 `compat-libcap1` 在Linux 8中。

```
yum install /tmp/archive/compat-libcap1-1.10-7.el7.x86_64.rpm
```

5. 从NetApp下载并安装NetApp主机实用程序。

```
yum install /tmp/archive/netapp_linux_unified_host_utilities-7-
1.x86_64.rpm
```

6. 安装 `policycoreutils-python-utils`、在EC2实例中不可用。

```
yum install /tmp/archive/policycoreutils-python-utils-2.9-
9.el8.noarch.rpm
```

7. 安装Open JDK 1.8版。

```
yum install java-1.8.0-openjdk.x86_64
```

8. 安装NFS-utils。

```
yum install nfs-utils
```

9. 在当前系统中禁用透明页面。

```
echo never > /sys/kernel/mm/transparent_hugepage/enabled
echo never > /sys/kernel/mm/transparent_hugepage/defrag
```

在中添加以下行 `/etc/rc.local` 以禁用 `transparent_hugepage` 重新启动后：

```
  # Disable transparent hugepages
          if test -f /sys/kernel/mm/transparent_hugepage/enabled;
then
            echo never > /sys/kernel/mm/transparent_hugepage/enabled
          fi
          if test -f /sys/kernel/mm/transparent_hugepage/defrag;
then
            echo never > /sys/kernel/mm/transparent_hugepage/defrag
          fi
```

10. 通过更改禁用SELinux `SELINUX=enforcing` to `SELINUX=disabled`。要使更改生效、必须重新启动主机。

```
vi /etc/sysconfig/selinux
```

11. 将以下行添加到 `limit.conf` 设置不带引号的文件描述符限制和堆栈大小 " "。

```
vi /etc/security/limits.conf
  "*               hard    nofile           65536"
  "*               soft    stack            10240"
```

12. 按照以下说明向EC2实例添加交换空间： "如何使用交换文件分配内存以用作Amazon EC2实例中的交换空间？" 要添加的确切空间量取决于RAM大小、最高可达16G。
13. 添加要用于ASM sysasm组的ASM组

```
groupadd asm
```

14. 修改Oracle用户以将ASM添加为二级组(Oracle用户应在安装Oracle预安装RPM后创建)。

```
usermod -a -G asm oracle
```

15. 重新启动EC2实例。

配置和导出要挂载到**EC2**实例主机的**NFS**卷

使用FSx集群管理IP通过ssh以fsxadmin用户身份登录FSx集群、从命令行配置三个卷、以托管Oracle数据库二进制文件、数据和日志文件。

1. 以fsxadmin用户身份通过SSH登录到FSX集群。

```
ssh fsxadmin@172.30.15.53
```

2. 执行以下命令为Oracle二进制文件创建卷。

```
vol create -volume ora_01_biny -aggregate aggr1 -size 50G -state
online  -type RW -junction-path /ora_01_biny -snapshot-policy none
-tiering-policy snapshot-only
```

3. 执行以下命令为Oracle数据创建卷。

```
vol create -volume ora_01_data -aggregate aggr1 -size 100G -state
online  -type RW -junction-path /ora_01_data -snapshot-policy none
-tiering-policy snapshot-only
```

4. 执行以下命令为Oracle日志创建卷。

```
vol create -volume ora_01_logs -aggregate aggr1 -size 100G -state
online  -type RW -junction-path /ora_01_logs -snapshot-policy none
-tiering-policy snapshot-only
```

5. 验证已创建的数据库卷。

```
vol show
```

这将返回：

```
FsxId02ad7bf3476b741df::> vol show
  (vol show)
FsxId06c3c8b2a7bd56458::> vol show
Vserver   Volume       Aggregate    State      Type       Size
Available Used%
--------- ------------ ------------ ---------- ---- ----------
---------- -----
svm_ora   ora_01_biny  aggr1        online     RW         50GB
47.50GB    0%
svm_ora   ora_01_data  aggr1        online     RW        100GB
95.00GB    0%
svm_ora   ora_01_logs  aggr1        online     RW        100GB
95.00GB    0%
svm_ora   svm_ora_root aggr1        online     RW          1GB
972.1MB    0%
4 entries were displayed.
```

数据库存储配置

现在、为EC2实例主机上的Oracle网格基础架构和数据库安装导入并设置FSX存储。

1. 以EC2用户身份使用SSH密钥和EC2实例IP地址通过SSH登录到EC2实例。

```
ssh -i ora_01.pem ec2-user@172.30.15.58
```

2. 创建/u01目录以挂载Oracle二进制文件系统

```
sudo mkdir /u01
```

3. 将二进制卷挂载到 `/u01`，已更改为FSx NFS lip地址。如果您已通过NetApp自动化工具包部署FSx集群、则在资源配置执行结束时、输出中将列出FSx虚拟存储服务器NFS lif IP地址。否则、可以从AWS FSx控制台UI中检索。

```
sudo mount -t nfs 172.30.15.19:/ora_01_biny /u01 -o
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
```

4. 更改 `/u01` Oracle用户及其关联主组的挂载点所有权。

```
sudo chown oracle:oinstall /u01
```

5. 创建/oradATA目录以挂载Oracle数据文件系统

```
sudo mkdir /oradata
```

6. 将数据卷挂载到 `/oradata`，已更改为FSx NFS lip地址

```
sudo mount -t nfs 172.30.15.19:/ora_01_data /oradata -o
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
```

7. 更改 `/oradata` Oracle用户及其关联主组的挂载点所有权。

```
sudo chown oracle:oinstall /oradata
```

8. 创建/orlogs目录以挂载Oracle日志文件系统

```
sudo mkdir /oralogs
```

9. 将日志卷挂载到 `/oralogs`，已更改为FSx NFS lip地址

```
sudo mount -t nfs 172.30.15.19:/ora_01_logs /oralogs -o
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
```

10. 更改 `/oralogs` Oracle用户及其关联主组的挂载点所有权。

```
sudo chown oracle:oinstall /oralogs
```

11. 将挂载点添加到 `/etc/fstab`。

```
sudo vi /etc/fstab
```

添加以下行。

```
172.30.15.19:/ora_01_biny       /u01            nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536   0
0
172.30.15.19:/ora_01_data       /oradata        nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536   0
0
172.30.15.19:/ora_01_logs       /oralogs        nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536   0
0
```

12. sudo to Oracle user、创建ASM文件夹以存储ASM磁盘文件

```
sudo su
su - oracle
mkdir /oradata/asm
mkdir /oralogs/asm
```

13. 作为Oracle用户、创建ASM数据磁盘文件、将计数更改为与具有块大小的磁盘大小匹配。

```
dd if=/dev/zero of=/oradata/asm/nfs_data_disk01 bs=1M count=20480
oflag=direct
dd if=/dev/zero of=/oradata/asm/nfs_data_disk02 bs=1M count=20480
oflag=direct
dd if=/dev/zero of=/oradata/asm/nfs_data_disk03 bs=1M count=20480
oflag=direct
dd if=/dev/zero of=/oradata/asm/nfs_data_disk04 bs=1M count=20480
oflag=direct
```

14. 作为root用户、将数据磁盘文件权限更改为640

```
chmod 640 /oradata/asm/*
```

15. 作为Oracle用户、创建ASM日志磁盘文件、并更改为计数、以便与具有块大小的磁盘大小匹配。

```
dd if=/dev/zero of=/oralogs/asm/nfs_logs_disk01 bs=1M count=40960
oflag=direct
dd if=/dev/zero of=/oralogs/asm/nfs_logs_disk02 bs=1M count=40960
oflag=direct
```

16. 作为root用户、更改会将磁盘文件权限记录为640

```
chmod 640 /oralogs/asm/*
```

17. 重新启动EC2实例主机。

Oracle网格基础架构安装

1. 通过SSH以EC2-user身份登录到EC2实例、并通过取消注释启用密码身份验证 `PasswordAuthentication yes` 然后进行注释 `PasswordAuthentication no`。

```
sudo vi /etc/ssh/sshd_config
```

2. 重新启动sshd服务。

```
sudo systemctl restart sshd
```

3. 重置Oracle用户密码。

```
sudo passwd oracle
```

4. 以Oracle Restart软件所有者用户(Oracle)身份登录。按如下所示创建Oracle目录：

```
mkdir -p /u01/app/oracle
mkdir -p /u01/app/oraInventory
```

5. 更改目录权限设置。

```
chmod -R 775 /u01/app
```

6. 创建网格主目录并进行更改。

```
mkdir -p /u01/app/oracle/product/19.0.0/grid
cd /u01/app/oracle/product/19.0.0/grid
```

7. 解压缩网格安装文件。

```
unzip -q /tmp/archive/LINUX.X64_193000_grid_home.zip
```

8. 从网格主页中、删除 `OPatch` 目录。

```
rm -rf OPatch
```

9. 从网格主页复制 `p6880880_190000_Linux-x86-64.zip` 到grid_home、然后将其解压缩。

```
cp /tmp/archive/p6880880_190000_Linux-x86-64.zip .
unzip p6880880_190000_Linux-x86-64.zip
```

10. 从网格主页修改 `cv/admin/cvu_config`、取消注释并替换 `CV_ASSUME_DISTID=OEL5` 使用 `CV_ASSUME_DISTID=OL7`。

```
vi cv/admin/cvu_config
```

11. 准备 `gridsetup.rsp` 文件以进行静默安装、并将rsp文件置于中 `/tmp/archive` 目录。rsp文件应涵盖A、B和G部分、并提供以下信息：

```
INVENTORY_LOCATION=/u01/app/oraInventory
oracle.install.option=HA_CONFIG
ORACLE_BASE=/u01/app/oracle
oracle.install.asm.OSDBA=dba
oracle.install.asm.OSOPER=oper
oracle.install.asm.OSASM=asm
oracle.install.asm.SYSASMPassword="SetPWD"
oracle.install.asm.diskGroup.name=DATA
oracle.install.asm.diskGroup.redundancy=EXTERNAL
oracle.install.asm.diskGroup.AUSize=4
oracle.install.asm.diskGroup.disks=/oradata/asm/*,/oralogs/asm/*
oracle.install.asm.diskGroup.diskDiscoveryString=/oradata/asm/nfs_da
ta_disk01,/oradata/asm/nfs_data_disk02,/oradata/asm/nfs_data_disk03,
/oradata/asm/nfs_data_disk04
oracle.install.asm.monitorPassword="SetPWD"
oracle.install.asm.configureAFD=false
```

12. 以root用户身份登录到EC2实例。
13. 安装 `cvuqdisk-1.0.10-1.rpm`。

```
rpm -ivh /u01/app/oracle/product/19.0.0/grid/cv/rpm/cvuqdisk-1.0.10-
1.rpm
```

14. 以Oracle用户身份登录到EC2实例、然后在中提取修补程序 `/tmp/archive` 文件夹。

```
unzip p34762026_190000_Linux-x86-64.zip
```

15. 从Grid home /u01/app/oracle/product/19.0.0/grid中、以Oracle用户身份启动 `gridSetup.sh` 用于网格基础架构安装。

```
 ./gridSetup.sh -applyRU /tmp/archive/34762026/ -silent
-responseFile /tmp/archive/gridsetup.rsp
```

忽略有关网格基础架构的错误组的警告。我们正在使用一个Oracle用户来管理Oracle Restart、因此这是预期的。

16. 以root用户身份执行以下脚本：

```
/u01/app/oraInventory/orainstRoot.sh

/u01/app/oracle/product/19.0.0/grid/root.sh
```

17. 以Oracle用户身份执行以下命令以完成配置：

```
/u01/app/oracle/product/19.0.0/grid/gridSetup.sh -executeConfigTools
-responseFile /tmp/archive/gridsetup.rsp -silent
```

18. 以Oracle用户身份创建日志磁盘组。

```
bin/asmca -silent -sysAsmPassword 'yourPWD' -asmsnmpPassword
'yourPWD' -createDiskGroup -diskGroupName LOGS -disk
'/oralogs/asm/nfs_logs_disk*' -redundancy EXTERNAL -au_size 4
```

19. 以Oracle用户身份、在安装配置后验证网格服务。

```
bin/crsctl stat res -t
+
Name                 Target  State        Server
State details
Local Resources
ora.DATA.dg          ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.LISTENER.lsnr    ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.LOGS.dg          ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.asm              ONLINE  ONLINE       ip-172-30-15-58
Started,STABLE
ora.ons              OFFLINE OFFLINE      ip-172-30-15-58
STABLE
Cluster Resources
ora.cssd             ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.diskmon          OFFLINE OFFLINE
STABLE
ora.driver.afd       ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.evmd             ONLINE  ONLINE       ip-172-30-15-58
STABLE
```

Oracle数据库安装

1. 以Oracle用户身份登录并取消设置 `$ORACLE_HOME` 和 `$ORACLE_SID` 如果已设置。

```
unset ORACLE_HOME
unset ORACLE_SID
```

2. 创建Oracle DB主目录并进行更改。

```
mkdir /u01/app/oracle/product/19.0.0/db1
cd /u01/app/oracle/product/19.0.0/db1
```

3. 解压缩Oracle数据库安装文件。

```
unzip -q /tmp/archive/LINUX.X64_193000_db_home.zip
```

4. 从数据库主目录中、删除 `OPatch` 目录。

```
rm -rf OPatch
```

5. 从数据库主页复制 `p6880880_190000_Linux-x86-64.zip` to `grid_home`、然后将其解压缩。

```
cp /tmp/archive/p6880880_190000_Linux-x86-64.zip .
unzip p6880880_190000_Linux-x86-64.zip
```

6. 在数据库主页中、修改 `cv/admin/cvu_config`和取消注释并替换 `CV_ASSUME_DISTID=OEL5` 使用 `CV_ASSUME_DISTID=OL7`。

```
vi cv/admin/cvu_config
```

7. 从 `/tmp/archive` 目录中、解压缩DB 19.18 RU修补程序。

```
unzip p34765931_190000_Linux-x86-64.zip
```

8. 在中准备DB静默安装rsp文件 `/tmp/archive/dbinstall.rsp` 具有以下值的目录：

```
oracle.install.option=INSTALL_DB_SWONLY
UNIX_GROUP_NAME=oinstall
INVENTORY_LOCATION=/u01/app/oraInventory
ORACLE_HOME=/u01/app/oracle/product/19.0.0/db1
ORACLE_BASE=/u01/app/oracle
oracle.install.db.InstallEdition=EE
oracle.install.db.OSDBA_GROUP=dba
oracle.install.db.OSOPER_GROUP=oper
oracle.install.db.OSBACKUPDBA_GROUP=oper
oracle.install.db.OSDGDBA_GROUP=dba
oracle.install.db.OSKMDBA_GROUP=dba
oracle.install.db.OSRACDBA_GROUP=dba
oracle.install.db.rootconfig.executeRootScript=false
```

9. 从db1 home /u01/app/oracle/product/19.0.0/db1中、执行无提示纯软件DB安装。

```
 ./runInstaller -applyRU /tmp/archive/34765931/ -silent
-ignorePrereqFailure -responseFile /tmp/archive/dbinstall.rsp
```

10. 以root用户身份运行 `root.sh` 在仅安装软件后编写脚本。

```
/u01/app/oracle/product/19.0.0/db1/root.sh
```

11. 以Oracle用户身份创建 `dbca.rsp` 包含以下条目的文件：

```
gdbName=db1.demo.netapp.com
sid=db1
createAsContainerDatabase=true
numberOfPDBs=3
pdbName=db1_pdb
useLocalUndoForPDBs=true
pdbAdminPassword="yourPWD"
templateName=General_Purpose.dbc
sysPassword="yourPWD"
systemPassword="yourPWD"
dbsnmpPassword="yourPWD"
storageType=ASM
diskGroupName=DATA
characterSet=AL32UTF8
nationalCharacterSet=AL16UTF16
listeners=LISTENER
databaseType=MULTIPURPOSE
automaticMemoryManagement=false
totalMemory=8192
```

根据EC2实例主机中的可用内存设置总内存。Oracle会分配75%的 `totalMemory` 数据库实例SGA或缓冲区缓存。

12. 以Oracle用户身份、使用dbca启动数据库创建。

```
bin/dbca -silent -createDatabase -responseFile /tmp/archive/dbca.rsp

output:
Prepare for db operation
7% complete
Registering database with Oracle Restart
11% complete
Copying database files
33% complete
Creating and starting Oracle instance
35% complete
38% complete
42% complete
45% complete
48% complete
Completing Database Creation
53% complete
55% complete
56% complete
Creating Pluggable Databases
60% complete
64% complete
69% complete
78% complete
Executing Post Configuration Actions
100% complete
Database creation complete. For details check the logfiles at:
 /u01/app/oracle/cfgtoollogs/dbca/db1.
Database Information:
Global Database Name:db1.demo.netapp.com
System Identifier(SID):db1
Look at the log file "/u01/app/oracle/cfgtoollogs/dbca/db1/db1.log"
for further details.
```

13. 以Oracle用户身份、在创建数据库后验证Oracle Restart HA服务。

```
[oracle@ip-172-30-15-58 db1]$ ../grid/bin/crsctl stat res -t
--------------------------------------------------------------------------------
------------
Name           Target  State        Server                   State
details
--------------------------------------------------------------------------------
------------
Local Resources
--------------------------------------------------------------------------------
------------
ora.DATA.dg
               ONLINE  ONLINE       ip-172-30-15-58          STABLE
ora.LISTENER.lsnr
               ONLINE  ONLINE       ip-172-30-15-58          STABLE
ora.LOGS.dg
               ONLINE  ONLINE       ip-172-30-15-58          STABLE
ora.asm
               ONLINE  ONLINE       ip-172-30-15-58
Started,STABLE
ora.ons
               OFFLINE OFFLINE      ip-172-30-15-58          STABLE
--------------------------------------------------------------------------------
------------
Cluster Resources
--------------------------------------------------------------------------------
------------
ora.cssd
      1        ONLINE  ONLINE       ip-172-30-15-58          STABLE
ora.db1.db
      1        ONLINE  ONLINE       ip-172-30-15-58
Open,HOME=/u01/app/o

racle/product/19.0.0

/db1,STABLE
ora.diskmon
      1        OFFLINE OFFLINE                               STABLE
ora.evmd
      1        ONLINE  ONLINE       ip-172-30-15-58          STABLE
--------------------------------------------------------------------------------
------------
[oracle@ip-172-30-15-58 db1]$
```

14. 设置Oracle用户 `.bash_profile`。

```
vi ~/.bash_profile
```

15. 添加以下条目：

```
export ORACLE_HOME=/u01/app/oracle/product/19.0.0/db1
export ORACLE_SID=db1
export PATH=$PATH:$ORACLE_HOME/bin
alias asm='export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/grid;export
ORACLE_SID=+ASM;export PATH=$PATH:$ORACLE_HOME/bin'
```

16. 验证已创建的CDB/PDB。

```
. ~/.bash_profile

sqlplus / as sysdba

SQL> select name, open_mode from v$database;

NAME      OPEN_MODE

DB1       READ WRITE

SQL> select name from v$datafile;

NAME

+DATA/DB1/DATAFILE/system.256.1132176177
+DATA/DB1/DATAFILE/sysaux.257.1132176221
+DATA/DB1/DATAFILE/undotbs1.258.1132176247
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/system.265.11321
77009
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/sysaux.266.11321
77009
+DATA/DB1/DATAFILE/users.259.1132176247
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/undotbs1.267.113
2177009
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/system.271.11321
77853
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/sysaux.272.11321
77853
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/undotbs1.270.113
2177853
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/users.274.113217
```

```
7871

NAME

+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/system.276.11321
77871
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/sysaux.277.11321
77871
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/undotbs1.275.113
2177871
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/users.279.113217
7889
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/system.281.11321
77889
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/sysaux.282.11321
77889
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/undotbs1.280.113
2177889
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/users.284.113217
7907

19 rows selected.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED

         2 PDB$SEED                       READ ONLY  NO
         3 DB1_PDB1                       READ WRITE NO
         4 DB1_PDB2                       READ WRITE NO
         5 DB1_PDB3                       READ WRITE NO
SQL>
```

17. 以Oracle用户身份、更改为Oracle数据库主目录/u01/app/oracle/product/19.0.0/db1并启用DNFS

```
cd /u01/app/oracle/product/19.0.0/db1

mkdir rdbms/lib/odm

cp lib/libnfsodm19.so rdbms/lib/odm/
```

18. 在oracle_home中配置oranfstab文件

```
vi $ORACLE_HOME/dbs/oranfstab

add following entries:

server: fsx_01
local: 172.30.15.58 path: 172.30.15.19
nfs_version: nfsv3
export: /ora_01_biny mount: /u01
export: /ora_01_data mount: /oradata
export: /ora_01_logs mount: /oralogs
```

19. 以Oracle用户身份、从sqlplus登录到数据库、并将数据库恢复大小和位置设置为+logs.磁盘组。

```
. ~/.bash_profile

sqlplus / as sysdba

alter system set db_recovery_file_dest_size = 80G scope=both;

alter system set db_recovery_file_dest = '+LOGS' scope=both;
```

20. 启用归档日志模式并重新启动Oracle DB实例

```
shutdown immediate;

startup mount;

alter database archivelog;

alter database open;

alter system switch logfile;
```

21. 在实例重新启动后验证数据库日志模式和DNFS

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

NAME      LOG_MODE
--------- ------------
DB1       ARCHIVELOG

SQL> select svrname, dirname from v$dnfs_servers;

SVRNAME
------------------------------------------------------------------------
------------
DIRNAME
------------------------------------------------------------------------
------------
fsx_01
/ora_01_data

fsx_01
/ora_01_biny

fsx_01
/ora_01_logs
```

22. 验证Oracle ASM

```
[oracle@ip-172-30-15-58 db1]$ asm
[oracle@ip-172-30-15-58 db1]$ sqlplus / as sysasm

SQL*Plus: Release 19.0.0.0.0 - Production on Tue May 9 20:39:39 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> set lin 200
SQL> col path form a30
SQL> select name, path, header_status, mount_status, state from
v$asm_disk;

NAME                           PATH
```

```
HEADER_STATU MOUNT_S STATE
------------------------------ ------------------------------
------------ ------- --------
DATA_0002                      /oradata/asm/nfs_data_disk01   MEMBER
CACHED  NORMAL
DATA_0000                      /oradata/asm/nfs_data_disk02   MEMBER
CACHED  NORMAL
DATA_0001                      /oradata/asm/nfs_data_disk03   MEMBER
CACHED  NORMAL
DATA_0003                      /oradata/asm/nfs_data_disk04   MEMBER
CACHED  NORMAL
LOGS_0000                      /oralogs/asm/nfs_logs_disk01   MEMBER
CACHED  NORMAL
LOGS_0001                      /oralogs/asm/nfs_logs_disk02   MEMBER
CACHED  NORMAL

6 rows selected.


SQL> select name, state, ALLOCATION_UNIT_SIZE, TOTAL_MB, FREE_MB
from v$asm_diskgroup;

NAME                           STATE       ALLOCATION_UNIT_SIZE
TOTAL_MB    FREE_MB
------------------------------ ----------- --------------------
---------- ----------
DATA                           MOUNTED                  4194304
81920      73536
LOGS                           MOUNTED                  4194304
81920      81640

This completes Oracle 19c version 19.18 Restart deployment on an
Amazon FSx for ONTAP and EC2 compute instance with NFS/ASM. If
desired, NetApp recommends relocating the Oracle control file and
online log files to the +LOGS disk group.
```

自动化部署选项

NetApp将随Ansible发布一个完全自动化的解决方案 部署工具包、以便于实施此解决方案。请查看此工具包的可用性。发布后、此处将发布一个链接。

使用SnapCenter 服务备份、还原和克隆Oracle数据库

目前、只有传统的SnapCenter服务器UI工具(请参见)才支持使用NFS和ASM存储选项的Oracle数据库 "采用 SnapCenter 的混合云数据库解决方案" 有关使用NetApp SnapCenter UI工具进行Oracle数据库备份、还原和克

隆的详细信息。

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- 在安装新数据库的情况下为独立服务器安装Oracle网格基础架构

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-oracle-grid-infrastructure-for-a-standalone-server-with-a-new-database-installation.html#GUID-0B1CEE8C-C893-46AA-8A6A-7B5FAAEC72B3"

- 使用响应文件安装和配置Oracle数据库

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-and-configuring-oracle-database-using-response-files.html#GUID-D53355E9-E901-4224-9A2A-B882070EDDF7"

- 适用于 NetApp ONTAP 的 Amazon FSX

  "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

TR-4965：《使用iSCSI/ASM在AWS FSX/EC2中部署和保护Oracle数据库》

NetApp公司Allen Cao、Niyaz Mohamed

目的

ASM (Automatic Storage Management、自动存储管理)是许多Oracle安装中常用的Oracle存储卷管理器。这也是Oracle建议的存储管理解决方案。它提供了传统卷管理器和文件系统的替代方案。自Oracle 11g版以来、ASM一直采用网格基础架构而非数据库打包。因此、要在不使用RAC的情况下使用Oracle ASM进行存储管理、您必须在独立服务器中安装Oracle网格基础架构、也称为Oracle Restart。这样做无疑会增加Oracle数据库部署的复杂性。但是、顾名思义、在以重新启动模式部署Oracle时、出现故障的Oracle服务会由网格基础架构自动重新启动、或者在主机重新启动后无需用户干预、从而提供一定程度的高可用性或HA功能。

在本文档中、我们将演示如何在使用EC2计算实例的Amazon FSX for ONTAP 存储环境中部署采用iSCSI协议的Oracle数据库和Oracle ASM。我们还演示了如何通过NetApp BlueXP控制台使用NetApp SnapCenter 服务来备份、还原和克隆Oracle数据库、以便在AWS公共云中进行开发/测试或其他存储效率数据库操作用例。

此解决方案 可解决以下使用情形：

- 在Amazon FSX中部署Oracle数据库、用于使用iSCSI/ASM的ONTAP 存储和EC2计算实例
- 使用iSCSI/ASM在公共AWS云中测试和验证Oracle工作负载

- 测试和验证AWS中部署的Oracle数据库重新启动功能

audience

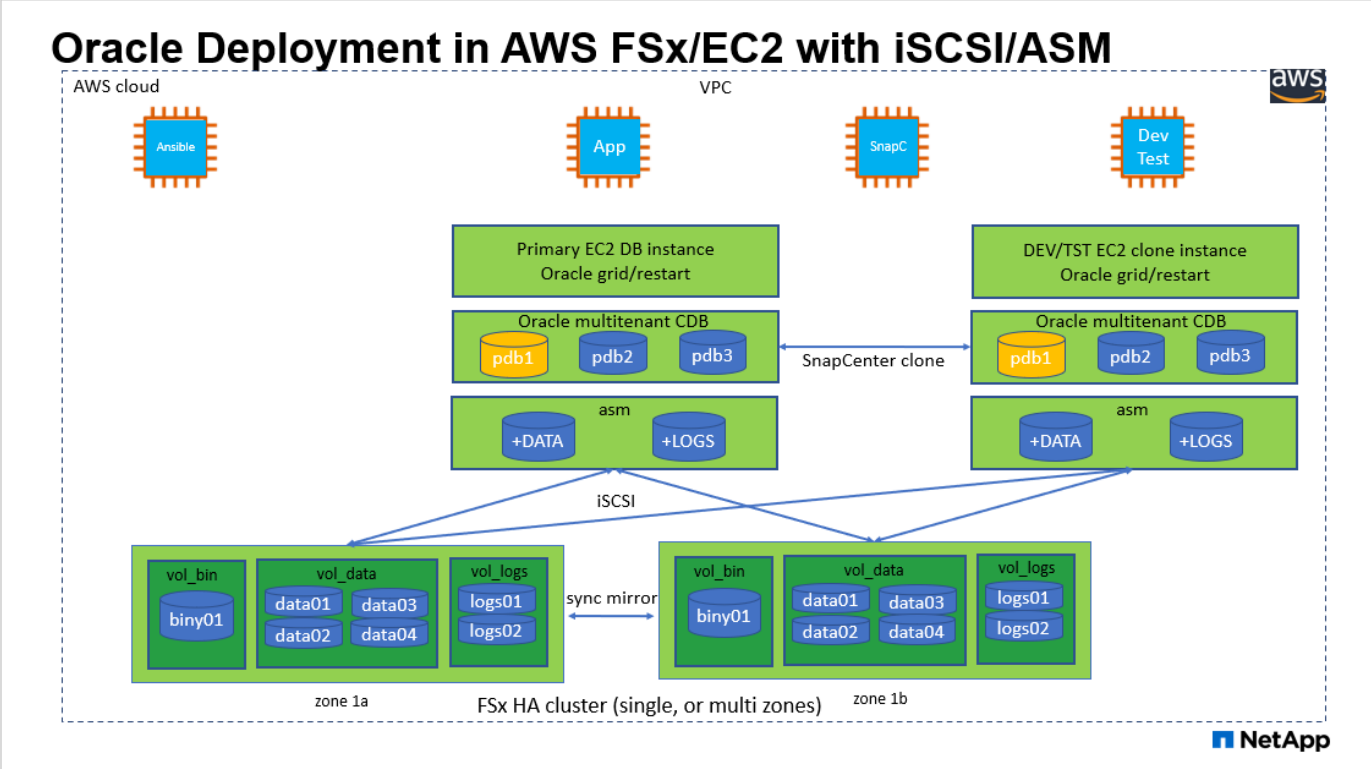
此解决方案 适用于以下人员：

- 希望使用iSCSI/ASM在AWS公共云中部署Oracle的DBA。
- 一名数据库解决方案 架构师、希望在AWS公共云中测试Oracle工作负载。
- 希望部署和管理部署到AWS FSX存储的Oracle数据库的存储管理员。
- 希望在AWS FSX/EC2中建立Oracle数据库的应用程序所有者。

解决方案 测试和验证环境

此解决方案 的测试和验证是在AWS FSX和EC2环境中执行的、该环境可能与最终部署环境不匹配。有关详细信息，请参见一节 [Key Factors for Deployment Consideration]。

架构



硬件和软件组件

* 硬件 *		
FSX ONTAP 存储	AWS提供的当前版本	一个FSX HA集群位于同一VPC和可用性区域中
用于计算的EC2实例	t2.xlarge/4vCPU/16G	两个EC2 t2 xlarge EC2实例、一个用作主数据库服务器、另一个用作克隆数据库服务器

软件		
RedHat Linux	rhel-8.6.0_hvm-20220503-x86_64-2-Hourly2-gp2	已部署RedHat订阅以进行测试
Oracle网格基础架构	版本19.18	已应用RU修补程序p34762026_190000_Linux-x86-64.zip
Oracle 数据库	版本19.18	已应用RU修补程序p34765931_190000_Linux-x86-64.zip
Oracle OPatch	版本12.2.0.1.36	最新修补程序p6880880_190000_Linux-x86-64.zip
SnapCenter 服务	version	v2.3.1.2324

部署注意事项的关键因素

- * EC2计算实例。*在这些测试和验证中、我们对Oracle数据库计算实例使用了AWS EC2 T2.xlarge实例类型。NetApp建议在生产部署中使用M5类型的EC2实例作为Oracle的计算实例、因为它针对数据库工作负载进行了优化。您需要根据实际工作负载要求根据vCPU数量和RAM量适当调整EC2实例的大小。
- * FSX存储HA集群单区域或多区域部署。*在这些测试和验证中、我们在一个AWS可用性区域中部署了一个FSX HA集群。对于生产部署、NetApp建议在两个不同的可用性区域中部署一个FSX HA对。FSX HA集群始终配置在一个HA对中、该HA对在一对主动-被动文件系统中进行同步镜像、以提供存储级别的冗余。多区域部署可在单个AWS区域发生故障时进一步提高高可用性。
- * FSX存储集群规模估算。*适用于ONTAP 存储文件系统的Amazon FSX可提供高达160、000个原始SSD IOPS、高达4 Gbps吞吐量以及最大192 TiB容量。但是、您可以根据部署时的实际要求、根据已配置的IOPS、吞吐量和存储限制(最小1、024 GiB)来调整集群的大小。可以动态调整容量、而不会影响应用程序可用性。
- * Oracle数据和日志布局。*在测试和验证中、我们分别为数据和日志部署了两个ASM磁盘组。在+data ASM磁盘组中、我们在一个数据卷中配置了四个LUN。在+logs ASM磁盘组中、我们在一个日志卷中配置了两个LUN。通常、在适用于ONTAP 的Amazon FSX卷中部署多个LUN可提高性能。
- * iSCSI配置。* EC2实例数据库服务器使用iSCSI协议连接到FSX存储。EC2实例通常使用一个网络接口或ENI进行部署。单个NIC接口可同时传输iSCSI和应用程序流量。请务必通过仔细分析Oracle AWR报告来衡量Oracle数据库峰值I/O吞吐量需求、以便选择既满足应用程序流量吞吐量要求又符合iSCSI流量吞吐量要求的正确EC2计算实例。NetApp还建议为正确配置多路径的两个FSX iSCSI端点分配四个iSCSI连接。
- *要为创建的每个Oracle ASM磁盘组使用的Oracle ASM冗余级别。*由于FSX已在FSX集群级别镜像存储、因此应使用外部冗余、这意味着该选项不允许Oracle ASM镜像磁盘组的内容。
- 数据库备份。 NetApp提供SaaS版本的SnapCenter 软件服务、用于在云中备份、还原和克隆数据库、该服务可通过NetApp BlueXP控制台UI访问。NetApp建议实施此类服务、以实现快速(不到一分钟)的快照备份、快速(几分钟)的数据库还原和数据库克隆。

解决方案 部署

下一节介绍了分步部署过程。

部署的前提条件

部署需要满足以下前提条件。

1. 已设置AWS帐户、并已在您的AWS帐户中创建必要的VPC和网段。
2. 在AWS EC2控制台中、您必须部署两个EC2 Linux实例、一个用作主Oracle数据库服务器、另一个用作可选的克隆目标数据库服务器。有关环境设置的详细信息、请参见上一节中的架构图。另请查看 "Linux实例用户指南" 有关详细信息 …
3. 在AWS EC2控制台中、部署适用于ONTAP 存储HA集群的Amazon FSX以托管Oracle数据库卷。如果您不熟悉FSX存储的部署、请参见相关文档 "为ONTAP 文件系统创建FSX" 了解分步说明。
4. 可以使用以下Terraform自动化工具包执行步骤2和步骤3、该工具包会创建一个名为的EC2实例 `ora_01` 和名为的FSX文件系统 `fsx_01`。执行前、请仔细阅读该说明并根据您的环境更改变量。

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

确保您已在EC2实例根卷中至少分配50G、以便有足够的空间来暂存Oracle安装文件。

EC2实例内核配置

在配置了前提条件的情况下、以EC2-user身份登录到EC2实例、并使用sudo to root user配置Linux内核以进行Oracle安装。

1. 创建暂存目录 `/tmp/archive` 文件夹并设置 `777` 权限。

```
mkdir /tmp/archive

chmod 777 /tmp/archive
```

2. 将Oracle二进制安装文件和其他所需的rpm文件下载并暂存到 `/tmp/archive` 目录。

   请参见以下要在中说明的安装文件列表 `/tmp/archive` 在EC2实例上。

```
[ec2-user@ip-172-30-15-58 ~]$ ls -l /tmp/archive
total 10537316
-rw-rw-r--. 1 ec2-user ec2-user      19112 Mar 21 15:57 compat-
libcap1-1.10-7.el7.x86_64.rpm
-rw-rw-r--  1 ec2-user ec2-user 3059705302 Mar 21 22:01
LINUX.X64_193000_db_home.zip
-rw-rw-r--  1 ec2-user ec2-user 2889184573 Mar 21 21:09
LINUX.X64_193000_grid_home.zip
-rw-rw-r--. 1 ec2-user ec2-user     589145 Mar 21 15:56
netapp_linux_unified_host_utilities-7-1.x86_64.rpm
-rw-rw-r--. 1 ec2-user ec2-user      31828 Mar 21 15:55 oracle-
database-preinstall-19c-1.0-2.el8.x86_64.rpm
-rw-rw-r--  1 ec2-user ec2-user 2872741741 Mar 21 22:31
p34762026_190000_Linux-x86-64.zip
-rw-rw-r--  1 ec2-user ec2-user 1843577895 Mar 21 22:32
p34765931_190000_Linux-x86-64.zip
-rw-rw-r--  1 ec2-user ec2-user  124347218 Mar 21 22:33
p6880880_190000_Linux-x86-64.zip
-rw-r--r--  1 ec2-user ec2-user     257136 Mar 22 16:25
policycoreutils-python-utils-2.9-9.el8.noarch.rpm
```

3. 安装Oracle 19c预安装RPM、以满足大多数内核配置要求。

```
yum install /tmp/archive/oracle-database-preinstall-19c-1.0-
2.el8.x86_64.rpm
```

4. 下载并安装缺少的 `compat-libcap1` 在Linux 8中。

```
yum install /tmp/archive/compat-libcap1-1.10-7.el7.x86_64.rpm
```

5. 从NetApp下载并安装NetApp主机实用程序。

```
yum install /tmp/archive/netapp_linux_unified_host_utilities-7-
1.x86_64.rpm
```

6. 安装 `policycoreutils-python-utils`、在EC2实例中不可用。

```
yum install /tmp/archive/policycoreutils-python-utils-2.9-
9.el8.noarch.rpm
```

7. 安装Open JDK 1.8版。

```
yum install java-1.8.0-openjdk.x86_64
```

8. 安装iSCSI启动程序实用程序。

```
yum install iscsi-initiator-utils
```

9. 安装 `sg3_utils`。

```
yum install sg3_utils
```

10. 安装 `device-mapper-multipath`。

```
yum install device-mapper-multipath
```

11. 在当前系统中禁用透明页面。

```
echo never > /sys/kernel/mm/transparent_hugepage/enabled
echo never > /sys/kernel/mm/transparent_hugepage/defrag
```

在中添加以下行 `/etc/rc.local` 以禁用 `transparent_hugepage` 重新启动后：

```
  # Disable transparent hugepages
          if test -f /sys/kernel/mm/transparent_hugepage/enabled;
then
            echo never > /sys/kernel/mm/transparent_hugepage/enabled
          fi
          if test -f /sys/kernel/mm/transparent_hugepage/defrag;
then
            echo never > /sys/kernel/mm/transparent_hugepage/defrag
          fi
```

12. 通过更改禁用SELinux `SELINUX=enforcing` to `SELINUX=disabled`。要使更改生效、必须重新启动主机。

```
vi /etc/sysconfig/selinux
```

13. 将以下行添加到 `limit.conf` 设置不带引号的文件描述符限制和堆栈大小 " "。

```
vi /etc/security/limits.conf
  "*               hard    nofile          65536"
  "*               soft    stack           10240"
```

14. 按照以下说明向EC2实例添加交换空间： "如何使用交换文件分配内存以用作Amazon EC2实例中的交换空间？" 要添加的确切空间量取决于RAM大小、最高可达16G。

15. 更改 `node.session.timeo.replacement_timeout` 在中 `iscsi.conf` 配置文件的时间从120秒到5秒不等。

```
vi /etc/iscsi/iscsid.conf
```

16. 在EC2实例上启用并启动iSCSI服务。

```
systemctl enable iscsid
systemctl start iscsid
```

17. 检索要用于数据库LUN映射的iSCSI启动程序地址。

```
cat /etc/iscsi/initiatorname.iscsi
```

18. 添加要用于ASM sysASM组的ASM组。

```
groupadd asm
```

19. 修改Oracle用户以将ASM添加为二级组(Oracle用户应在安装Oracle预安装RPM后创建)。

```
usermod -a -G asm oracle
```

20. 停止并禁用处于活动状态的Linux防火墙。

```
systemctl stop firewalld
systemctl disable firewalld
```

21. 重新启动EC2实例。

配置数据库卷和**LUN**并将其映射到**EC2**实例主机

使用FSx集群管理IP通过ssh以fsxadmin用户身份登录FSx集群、从命令行配置三个卷、以托管Oracle数据库二进制文件、数据和日志文件。

1. 以fsxadmin用户身份通过SSH登录到FSX集群。

```
ssh fsxadmin@172.30.15.53
```

2. 执行以下命令为Oracle二进制文件创建卷。

```
vol create -volume ora_01_biny -aggregate aggr1 -size 50G -state
online  -type RW -snapshot-policy none -tiering-policy snapshot-only
```

3. 执行以下命令为Oracle数据创建卷。

```
vol create -volume ora_01_data -aggregate aggr1 -size 100G -state
online  -type RW -snapshot-policy none -tiering-policy snapshot-only
```

4. 执行以下命令为Oracle日志创建卷。

```
vol create -volume ora_01_logs -aggregate aggr1 -size 100G -state
online  -type RW -snapshot-policy none -tiering-policy snapshot-only
```

5. 在数据库二进制卷中创建二进制LUN。

```
lun create -path /vol/ora_01_biny/ora_01_biny_01 -size 40G -ostype
linux
```

6. 在数据库数据卷中创建数据LUN。

```
lun create -path /vol/ora_01_data/ora_01_data_01 -size 20G -ostype
linux

lun create -path /vol/ora_01_data/ora_01_data_02 -size 20G -ostype
linux

lun create -path /vol/ora_01_data/ora_01_data_03 -size 20G -ostype
linux

lun create -path /vol/ora_01_data/ora_01_data_04 -size 20G -ostype
linux
```

7. 在数据库日志卷中创建日志LUN。

```
lun create -path /vol/ora_01_logs/ora_01_logs_01 -size 40G -ostype
linux

lun create -path /vol/ora_01_logs/ora_01_logs_02 -size 40G -ostype
linux
```

8. 使用从上述EC2内核配置的步骤14中检索到的启动程序为EC2实例创建一个igroup。

```
igroup create -igroup ora_01 -protocol iscsi -ostype linux
-initiator iqn.1994-05.com.redhat:f65fed7641c2
```

9. 将LUN映射到上述创建的igroup。按顺序增加卷中每个附加LUN的LUN ID。

```
lun map -path /vol/ora_01_biny/ora_01_biny_01 -igroup ora_01
-vserver svm_ora -lun-id 0
lun map -path /vol/ora_01_data/ora_01_data_01 -igroup ora_01
-vserver svm_ora -lun-id 1
lun map -path /vol/ora_01_data/ora_01_data_02 -igroup ora_01
-vserver svm_ora -lun-id 2
lun map -path /vol/ora_01_data/ora_01_data_03 -igroup ora_01
-vserver svm_ora -lun-id 3
lun map -path /vol/ora_01_data/ora_01_data_04 -igroup ora_01
-vserver svm_ora -lun-id 4
lun map -path /vol/ora_01_logs/ora_01_logs_01 -igroup ora_01
-vserver svm_ora -lun-id 5
lun map -path /vol/ora_01_logs/ora_01_logs_02 -igroup ora_01
-vserver svm_ora -lun-id 6
```

10. 验证LUN映射。

```
mapping show
```

这将返回：

```
FsxId02ad7bf3476b741df::> mapping show
  (lun mapping show)
Vserver    Path                                      Igroup   LUN ID
Protocol
---------- ----------------------------------------  -------  ------
--------
svm_ora    /vol/ora_01_biny/ora_01_biny_01           ora_01        0
iscsi
svm_ora    /vol/ora_01_data/ora_01_data_01           ora_01        1
iscsi
svm_ora    /vol/ora_01_data/ora_01_data_02           ora_01        2
iscsi
svm_ora    /vol/ora_01_data/ora_01_data_03           ora_01        3
iscsi
svm_ora    /vol/ora_01_data/ora_01_data_04           ora_01        4
iscsi
svm_ora    /vol/ora_01_logs/ora_01_logs_01           ora_01        5
iscsi
svm_ora    /vol/ora_01_logs/ora_01_logs_02           ora_01        6
iscsi
```

数据库存储配置

现在、为EC2实例主机上的Oracle网格基础架构和数据库安装导入并设置FSX存储。

1. 以EC2用户身份使用SSH密钥和EC2实例IP地址通过SSH登录到EC2实例。

```
ssh -i ora_01.pem ec2-user@172.30.15.58
```

2. 使用任一SVM iSCSI IP地址发现FSX iSCSI端点。然后、更改为您的环境专用门户地址。

```
sudo iscsiadm iscsiadm --mode discovery --op update --type
sendtargets --portal 172.30.15.51
```

3. 登录到每个目标以建立iSCSI会话。

```
sudo iscsiadm --mode node -l all
```

命令的预期输出为：

```
[ec2-user@ip-172-30-15-58 ~]$ sudo iscsiadm --mode node -l all
Logging in to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
172.30.15.51,3260]
Logging in to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
172.30.15.13,3260]
Login to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
172.30.15.51,3260] successful.
Login to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
172.30.15.13,3260] successful.
```

4. 查看并验证活动iSCSI会话的列表。

```
sudo iscsiadm --mode session
```

返回iSCSI会话。

```
[ec2-user@ip-172-30-15-58 ~]$ sudo iscsiadm --mode session
tcp: [1] 172.30.15.51:3260,1028 iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3 (non-flash)
tcp: [2] 172.30.15.13:3260,1029 iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3 (non-flash)
```

5. 验证LUN是否已导入到主机中。

```
sudo sanlun lun show
```

此操作将从FSX返回Oracle LUN的列表。

```
[ec2-user@ip-172-30-15-58 ~]$ sudo sanlun lun show
controller(7mode/E-Series)/                                    device
host                  lun
vserver(cDOT/FlashRay)        lun-pathname
filename        adapter    protocol   size    product

svm_ora                       /vol/ora_01_logs/ora_01_logs_02
/dev/sdn        host3      iSCSI      40g     cDOT
svm_ora                       /vol/ora_01_logs/ora_01_logs_01
/dev/sdm        host3      iSCSI      40g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_03
/dev/sdk        host3      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_04
/dev/sdl        host3      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_01
/dev/sdi        host3      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_02
/dev/sdj        host3      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_biny/ora_01_biny_01
/dev/sdh        host3      iSCSI      40g     cDOT
svm_ora                       /vol/ora_01_logs/ora_01_logs_02
/dev/sdg        host2      iSCSI      40g     cDOT
svm_ora                       /vol/ora_01_logs/ora_01_logs_01
/dev/sdf        host2      iSCSI      40g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_04
/dev/sde        host2      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_02
/dev/sdc        host2      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_03
/dev/sdd        host2      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_01
/dev/sdb        host2      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_biny/ora_01_biny_01
/dev/sda        host2      iSCSI      40g     cDOT
```

6. 配置 `multipath.conf` 包含以下默认条目和黑名单条目的文件。

```
sudo vi /etc/multipath.conf

defaults {
    find_multipaths yes
    user_friendly_names yes
}

blacklist {
    devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
    devnode "^hd[a-z]"
    devnode "^cciss.*"
}
```

7. 启动多路径服务。

```
sudo systemctl start multipathd
```

现在、多路径设备将显示在中 `/dev/mapper` 目录。

```
[ec2-user@ip-172-30-15-58 ~]$ ls -l /dev/mapper
total 0
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e68512d -> ../dm-0
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685141 -> ../dm-1
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685142 -> ../dm-2
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685143 -> ../dm-3
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685144 -> ../dm-4
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685145 -> ../dm-5
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685146 -> ../dm-6
crw------- 1 root root 10, 236 Mar 21 18:19 control
```

8. 通过SSH以fsxadmin用户身份登录到FSX集群、以检索每个LUN的序列号十六进制值、该值以6c574xxx开头…、十六进制值以3600a0980开头、即AWS供应商ID。

```
lun show -fields serial-hex
```

并返回如下内容：

```
FsxId02ad7bf3476b741df::> lun show -fields serial-hex
vserver path                            serial-hex
------- ------------------------------- ------------------------
svm_ora /vol/ora_01_biny/ora_01_biny_01 6c574235472455534e68512d
svm_ora /vol/ora_01_data/ora_01_data_01 6c574235472455534e685141
svm_ora /vol/ora_01_data/ora_01_data_02 6c574235472455534e685142
svm_ora /vol/ora_01_data/ora_01_data_03 6c574235472455534e685143
svm_ora /vol/ora_01_data/ora_01_data_04 6c574235472455534e685144
svm_ora /vol/ora_01_logs/ora_01_logs_01 6c574235472455534e685145
svm_ora /vol/ora_01_logs/ora_01_logs_02 6c574235472455534e685146
7 entries were displayed.
```

9. 更新 `/dev/multipath.conf` 文件、用于为多路径设备添加用户友好名称。

```
sudo vi /etc/multipath.conf
```

包含以下条目：

```
multipaths {
        multipath {
                wwid            3600a09806c574235472455534e68512d
                alias           ora_01_biny_01
        }
        multipath {
                wwid            3600a09806c574235472455534e685141
                alias           ora_01_data_01
        }
        multipath {
                wwid            3600a09806c574235472455534e685142
                alias           ora_01_data_02
        }
        multipath {
                wwid            3600a09806c574235472455534e685143
                alias           ora_01_data_03
        }
        multipath {
                wwid            3600a09806c574235472455534e685144
                alias           ora_01_data_04
        }
        multipath {
                wwid            3600a09806c574235472455534e685145
                alias           ora_01_logs_01
        }
        multipath {
                wwid            3600a09806c574235472455534e685146
                alias           ora_01_logs_02
        }
}
```

10. 重新启动多路径服务以验证下的设备 `/dev/mapper` 已更改为LUN名称与串行十六进制ID。

```
sudo systemctl restart multipathd
```

检查 `/dev/mapper` 返回如下内容：

```
[ec2-user@ip-172-30-15-58 ~]$ ls -l /dev/mapper
total 0
crw------- 1 root root 10, 236 Mar 21 18:19 control
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_biny_01 -> ../dm-
0
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_data_01 -> ../dm-
1
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_data_02 -> ../dm-
2
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_data_03 -> ../dm-
3
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_data_04 -> ../dm-
4
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_logs_01 -> ../dm-
5
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_logs_02 -> ../dm-
6
```

11. 使用一个主分区对二进制LUN进行分区。

```
sudo fdisk /dev/mapper/ora_01_biny_01
```

12. 使用XFS文件系统格式化分区的二进制LUN。

```
sudo mkfs.xfs /dev/mapper/ora_01_biny_01p1
```

13. 将二进制LUN挂载到 `/u01`。

```
sudo mount -t xfs /dev/mapper/ora_01_biny_01p1 /u01
```

14. 更改 `/u01` 将点所有权挂载到Oracle用户及其所属的主组。

```
sudo chown oracle:oinstall /u01
```

15. 查找二进制LUN的UUI。

```
sudo blkid /dev/mapper/ora_01_biny_01p1
```

16. 将挂载点添加到 `/etc/fstab`。

```
sudo vi /etc/fstab
```

添加以下行。

```
UUID=d89fb1c9-4f89-4de4-b4d9-17754036d11d       /u01    xfs
defaults,nofail 0       2
```

请务必仅使用UUID和nodfail选项挂载二进制文件、以避免在EC2实例重新启动期间可能出现根锁定问题。

17. 以root用户身份为Oracle设备添加udev规则。

```
vi /etc/udev/rules.d/99-oracle-asmdevices.rules
```

包括以下条目：

```
ENV{DM_NAME}=="ora*", GROUP:="oinstall", OWNER:="oracle",
MODE:="660"
```

18. 以root用户身份重新加载udev规则。

```
udevadm control --reload-rules
```

19. 以root用户身份触发udev规则。

```
udevadm trigger
```

20. 以root用户身份重新加载multipathd。

```
systemctl restart multipathd
```

21. 重新启动EC2实例主机。

Oracle网格基础架构安装

1. 通过SSH以EC2-user身份登录到EC2实例、并通过取消注释启用密码身份验证 `PasswordAuthentication yes` 然后进行注释 `PasswordAuthentication no`。

```
sudo vi /etc/ssh/sshd_config
```

2. 重新启动sshd服务。

```
sudo systemctl restart sshd
```

3. 重置Oracle用户密码。

```
sudo passwd oracle
```

4. 以Oracle Restart软件所有者用户(Oracle)身份登录。按如下所示创建Oracle目录：

```
mkdir -p /u01/app/oracle
mkdir -p /u01/app/oraInventory
```

5. 更改目录权限设置。

```
chmod -R 775 /u01/app
```

6. 创建网格主目录并进行更改。

```
mkdir -p /u01/app/oracle/product/19.0.0/grid
cd /u01/app/oracle/product/19.0.0/grid
```

7. 解压缩网格安装文件。

```
unzip -q /tmp/archive/LINUX.X64_193000_grid_home.zip
```

8. 从网格主页中、删除 `OPatch` 目录。

```
rm -rf OPatch
```

9. 从网格主页解压缩 `p6880880_190000_Linux-x86-64.zip`。

```
unzip -q /tmp/archive/p6880880_190000_Linux-x86-64.zip
```

10. 从网格主页修改 `cv/admin/cvu_config`、取消注释并替换 `CV_ASSUME_DISTID=OEL5` 使用 `CV_ASSUME_DISTID=OL7`。

```
vi cv/admin/cvu_config
```

11. 准备 `gridsetup.rsp` 文件以进行静默安装、并将rsp文件置于中 `/tmp/archive` 目录。rsp文件应涵盖A、B和G部分、并提供以下信息：

```
INVENTORY_LOCATION=/u01/app/oraInventory
oracle.install.option=HA_CONFIG
ORACLE_BASE=/u01/app/oracle
oracle.install.asm.OSDBA=dba
oracle.install.asm.OSOPER=oper
oracle.install.asm.OSASM=asm
oracle.install.asm.SYSASMPassword="SetPWD"
oracle.install.asm.diskGroup.name=DATA
oracle.install.asm.diskGroup.redundancy=EXTERNAL
oracle.install.asm.diskGroup.AUSize=4
oracle.install.asm.diskGroup.disks=/dev/mapper/ora_01_data_01,/dev/m
apper/ora_01_data_02,/dev/mapper/ora_01_data_03,/dev/mapper/ora_01_d
ata_04
oracle.install.asm.diskGroup.diskDiscoveryString=/dev/mapper/*
oracle.install.asm.monitorPassword="SetPWD"
oracle.install.asm.configureAFD=true
```

12. 以root用户身份登录到EC2实例并进行设置 `ORACLE_HOME` 和 `ORACLE_BASE`。

```
export ORACLE_HOME=/u01/app/oracle/product/19.0.0/grid
export ORACLE_BASE=/tmp
cd /u01/app/oracle/product/19.0.0/grid/bin
```

13. 配置磁盘设备以与Oracle ASM筛选器驱动程序结合使用。

```
 ./asmcmd afd_label DATA01 /dev/mapper/ora_01_data_01 --init

 ./asmcmd afd_label DATA02 /dev/mapper/ora_01_data_02 --init

 ./asmcmd afd_label DATA03 /dev/mapper/ora_01_data_03 --init

 ./asmcmd afd_label DATA04 /dev/mapper/ora_01_data_04 --init

 ./asmcmd afd_label LOGS01 /dev/mapper/ora_01_logs_01 --init

 ./asmcmd afd_label LOGS02 /dev/mapper/ora_01_logs_02 --init
```

14. 安装 `cvuqdisk-1.0.10-1.rpm`。

```
rpm -ivh /u01/app/oracle/product/19.0.0/grid/cv/rpm/cvuqdisk-1.0.10-
1.rpm
```

15. 未设置 `$ORACLE_BASE`。

```
unset ORACLE_BASE
```

16. 以Oracle用户身份登录到EC2实例、然后在中提取修补程序 `/tmp/archive` 文件夹。

```
unzip /tmp/archive/p34762026_190000_Linux-x86-64.zip -d /tmp/archive
```

17. 从Grid home /u01/app/oracle/product/19.0.0/grid中、以Oracle用户身份启动 `gridSetup.sh` 用于网格基础架构安装。

```
 ./gridSetup.sh -applyRU /tmp/archive/34762026/ -silent
-responseFile /tmp/archive/gridsetup.rsp
```

忽略有关网格基础架构的错误组的警告。我们正在使用一个Oracle用户来管理Oracle Restart、因此这是预期的。

18. 以root用户身份执行以下脚本：

```
/u01/app/oraInventory/orainstRoot.sh

/u01/app/oracle/product/19.0.0/grid/root.sh
```

19. 以root用户身份重新加载multipathd。

```
systemctl restart multipathd
```

20. 以Oracle用户身份执行以下命令以完成配置：

```
/u01/app/oracle/product/19.0.0/grid/gridSetup.sh -executeConfigTools
-responseFile /tmp/archive/gridsetup.rsp -silent
```

21. 以Oracle用户身份创建日志磁盘组。

```
bin/asmca -silent -sysAsmPassword 'yourPWD' -asmsnmpPassword
'yourPWD' -createDiskGroup -diskGroupName LOGS -disk 'AFD:LOGS*'
-redundancy EXTERNAL -au_size 4
```

22. 以Oracle用户身份、在安装配置后验证网格服务。

```
bin/crsctl stat res -t
+
Name                Target  State        Server
State details
Local Resources
ora.DATA.dg         ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.LISTENER.lsnr   ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.LOGS.dg         ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.asm             ONLINE  ONLINE       ip-172-30-15-58
Started,STABLE
ora.ons             OFFLINE OFFLINE      ip-172-30-15-58
STABLE
Cluster Resources
ora.cssd            ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.diskmon         OFFLINE OFFLINE
STABLE
ora.driver.afd      ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.evmd            ONLINE  ONLINE       ip-172-30-15-58
STABLE
```

23. Valiate ASM筛选器驱动程序状态。

```
[oracle@ip-172-30-15-58 grid]$ export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/grid
[oracle@ip-172-30-15-58 grid]$ export ORACLE_SID=+ASM
[oracle@ip-172-30-15-58 grid]$ export PATH=$PATH:$ORACLE_HOME/bin
[oracle@ip-172-30-15-58 grid]$ asmcmd
ASMCMD> lsdg
State    Type    Rebal  Sector  Logical_Sector  Block       AU
Total_MB  Free_MB  Req_mir_free_MB  Usable_file_MB  Offline_disks
Voting_files  Name
MOUNTED  EXTERN  N         512             512   4096  1048576
81920    81847                0           81847              0
N  DATA/
MOUNTED  EXTERN  N         512             512   4096  1048576
81920    81853                0           81853              0
N  LOGS/
ASMCMD> afd_state
ASMCMD-9526: The AFD state is 'LOADED' and filtering is 'ENABLED' on
host 'ip-172-30-15-58.ec2.internal'
```

Oracle数据库安装

1. 以Oracle用户身份登录并取消设置 `$ORACLE_HOME` 和 `$ORACLE_SID` 如果已设置。

```
unset ORACLE_HOME
unset ORACLE_SID
```

2. 创建Oracle DB主目录并进行更改。

```
mkdir /u01/app/oracle/product/19.0.0/db1
cd /u01/app/oracle/product/19.0.0/db1
```

3. 解压缩Oracle数据库安装文件。

```
unzip -q /tmp/archive/LINUX.X64_193000_db_home.zip
```

4. 从数据库主目录中、删除 `OPatch` 目录。

```
rm -rf OPatch
```

5. 从DB主目录中、解压缩 `p6880880_190000_Linux-x86-64.zip`。

```
unzip -q /tmp/archive/p6880880_190000_Linux-x86-64.zip
```

6. 在数据库主页中、修改 `cv/admin/cvu_config`和取消注释并替换 `CV_ASSUME_DISTID=OEL5` 使用 `CV_ASSUME_DISTID=OL7`。

```
vi cv/admin/cvu_config
```

7. 从 `/tmp/archive` 目录中、解压缩DB 19.18 RU修补程序。

```
unzip p34765931_190000_Linux-x86-64.zip
```

8. 在中准备DB静默安装rsp文件 `/tmp/archive/dbinstall.rsp` 具有以下值的目录：

```
oracle.install.option=INSTALL_DB_SWONLY
UNIX_GROUP_NAME=oinstall
INVENTORY_LOCATION=/u01/app/oraInventory
ORACLE_HOME=/u01/app/oracle/product/19.0.0/db1
ORACLE_BASE=/u01/app/oracle
oracle.install.db.InstallEdition=EE
oracle.install.db.OSDBA_GROUP=dba
oracle.install.db.OSOPER_GROUP=oper
oracle.install.db.OSBACKUPDBA_GROUP=oper
oracle.install.db.OSDGDBA_GROUP=dba
oracle.install.db.OSKMDBA_GROUP=dba
oracle.install.db.OSRACDBA_GROUP=dba
oracle.install.db.rootconfig.executeRootScript=false
```

9. 从db1 home /u01/app/oracle/product/19.0.0/db1中、执行无提示纯软件DB安装。

```
 ./runInstaller -applyRU /tmp/archive/34765931/ -silent
-ignorePrereqFailure -responseFile /tmp/archive/dbinstall.rsp
```

10. 以root用户身份运行 `root.sh` 在仅安装软件后编写脚本。

```
/u01/app/oracle/product/19.0.0/db1/root.sh
```

11. 以Oracle用户身份创建 `dbca.rsp` 包含以下条目的文件：

```
gdbName=db1.demo.netapp.com
sid=db1
createAsContainerDatabase=true
numberOfPDBs=3
pdbName=db1_pdb
useLocalUndoForPDBs=true
pdbAdminPassword="yourPWD"
templateName=General_Purpose.dbc
sysPassword="yourPWD"
systemPassword="yourPWD"
dbsnmpPassword="yourPWD"
datafileDestination=+DATA
recoveryAreaDestination=+LOGS
storageType=ASM
diskGroupName=DATA
characterSet=AL32UTF8
nationalCharacterSet=AL16UTF16
listeners=LISTENER
databaseType=MULTIPURPOSE
automaticMemoryManagement=false
totalMemory=8192
```

12. 以Oracle用户身份、使用dbca启动数据库创建。

```
bin/dbca -silent -createDatabase -responseFile /tmp/archive/dbca.rsp

output:
Prepare for db operation
7% complete
Registering database with Oracle Restart
11% complete
Copying database files
33% complete
Creating and starting Oracle instance
35% complete
38% complete
42% complete
45% complete
48% complete
Completing Database Creation
53% complete
55% complete
56% complete
Creating Pluggable Databases
60% complete
64% complete
69% complete
78% complete
Executing Post Configuration Actions
100% complete
Database creation complete. For details check the logfiles at:
 /u01/app/oracle/cfgtoollogs/dbca/db1.
Database Information:
Global Database Name:db1.demo.netapp.com
System Identifier(SID):db1
Look at the log file "/u01/app/oracle/cfgtoollogs/dbca/db1/db1.log"
for further details.
```

13. 以Oracle用户身份、在创建数据库后验证Oracle Restart HA服务。

```
[oracle@ip-172-30-15-58 db1]$ ../grid/bin/crsctl stat res -t

Name            Target  State        Server                   State
details

Local Resources

ora.DATA.dg     ONLINE  ONLINE       ip-172-30-15-58          STABLE
ora.LISTENER.lsnr   ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.LOGS.dg     ONLINE  ONLINE       ip-172-30-15-58          STABLE
ora.asm     ONLINE  ONLINE       ip-172-30-15-58
Started,STABLE
ora.ons     OFFLINE OFFLINE      ip-172-30-15-58          STABLE

Cluster Resources

ora.cssd            ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.db1.db      ONLINE  ONLINE       ip-172-30-15-58
Open,HOME=/u01/app/oracle/product/19.0.0/db1,STABLE
ora.diskmon     OFFLINE OFFLINE                               STABLE
ora.driver.afd  ONLINE  ONLINE       ip-172-30-15-58          STABLE
ora.evmd        ONLINE  ONLINE       ip-172-30-15-58          STABLE
```

14. 设置Oracle用户 `.bash_profile`。

```
vi ~/.bash_profile
```

15. 添加以下条目：

```
export ORACLE_HOME=/u01/app/oracle/product/19.0.0/db1
export ORACLE_SID=db1
export PATH=$PATH:$ORACLE_HOME/bin
alias asm='export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/grid;export
ORACLE_SID=+ASM;export PATH=$PATH:$ORACLE_HOME/bin'
```

16. 验证已创建的CDB/PDB。

```
/home/oracle/.bash_profile

sqlplus / as sysdba
```

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

NAME      OPEN_MODE

DB1       READ WRITE

SQL> select name from v$datafile;

NAME

+DATA/DB1/DATAFILE/system.256.1132176177
+DATA/DB1/DATAFILE/sysaux.257.1132176221
+DATA/DB1/DATAFILE/undotbs1.258.1132176247
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/system.265.11321
77009
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/sysaux.266.11321
77009
+DATA/DB1/DATAFILE/users.259.1132176247
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/undotbs1.267.113
2177009
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/system.271.11321
77853
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/sysaux.272.11321
77853
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/undotbs1.270.113
2177853
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/users.274.113217
7871

NAME

+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/system.276.11321
77871
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/sysaux.277.11321
77871
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/undotbs1.275.113
2177871
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/users.279.113217
7889
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/system.281.11321
77889
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/sysaux.282.11321
77889
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/undotbs1.280.113
2177889
```

```
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/users.284.113217
7907

19 rows selected.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED

         2 PDB$SEED                       READ ONLY  NO
         3 DB1_PDB1                       READ WRITE NO
         4 DB1_PDB2                       READ WRITE NO
         5 DB1_PDB3                       READ WRITE NO
SQL>
```

17. 将数据库恢复目标大小设置为+logs.磁盘组大小。

```
alter system set db_recovery_file_dest_size = 80G scope=both;
```

18. 使用sqlplus登录到数据库并启用归档日志模式。

```
sqlplus /as sysdba.

shutdown immediate;

startup mount;

alter database archivelog;

alter database open;
```

至此、在适用于ONTAP 和EC2计算实例的Amazon FSX上完成Oracle 19c版本19.18重新启动部署。如果需要、NetApp建议将Oracle控制文件和联机日志文件重新定位到+logs磁盘组。

自动化部署选项

NetApp将随Ansible发布一个完全自动化的解决方案 部署工具包、以便于实施此解决方案。请查看此工具包的可用性。发布后、此处将发布一个链接。

使用SnapCenter 服务备份、还原和克隆Oracle数据库

请参见 "适用于Oracle的SnapCenter 服务" 有关使用NetApp BlueXP控制台备份、还原和克隆Oracle数据库的详细信息。

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- 在安装新数据库的情况下为独立服务器安装Oracle网格基础架构

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-oracle-grid-infrastructure-for-a-standalone-server-with-a-new-database-installation.html#GUID-0B1CEE8C-C893-46AA-8A6A-7B5FAAEC72B3"

- 使用响应文件安装和配置Oracle数据库

  "https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-and-configuring-oracle-database-using-response-files.html#GUID-D53355E9-E901-4224-9A2A-B882070EDDF7"

- 适用于 NetApp ONTAP 的 Amazon FSX

  "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

在AWS EC2和FSX上部署Oracle数据库最佳实践

WP-7357：《在EC2和FSX上部署Oracle数据库最佳实践》简介

NetApp公司的Allen Cao、Niyaz Mohamed、Jeffrey Steiner

许多任务关键型企业级Oracle数据库仍托管在内部、许多企业都希望将这些Oracle数据库迁移到公有 云。这些Oracle数据库通常以应用程序为中心、因此需要用户专用配置、而许多数据库即服务公共云产品都缺少这一功能。因此、当前的数据库环境要求基于公共云的Oracle数据库解决方案 、该数据库是基于高性能、可扩展的计算和存储服务构建的、可满足独特的需求。AWS EC2计算实例和AWS FSX存储服务可能是这个难题的缺失部分、您可以利用这些信息构建任务关键型Oracle数据库工作负载并将其迁移到公有 云。

Amazon Elastic Compute Cloud (Amazon EC2)是一种Web服务、可在云中提供安全、可调整大小的计算容量。它旨在使企业更轻松地进行网络级云计算。通过简单的Amazon EC2 Web服务界面、您可以轻松获得和配置容量、而不会产生任何摩擦。它可以让您完全控制计算资源、并在经过Amazon验证的计算环境中运行。

Amazon FSX for ONTAP 是一种AWS存储服务、它使用行业领先的NetApp ONTAP 块和文件存储、可公开NFS、SMB和iSCSI。借助如此强大的存储引擎、将任务关键型Oracle数据库应用程序重新定位到AWS从未如此简单、其响应时间为亚毫秒级、吞吐量高达多Gbps、每个数据库实例的IOPS超过100、000次。更好的是、FSX存储服务还附带了原生 复制功能、您可以轻松地将内部Oracle数据库迁移到AWS、或者将任务关键型Oracle数据库复制到二级AWS可用性区域以实现HA或DR。

本文档的目标是、提供有关如何使用FSX存储和EC2实例部署和配置Oracle数据库的分步流程、过程和最佳实践指导、该实例可提供与内部系统类似的性能。NetApp还提供了一个自动化工具包、用于自动执行在AWS公有 云

中部署、配置和管理Oracle数据库工作负载所需的大部分任务。

要了解有关解决方案 和用例的更多信息、请观看以下概述视频：

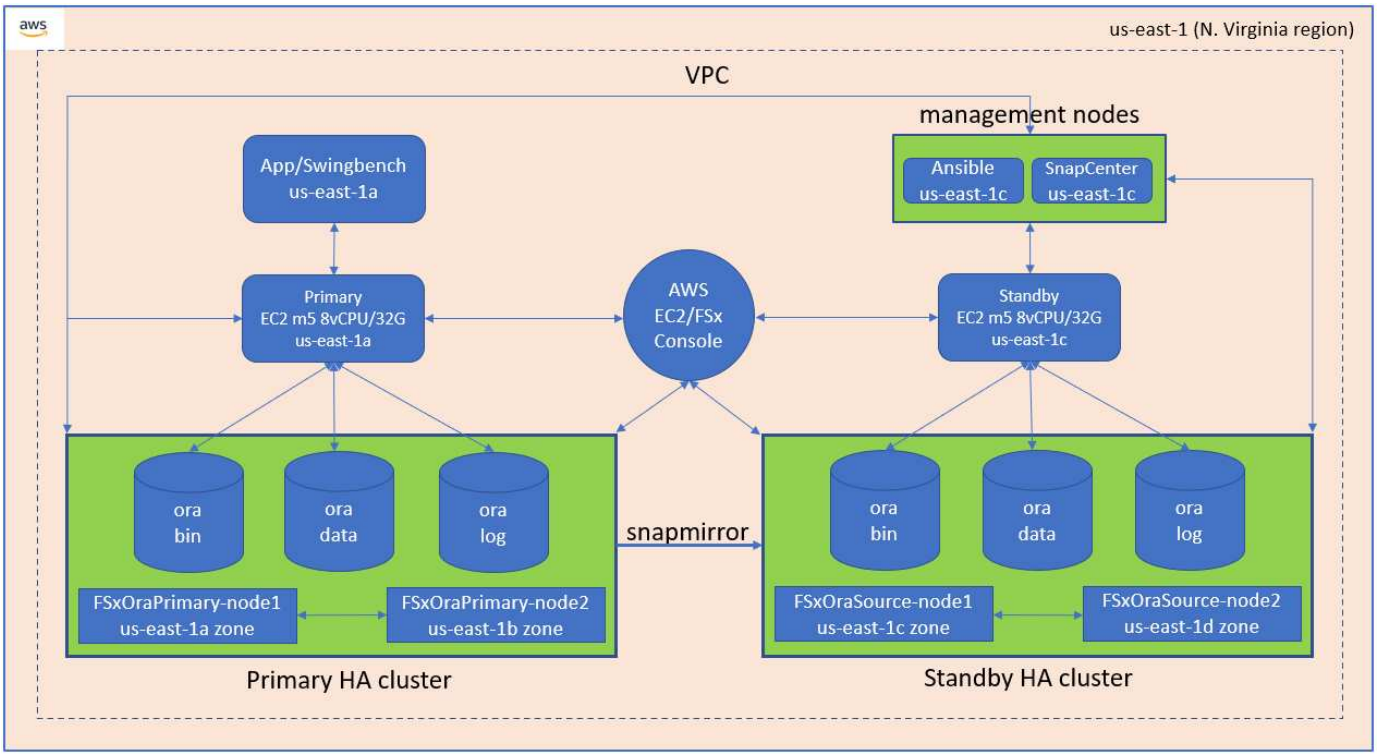
"利用AWS和FSX ONTAP 中的混合云打造现代化的Oracle数据库、第1部分—用例和解决方案 架构"

解决方案架构

以下架构图展示了在具有FSX存储服务的AWS EC2实例上部署高可用性Oracle数据库的情况。可以设置一个类似的部署方案、但在另一个区域中使用备用模式进行灾难恢复。

在环境中、Oracle计算实例通过AWS EC2实例控制台进行部署。控制台提供了多种EC2实例类型。NetApp建议使用RedHat Enterprise Linux 8部署面向数据库的EC2实例类型、例如M5 Ami映像以及高达10Gps的网络带宽。

而FSX卷上的Oracle数据库存储则通过AWS FSX控制台或CLI进行部署。随后、Oracle二进制卷、数据卷或日志卷将显示并挂载到EC2实例Linux主机上。根据使用的底层存储协议、每个数据或日志卷可以分配多个LUN。



FSX存储集群采用双冗余设计、因此主存储集群和备用存储集群都部署在两个不同的可用性区域中。对于所有Oracle二进制卷、数据卷和日志卷、数据库卷会按用户可配置的间隔从主FSX集群复制到备用FSX集群。

此高可用性Oracle环境可通过Ansible控制器节点以及SnapCenter 备份服务器和UI工具进行管理。Oracle安装、配置和复制可使用基于Ansible攻略手册的工具套件实现自动化。可以并行执行对Oracle EC2实例内核操作系统或Oracle修补的任何更新、以使主系统和备用系统保持同步。事实上、初始自动化设置可以轻松地进行扩展、以便在需要时执行某些重复的日常Oracle任务。

SnapCenter 提供了用于Oracle数据库时间点恢复或在主分区或备用分区(如果需要)克隆数据库的工作流。通过SnapCenter UI、您可以根据RTO或RPO目标将Oracle数据库备份和复制配置为备用FSX存储、以实现高可用性或灾难恢复。

解决方案 提供了一个替代过程、可提供与Oracle RAC和Data Guard部署中提供的功能类似的功能。

部署Oracle数据库时需要考虑的因素

公有 云为计算和存储提供了多种选择、使用正确类型的计算实例和存储引擎是开始部署数据库的好地方。您还应选择针对Oracle数据库进行优化的计算和存储配置。

以下各节介绍了在使用FSX存储的EC2实例上的AWS公有 云中部署Oracle数据库时的主要注意事项。

虚拟机性能

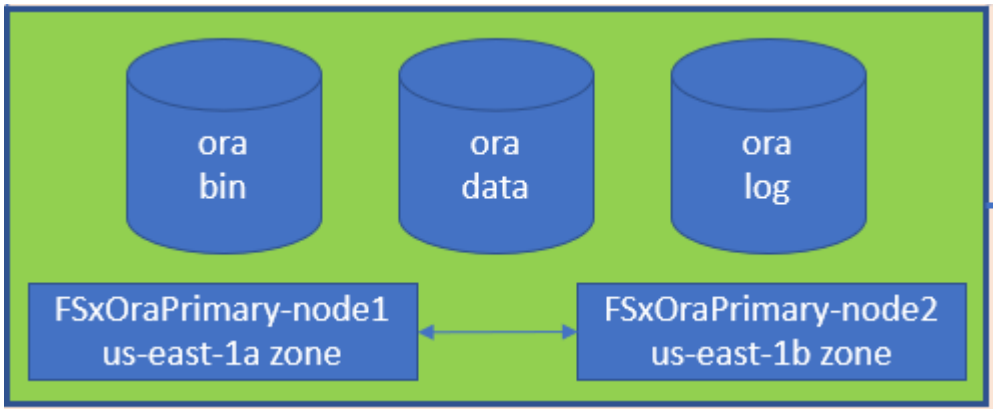
选择合适的 VM 大小对于公有云中关系数据库的最佳性能非常重要。为了提高性能、NetApp建议在Oracle部署中使用EC2 M5系列实例、该实例针对数据库工作负载进行了优化。AWS还会使用相同的实例类型为Oracle的RDS实例提供支持。

- 根据工作负载特征选择正确的vCPU和RAM组合。
- 向VM添加交换空间。默认的EC2实例部署不会创建交换空间、而交换空间对于数据库来说并不是最佳选择。

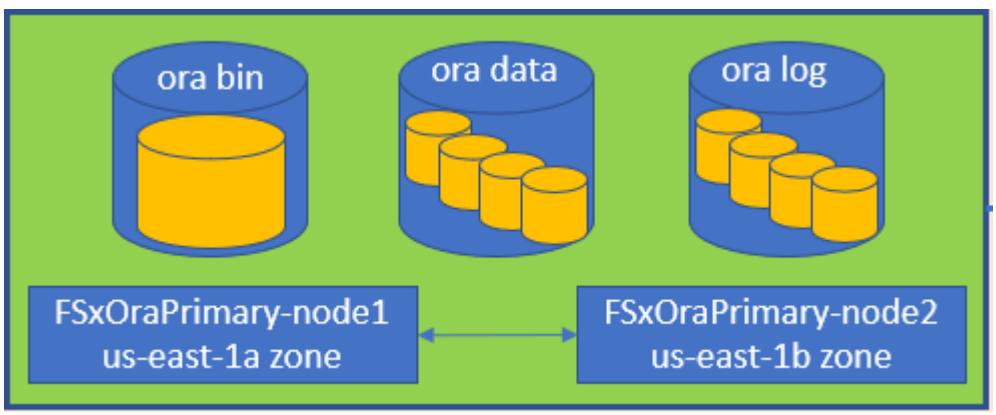
存储布局和设置

NetApp建议采用以下存储布局：

- 对于NFS存储、建议的卷布局为三个卷：一个用于Oracle二进制文件；一个用于Oracle数据和一个重复的控制文件；一个用于Oracle活动日志、归档日志和控制文件。



- 对于iSCSI存储、建议的卷布局为三个卷：一个用于Oracle二进制文件；一个用于Oracle数据和一个重复的控制文件；一个用于Oracle活动日志、归档日志和控制文件。但是、理想情况下、每个数据卷和日志卷都应包含四个LUN。在HA集群节点上、LUN的平衡性最佳。



- 对于存储IOPS和吞吐量、您可以为FSX存储集群的已配置IOPS和吞吐量选择阈值、并且可以随时在工作负载发生变化时动态调整这些参数。
  - 自动IOPS设置是、在已分配存储容量或用户定义的存储中、每个GiB三个IOPS、最多80、000个。
  - 吞吐量级别将按以下方式递增：128、256、512、1024、2045 Mbps。

查看 "适用于NetApp ONTAP 性能的Amazon FSX" 估算吞吐量和IOPS规模时的文档。

NFS 配置

最常见的操作系统Linux具有原生 NFS功能。Oracle提供的Direct NFS (DNFS)客户端本机集成到Oracle中。Oracle支持NFSv3已超过20年。所有Oracle版本的NFSv3均支持DNFS。遵循NFSv4标准的所有操作系统均支持NFSv4。NFSv4的DNFS支持需要Oracle 12.1.0.2或更高版本。NFSv4.1需要特定的操作系统支持。有关受支持的操作系统、请参见NetApp互操作性表工具(IMT)。NFSv4.1的DNFS支持需要Oracle 19.3.0.0或更高版本。

使用NetApp自动化工具包自动部署Oracle会自动在NFSv3上配置DNFS。

需要考虑的其他因素：

- TCP插槽表是主机总线适配器(host-bus-adapter、HBA)队列深度的NFS等效项。这些表可控制任何时候都可以处理的NFS操作的数量。默认值通常为16、该值太低、无法实现最佳性能。在较新的Linux内核上会出现相反的问题、这会自动将TCP插槽表限制增加到使NFS服务器充满请求的级别。

  为了获得最佳性能并防止出现性能问题、请将控制TCP插槽表的内核参数调整为128。

  ```
  sysctl -a | grep tcp.*.slot_table
  ```

- 下表提供了适用于Linux NFSv3 -单个实例的建议NFS挂载选项。

File Type	Mount Options
• Control files • Data files • Redo logs	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
• ORACLE_HOME • ORACLE_BASE	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536

在使用DNFS之前、请验证是否已安装Oracle文档1495104.1中所述的修补程序。NetApp针对NFSv3和NFSv4的支持列表不包括特定的操作系统。支持所有符合RFC的操作系统。在联机IMT 中搜索NFSv3或NFSv4支持时、请勿选择特定操作系统、因为不会显示任何匹配项。常规策略隐式支持所有操作系统。

高可用性

如解决方案 架构所示、HA基于存储级别复制构建。因此、Oracle的启动和可用性取决于计算和存储的启动和恢复速度。请参见以下主要因素：

- 准备好备用计算实例、并通过向两个主机进行Ansible并行更新与主实例同步。

- 从主系统复制二进制卷以供备用、这样您就不需要在最后一分钟安装Oracle并确定需要安装和修补的内容。
- 复制频率决定了恢复Oracle数据库以提供服务的速度。复制频率与存储消耗之间存在一定的权衡。
- 利用自动化功能快速恢复并切换到备用模式、不会出现人为错误。为此、NetApp提供了一个自动化工具包。

AWS EC2和FSX上的Oracle分步部署过程

本节介绍使用FSX存储部署Oracle RDS自定义数据库的部署过程。

通过EC2控制台部署适用于Oracle的EC2 Linux实例

如果您是AWS的新用户、则首先需要设置AWS环境。AWS网站登录页面上的文档选项卡提供了有关如何部署Linux EC2实例的EC2说明链接、该实例可用于通过AWS EC2控制台托管Oracle数据库。以下部分总结了这些步骤。有关详细信息、请参见链接的AWS EC2专用文档。

设置AWS EC2环境

您必须创建一个AWS帐户来配置必要的资源、以便在EC2和FSX服务上运行Oracle环境。以下AWS文档提供了必要的详细信息：

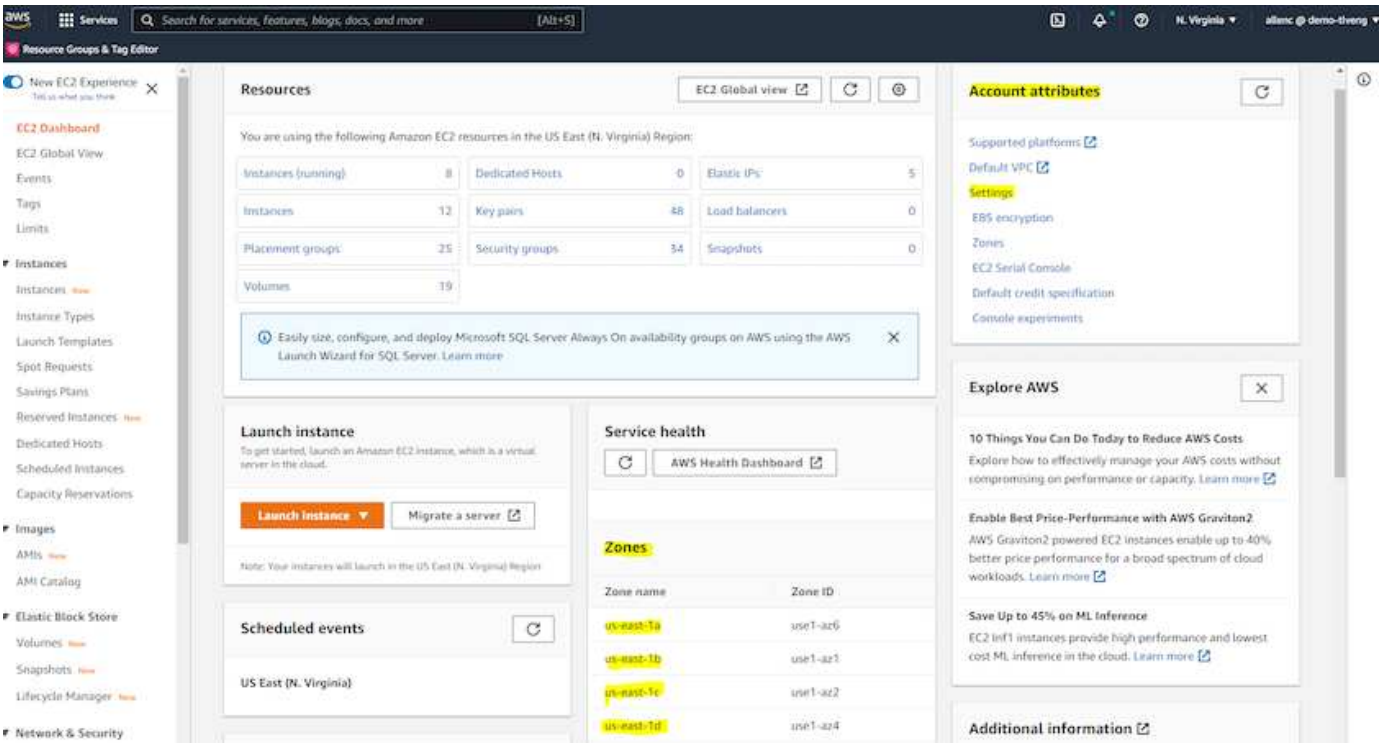
- "设置为使用Amazon EC2"

主要主题：

- 注册AWS。
- 创建密钥对。
- 创建安全组。

在AWS帐户属性中启用多个可用性区域

对于架构图中所示的Oracle高可用性配置、您必须在一个区域中至少启用四个可用性区域。多个可用性区域也可以位于不同区域、以满足灾难恢复所需的距离。



创建并连接到**EC2**实例以托管**Oracle**数据库

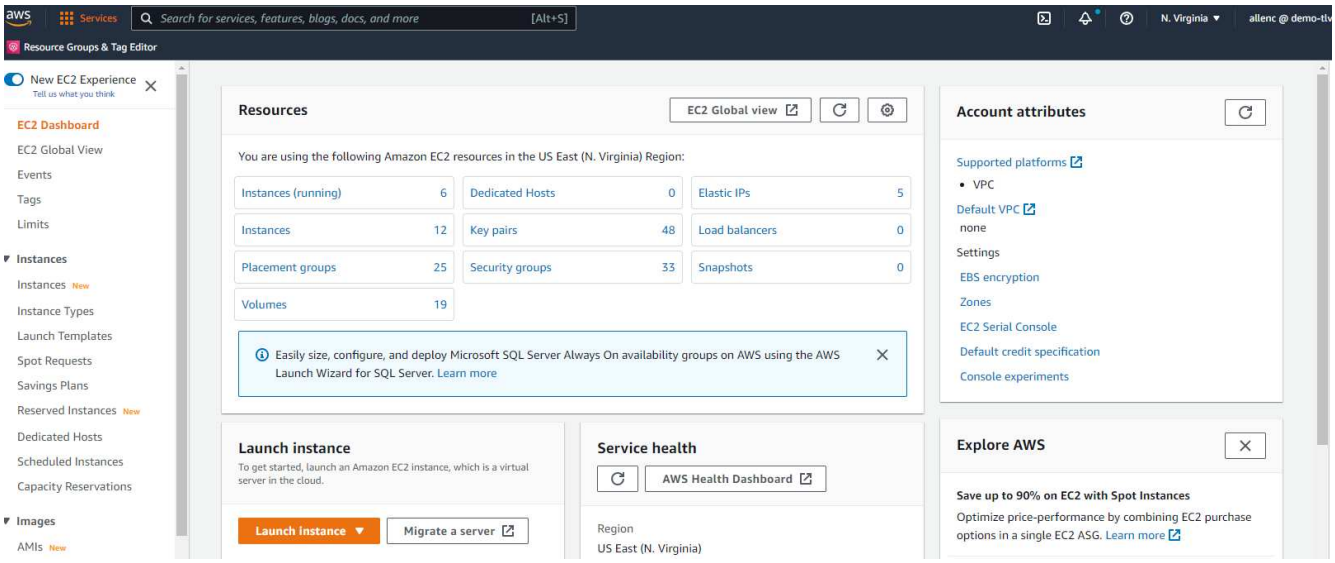
请参见教程 "开始使用Amazon EC2 Linux实例" 分步部署过程和最佳实践。

主要主题：

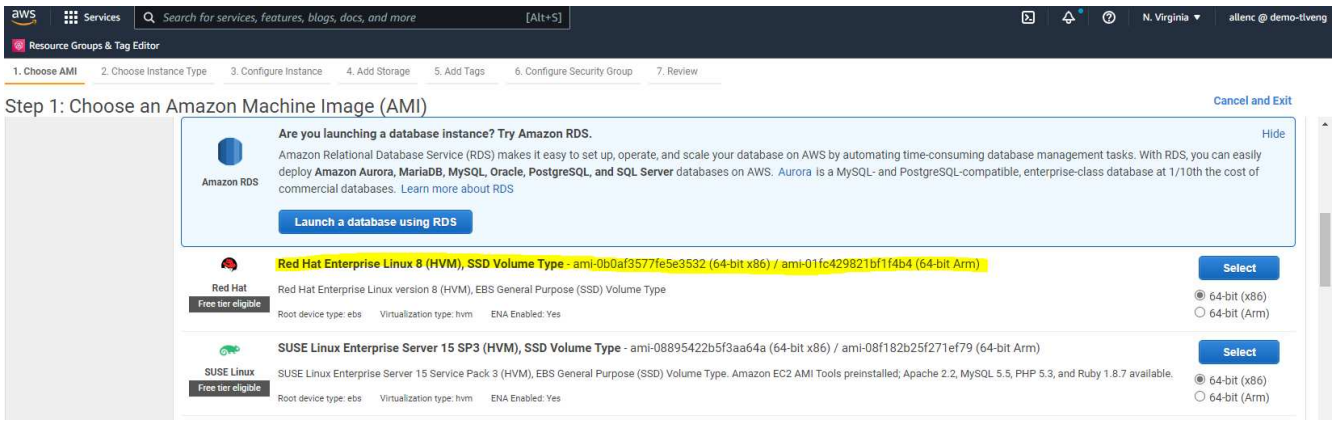
- 概述。
- 前提条件。
- 第1步：启动实例。
- 第2步：连接到实例。
- 第3步：清理实例。

以下屏幕截图展示了如何使用EC2控制台部署M5类型的Linux实例以运行Oracle。

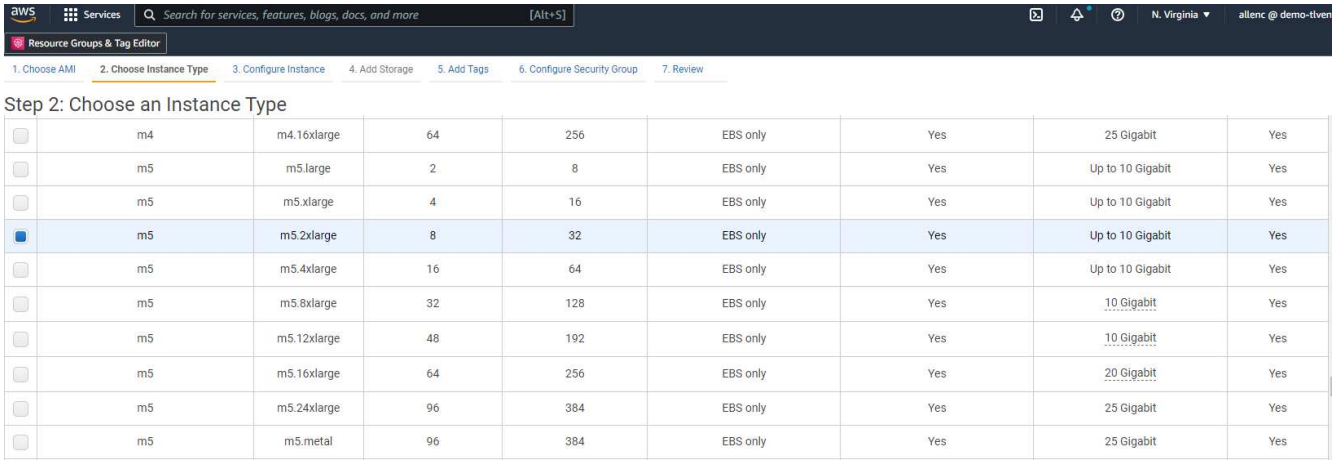
1. 在EC2信息板中、单击黄色的Launch Instance按钮以启动EC2实例部署工作流。



2. 在第1步中、选择"Red Hat Enterprise Linux 8 (HVM)、SSD卷类型- AMI-0b0af3577fe5e3532 (64位x86)/AMI-01fc429821bf1f4b4 (64位ARM)"。

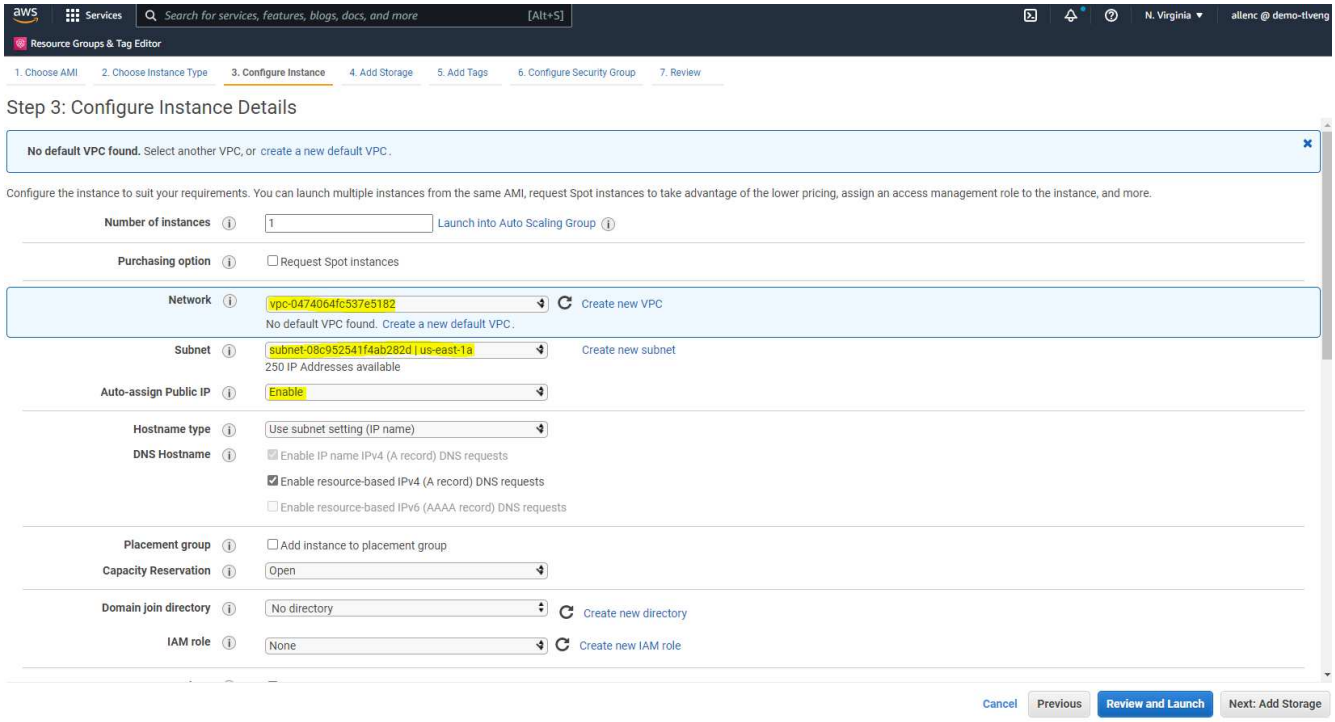


3. 在步骤2中、根据Oracle数据库工作负载选择一个M5实例类型、并分配适当的CPU和内存。单击"下一步：配置实例详细信息"。

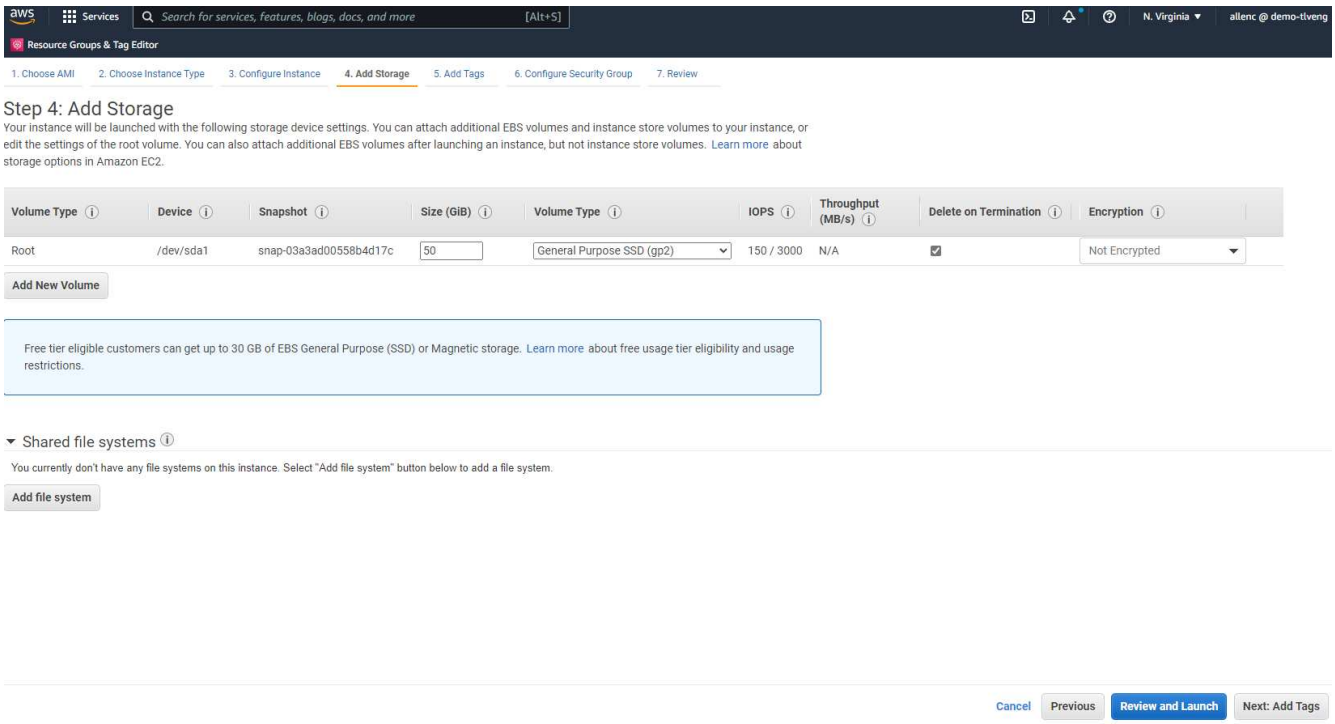


	m4	m4.16xlarge	64	256	EBS only	Yes	25 Gigabit	Yes
	m5	m5.large	2	8	EBS only	Yes	Up to 10 Gigabit	Yes
	m5	m5.xlarge	4	16	EBS only	Yes	Up to 10 Gigabit	Yes
■	m5	m5.2xlarge	8	32	EBS only	Yes	Up to 10 Gigabit	Yes
	m5	m5.4xlarge	16	64	EBS only	Yes	Up to 10 Gigabit	Yes
	m5	m5.8xlarge	32	128	EBS only	Yes	10 Gigabit	Yes
	m5	m5.12xlarge	48	192	EBS only	Yes	10 Gigabit	Yes
	m5	m5.16xlarge	64	256	EBS only	Yes	20 Gigabit	Yes
	m5	m5.24xlarge	96	384	EBS only	Yes	25 Gigabit	Yes
	m5	m5.metal	96	384	EBS only	Yes	25 Gigabit	Yes

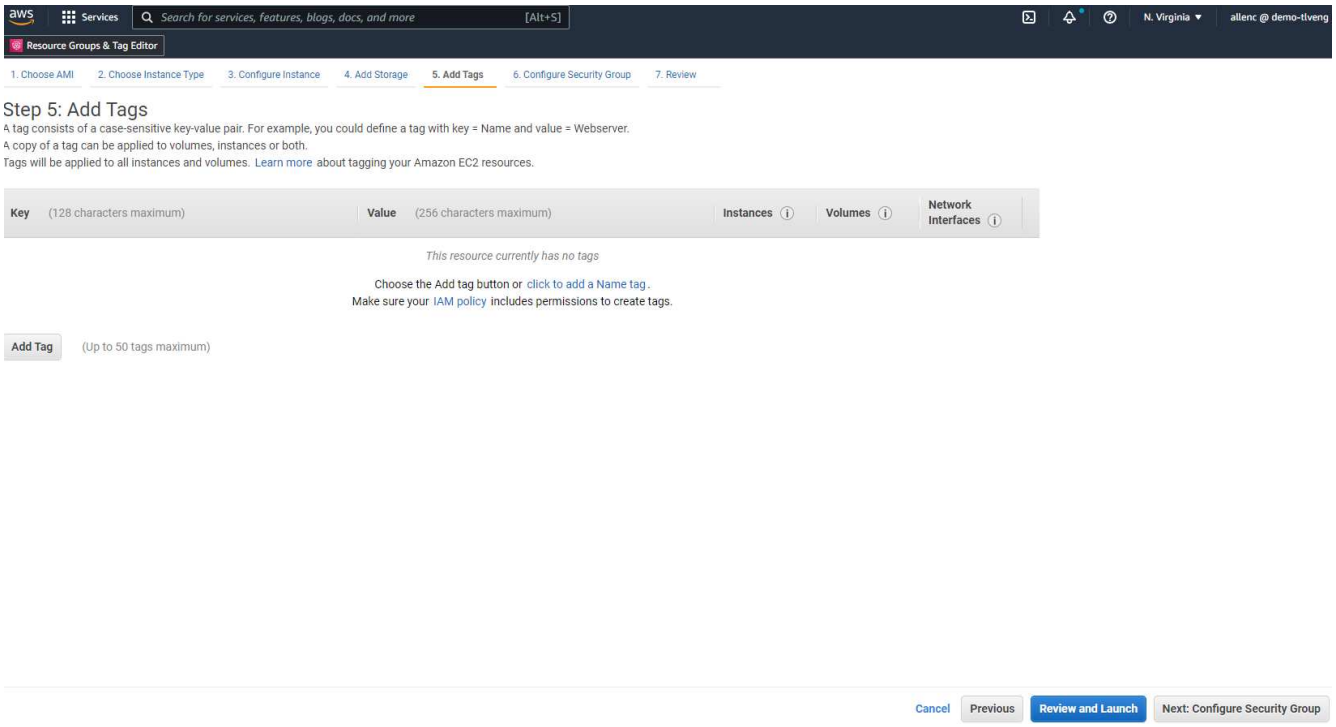
4. 在步骤3中、选择应放置实例的VPC和子网、并启用公有 IP分配。单击"下一步：添加存储"。



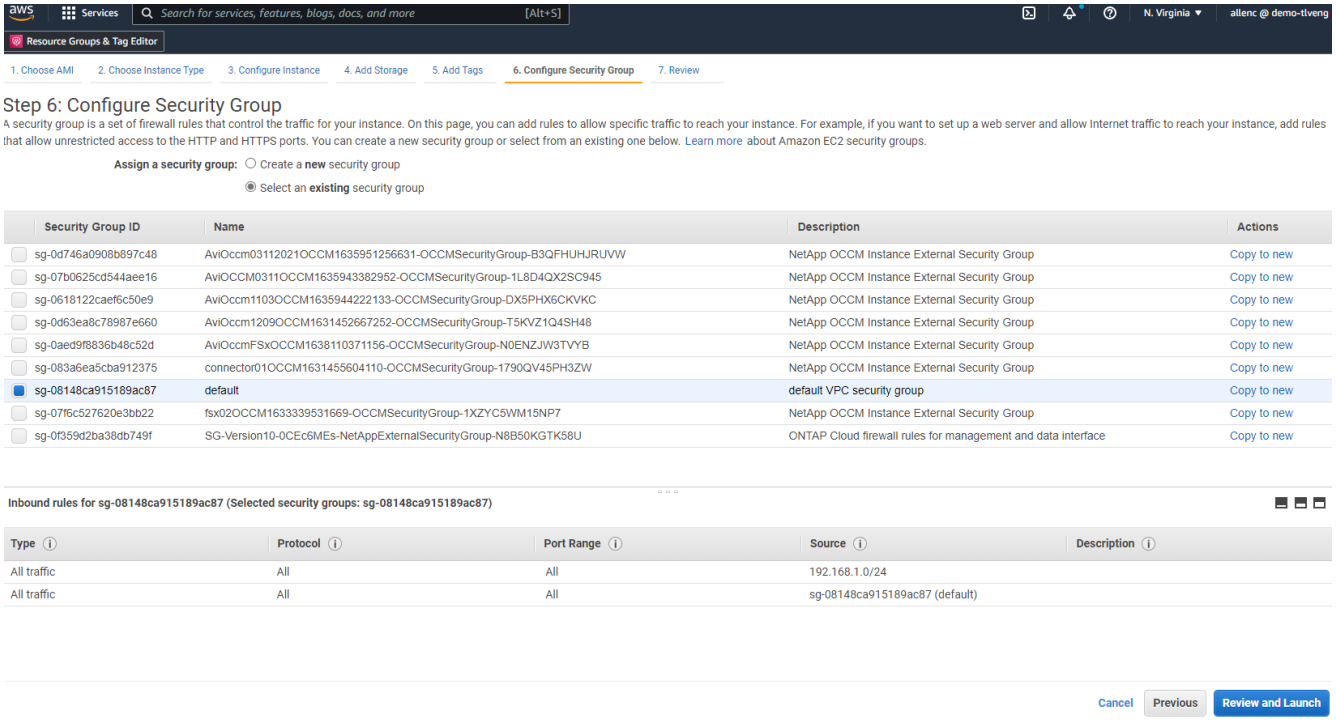
5. 在步骤4中、为根磁盘分配足够的空间。您可能需要空间来添加交换。默认情况下、EC2实例分配的交换空间为零、这不是运行Oracle的最佳选择。



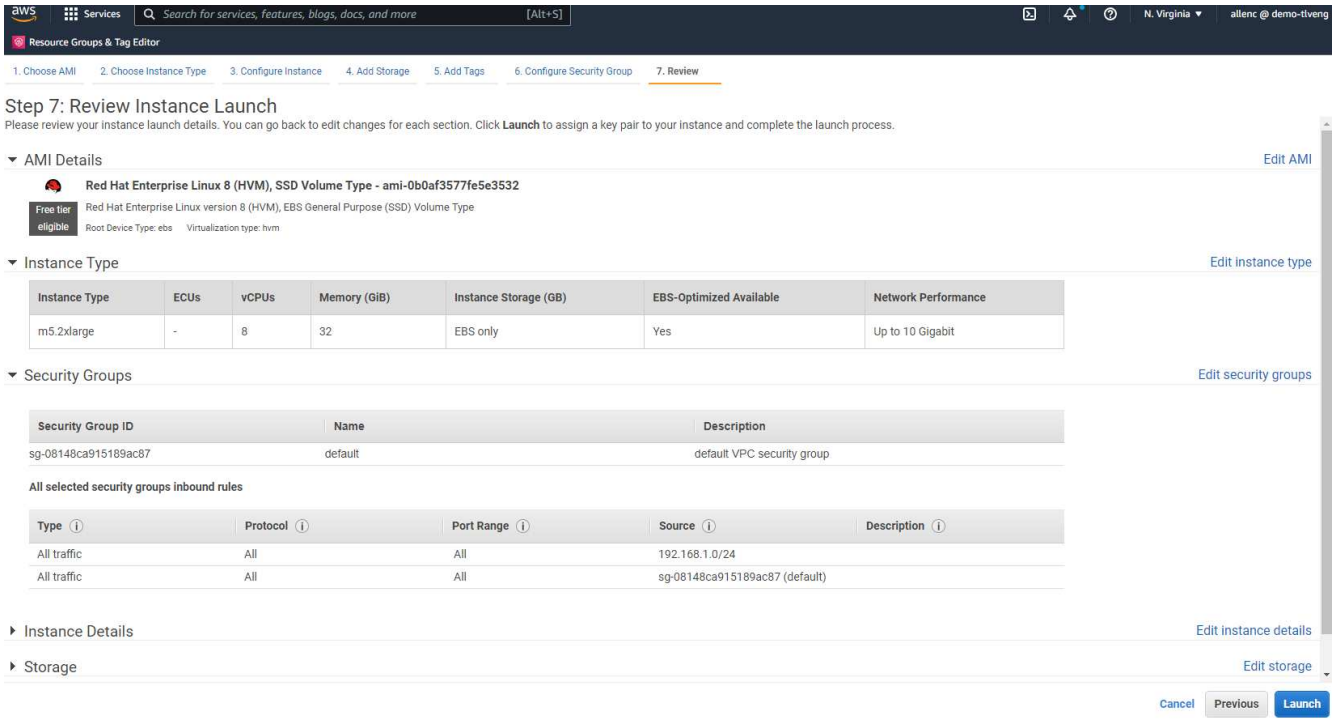
6. 在步骤5中、根据需要添加用于实例标识的标记。

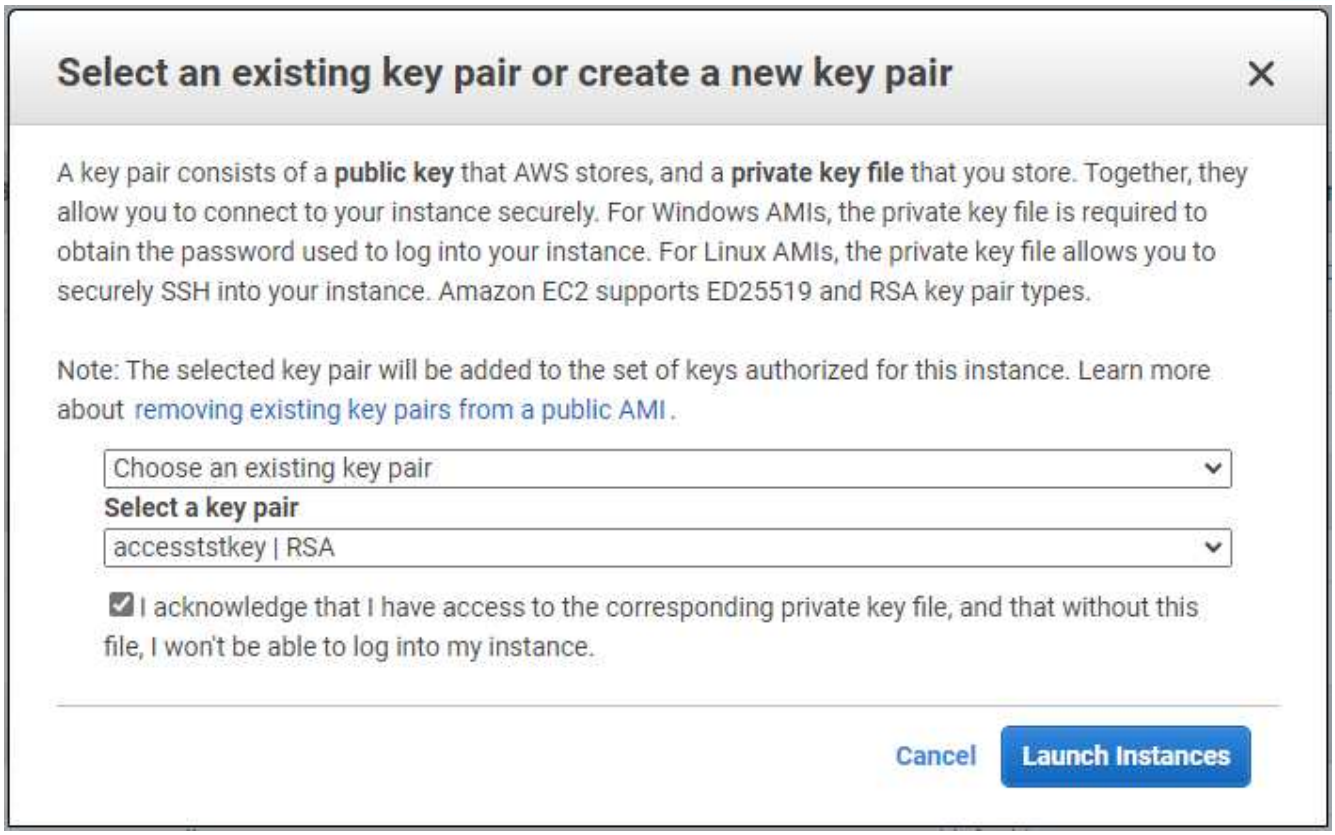


7. 在第6步中、选择现有安全组或使用所需的实例入站和出站策略创建一个新安全组。



8. 在第7步中、查看实例配置摘要、然后单击启动以启动实例部署。系统将提示您创建密钥对或选择密钥对以访问实例。





9. 使用SSH密钥对登录到EC2实例。根据需要更改密钥名称和实例IP地址。

```
ssh -i ora-db1v2.pem ec2-user@54.80.114.77
```

您需要在其指定可用性区域中创建两个EC2实例作为主Oracle服务器和备用Oracle服务器、如架构图所示。

为ONTAP 文件系统配置FSX以存储Oracle数据库

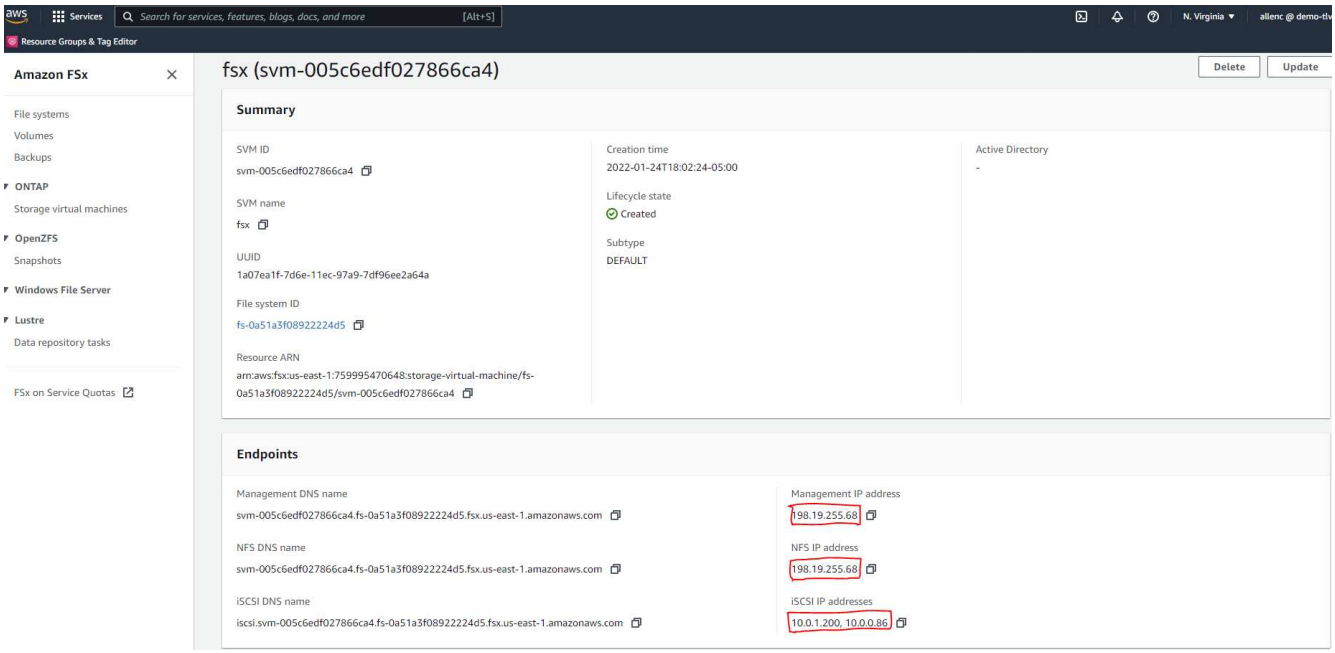
EC2实例部署会为操作系统分配EBS根卷。适用于ONTAP 文件系统的FSX可提供Oracle数据库存储卷、包括Oracle二进制卷、数据卷和日志卷。FSX存储NFS卷可以从AWS FSX控制台或Oracle安装进行配置、也可以通过配置自动化在自动化参数文件中按照用户配置的方式分配卷。

为ONTAP 文件系统创建FSX

已参考此文档 "管理适用于ONTAP 文件系统的FSX" 用于为ONTAP 文件系统创建FSX。

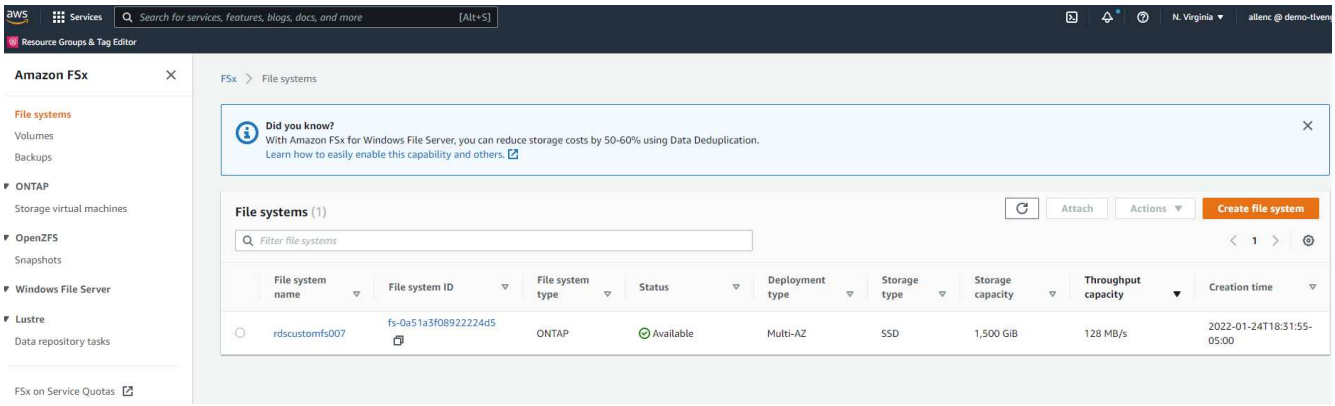
主要注意事项：

- SSD存储容量。最小1024 GiB、最大192 TiB。
- 已配置SSD IOPS。根据工作负载要求、每个文件系统最多可达到80、000 SSD IOPS。
- 吞吐量容量。
- 设置管理员fsxadmin/vsadmin密码。FSX配置自动化所需。
- 备份和维护。禁用自动每日备份；数据库存储备份通过SnapCenter 计划执行。
- 从SVM详细信息页面检索SVM管理IP地址以及特定于协议的访问地址。FSX配置自动化所需。

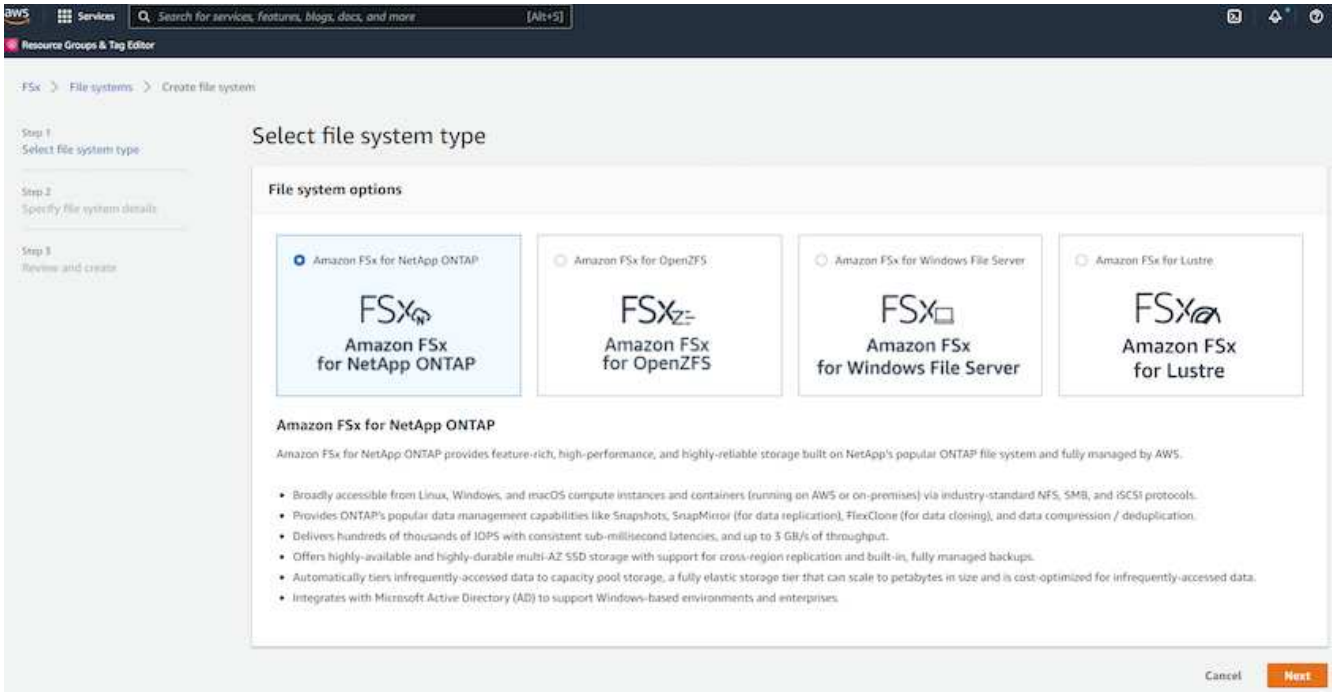


有关设置主HA FSX集群或备用HA FSX集群的步骤、请参见以下分步过程。

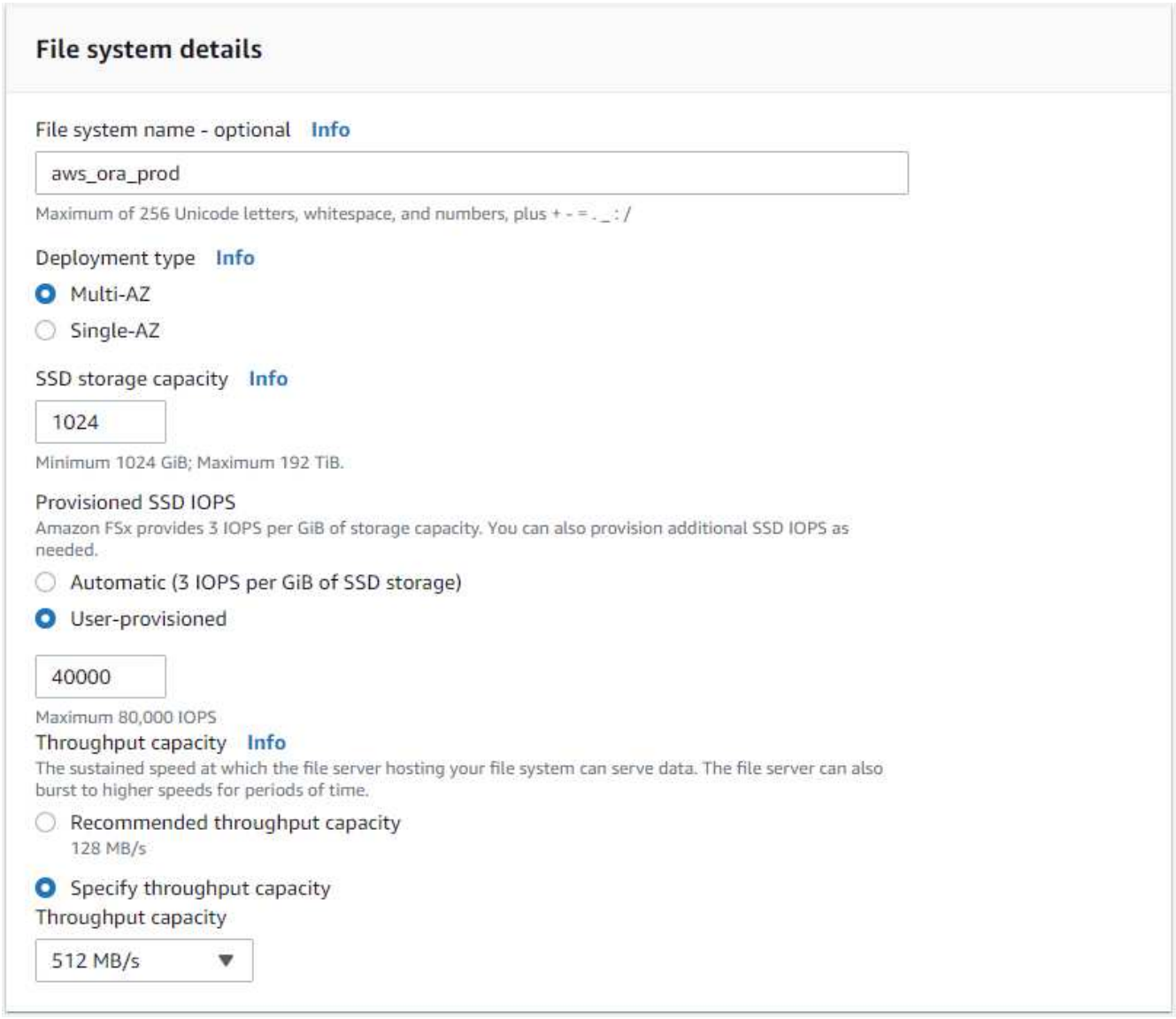
1. 在FSX控制台中、单击Create File System以启动FSX配置工作流。



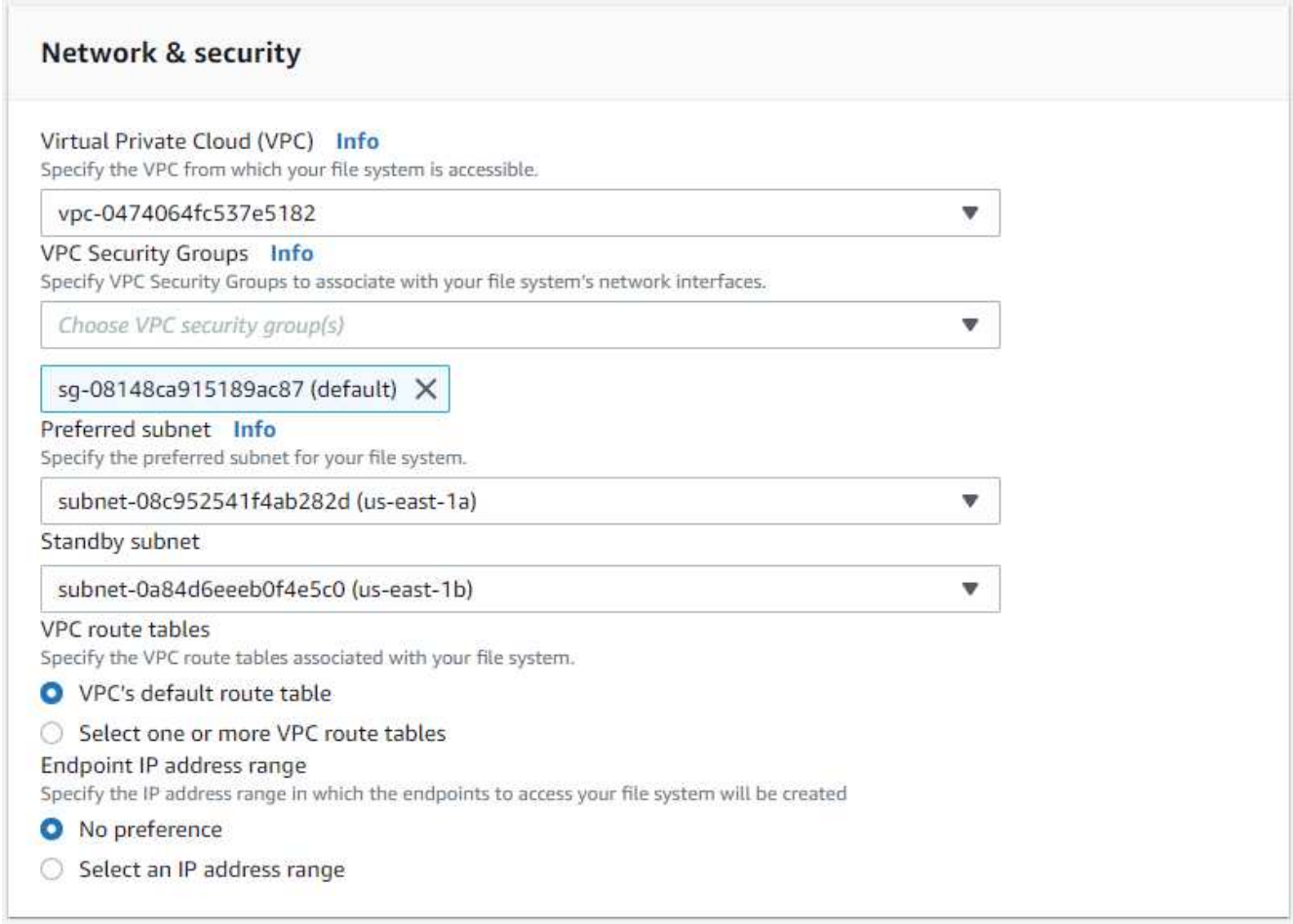
2. 选择适用于NetApp ONTAP 的Amazon FSX。然后单击下一步。



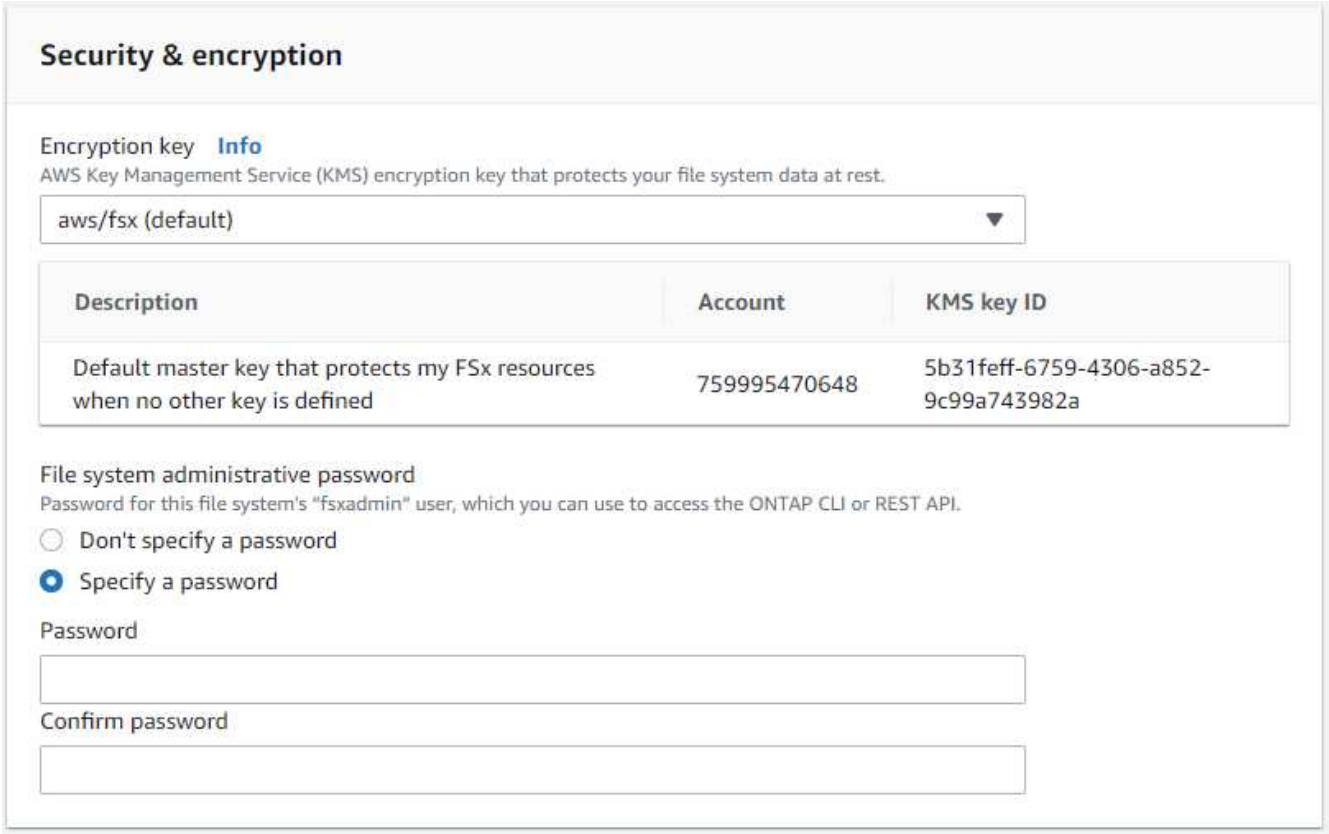
3. 选择标准创建、然后在文件系统详细信息中将文件系统命名为Multi-AZ HA。根据您的数据库工作负载、选择自动或用户配置的IOPS、最高可达80、000 SSD IOPS。FSX存储在后端提供高达2 TiB的NVMe缓存、可提供更高的测量IOPS。



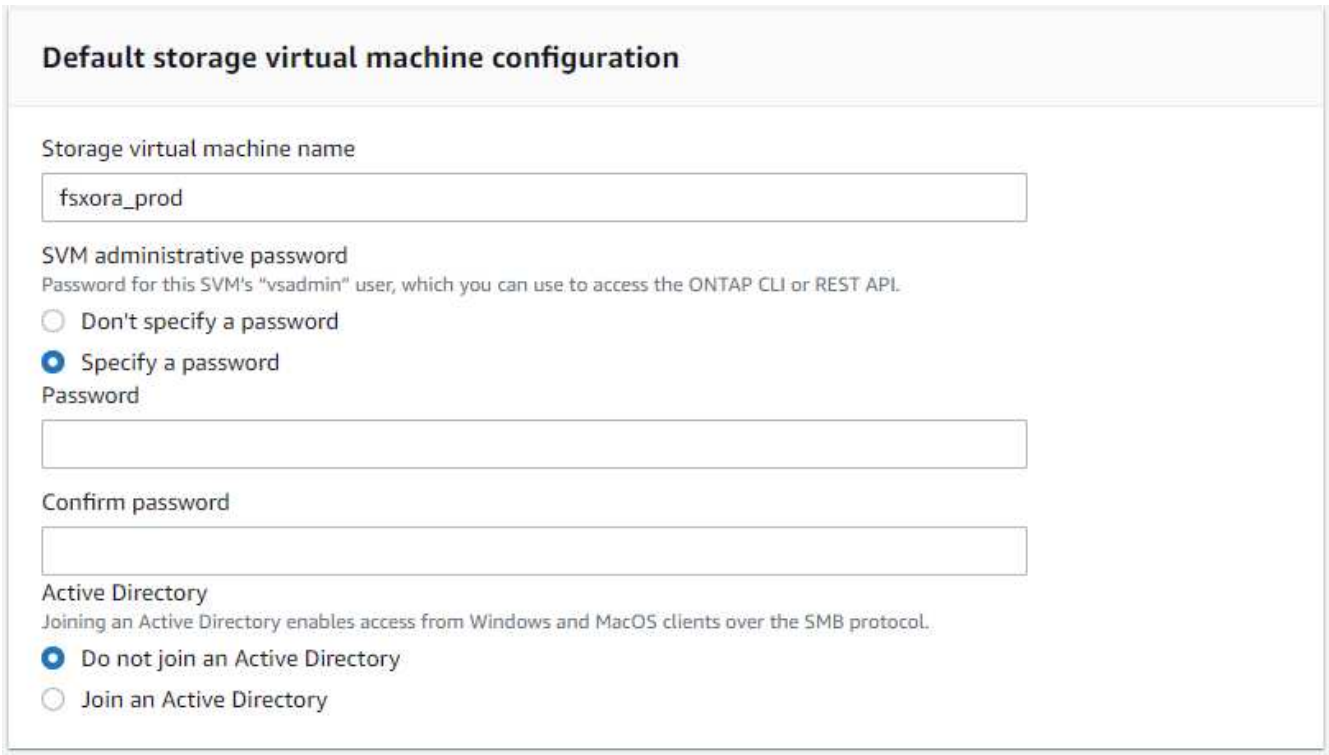
4. 在网络和安全部分中、选择VPC、安全组和子网。应在部署FSX之前创建这些卷。根据FSX集群的角色(主或备用)、将FSX存储节点置于相应的分区中。



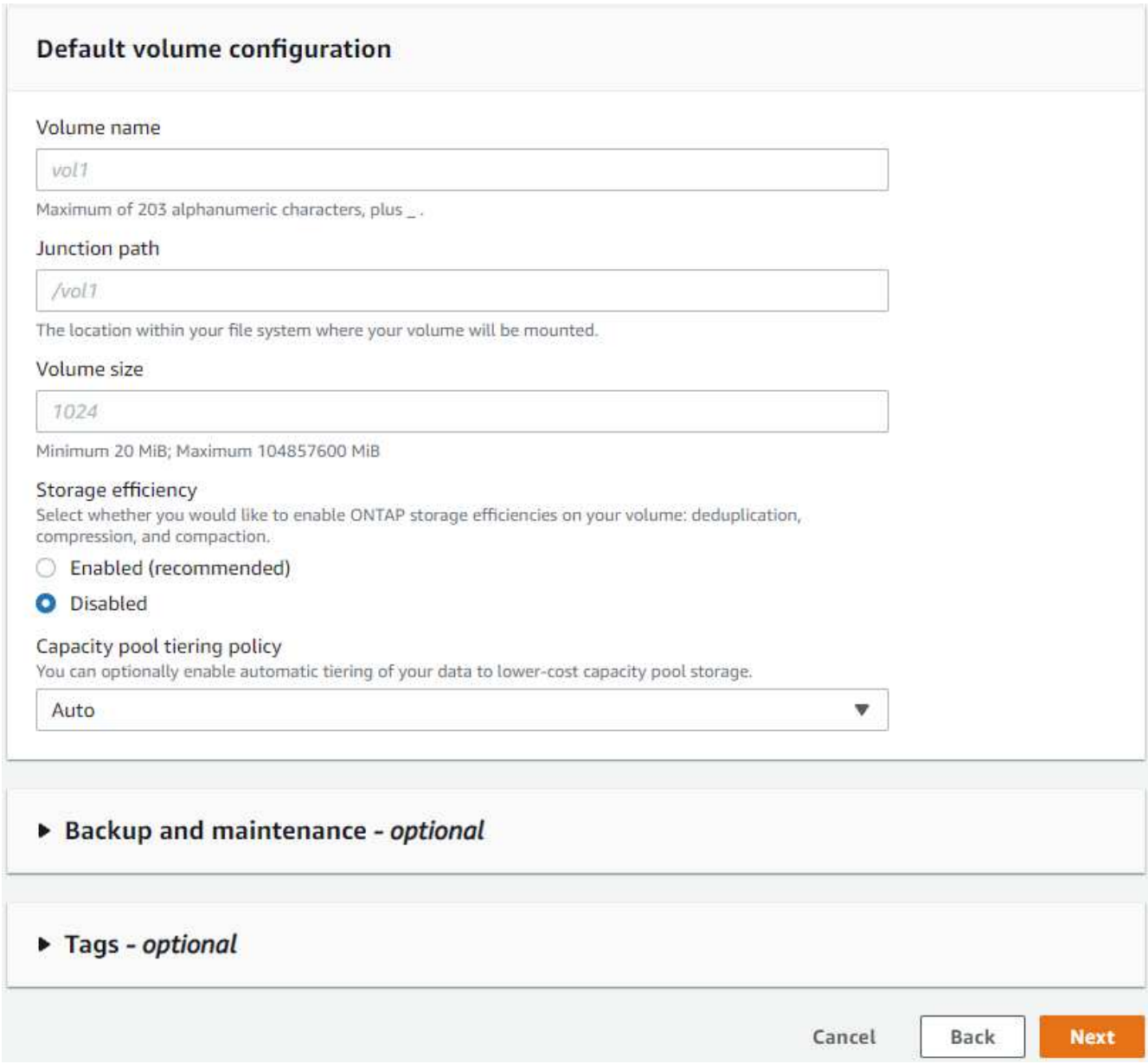
5. 在安全性和加密部分中、接受默认值、然后输入fsxadmin密码。



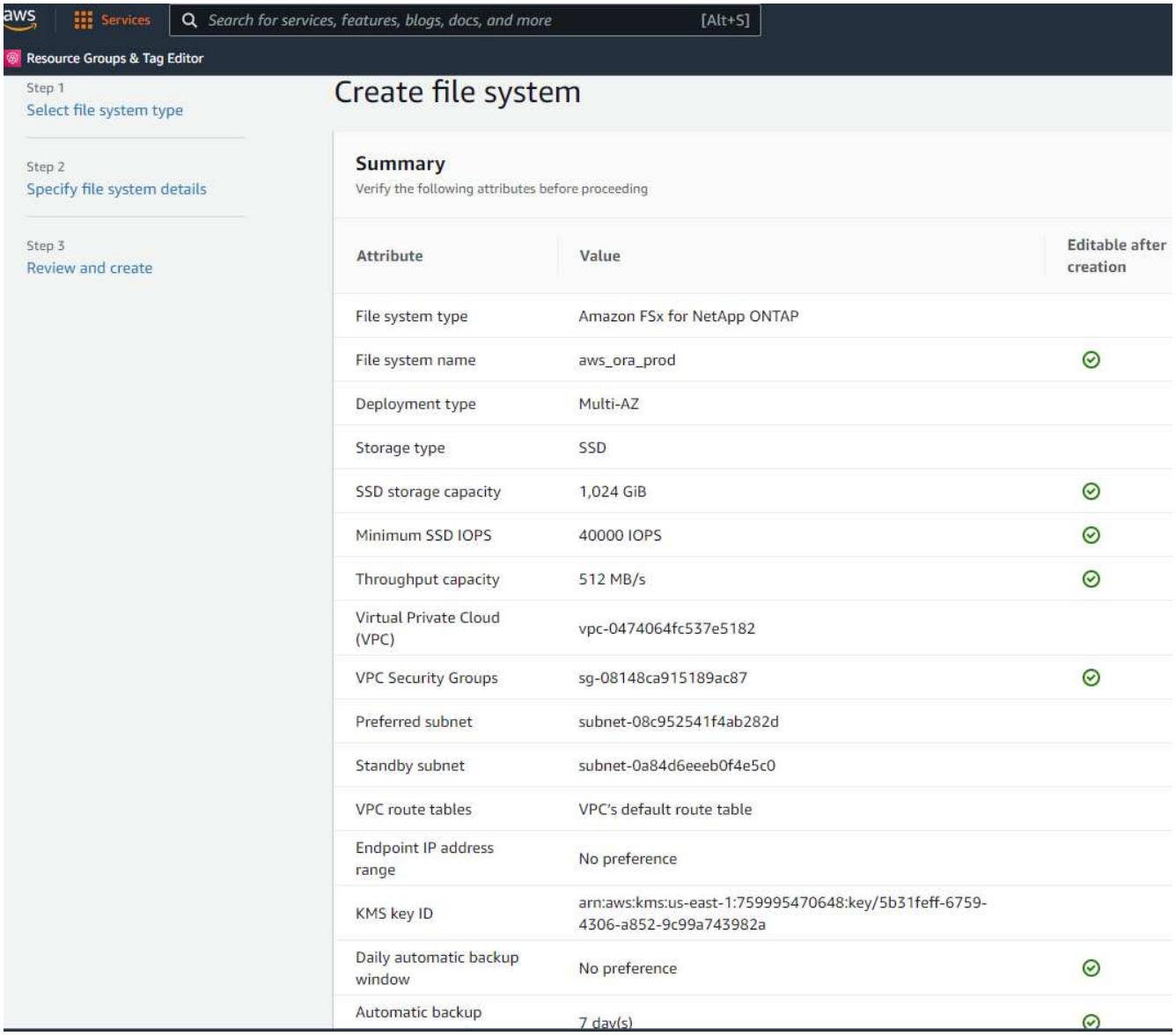
6. 输入SVM名称和vsadmin密码。



7. 将卷配置留空；此时不需要创建卷。



8. 查看摘要页面、然后单击创建文件系统以完成FSX文件系统配置。



为**Oracle**数据库配置数据库卷

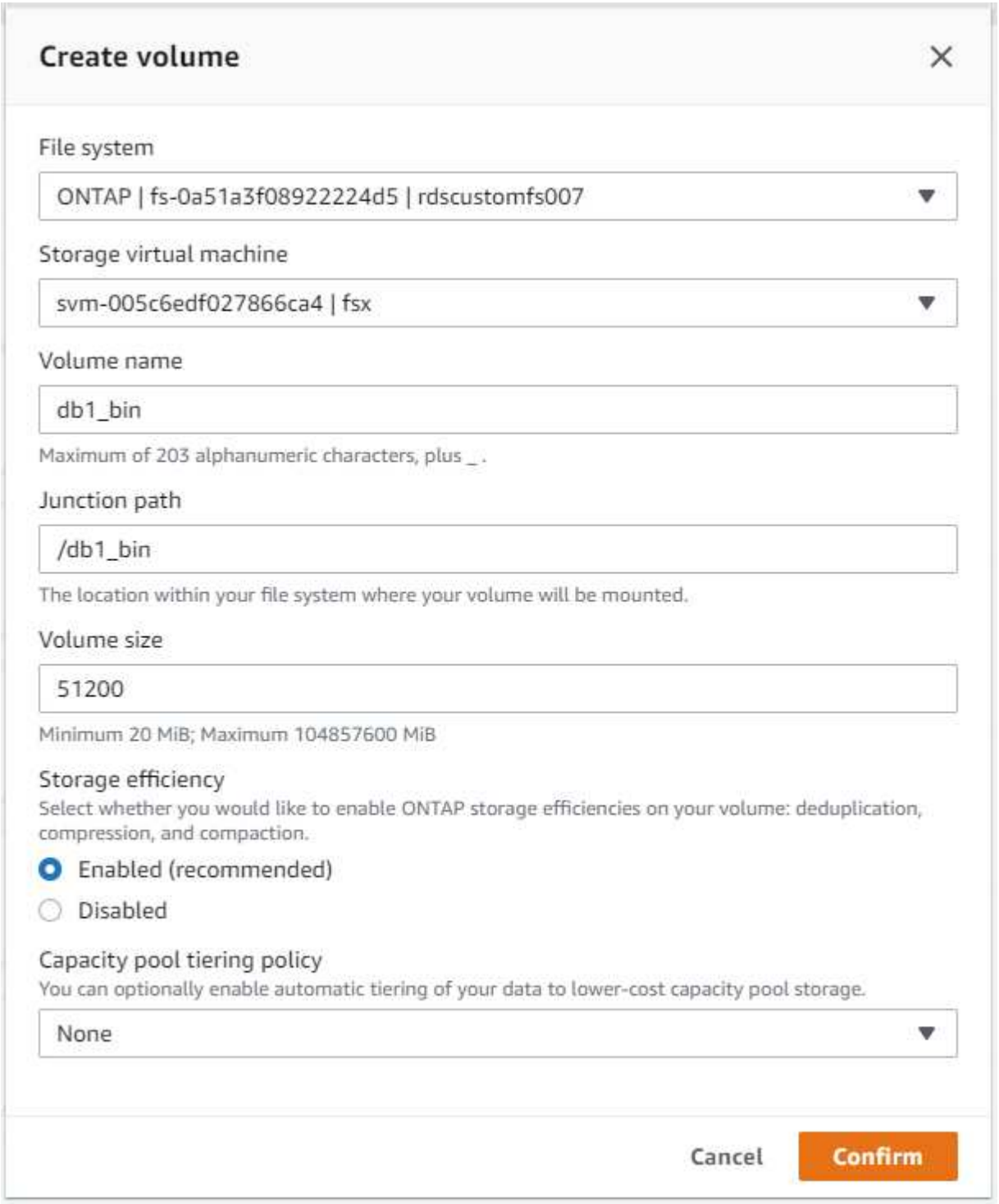
请参见 "管理ONTAP 卷的FSX—创建卷" 了解详细信息。

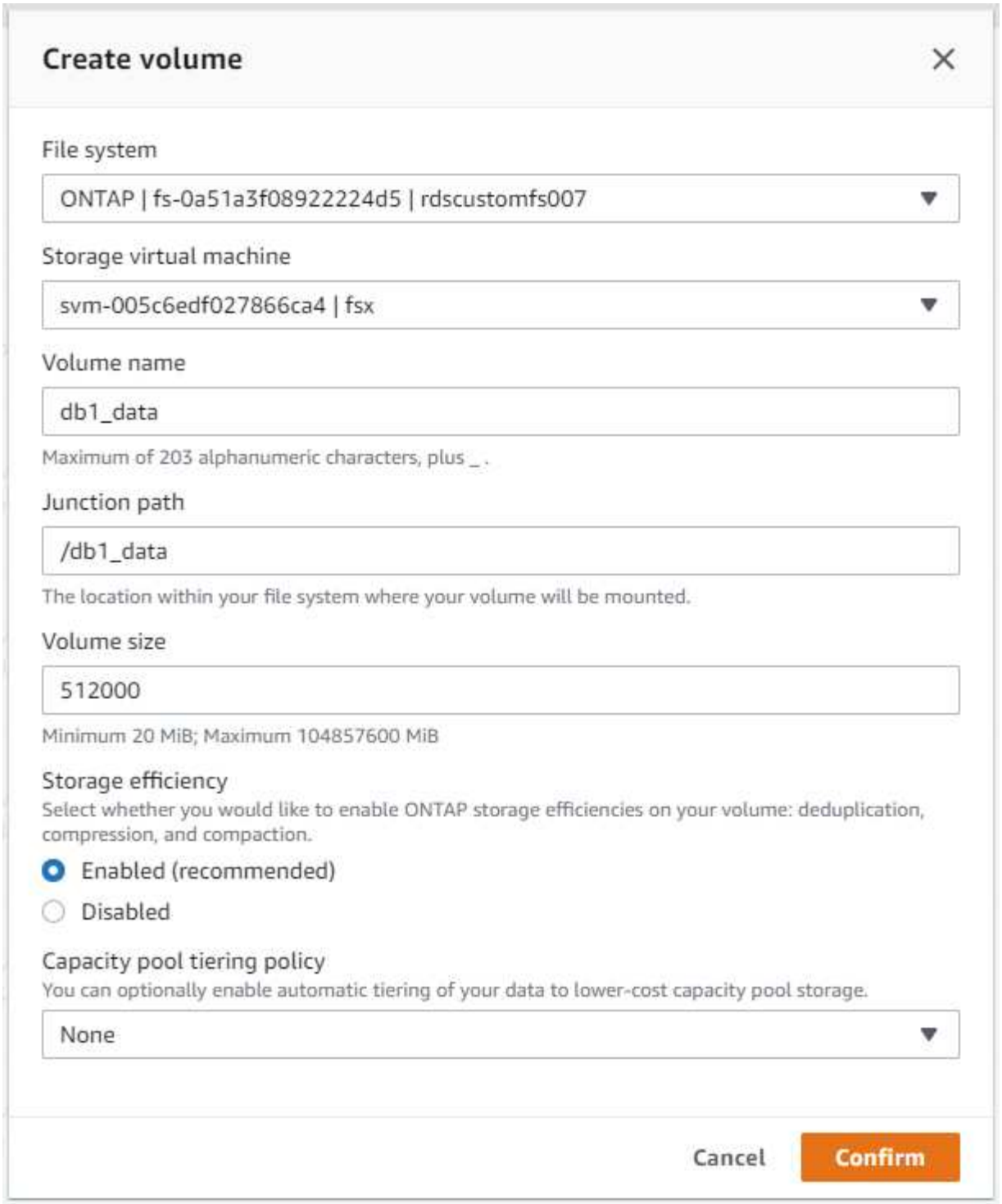
主要注意事项：

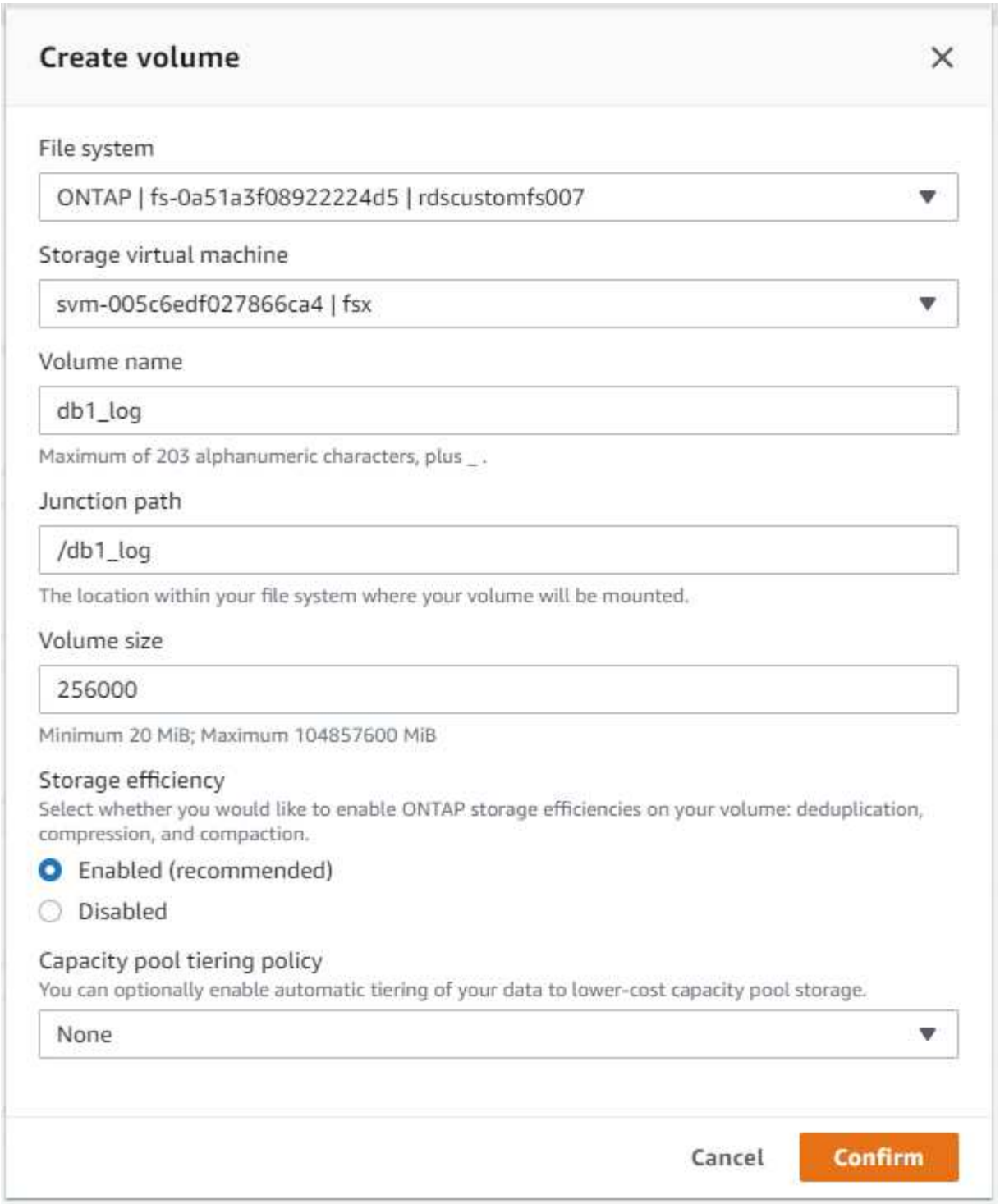
- 适当调整数据库卷的大小。
- 为性能配置禁用容量池分层策略。
- 为NFS存储卷启用Oracle DNFS。
- 为iSCSI存储卷设置多路径。

从**FSX**控制台创建数据库卷

在AWS FSX控制台中、您可以为Oracle数据库文件存储创建三个卷：一个用于Oracle二进制文件、一个用于Oracle数据、一个用于Oracle日志。请确保卷命名与Oracle主机名(在自动化工具包中的hosts文件中定义)匹配、以便正确识别。在此示例中、我们使用db1作为EC2 Oracle主机名、而不是使用典型的基于IP地址的主机名作为EC2实例。


Create volume
File system
ONTAP | fs-0a51a3f08922224d5 | rdscustomfs007
Storage virtual machine
svm-005c6edf027866ca4 | fsx
Volume name
db1_bin
Maximum of 203 alphanumeric characters, plus _ .
Junction path
/db1_bin
The location within your file system where your volume will be mounted.
Volume size
51200
Minimum 20 MiB; Maximum 104857600 MiB
Storage efficiency
Select whether you would like to enable ONTAP storage efficiencies on your volume: deduplication, compression, and compaction.
Enabled (recommended)
Disabled
Capacity pool tiering policy
You can optionally enable automatic tiering of your data to lower-cost capacity pool storage.
None
Cancel
Confirm


Create volume
File system
ONTAP | fs-0a51a3f08922224d5 | rdscustomfs007
Storage virtual machine
svm-005c6edf027866ca4 | fsx
Volume name
db1_data
Maximum of 203 alphanumeric characters, plus _ .
Junction path
/db1_data
The location within your file system where your volume will be mounted.
Volume size
512000
Minimum 20 MiB; Maximum 104857600 MiB
Storage efficiency
Select whether you would like to enable ONTAP storage efficiencies on your volume: deduplication, compression, and compaction.
Enabled (recommended)
Disabled
Capacity pool tiering policy
You can optionally enable automatic tiering of your data to lower-cost capacity pool storage.
None
Cancel
Confirm



FSX控制台当前不支持创建iSCSI LUN。对于适用于Oracle的iSCSI LUN部署、可以通过NetApp自动化工具包中的自动化for ONTAP 来创建卷和LUN。

在具有**FSX**数据库卷的**EC2**实例上安装和配置**Oracle**

NetApp自动化团队提供了一个自动化套件、用于根据最佳实践在EC2实例上运行Oracle安装和配置。当前版本的自动化套件支持采用默认RU修补程序19.8的基于NFS的Oracle 19c。如果需要、可以轻松地对该自动化套件进行调整、以支持其他RU修补程序。

准备Ansible控制器以运行自动化

请按照"创建并连接到EC2实例以托管Oracle数据库"以配置一个小型EC2 Linux实例以运行Ansible控制器。与使用RedHat相比、使用2vCPU和8G RAM的Amazon Linux T2.large应该足以满足要求。

检索NetApp Oracle部署自动化工具包

以EC2-user身份登录到步骤1中配置的EC2 Ansible控制器实例、然后从EC2-user主目录执行`git clone`命令克隆自动化代码的副本。

```
git clone https://github.com/NetApp-Automation/na_oracle19c_deploy.git
```

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

使用自动化工具包执行自动化Oracle 19c部署

请参见以下详细说明 "CLI 部署 Oracle 19c 数据库" 使用CLI自动化部署Oracle 19c。执行攻略手册时的命令语法略有变化、因为您使用的是SSH密钥对、而不是主机访问身份验证的密码。以下列表概括介绍了相关内容：

1. 默认情况下、EC2实例使用SSH密钥对进行访问身份验证。从Ansible控制器自动化根目录`/home/EC2-user/na_oracle19c_deploy`和`/home/EC2-user/na_RDS_FSx_oranfs_config`中、为在步骤中部署的Oracle主机创建SSH密钥`accesstkey.pem`的副本"创建并连接到EC2实例以托管Oracle数据库。 "
2. 以EC2-user身份登录到EC2实例数据库主机、然后安装python3库。

   ```
   sudo yum install python3
   ```

3. 从根磁盘驱动器创建16G交换空间。默认情况下、EC2实例创建的交换空间为零。请按照以下AWS文档操作： "如何使用交换文件分配内存以用作Amazon EC2实例中的交换空间？"。
4. 返回到Ansible控制器(`cd /home/EC2-user/na_RDS_FSx_oranfs_config`)、并根据相应要求和`linux_config`标记执行克隆前攻略手册。

   ```
   ansible-playbook -i hosts rds_preclone_config.yml -u ec2-user --private
   -key accesststkey.pem -e @vars/fsx_vars.yml -t requirements_config
   ```

   ```
   ansible-playbook -i hosts rds_preclone_config.yml -u ec2-user --private
   -key accesststkey.pem -e @vars/fsx_vars.yml -t linux_config
   ```

5. 切换到`/home/EC2-user/na_oracle19c_deploy-master`目录、阅读README文件、并使用相关全局参数填充全局`vars.yml`文件。
6. 使用`host_vars`目录中的相关参数填充`host_name.yml`文件。
7. 执行适用于Linux的攻略手册、并在系统提示输入vsadmin密码时按Enter键。

```
ansible-playbook -i hosts all_playbook.yml -u ec2-user --private-key
accesststkey.pem -t linux_config -e @vars/vars.yml
```

8. 执行适用于Oracle的攻略手册、并在系统提示您输入vsadmin密码时按Enter键。

```
ansible-playbook -i hosts all_playbook.yml -u ec2-user --private-key
accesststkey.pem -t oracle_config -e @vars/vars.yml
```

如果需要、将SSH密钥文件上的权限位更改为400。将Oracle主机(`host_vars`文件中的`Ansible`主机) IP地址更改为EC2实例公有 地址。

在主FSX HA集群和备用FSX HA集群之间设置SnapMirror

为了实现高可用性和灾难恢复、您可以在主FSX存储集群和备用FSX存储集群之间设置SnapMirror复制。与其他云存储服务不同、FSX支持用户按所需频率和复制吞吐量控制和管理存储复制。此外、它还允许用户在不影响可用性的情况下测试HA/DR。

以下步骤显示了如何在主FSX存储集群和备用FSX存储集群之间设置复制。

1. 设置主集群对等和备用集群对等。以fsxadmin用户身份登录到主集群、然后执行以下命令。此对等创建过程会在主集群和备用集群上执行create命令。将`standby-cluster_name`替换为适用于您的环境的名称。

```
cluster peer create -peer-addrs
standby_cluster_name,inter_cluster_ip_address -username fsxadmin
-initial-allowed-vserver-peers *
```

2. 在主集群和备用集群之间设置SVM对等关系。以vsadmin用户身份登录到主集群、然后执行以下命令。将`primary_vserver_name`、`standby-vserver_name`、`standby-cluster_name`替换为适用于您环境的名称。

```
vserver peer create -vserver primary_vserver_name -peer-vserver
standby_vserver_name -peer-cluster standby_cluster_name -applications
snapmirror
```

3. 验证集群和SVM对等项是否设置正确。

```
FsxId00164454fac5591e6::> cluster peer show
Peer Cluster Name         Cluster Serial Number Availability   Authentication
------------------------- --------------------- -------------- --------------
FsxId0b6a95149d07aa82e    1-80-000011           Available      ok

FsxId00164454fac5591e6::> vserver peer show
            Peer        Peer                           Peering        Remote
Vserver     Vserver     State        Peer Cluster      Applications   Vserver
----------- ----------- ------------ ----------------- -------------- ---------
svm_FSxOraSource
            svm_FSxOraTarget
                        peered       FsxId0b6a95149d07aa82e
                                                       snapmirror     svm_FSxOraTarget

FsxId00164454fac5591e6::>
```

4. 在备用FSX集群上为主FSX集群上的每个源卷创建目标NFS卷。根据您的环境需要替换卷名称。

```
vol create -volume dr_db1_bin -aggregate aggr1 -size 50G -state online
-policy default -type DP
```

```
vol create -volume dr_db1_data -aggregate aggr1 -size 500G -state online
-policy default -type DP
```

```
vol create -volume dr_db1_log -aggregate aggr1 -size 250G -state online
-policy default -type DP
```

5. 如果使用iSCSI协议进行数据访问、则还可以为Oracle二进制文件、Oracle数据和Oracle日志创建iSCSI卷和LUN。在卷中为快照留出大约10%的可用空间。

```
vol create -volume dr_db1_bin -aggregate aggr1 -size 50G -state online
-policy default -unix-permissions ---rwxr-xr-x -type RW
```

```
lun create -path /vol/dr_db1_bin/dr_db1_bin_01 -size 45G -ostype linux
```

```
vol create -volume dr_db1_data -aggregate aggr1 -size 500G -state online
-policy default -unix-permissions ---rwxr-xr-x -type RW
```

```
lun create -path /vol/dr_db1_data/dr_db1_data_01 -size 100G -ostype
linux
```

```
lun create -path /vol/dr_db1_data/dr_db1_data_02 -size 100G -ostype
linux
```

```
lun create -path /vol/dr_db1_data/dr_db1_data_03 -size 100G -ostype
linux
```

```
lun create -path /vol/dr_db1_data/dr_db1_data_04 -size 100G -ostype
linux
```

vol create -volume dr_db1_log -aggregate aggr1 -size 250G -state online -policy default -unix-permissions -rwxr-x -type rw

```
lun create -path /vol/dr_db1_log/dr_db1_log_01 -size 45G -ostype linux
```

```
lun create -path /vol/dr_db1_log/dr_db1_log_02 -size 45G -ostype linux
```

```
lun create -path /vol/dr_db1_log/dr_db1_log_03 -size 45G -ostype linux
```

```
lun create -path /vol/dr_db1_log/dr_db1_log_04 -size 45G -ostype linux
```

6. 对于iSCSI LUN、使用二进制LUN作为示例、为每个LUN的Oracle主机启动程序创建映射。将igroup替换为适合您环境的名称、并增加每个附加LUN的lun-id。

```
lun mapping create -path /vol/dr_db1_bin/dr_db1_bin_01 -igroup ip-10-0-
1-136 -lun-id 0
```

```
lun mapping create -path /vol/dr_db1_data/dr_db1_data_01 -igroup ip-10-
0-1-136 -lun-id 1
```

7. 在主数据库卷和备用数据库卷之间创建SnapMirror关系。替换您的环境的相应SVM名称

```
snapmirror create -source-path svm_FSxOraSource:db1_bin -destination
-path svm_FSxOraTarget:dr_db1_bin -vserver svm_FSxOraTarget -throttle
unlimited -identity-preserve false -policy MirrorAllSnapshots -type DP
```

```
snapmirror create -source-path svm_FSxOraSource:db1_data -destination
-path svm_FSxOraTarget:dr_db1_data -vserver svm_FSxOraTarget -throttle
unlimited -identity-preserve false -policy MirrorAllSnapshots -type DP
```

```
snapmirror create -source-path svm_FSxOraSource:db1_log -destination
-path svm_FSxOraTarget:dr_db1_log -vserver svm_FSxOraTarget -throttle
unlimited -identity-preserve false -policy MirrorAllSnapshots -type DP
```

可以使用适用于NFS数据库卷的NetApp自动化工具包自动设置此SnapMirror。该工具包可从NetApp公有 GitHub 站点下载。

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

在尝试进行设置和故障转移测试之前、请仔细阅读自述文件中的说明。

将Oracle二进制文件从主集群复制到备用集群可能会涉及Oracle许可证。有关说明、请联系您的Oracle许可证代表。另一种方法是在恢复和故障转移时安装和配置Oracle。

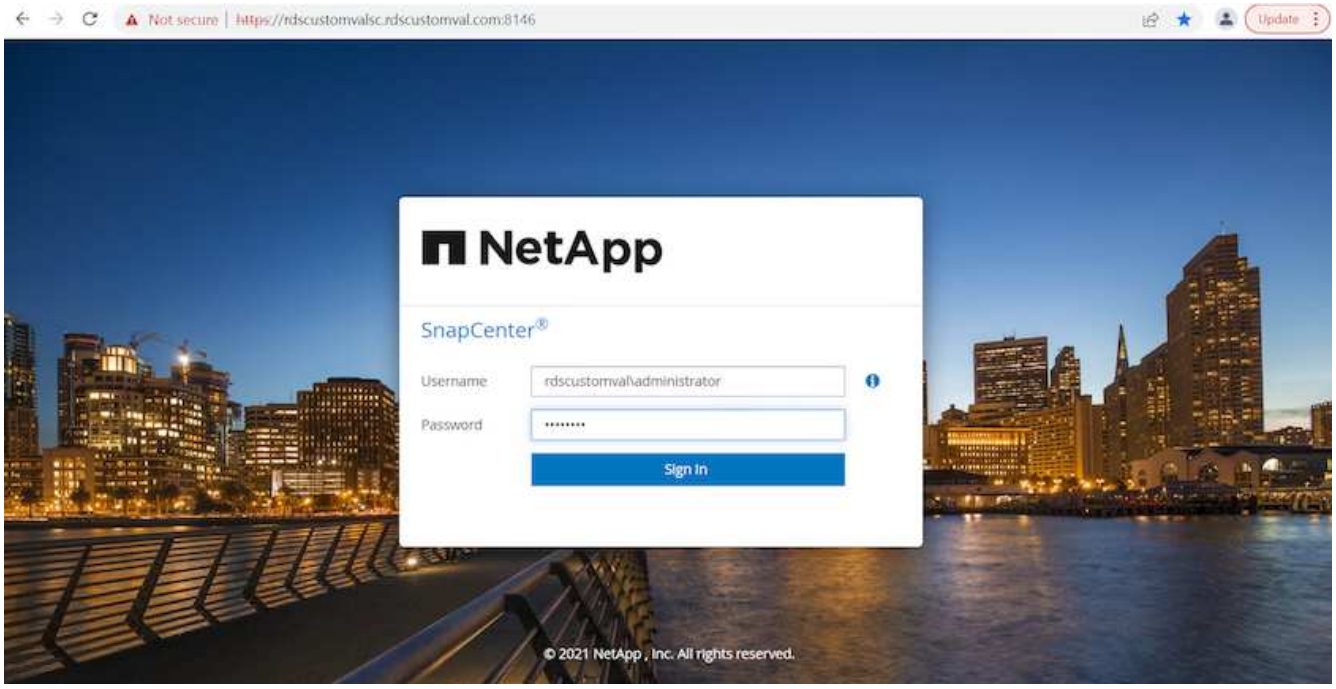
SnapCenter 部署

SnapCenter 安装

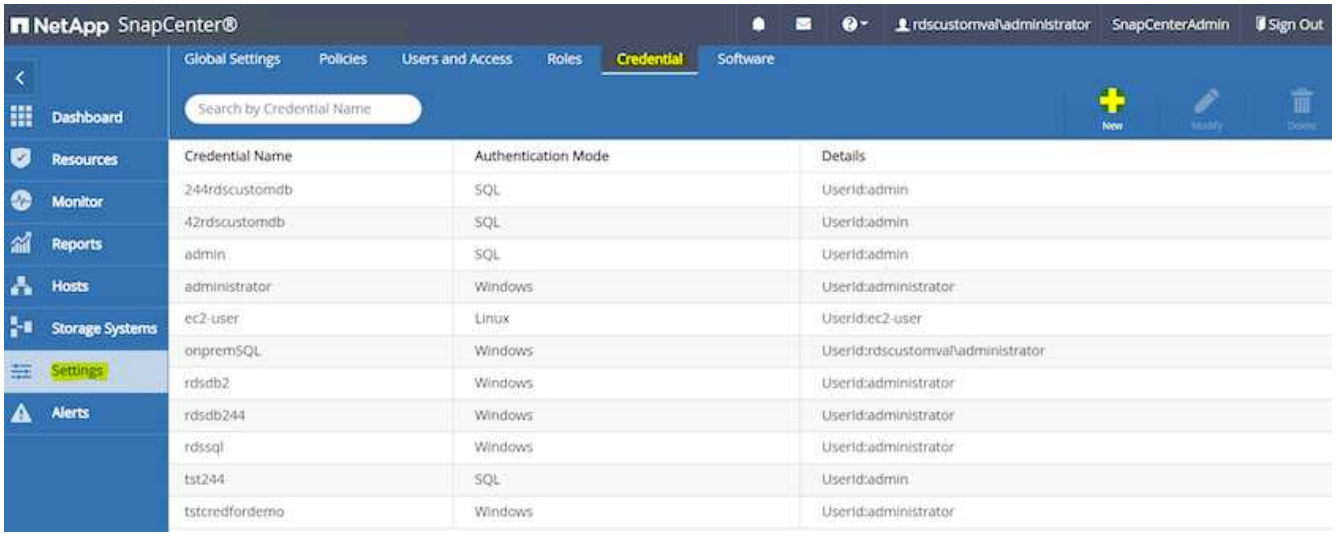
请遵循 "安装SnapCenter 服务器" 安装SnapCenter 服务器。本文档介绍如何安装独立的SnapCenter 服务器。SaaS版本的SnapCenter 正在进行测试审核、不久将推出。如果需要、请咨询NetApp代表以了解可用性。

为EC2 Oracle主机配置SnapCenter 插件

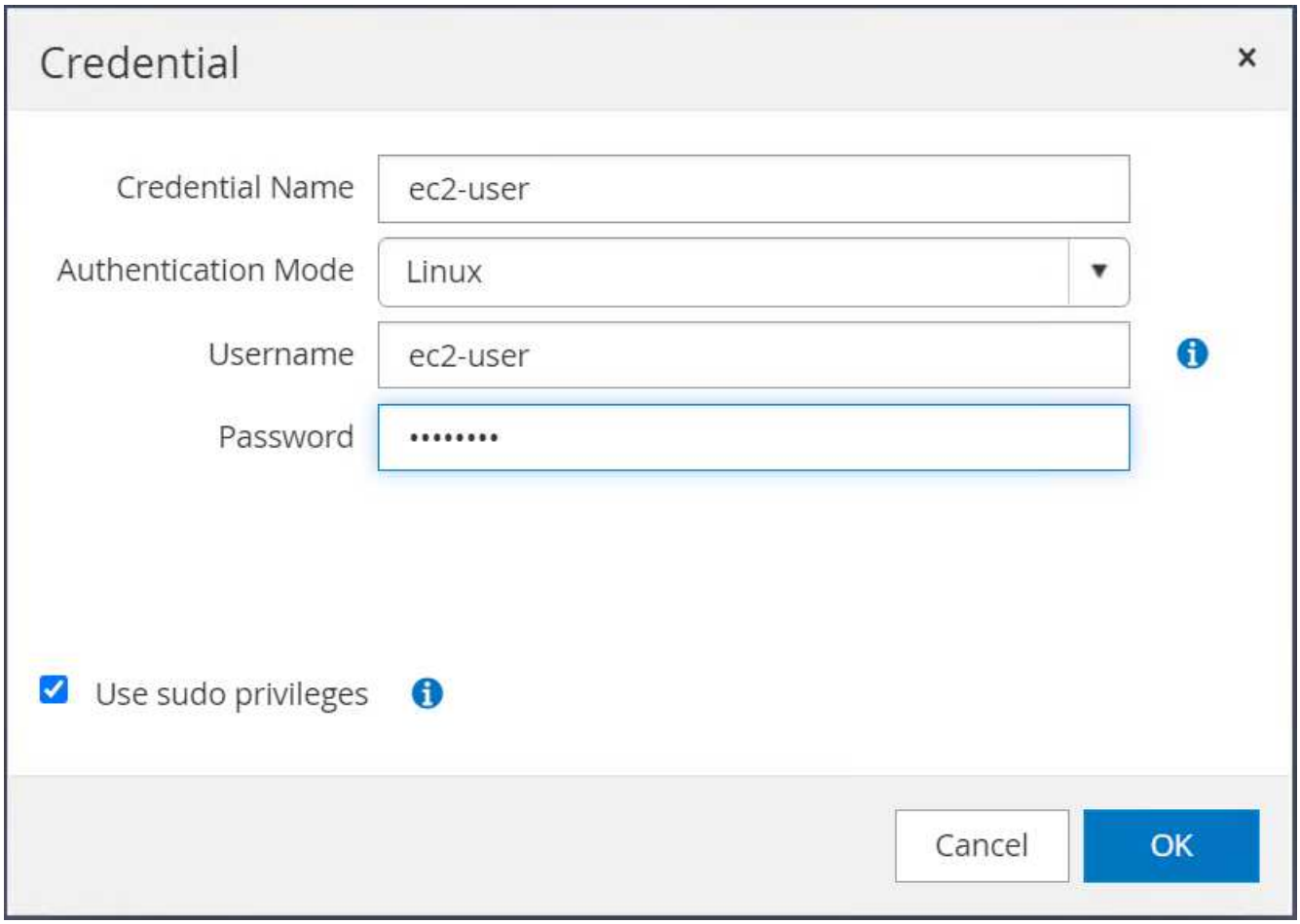
1. 自动安装SnapCenter 后、以安装SnapCenter 服务器的Window主机的管理用户身份登录到SnapCenter。



2. 从左侧菜单中、单击设置、然后单击凭据和新建、为SnapCenter 插件安装添加EC2-user凭据。



3. 通过编辑EC2实例主机上的`/etc/ssh/sshd_config`文件、重置EC2-user密码并启用密码SSH身份验证。
4. 验证是否已选中"Use sudo privileges"复选框。您只需在上一步中重置EC2-user密码即可。



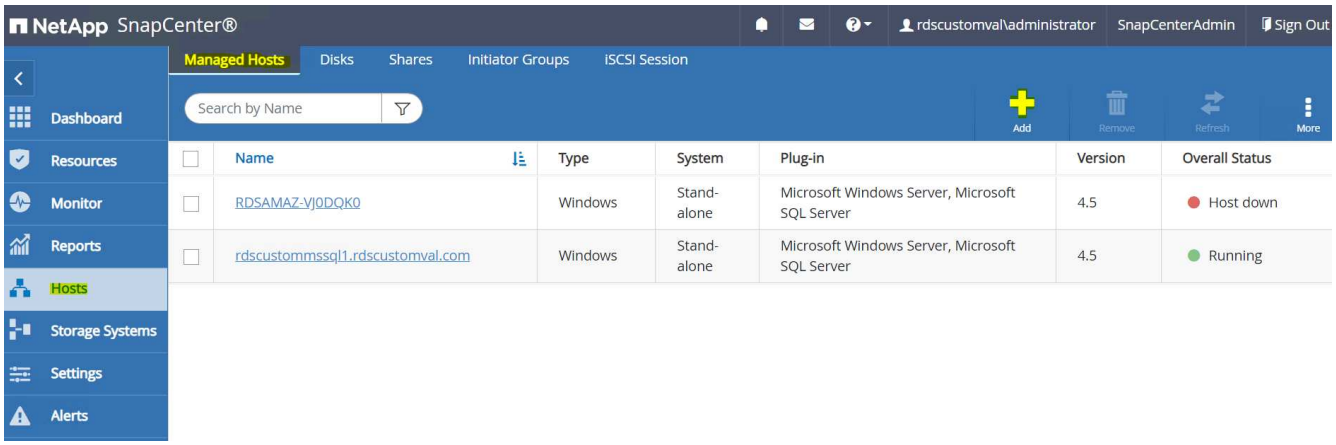
5. 将SnapCenter 服务器名称和IP地址添加到EC2实例主机文件以进行名称解析。

```
[ec2-user@ip-10-0-0-151 ~]$ sudo vi /etc/hosts
[ec2-user@ip-10-0-0-151 ~]$ cat /etc/hosts
127.0.0.1   localhost localhost.localdomain localhost4
localhost4.localdomain4
::1         localhost localhost.localdomain localhost6
localhost6.localdomain6
10.0.1.233  rdscustomvalsc.rdscustomval.com rdscustomvalsc
```

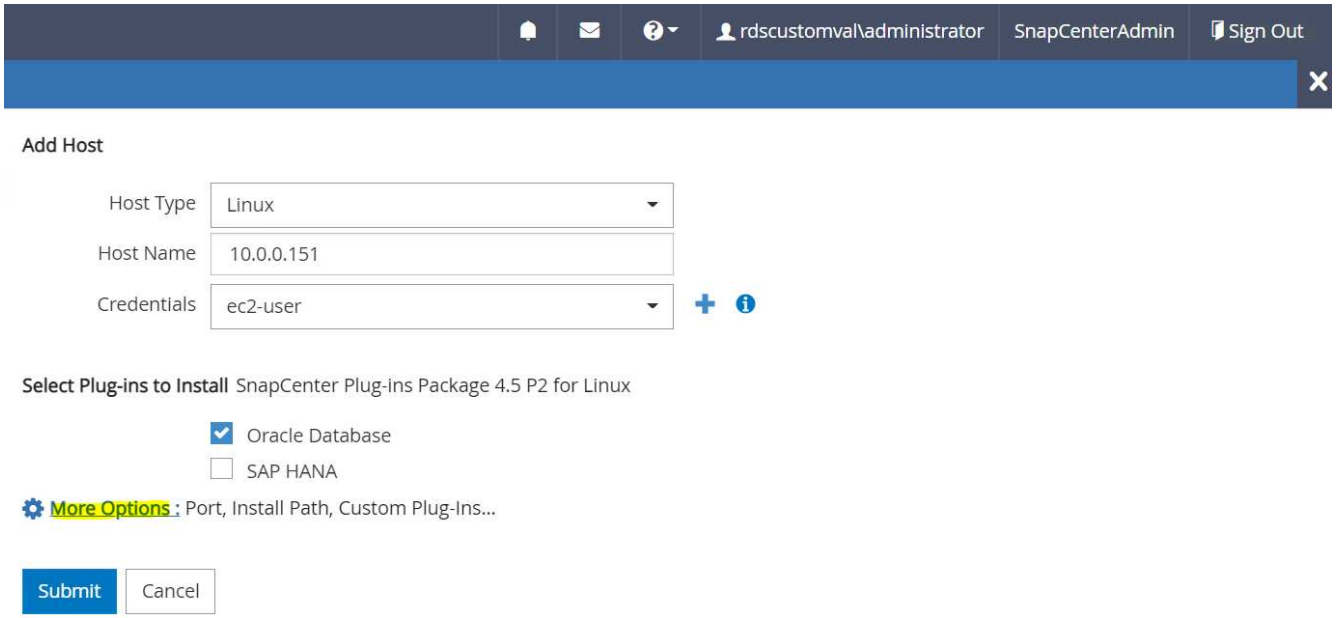
6. 在SnapCenter 服务器Windows主机上、将EC2实例主机IP地址添加到Windows主机文件`C：\Windows\System32\drivers\etc\hosts`。

```
10.0.0.151      ip-10-0-0-151.ec2.internal
```

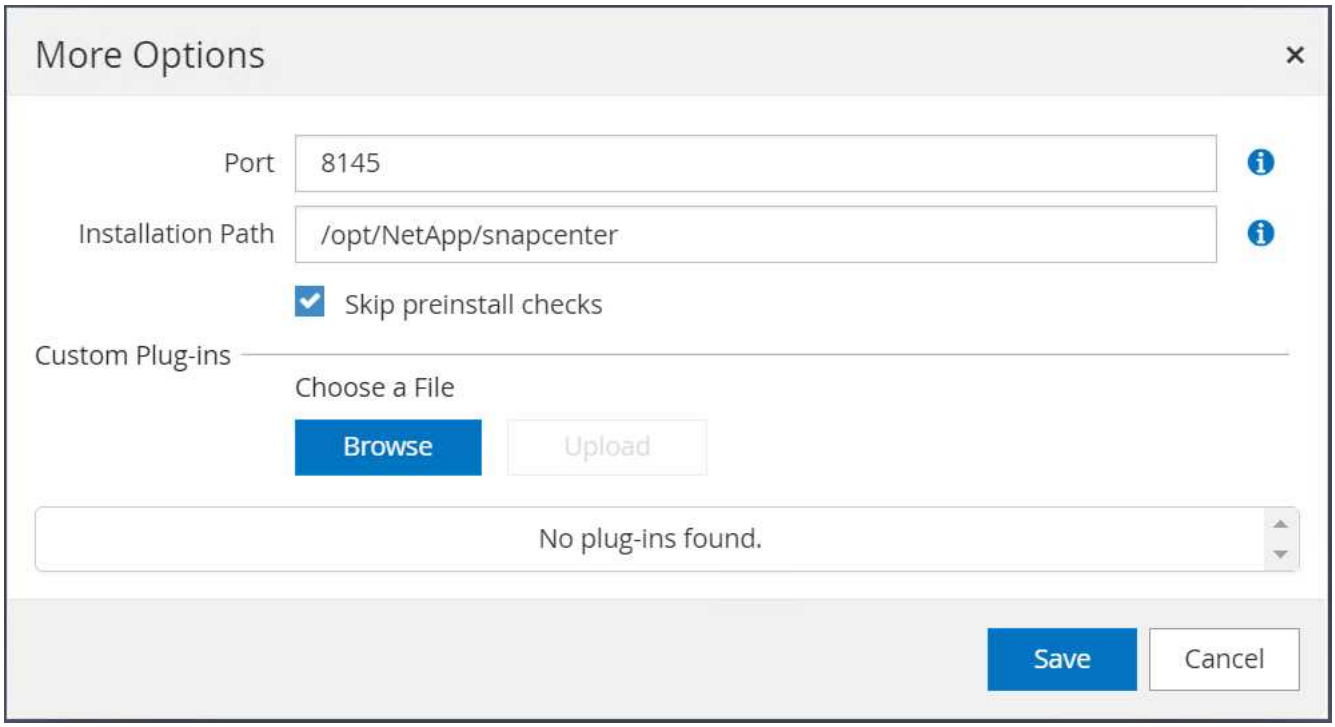
7. 在左侧菜单中、选择主机>受管主机、然后单击添加将EC2实例主机添加到SnapCenter。



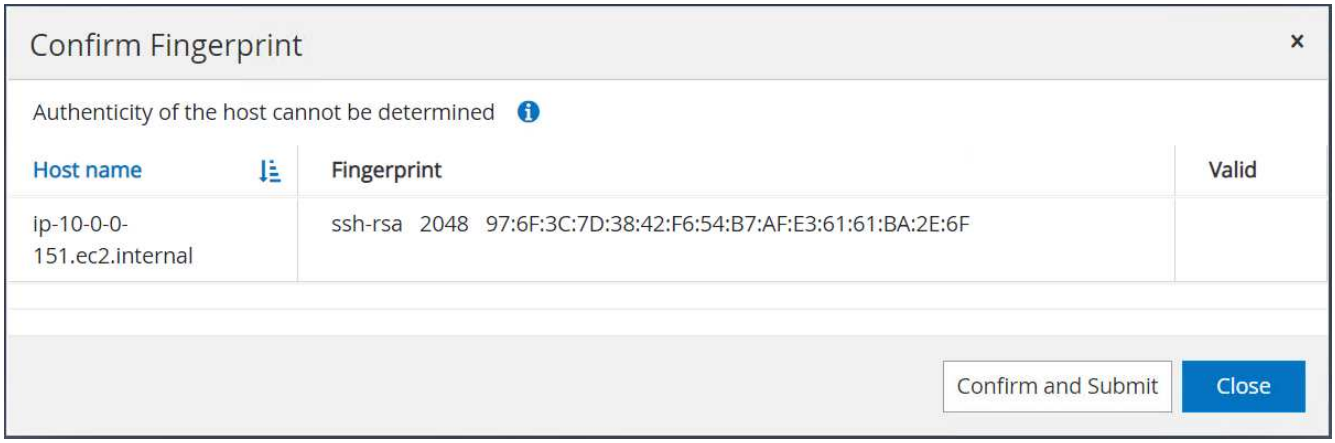
检查Oracle数据库、然后在提交之前、单击更多选项。



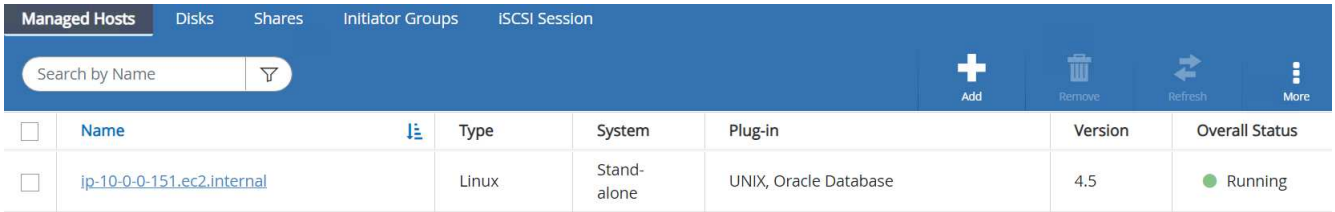
选中跳过预安装检查。确认跳过预安装检查、然后在保存后单击提交。



系统将提示您确认指纹、然后单击确认并提交。



成功配置插件后、受管主机的整体状态将显示为正在运行。



配置**Oracle**数据库的备份策略

请参见本节 "在 SnapCenter 中设置数据库备份策略" 有关配置Oracle数据库备份策略的详细信息。

通常、您需要为完整快照Oracle数据库备份创建一个策略、并为Oracle归档日志唯一快照备份创建一个策略。

您可以在备份策略中启用Oracle归档日志修剪、以控制日志归档空间。如果需要复制到HA或DR的备用位置、请选中"选择二级复制选项"中的"创建本地Snapshot副本后更新SnapMirror"。

配置**Oracle**数据库备份和计划

SnapCenter 中的数据库备份可由用户配置、可以单独设置、也可以作为资源组中的组进行设置。备份间隔取决于RTO和RPO目标。NetApp建议您每隔几小时运行一次完整的数据库备份、并以10到15分钟等较高的频率对日志备份进行归档、以实现快速恢复。

请参阅的Oracle部分 "实施备份策略以保护数据库" 有关实施一节中创建的备份策略的详细分步过程 配置Oracle数据库的备份策略 和用于备份作业计划。

下图举例说明了为备份Oracle数据库而设置的资源组。

EC2和FSX Oracle数据库管理

除了AWS EC2和FSX管理控制台之外、此Oracle环境还部署了Ansible控制节点和SnapCenter UI工具来进行数据库管理。

Ansible控制节点可用于管理Oracle环境配置、并可通过并行更新来保持主实例和备用实例同步、以便进行内核或修补程序更新。故障转移、重新同步和故障恢复可通过NetApp自动化工具包自动进行、以便通过Ansible实现快速应用程序恢复和可用性归档。可以使用攻略手册执行一些可重复的数据库管理任务、以减少人为错误。

SnapCenter UI工具可以使用适用于Oracle数据库的SnapCenter 插件执行数据库快照备份、时间点恢复、数据库克隆等操作。有关Oracle插件功能的详细信息、请参见 "适用于Oracle数据库的SnapCenter 插件概述"。

以下各节详细介绍了如何使用SnapCenter UI实现Oracle数据库管理的关键功能：

- 数据库快照备份
- 数据库时间点还原
- 创建数据库克隆

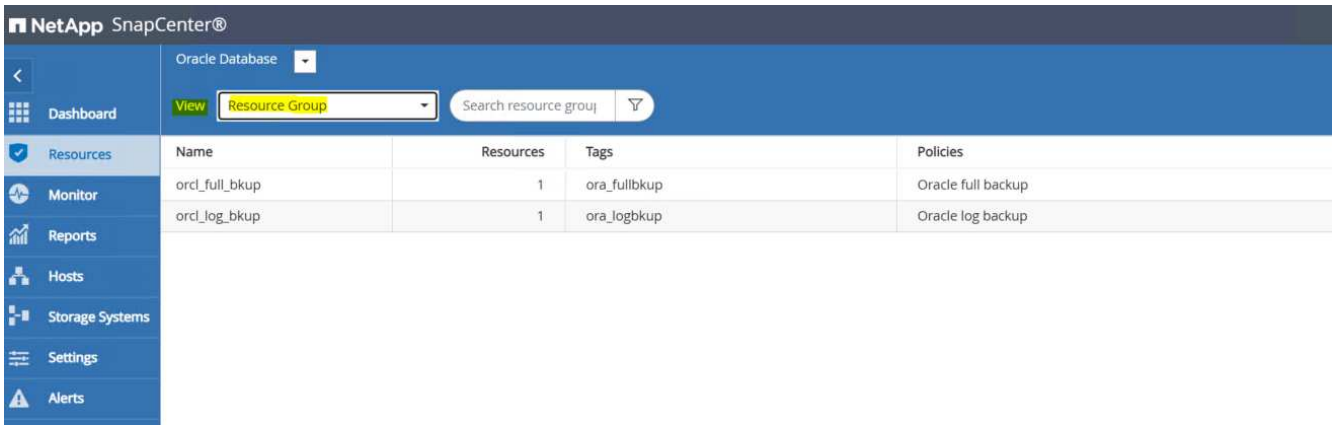
数据库克隆会在单独的EC2主机上创建主数据库的副本、以便在发生逻辑数据错误或损坏时进行数据恢复、克隆也可用于应用程序测试、调试、修补程序验证等。

正在创建快照

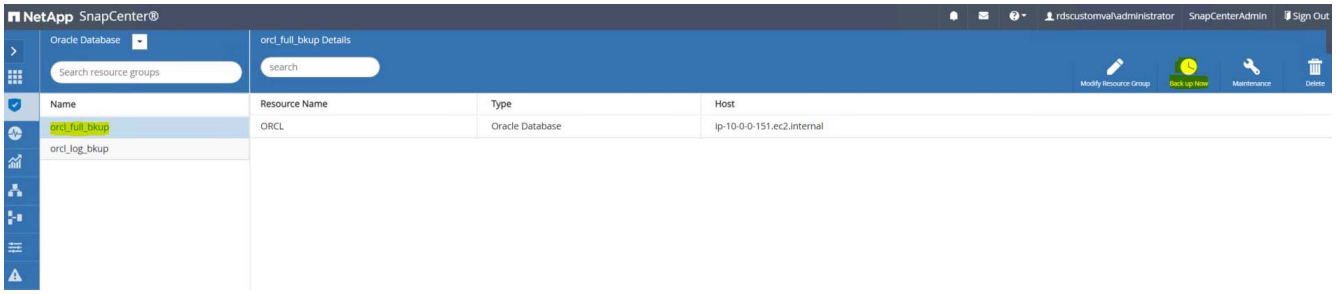
EC2/FSX Oracle数据库会按用户配置的间隔定期备份。用户也可以随时执行一次性快照备份。此适用场景 既可执行完整数据库快照备份、也可执行仅归档日志的快照备份。

完整的数据库快照包括所有Oracle文件、包括数据文件、控制文件和归档日志文件。

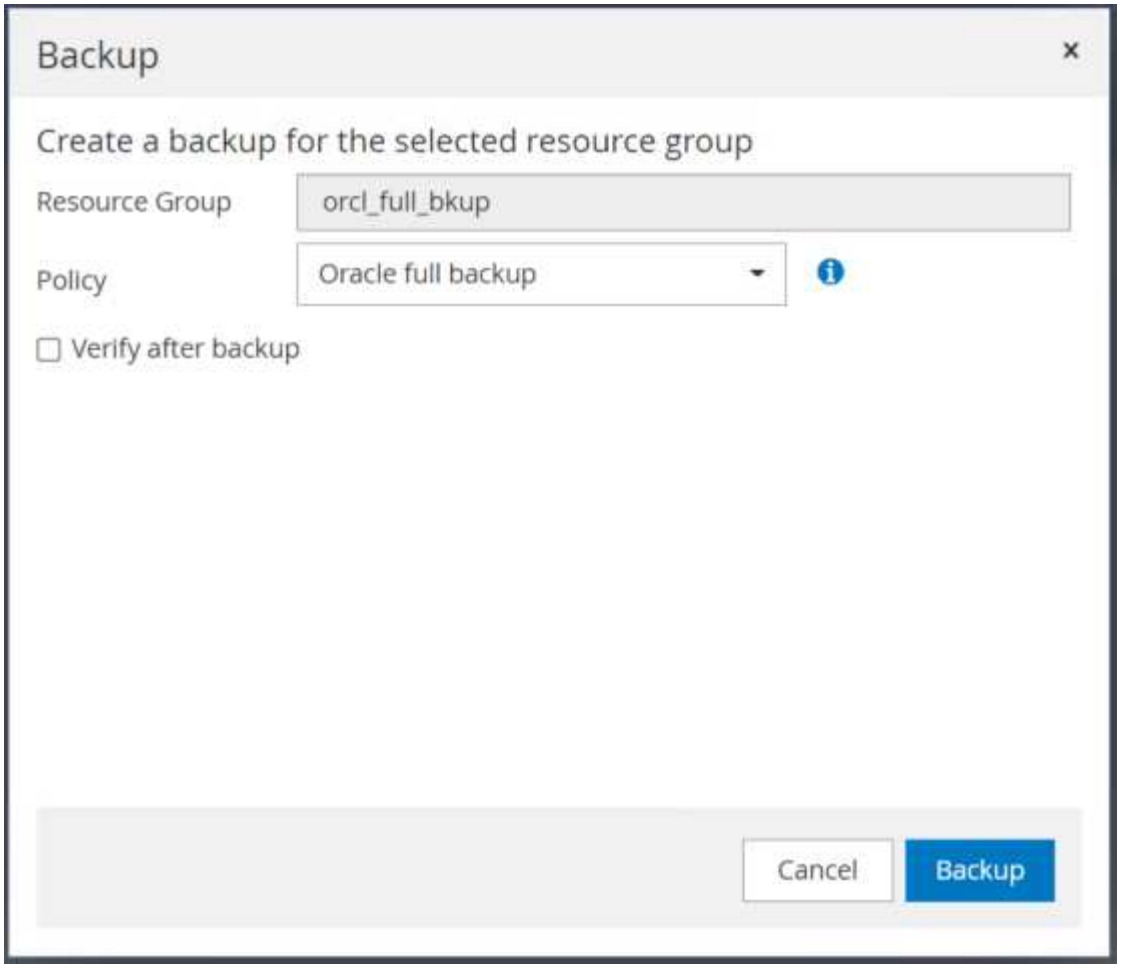
1. 登录到SnapCenter UI、然后单击左侧菜单中的"Resources"。从视图下拉列表中、更改为资源组视图。



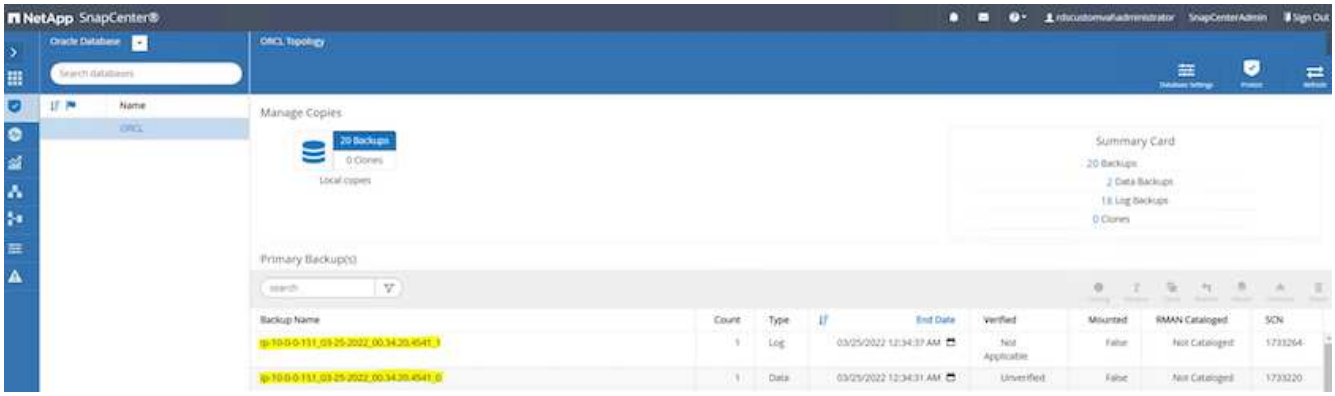
2. 单击完整备份资源名称、然后单击立即备份图标以启动附加临时备份。



3. 单击备份、然后确认备份以启动完整数据库备份。



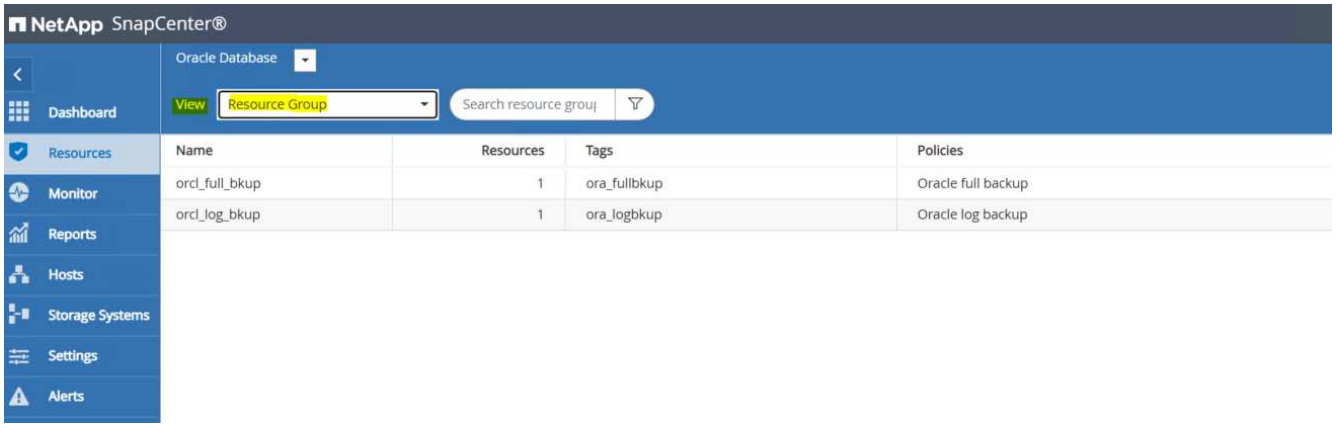
从数据库的资源视图中、打开数据库托管备份副本页面、验证一次性备份是否已成功完成。完整数据库备份会创建两个快照：一个用于数据卷、一个用于日志卷。



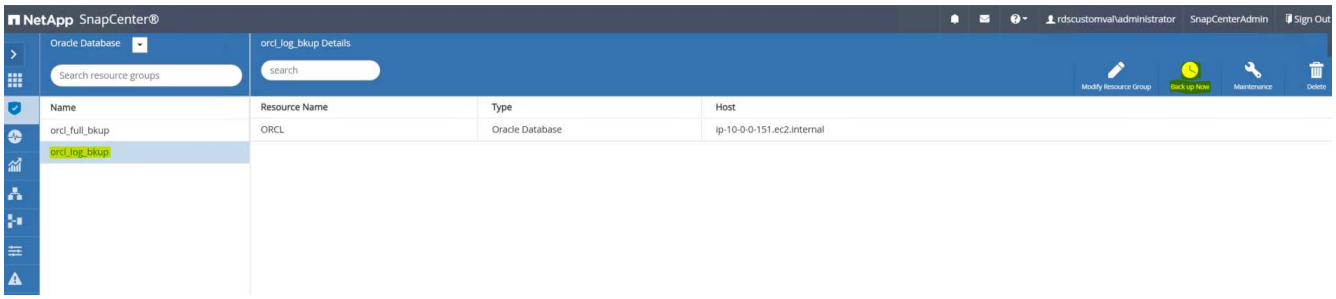
创建归档日志快照

只会为Oracle归档日志卷创建归档日志快照。

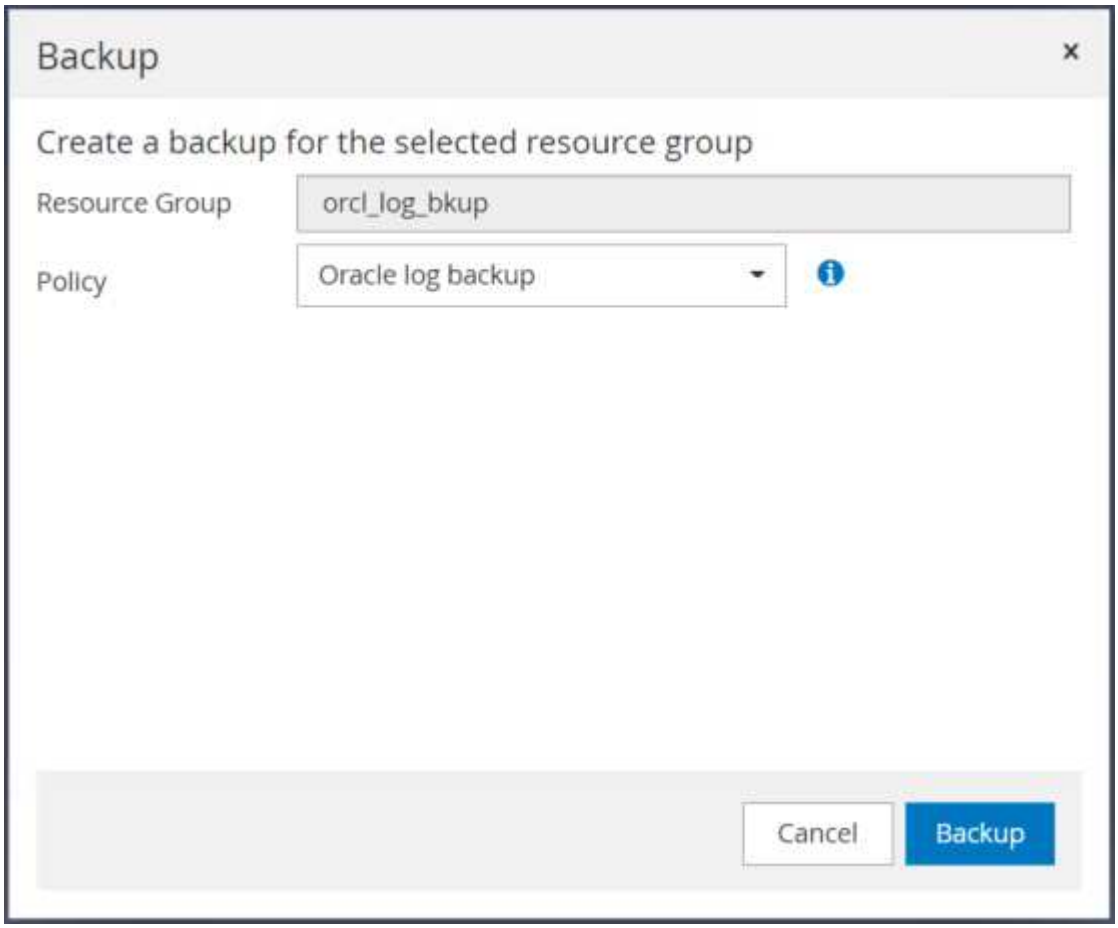
1. 登录到SnapCenter UI、然后单击左侧菜单栏中的"Resources"选项卡。从视图下拉列表中、更改为资源组视图。



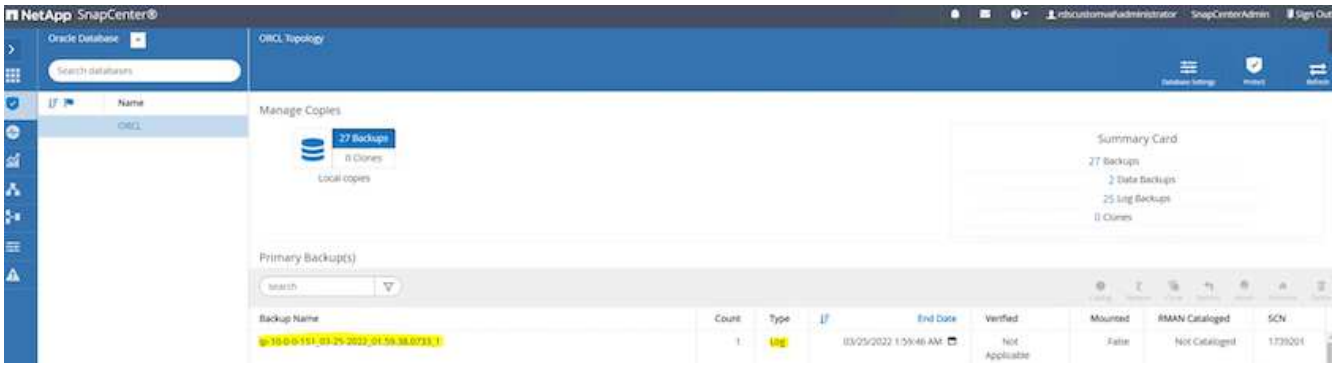
2. 单击日志备份资源名称、然后单击立即备份图标为归档日志启动附加临时备份。



3. 单击备份、然后确认备份以启动归档日志备份。



从数据库的资源视图中、打开数据库托管备份副本页面、验证一次性归档日志备份是否已成功完成。归档日志备份会为日志卷创建一个快照。

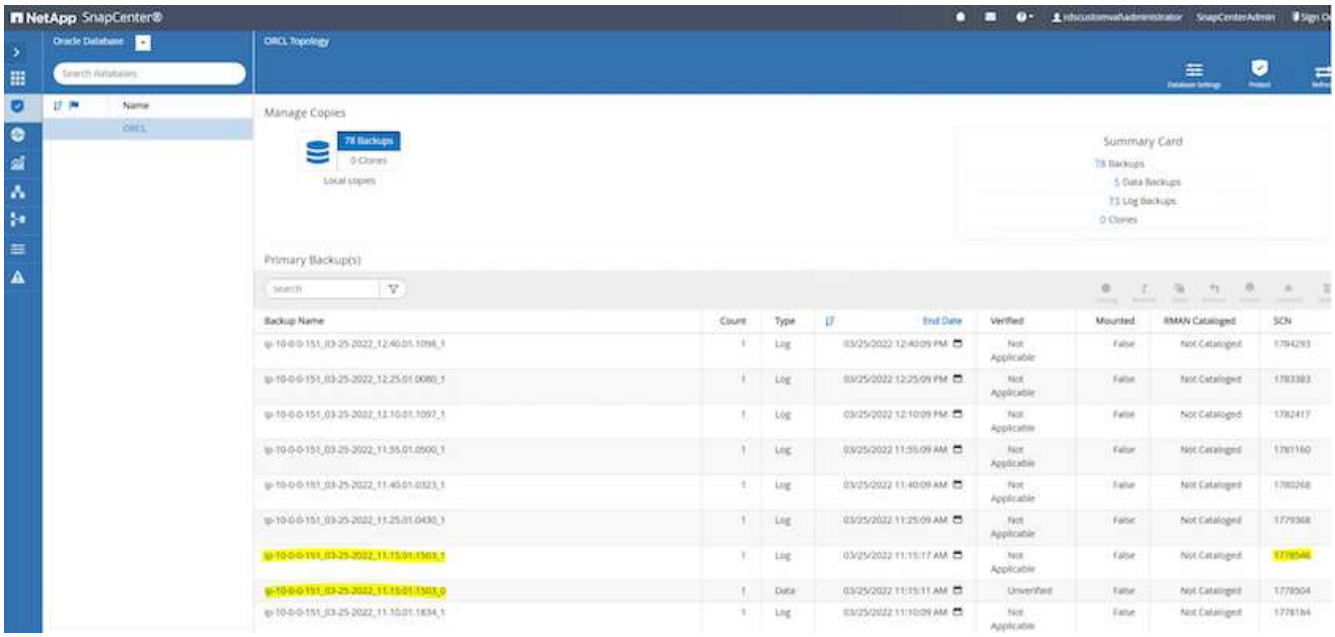


还原到某个时间点

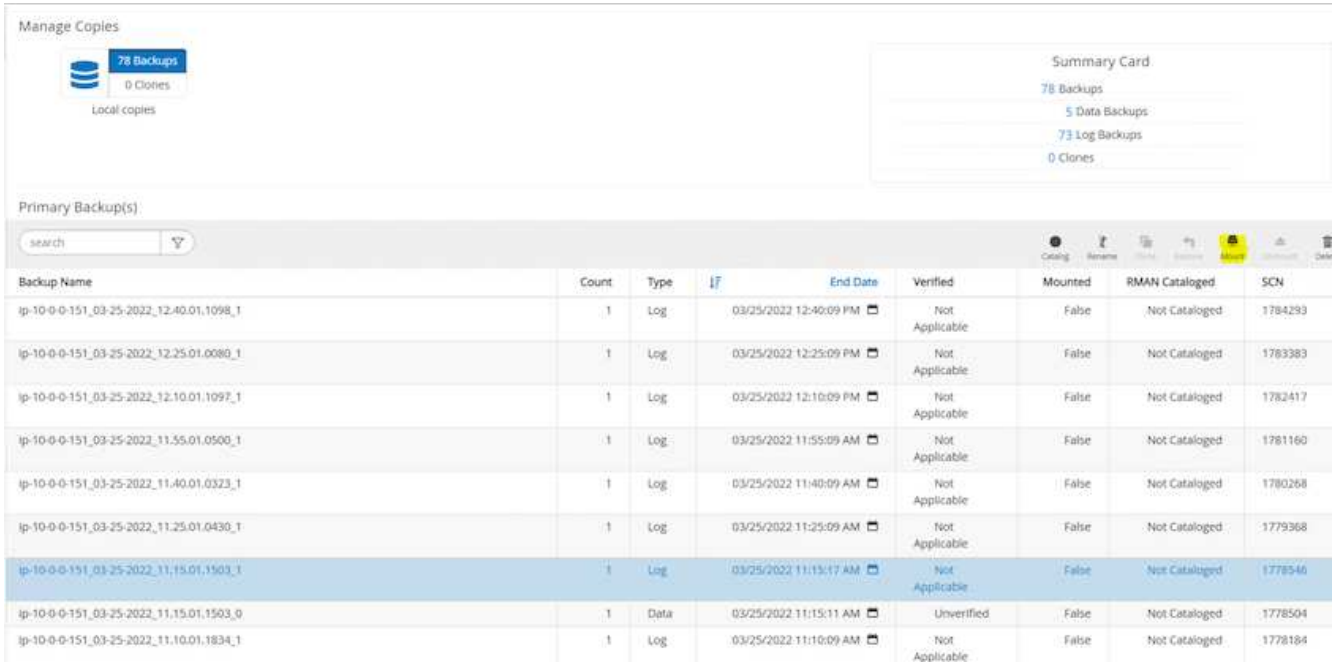
在同一个EC2实例主机上执行基于SnapCenter的时间点还原。完成以下步骤以执行还原：

1. 在SnapCenter 资源选项卡>数据库视图中、单击数据库名称以打开数据库备份。

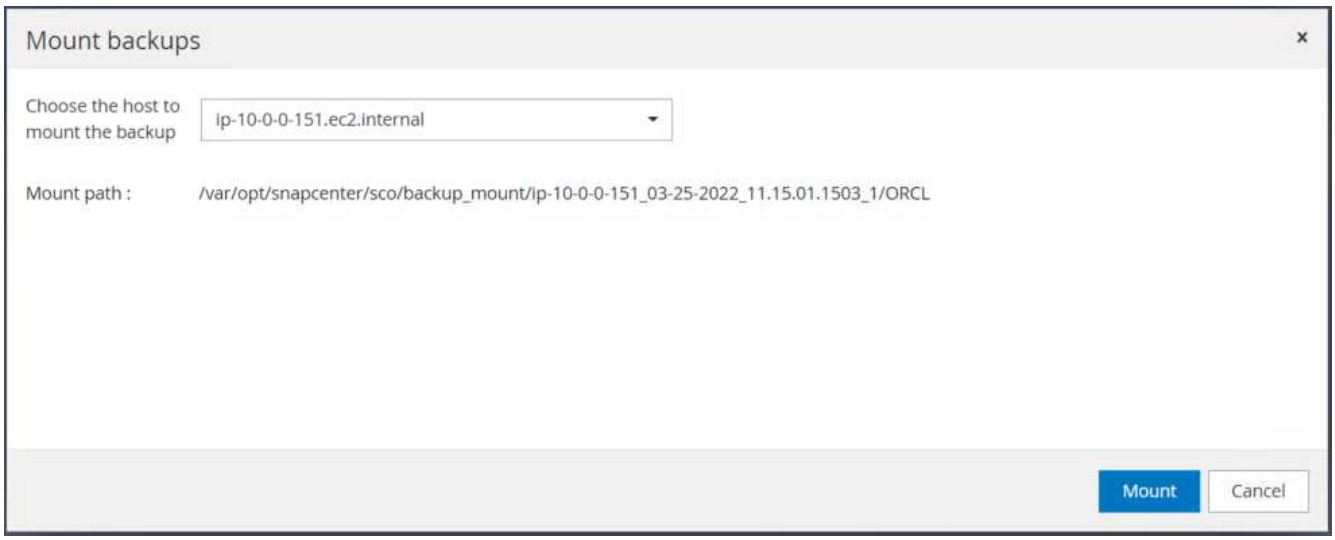
2. 选择数据库备份副本以及要还原的所需时间点。同时、记下时间点对应的SCN编号。可以使用时间或SCN执行时间点还原。



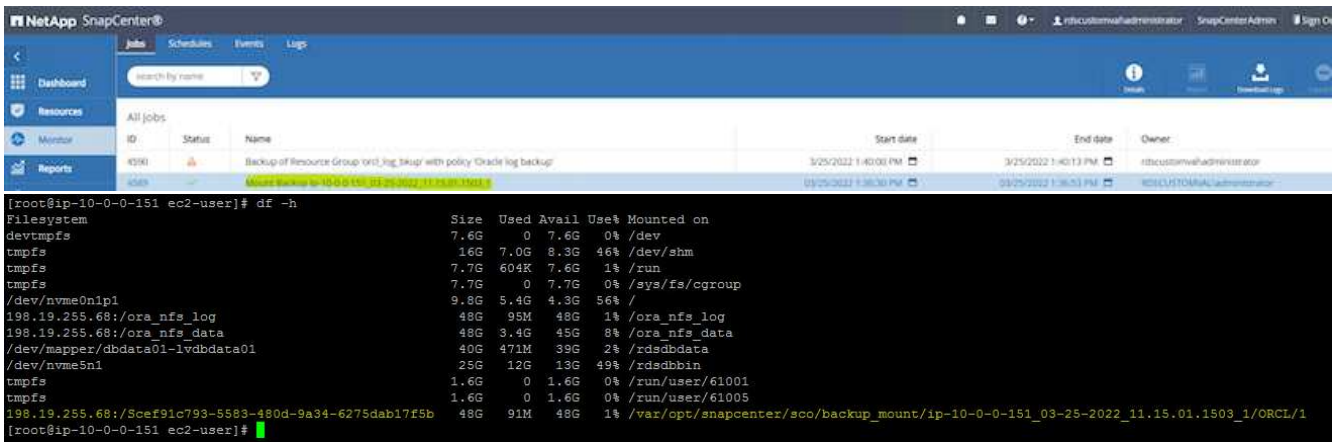
3. 突出显示日志卷快照、然后单击挂载按钮挂载此卷。



4. 选择要挂载日志卷的主EC2实例。



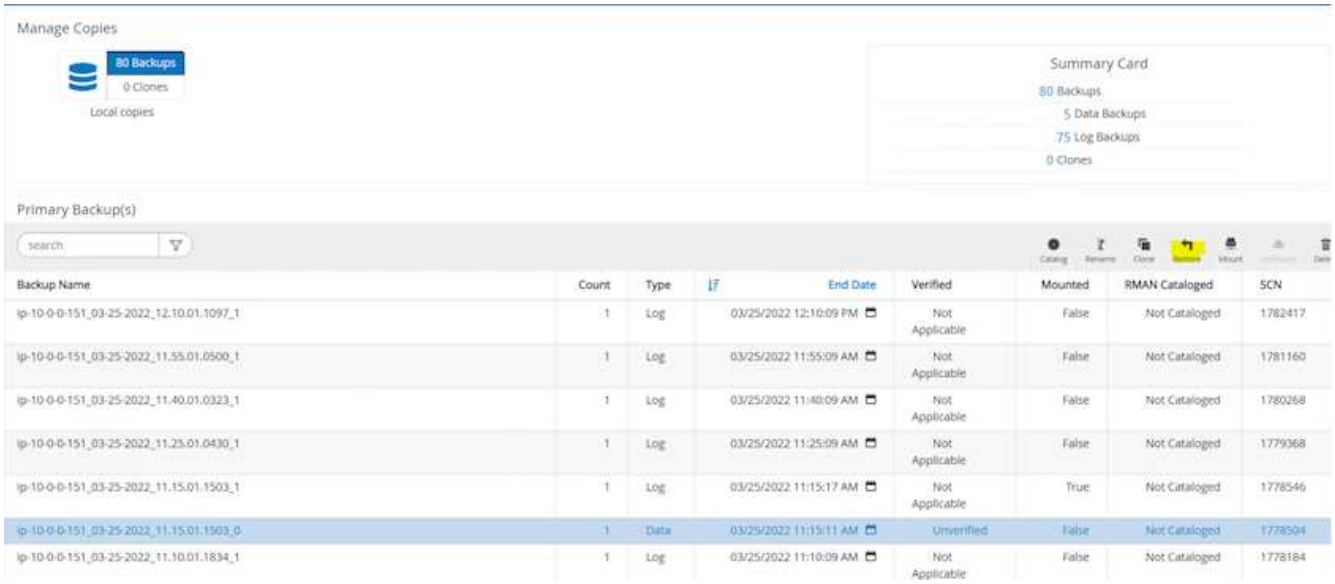
5. 验证挂载作业是否已成功完成。另外、请检查EC2实例主机以查看该日志卷已挂载以及挂载点路径。



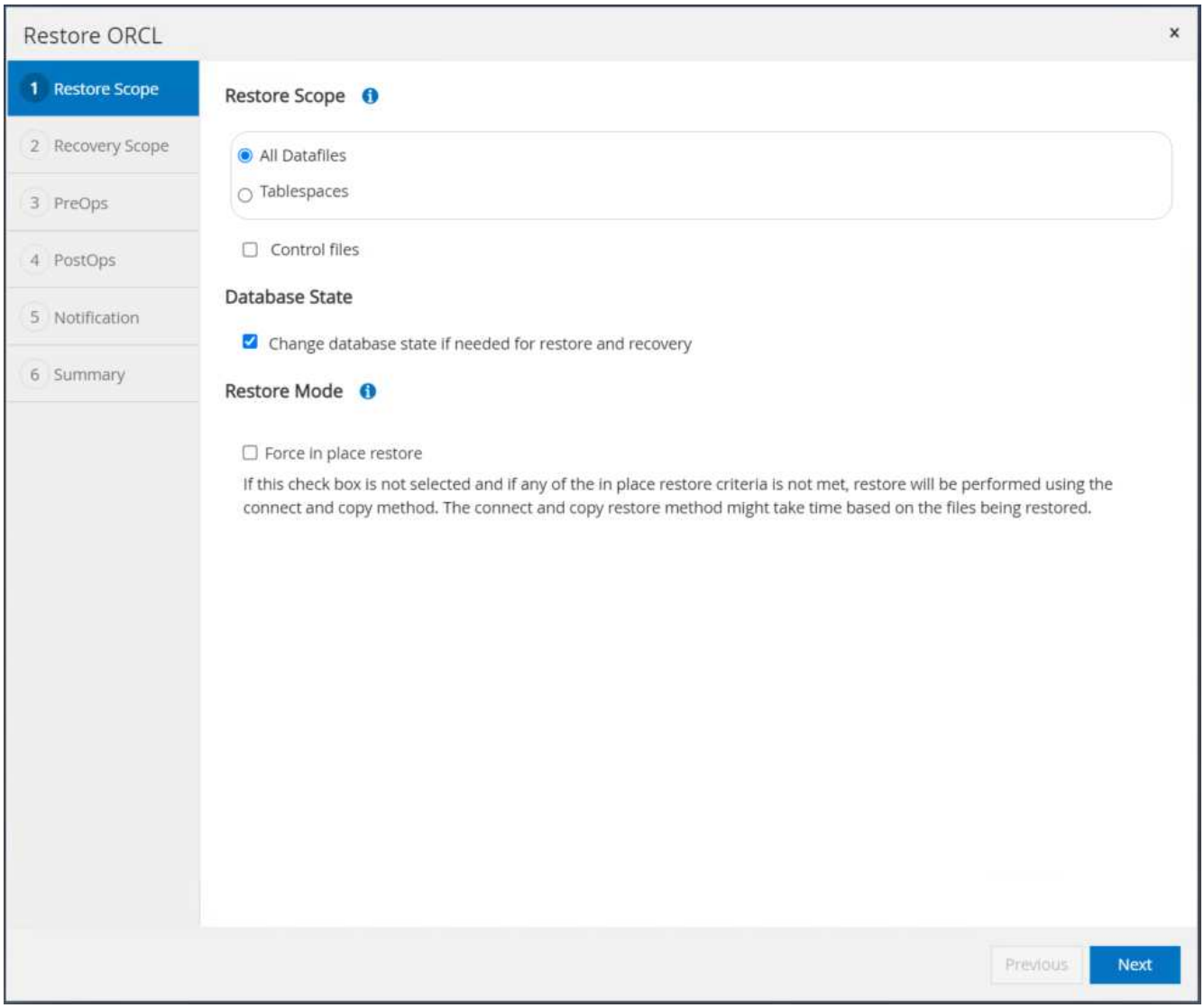
6. 将归档日志从挂载的日志卷复制到当前归档日志目录。

```
[ec2-user@ip-10-0-0-151 ~]$ cp /var/opt/snapcenter/sco/backup_mount/ip-10-0-0-151_03-25-2022_11.15.01.1503_1/ORCL/1/db/ORCL_A/arch/*.arc /ora_nfs_log/db/ORCL_A/arch/
```

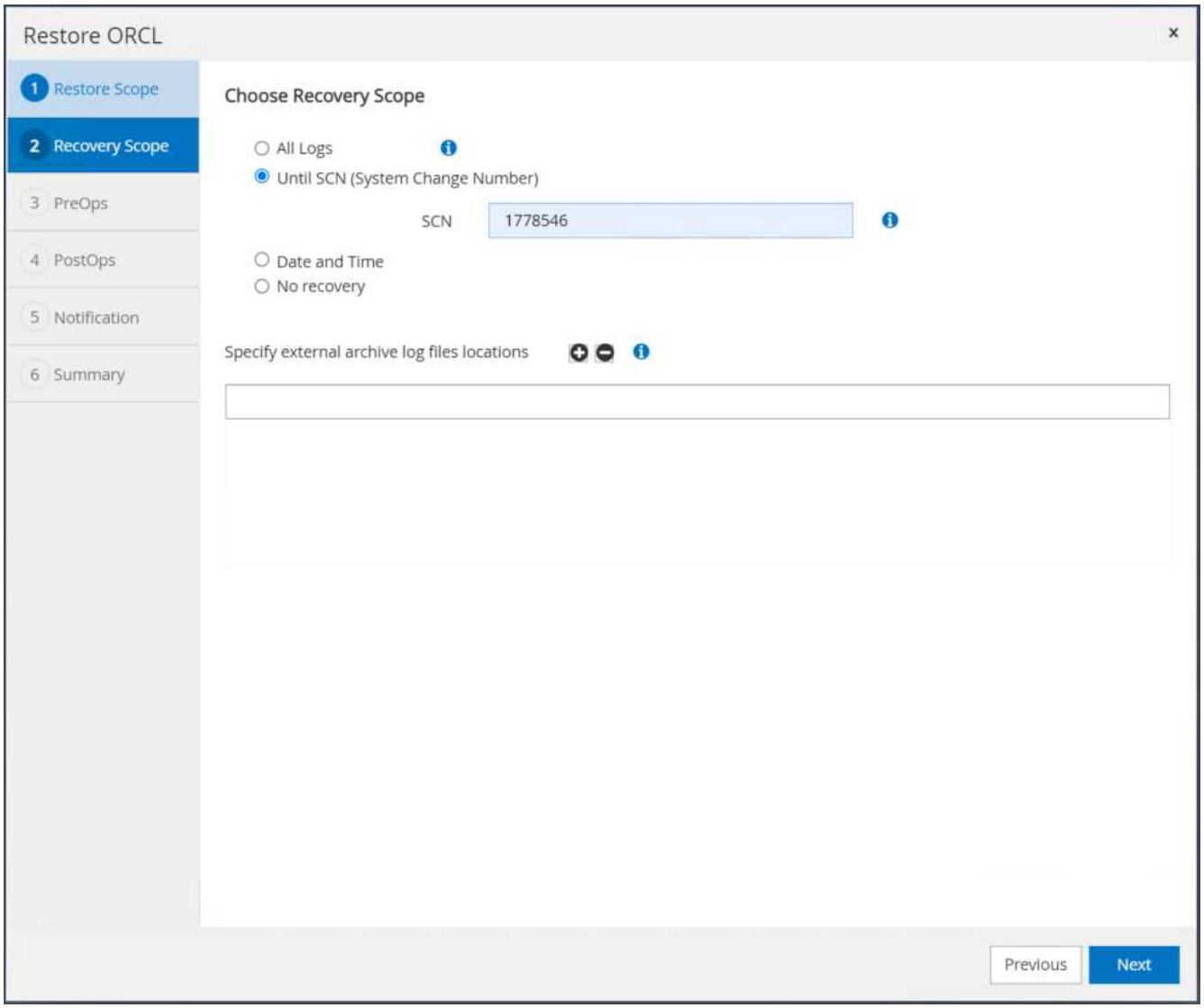
7. 返回到SnapCenter 资源选项卡>数据库备份页面、突出显示数据快照副本、然后单击还原按钮以启动数据库还原工作流。



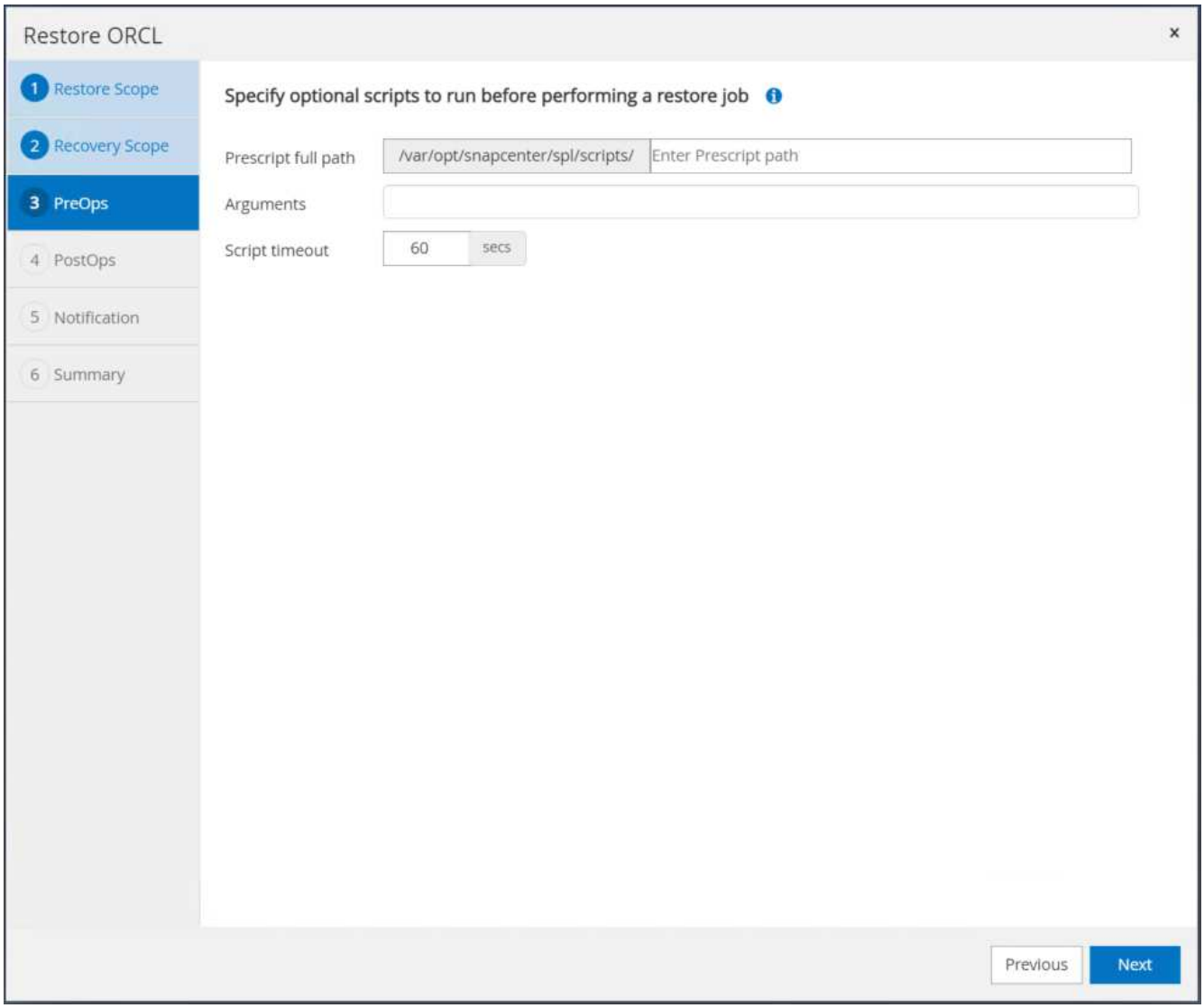
8. 选中"所有数据文件"和"根据还原和恢复需要更改数据库状态"、然后单击下一步。



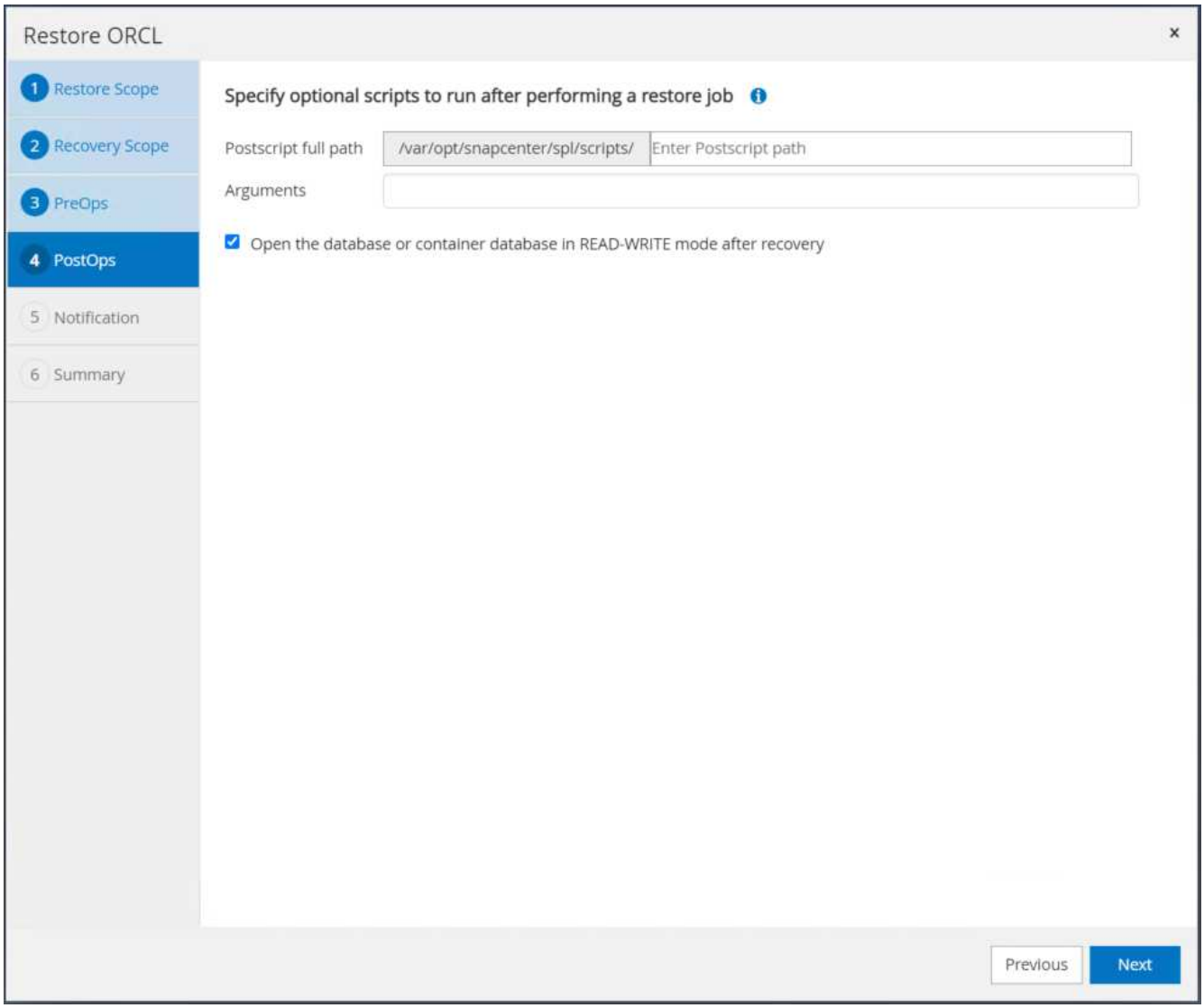
9. 使用SCN或时间选择所需的恢复范围。挂载的归档日志路径可以在"指定外部归档日志文件位置"中列出以供恢复、而不是像第6步所示将挂载的归档日志复制到当前日志目录。



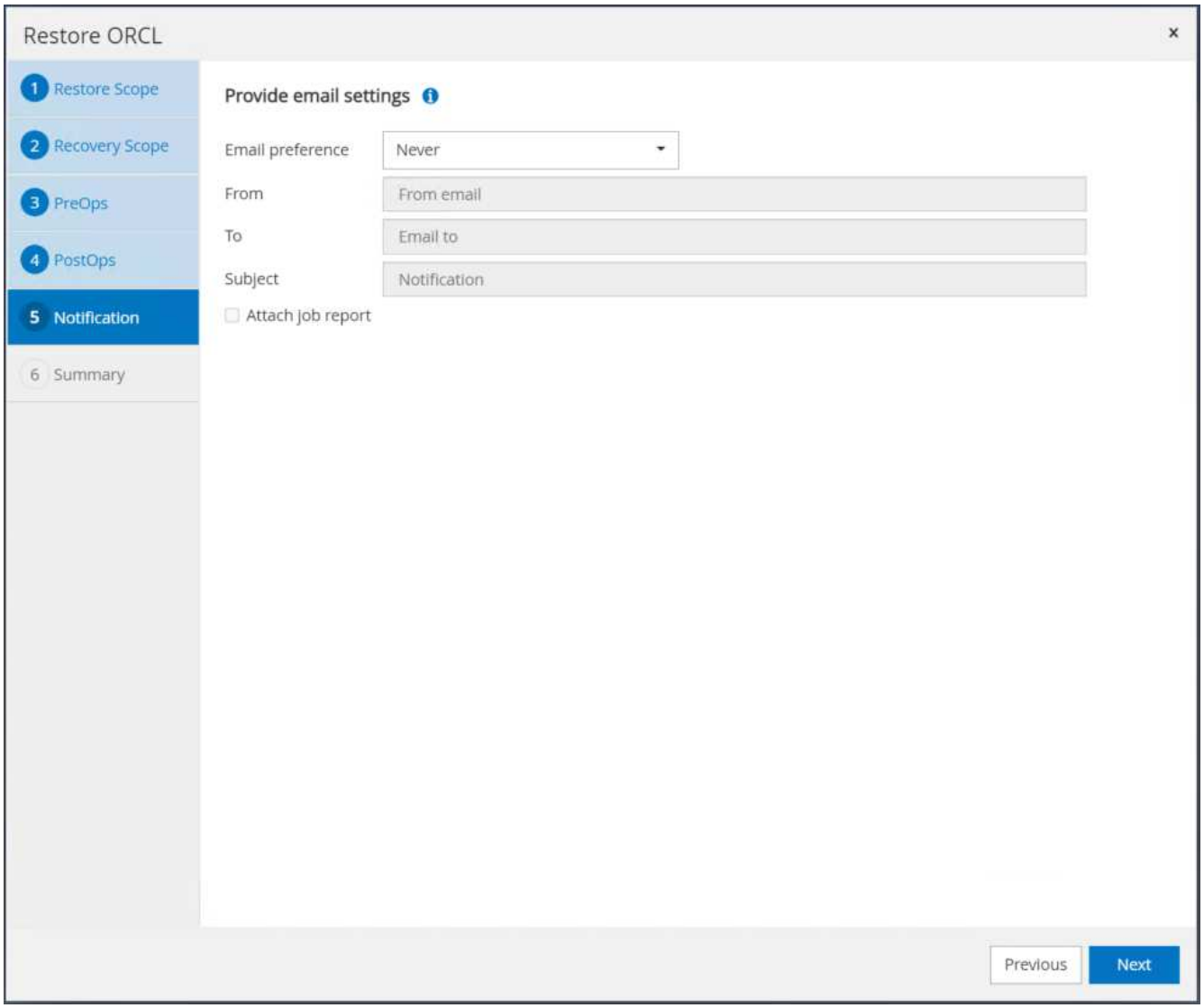
10. 如有必要、指定要运行的可选预处理程序。



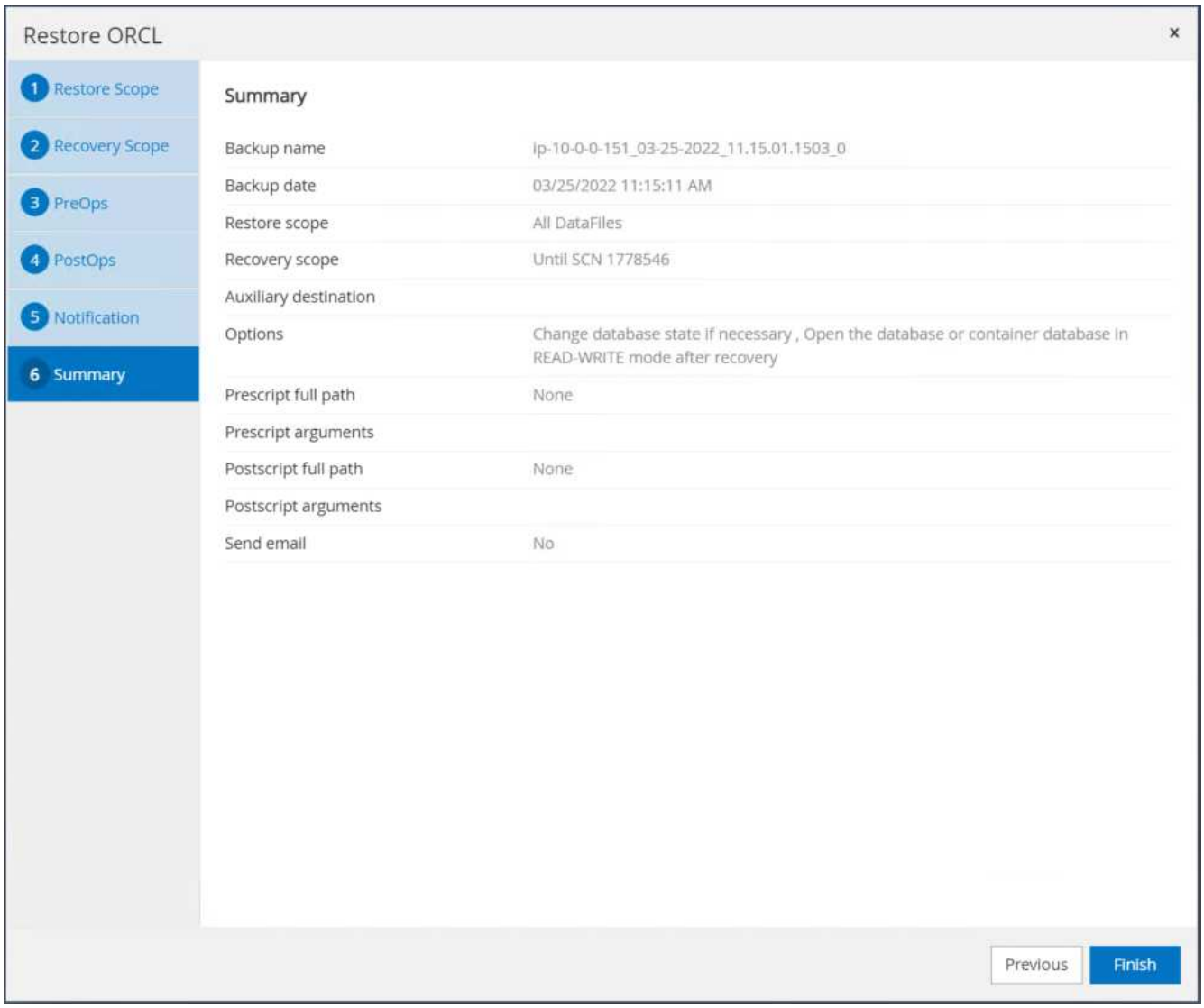
11. 指定一个可选的后处理脚本、以便在必要时运行。恢复后检查打开的数据库。



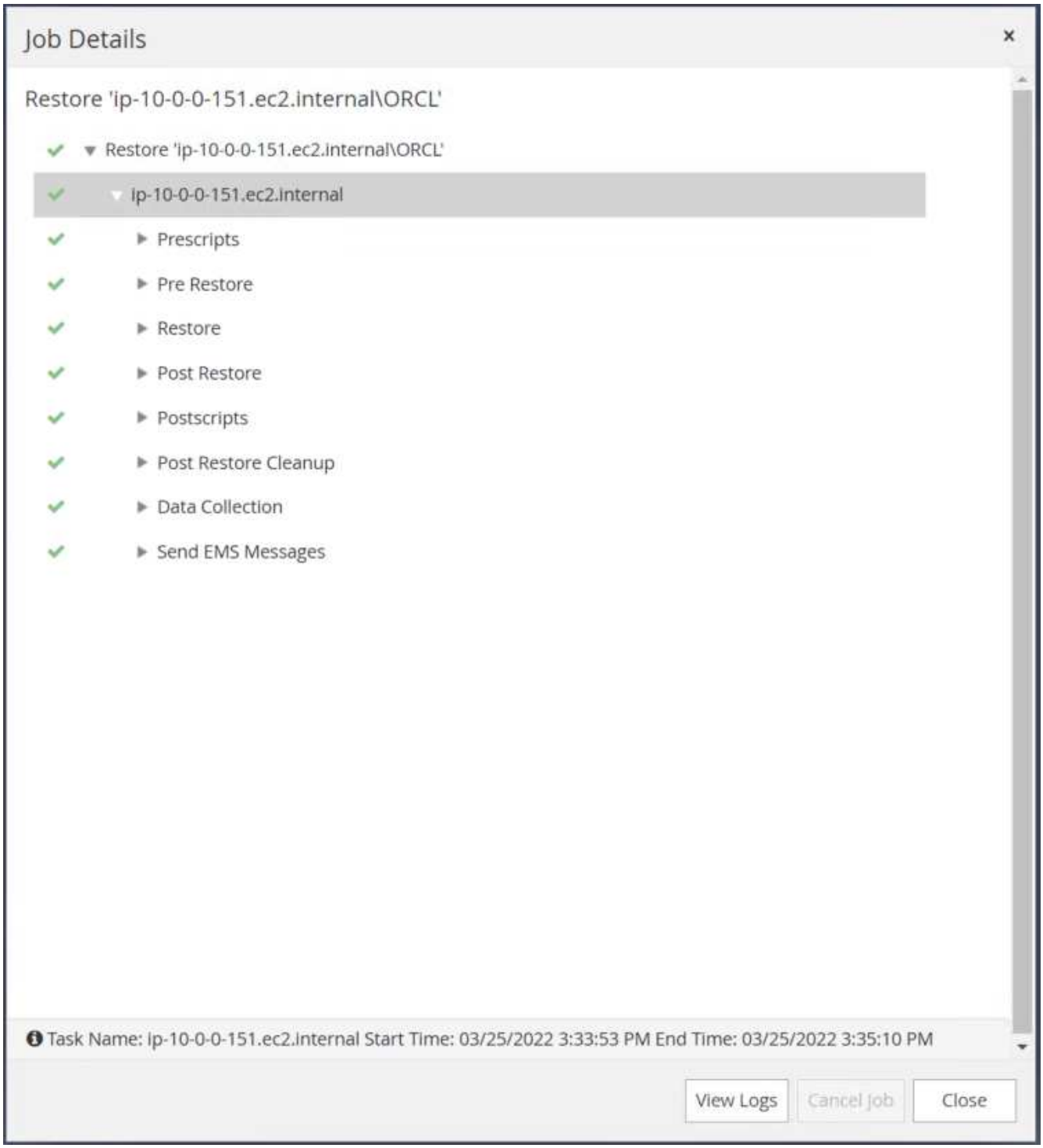
12. 如果需要作业通知、请提供SMTP服务器和电子邮件地址。



13. 还原作业摘要。单击完成以启动还原作业。



14. 验证是否已从SnapCenter 还原。



15. 验证从EC2实例主机还原的情况。

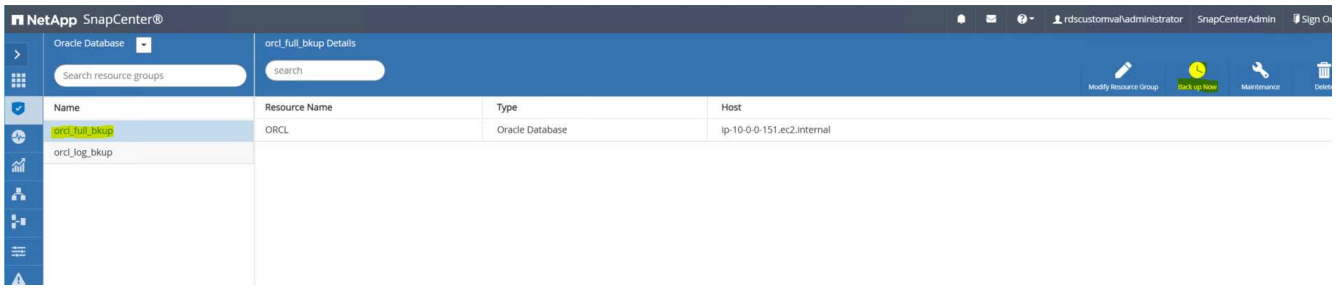
```
-bash-4.2$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Fri Mar 25 15:44:08 2022
Version 19.8.0.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL> select name, RESETLOGS_CHANGE#, RESETLOGS_TIME, open_mode from v$database;

NAME      RESETLOGS_CHANGE# RESETLOGS OPEN_MODE
--------- ----------------- --------- --------------------
ORCL                1778547 25-MAR-22 READ WRITE

SQL>
```

16. 要卸载还原日志卷、请反转步骤4中的步骤。

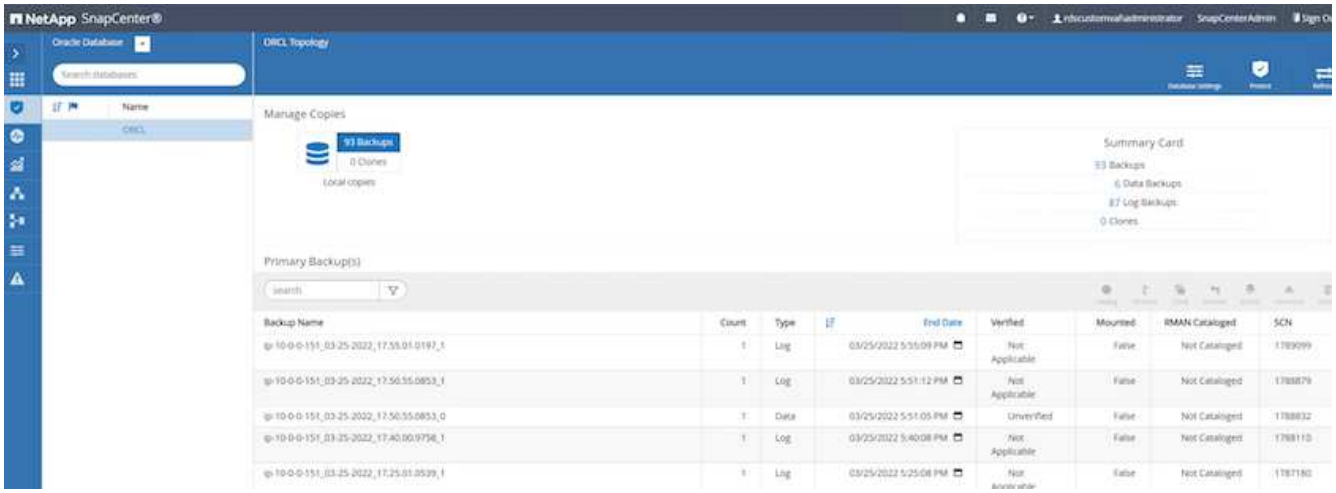
创建数据库克隆

下一节将演示如何使用SnapCenter 克隆工作流创建从主数据库到备用EC2实例的数据库克隆。

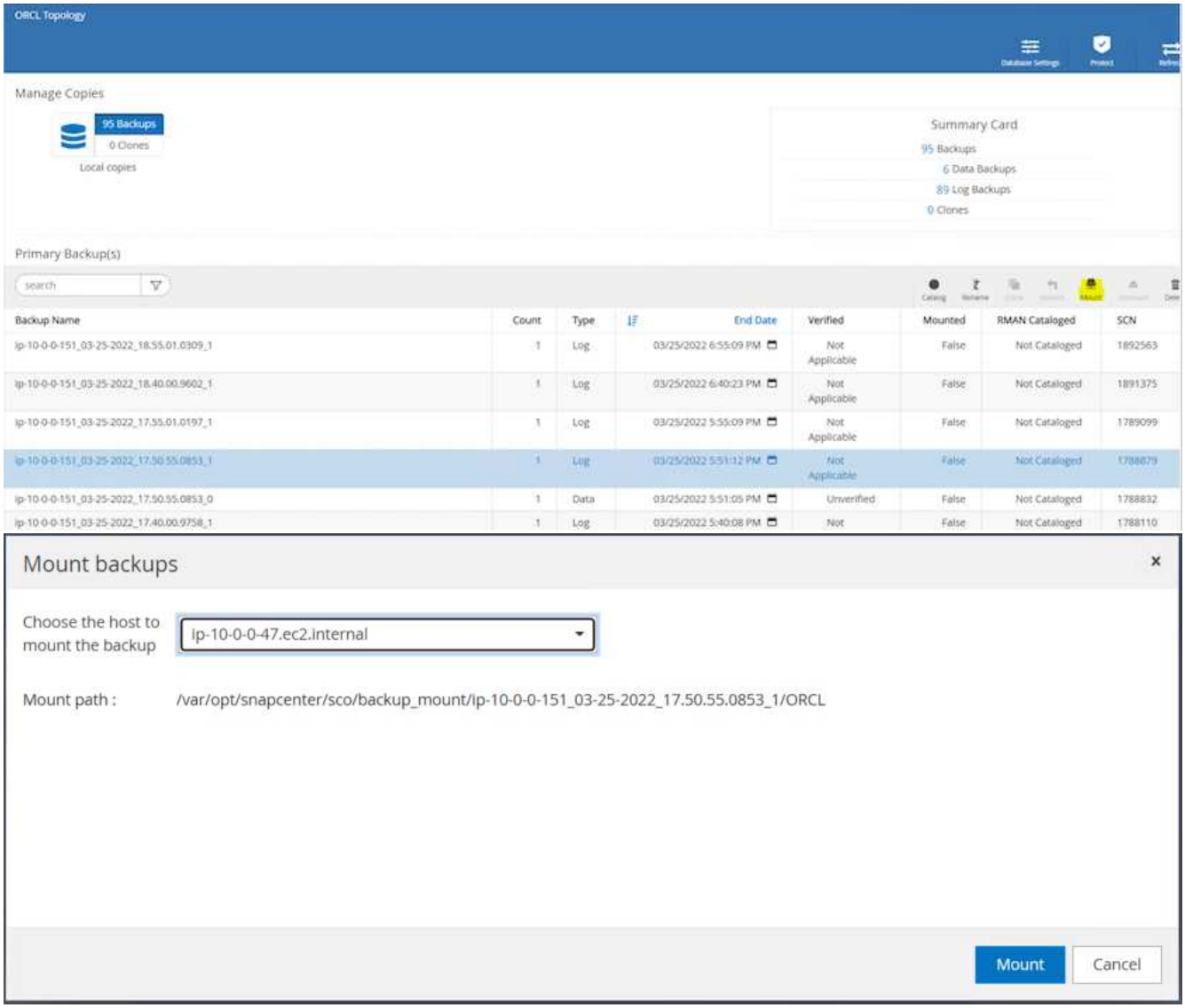
1. 使用完整备份资源组从SnapCenter 为主数据库创建完整快照备份。



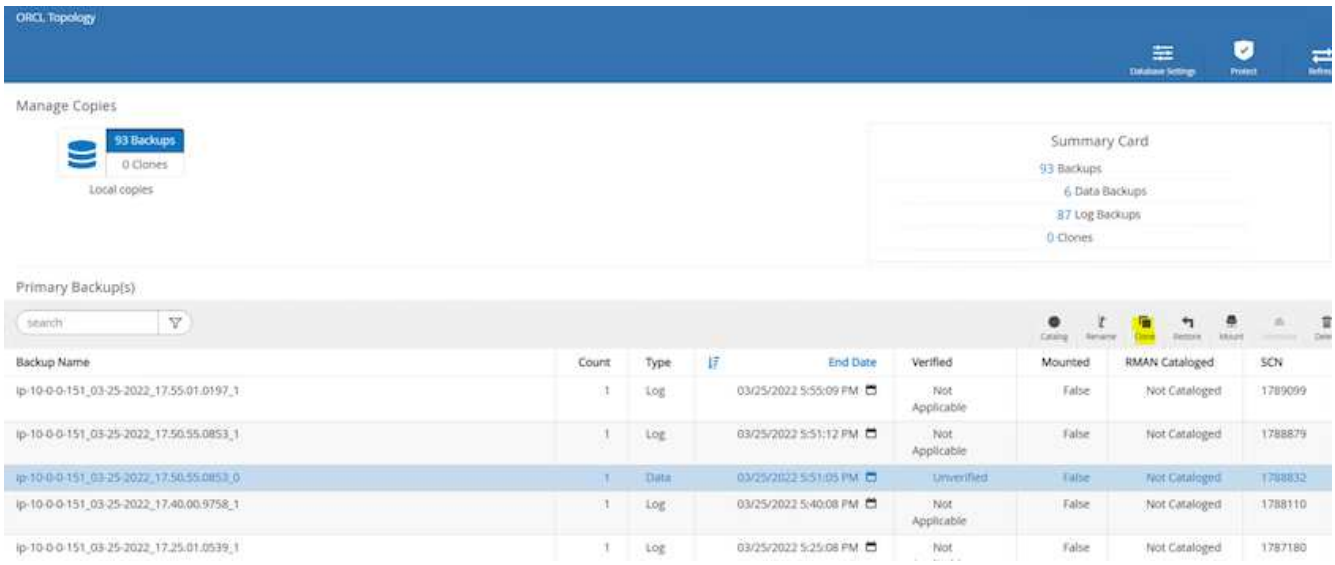
2. 从SnapCenter 资源选项卡>数据库视图中、打开要从中创建副本的主数据库的数据库备份管理页面。



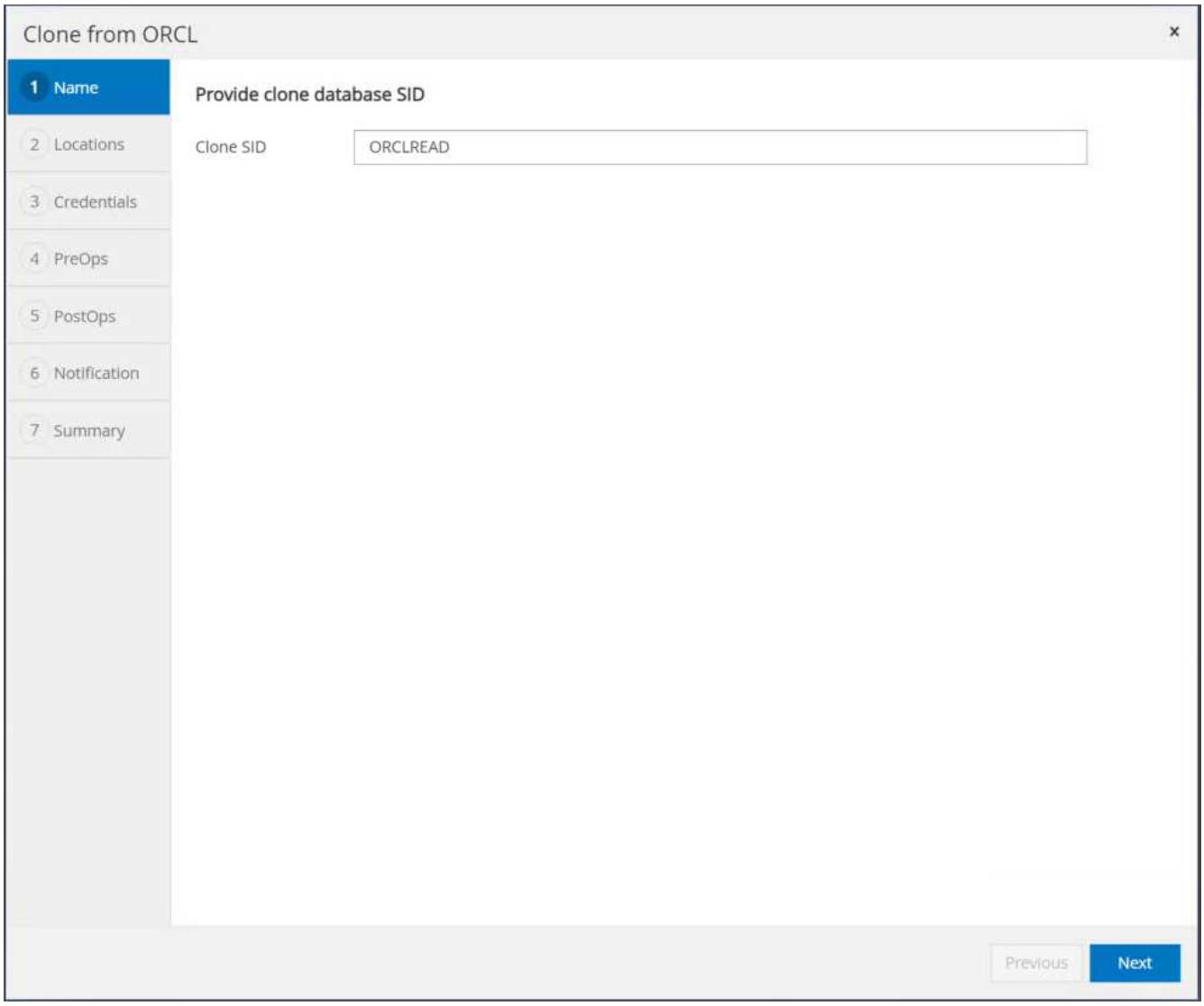
3. 将步骤4中创建的日志卷快照挂载到备用EC2实例主机。



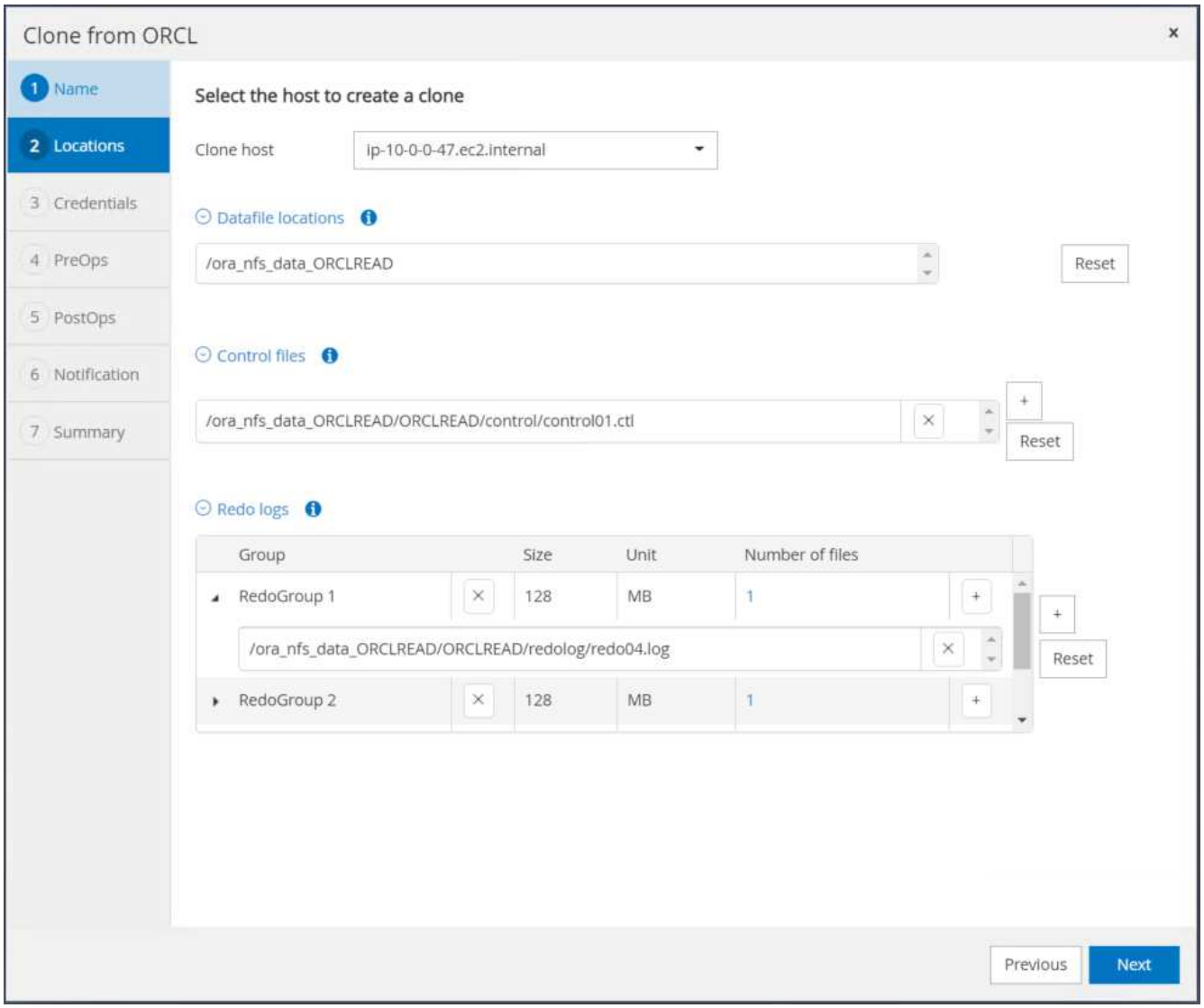
4. 突出显示要为副本克隆的Snapshot副本、然后单击克隆按钮启动克隆操作步骤。



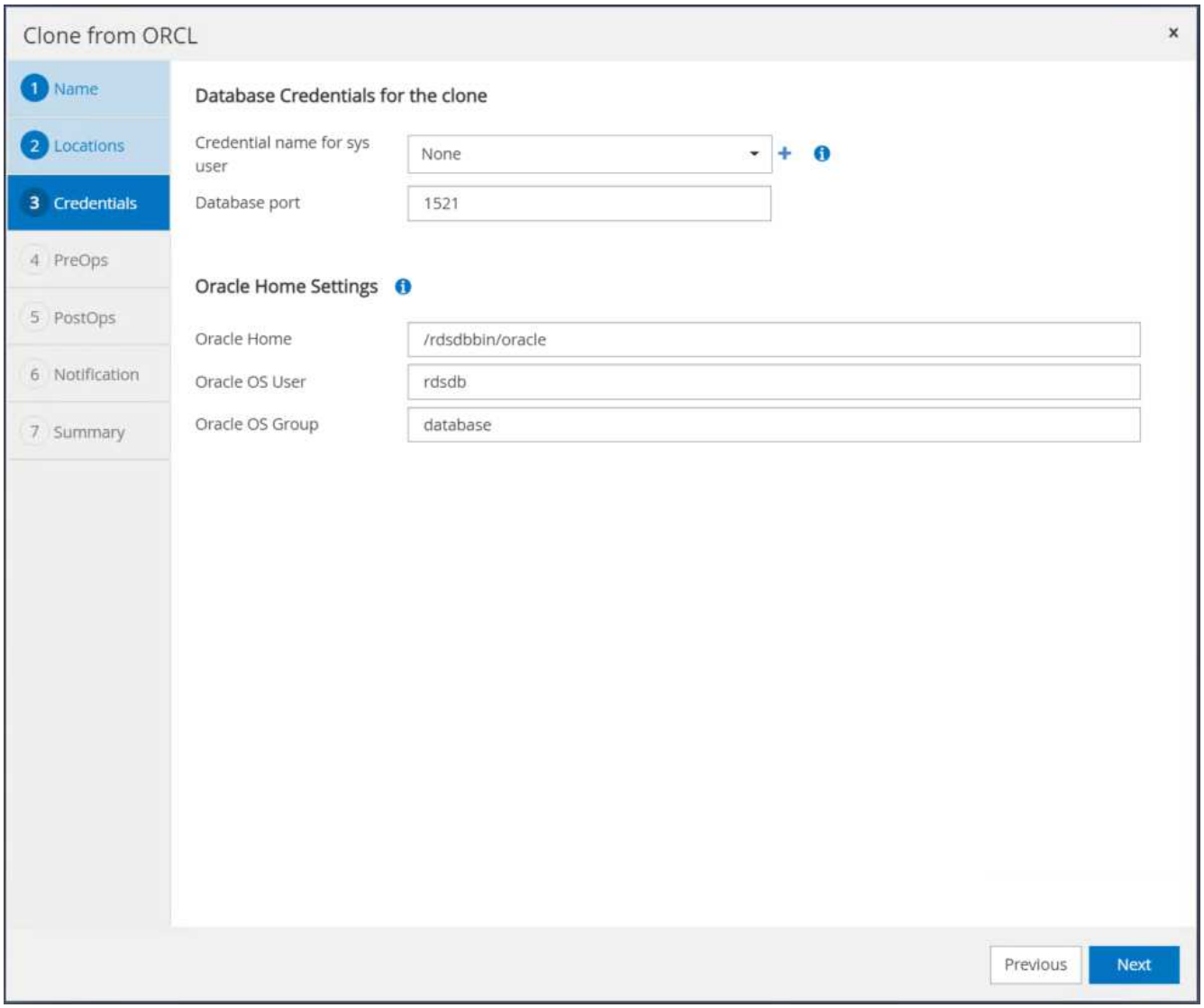
5. 更改副本副本名称、使其与主数据库名称不同。单击下一步。



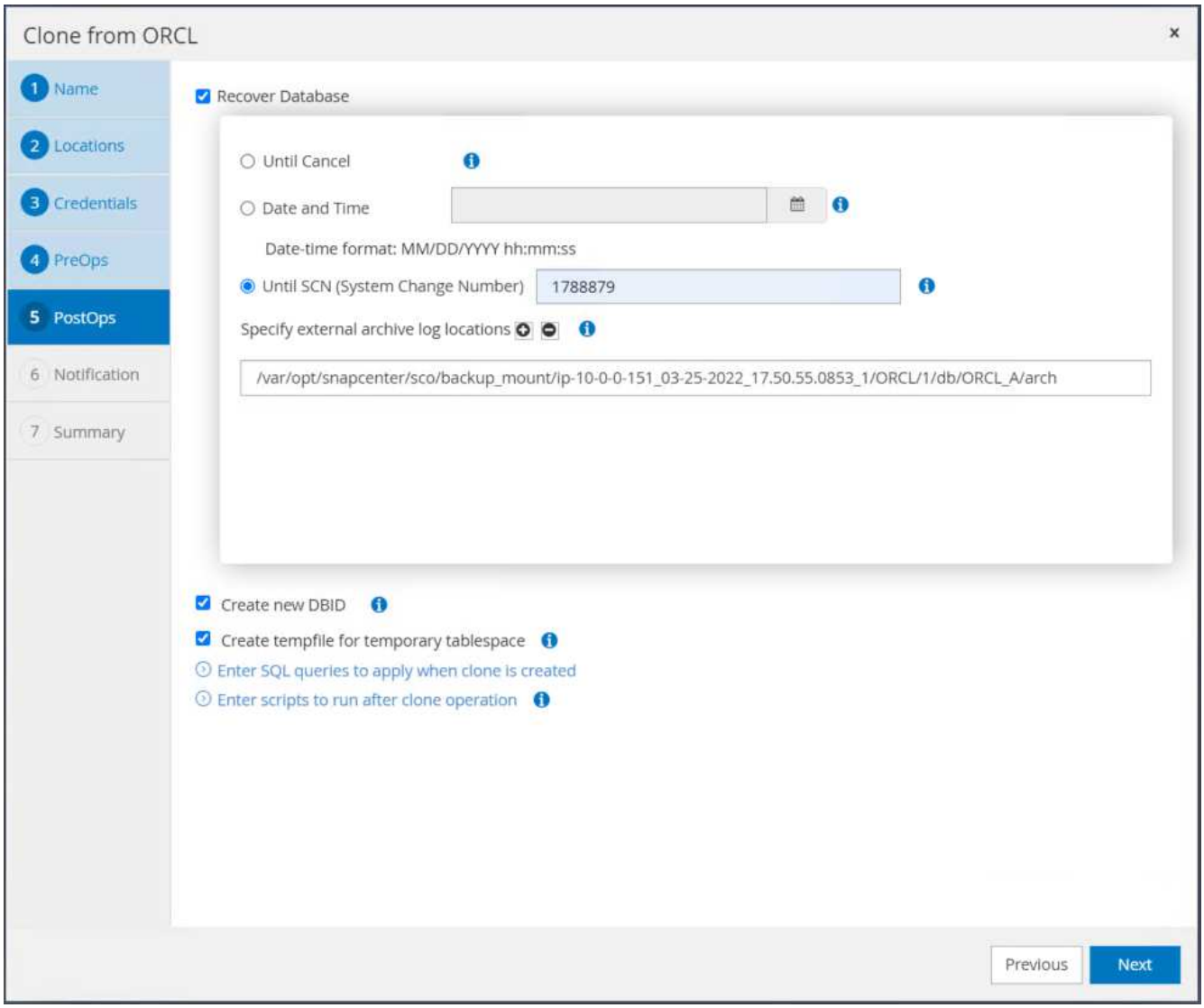
6. 将克隆主机更改为备用EC2主机、接受默认命名、然后单击下一步。



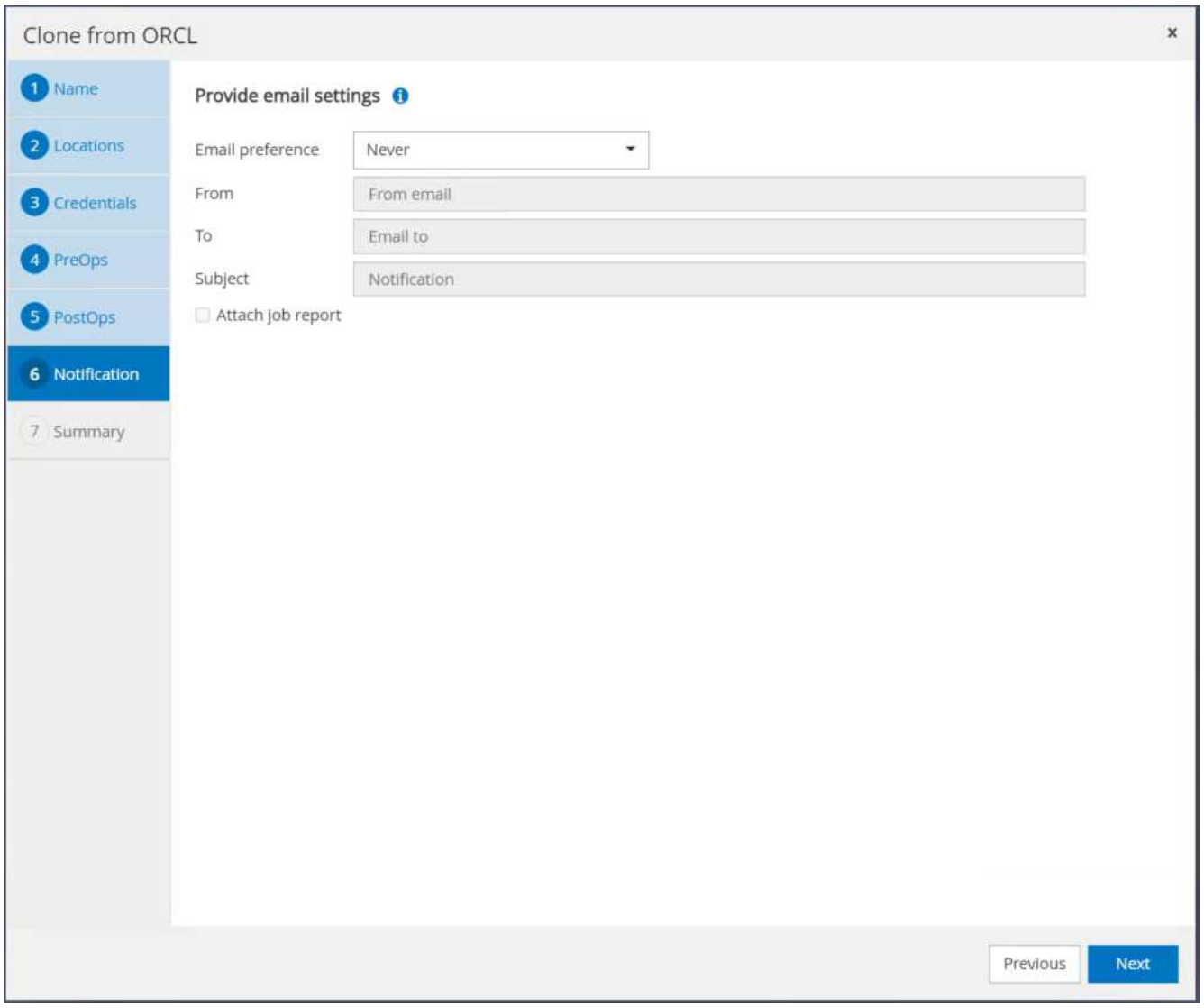
7. 更改Oracle主设置以与为目标Oracle服务器主机配置的设置相匹配、然后单击下一步。



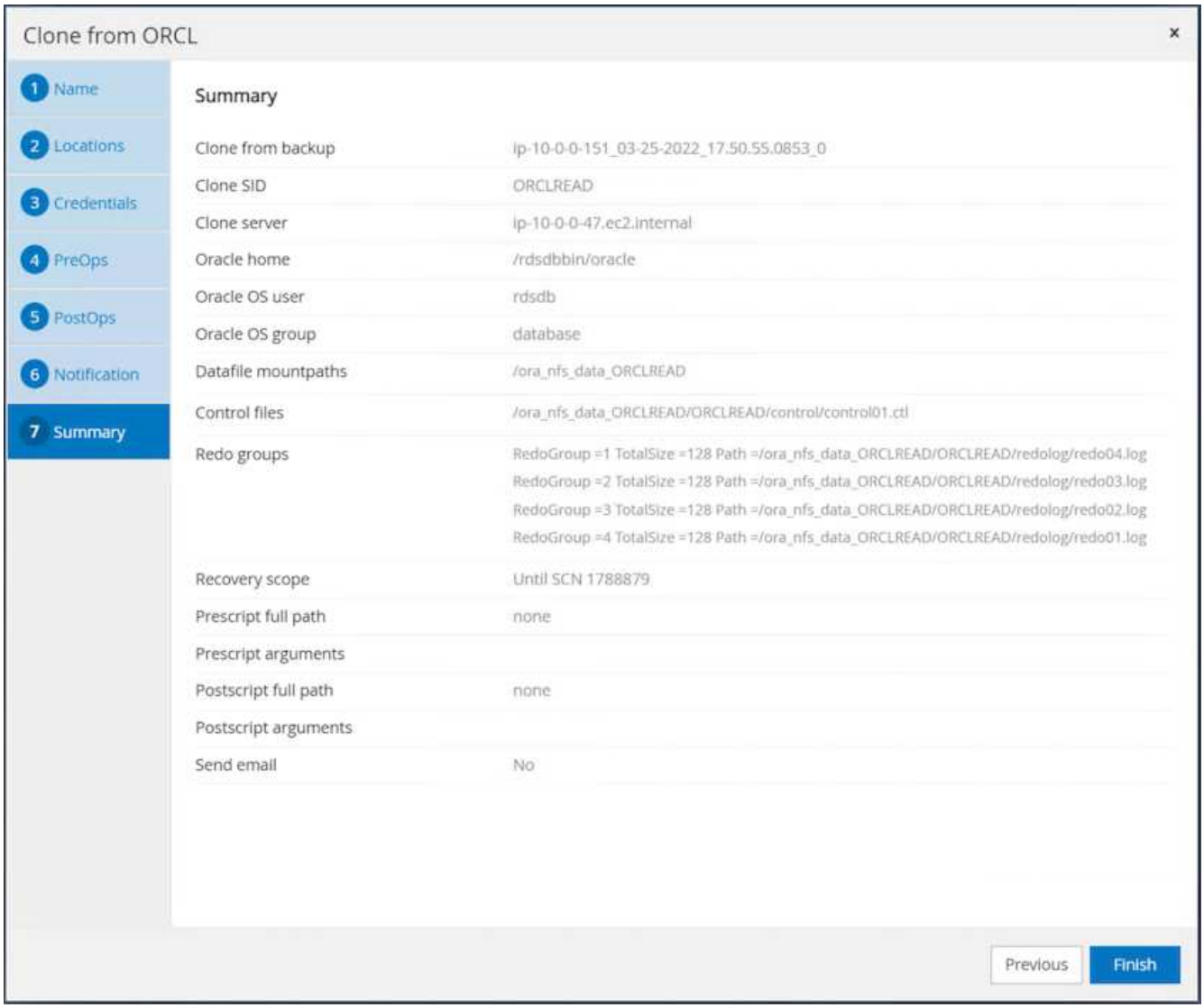
8. 使用时间或SCN和挂载的归档日志路径指定恢复点。



9. 根据需要发送SMTP电子邮件设置。

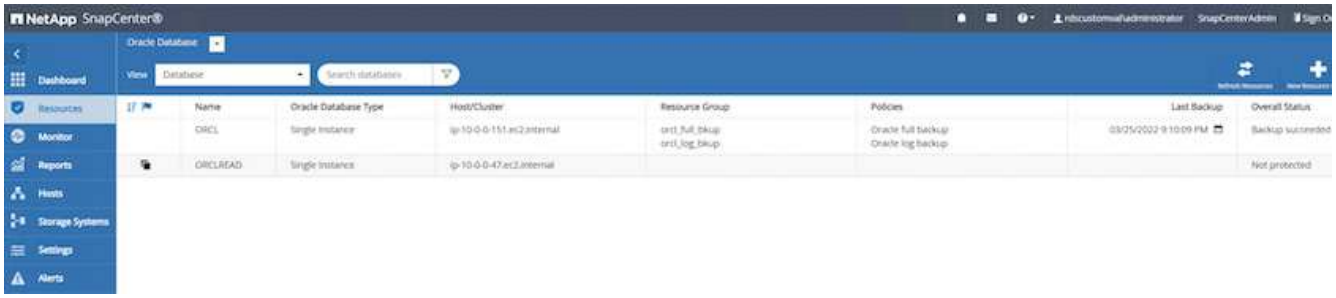


10. 克隆作业摘要、然后单击完成以启动克隆作业。



11. 通过查看克隆作业日志来验证副本克隆。

克隆的数据库会立即在SnapCenter 中注册。



12. 关闭Oracle归档日志模式。以Oracle用户身份登录到EC2实例并执行以下命令：

```
sqlplus / as sysdba
```

```
shutdown immediate;
```

```
startup mount;
```

```
alter database noarchivelog;
```

```
alter database open;
```

与主Oracle备份副本不同、也可以使用相同的过程从目标FSX集群上复制的二级备份副本创建克隆。

HA故障转移到备用并重新同步

备用Oracle HA集群可在主站点发生故障时提供高可用性、无论是在计算层还是存储层。解决方案 的一个重要优势是、用户可以随时或以任何频率测试和验证基础架构。故障转移可以由用户模拟、也可以由实际故障触发。故障转移过程完全相同、可以自动执行、以便快速恢复应用程序。

请参见以下故障转移过程列表：

1. 对于模拟故障转移、请运行日志快照备份、将最新事务刷新到备用站点、如一节所示 [创建归档日志快照]。对于因实际故障而触发的故障转移、最后一个可恢复的数据将通过上次成功计划的日志卷备份复制到备用站点。
2. 中断主FSX集群和备用FSX集群之间的SnapMirror。
3. 在备用EC2实例主机上挂载复制的备用数据库卷。
4. 如果复制的Oracle二进制文件用于Oracle恢复、请重新链接Oracle二进制文件。
5. 将备用Oracle数据库恢复到最后一个可用的归档日志。
6. 打开备用Oracle数据库以供应用程序和用户访问。
7. 对于实际主站点故障、备用Oracle数据库现在充当新的主站点、数据库卷可用于使用反向SnapMirror方法将故障主站点重建为新的备用站点。
8. 对于用于测试或验证的模拟主站点故障、请在完成测试练习后关闭备用Oracle数据库。然后、从备用EC2实例主机卸载备用数据库卷、并将复制从主站点重新同步到备用站点。

可以使用NetApp自动化工具包执行这些过程、该工具包可从公有 NetApp GitHub站点下载。

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

在尝试进行设置和故障转移测试之前、请仔细阅读自述文件说明。

将数据库从内部迁移到公有 云

数据库迁移无论如何都是一项极具挑战性的工作。将Oracle数据库从内部迁移到云也不例外。

以下各节提供了在使用AWS EC2计算和FSX存储平台将Oracle数据库迁移到AWS公有 云时需要考虑的关键因素。

ONTAP 存储可在内部使用

如果内部Oracle数据库位于ONTAP 存储阵列上、则可以使用AWS FSX ONTAP 存储中内置的NetApp SnapMirror技术更轻松地设置复制以进行数据库迁移。可以使用NetApp BlueXP控制台协调迁移过程。

1. 构建与内部实例匹配的目标计算EC2实例。
2. 从FSX控制台配置大小相等的匹配数据库卷。
3. 将FSX数据库卷挂载到EC2实例。
4. 在内部数据库卷与目标FSX数据库卷之间设置SnapMirror复制。初始同步可能需要一段时间才能移动主源数据、但后续的任何增量更新都要快得多。
5. 切换时、关闭主应用程序以停止所有事务。在Oracle sqlplus命令行界面中、执行Oracle联机日志切换并允许SnapMirror同步将最后一个归档日志推送到目标卷。
6. 拆分镜像卷、在目标上运行Oracle恢复、并启动数据库以进行服务。
7. 将应用程序指向云中的Oracle数据库。

以下视频演示了如何使用NetApp BlueXP控制台和SnapMirror复制将Oracle数据库从内部迁移到AWS FSX/EC2。

将内置Oracle数据库迁移到AWS

ONTAP 存储在内部不可用

如果内部Oracle数据库托管在ONTAP 以外的第三方存储上、则数据库迁移基于还原Oracle数据库备份副本。切换前、必须播放归档日志以使其保持最新。

AWS S3可用作数据库移动和迁移的暂存存储区域。有关此方法、请参见以下高级步骤：

1. 配置一个与内部实例相当的新的匹配EC2实例。
2. 从FSX存储配置等效的数据库卷、并将这些卷挂载到EC2实例。
3. 创建磁盘级Oracle备份副本。
4. 将备份副本移动到AWS S3存储。
5. 重新创建Oracle控制文件、并通过从S3存储中提取数据和归档日志来还原和恢复数据库。

6. 将目标Oracle数据库与内部源数据库同步。
7. 切换时、关闭应用程序和源Oracle数据库。复制最后几个归档日志并将其应用于目标Oracle数据库以使其保持最新。
8. 启动目标数据库以供用户访问。
9. 将应用程序重定向到目标数据库以完成切换。

使用**PDB**重新定位将内部**Oracle**数据库迁移到**AWS FSX/EC2**、并最大程度地提高可用性

这种迁移方法最适合已部署在PDB或CDB多租户模式下的Oracle数据库、并且ONTAP 存储在内部不可用。PDB重新定位方法利用Oracle PDB热克隆技术在源CDB和目标CDB之间移动PDB、同时最大程度地减少服务中断。

首先、在AWS FSX/EC2中创建CDB、并为其提供足够的存储空间来托管要从内部迁移的PDB。可以一次重新定位一个多个内部部署的PDB。

1. 如果内部数据库部署在单个实例中、而不是部署在多租户PDB或CDB模式中、请按照中的说明进行操作 "将单个实例非CDB转换为多租户CDB中的PDB" 将单个实例转换为多租户PDB或CDB。然后、按照下一步将转换后的PDB迁移到AWS FSX/EC2中的CDB。
2. 如果已在多租户PDB或CDB模式中部署内部数据库、请按照中的说明进行操作 "通过PDB重新定位将内部Oracle数据库迁移到云" 以执行迁移。

以下视频演示了如何使用PDB重新定位将Oracle数据库(PDB)迁移到FSX/EC2并最大程度地提高可用性。

"将内部Oracle PDB迁移到具有最大可用性的AWS CDB"

> 尽管第1步和第2步中的说明在Azure公共云环境中进行了说明、但这些过程适用于AWS云、不会发生任何更改。

NetApp解决方案自动化团队提供了一个迁移工具包、可帮助Oracle数据库从内部迁移到AWS云。使用以下命令下载用于PDB重新定位的Oracle数据库迁移工具包。

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

Azure NetApp Files 上的Oracle数据库部署和保护

TR-4954：《Azure NetApp Files 上的Oracle数据库部署和保护》

作者：NetApp公司Allen Cao、Niyaz Mohamed

概述

许多任务关键型Oracle企业数据库仍托管在内部、许多企业都希望将这些Oracle数据库迁移到公共云。这些Oracle数据库通常以应用程序为中心、因此需要用户专用配置、而许多数据库即服务公共云产品都缺少这一功能。因此、当前的数据库环境要求基于公共云的Oracle数据库解决方案 、该数据库是基于高性能、可扩展的计算和存储服务构建的、可满足独特的需求。Azure虚拟机计算实例和Azure NetApp Files 存储服务可能是这一难题中缺少的部分、您可以利用它构建任务关键型Oracle数据库工作负载并将其迁移到公共云。

Azure虚拟机

Azure虚拟机是Azure提供的多种按需、可扩展计算资源之一。通常、如果您需要比其他选项更好地控制计算环境、则可以选择虚拟机。Azure虚拟机提供了一种快速简单的方法来创建运行Oracle数据库所需的特定配置的计算机、无论该数据库是用于计算密集型工作负载还是内存密集型工作负载。Azure虚拟网络中的虚拟机可以轻松连接到您组织的网络、例如、通过安全的VPN通道。

Azure NetApp 文件 **(ANF)**

Azure NetApp Files 是一项完全受管的Microsoft服务、它可以让您的数据库工作负载以前所未有的速度更安全地迁移到云中。它旨在满足在云中运行高性能工作负载(例如Oracle数据库)的核心要求、并提供了能够反映实际IOPS需求范围、低延迟、高可用性、高持久性、大规模易管理性的性能层。以及快速高效的备份、恢复和克隆。之所以能够提供这些功能、是因为Azure NetApp Files 基于在Azure数据中心环境中运行的物理全闪存NetApp ONTAP 系统。Azure NetApp Files 完全集成到Azure DC和门户中、客户可以使用与任何其他Azure对象相同的舒适图形界面和API来创建和管理共享文件。借助Azure NetApp文件、您可以在不增加风险、成本或时间的情况下充分发挥Azure的全部功能、并信任Azure自带的唯一企业级文件服务。

结论

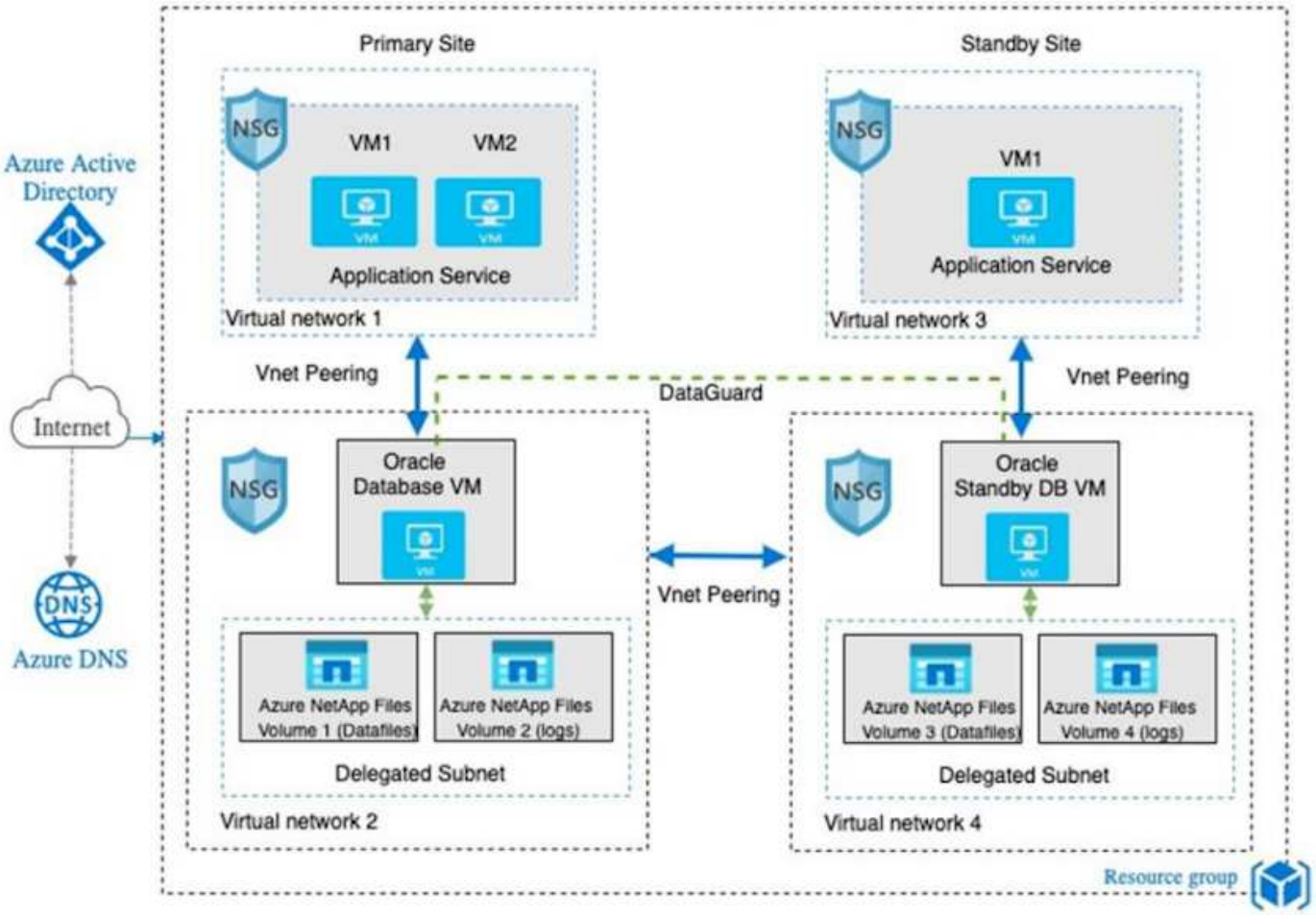
本文档详细介绍了如何使用Azure虚拟机和Azure NetApp Files 存储服务部署、配置和保护Oracle数据库、该服务可提供与内部系统类似的性能和持久性。有关最佳实践指导、请参见TR-4780 "Microsoft Azure上的Oracle数据库"。更重要的是、NetApp还提供了自动化工具包、可自动执行在Azure公共云中部署、配置、数据保护、迁移和管理Oracle数据库工作负载所需的大部分任务。这些自动化工具包可从NetApp公共GitHub站点下载：
"NetApp-Automation"。

解决方案架构

以下架构图展示了在Azure VM实例和Azure NetApp Files 存储上部署高可用性Oracle数据库的情况。

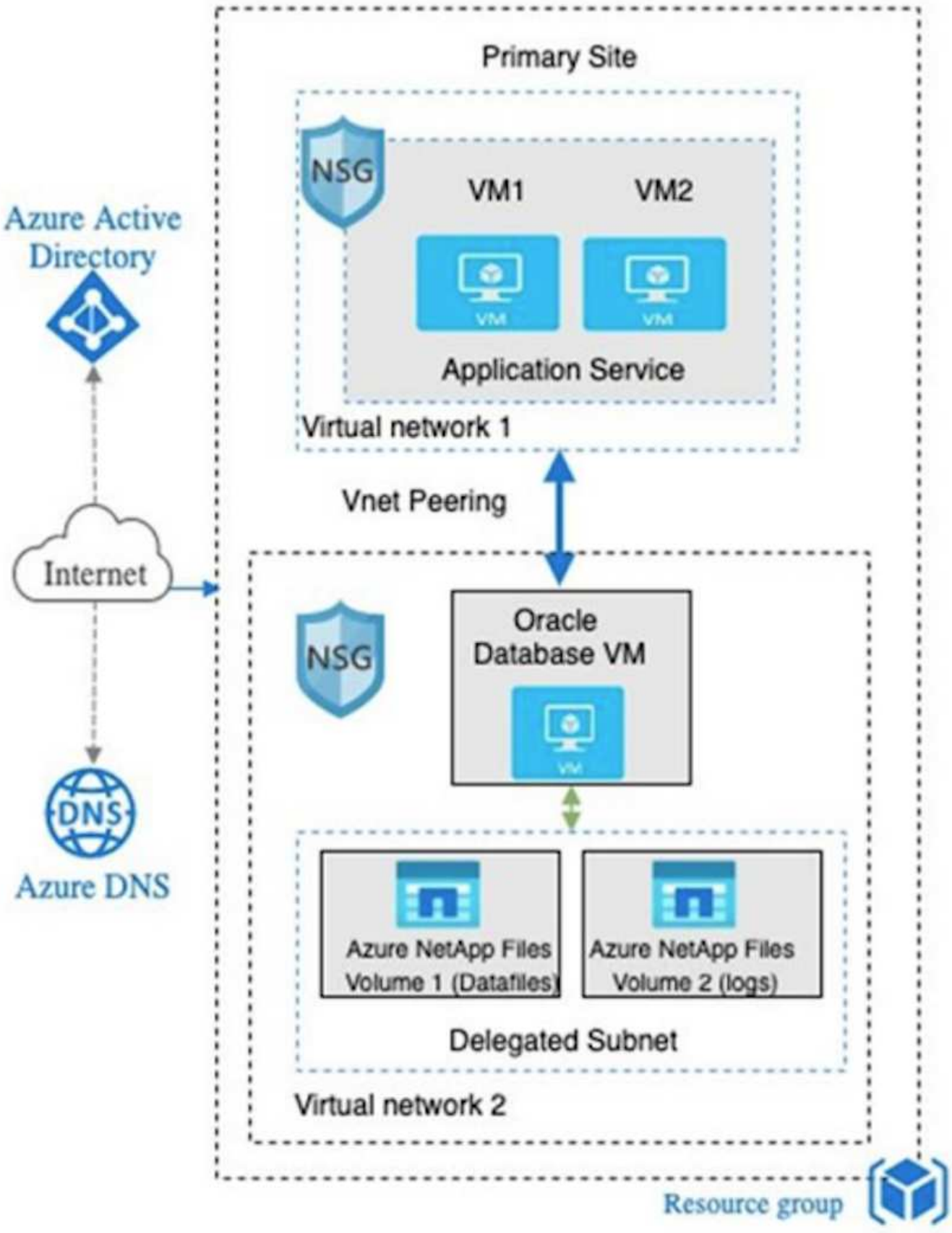
在环境中、Oracle计算实例通过Azure服务VM控制台进行部署。控制台提供了多种Azure实例类型。NetApp建议部署一个面向数据库的Azure VM实例、以满足您的预期工作负载。

另一方面、Oracle数据库存储则通过Azure控制台提供的Azure NetApp Files 服务进行部署。随后、Oracle二进制卷、数据卷或日志卷将显示并挂载到Azure VM实例Linux主机上。



在许多方面、在Azure云中实施Azure NetApp Files 与具有许多内置冗余功能(例如RAID和双控制器)的内部ONTAP 数据存储架构非常相似。对于灾难恢复、可以在不同区域设置备用站点、并且可以使用应用程序级复制(例如Oracle Data Guard)将数据库与主站点同步。

在我们对Oracle数据库部署和数据保护进行的测试验证中、Oracle数据库部署在一个Azure虚拟机上、如下图所示：



Azure Oracle环境可通过Ansible控制器节点进行管理、以便使用NetApp提供的用于数据库部署、备份、恢复和数据库迁移的工具包实现自动化。可以并行执行对Oracle Azure VM实例操作系统内核或Oracle修补的任何更

新、以使主系统和备用系统保持同步。事实上、初始工具包可以轻松扩展、以便在需要时执行日常Oracle任务。如果您需要有关设置CLI Ansible控制器的帮助、请参见 "NetApp 解决方案自动化" 开始使用。

部署Oracle数据库时需要考虑的因素

公有 云为计算和存储提供了多种选择、使用正确类型的计算实例和存储引擎是开始部署数据库的好地方。您还应选择针对Oracle数据库进行优化的计算和存储配置。

以下各节介绍在采用Azure NetApp Files 存储的Azure虚拟机实例上的Azure公共云中部署Oracle数据库时的主要注意事项。

VM类型和规模估算

选择合适的VM类型和大小对于优化公共云中关系数据库的性能非常重要。Azure虚拟机提供了各种计算实例、可用于托管Oracle数据库工作负载。请参见Microsoft文档 "Azure中虚拟机的大小" 不同类型的Azure虚拟机及其规模估算。一般来说、NetApp建议使用通用Azure虚拟机来部署中小型Oracle数据库。对于部署大型Oracle数据库、适合使用经过内存优化的Azure VM。利用更多可用RAM、可以配置更大的Oracle SGA或智能闪存缓存、以减少物理I/O、进而提高数据库性能。

Azure NetApp Files 用作连接到Azure虚拟机的NFS挂载、可提供更高的吞吐量、并通过本地存储克服存储优化的VM吞吐量限制。因此、在Azure NetApp Files 上运行Oracle可以减少可获得许可的Oracle CPU核心数量和许可成本。请参见 "TR-4780：Microsoft Azure上的Oracle数据库"第7节—Oracle许可的工作原理

需要考虑的其他因素包括：

- 根据工作负载特征选择正确的vCPU和RAM组合。随着VM上的RAM大小增加、vCPU核心数也会增加。由于Oracle许可证费用是按vCPU核心数收取的、因此应在某一时刻保持平衡。
- 向VM添加交换空间。默认Azure VM部署不会创建交换空间、而交换空间对于数据库来说并不是最佳选择。

Azure NetApp Files 性能

Azure NetApp Files 卷从客户必须在其Azure NetApp Files 存储帐户中配置的容量池中分配。每个容量池的分配如下：

- 定义整体性能功能的服务级别。
- 最初为此容量池配置的存储容量或分层。一种服务质量(QoS)级别、用于定义每个已配置空间的总最大吞吐量。

服务级别和初始配置的存储容量决定了特定Oracle数据库卷的性能级别。

1. Azure NetApp Files 的服务级别

Azure NetApp Files 支持三种服务级别："超"、"高级"和"标准"。

- *超存储。*此层可为分配的卷配额的每1 TiB提供高达128 MiB的吞吐量。
- *高级存储。*此层分配的卷配额每1 TiB可提供高达64 MiB的吞吐量。
- *标准存储。*此层可为分配的卷配额的每1 TiB提供高达16 MiB的吞吐量。

2.容量池和服务质量

每个所需的服务级别都与已配置容量相关、并包括一个服务质量(QoS)级别、用于定义已配置空间的总最大吞吐量。

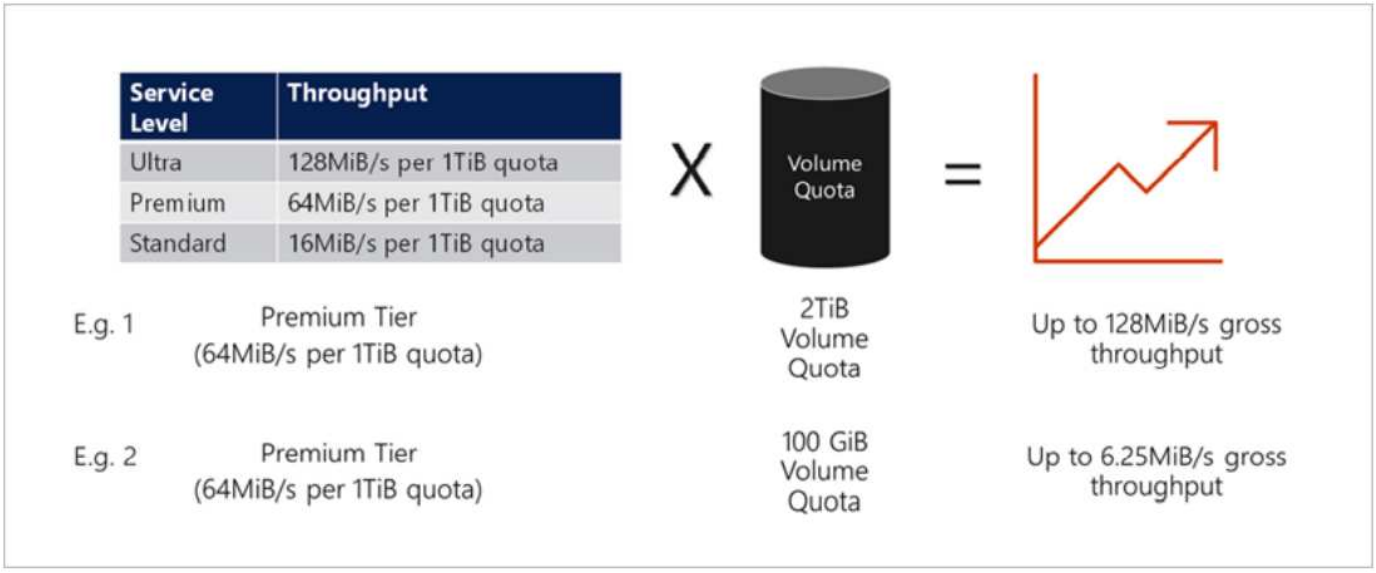
例如、具有高级服务级别的10 TiB配置单容量池可为该容量池中的所有卷提供10倍64 MBps的整体可用吞吐量、因此、640 MBps可提供40、000 (16 K)个IOPS或80、000 (8 K)个IOPS。

最小容量池大小为4 TiB。您可以根据工作负载需求的变化以1 TiB为增量更改容量池的大小、以管理存储需求和成本。

3.计算数据库卷的服务级别

Oracle数据库卷的吞吐量限制由以下因素组合决定：卷所属容量池的服务级别以及分配给卷的配额。

下图显示了如何计算Oracle数据库卷的吞吐量限制。



在示例1中、如果容量池中的高级存储层分配了2 TiB的配额、则该卷的吞吐量限制为128 MiBps (2 TiB * 64 MiBps)。无论容量池大小或实际卷占用情况如何、此方案都适用。

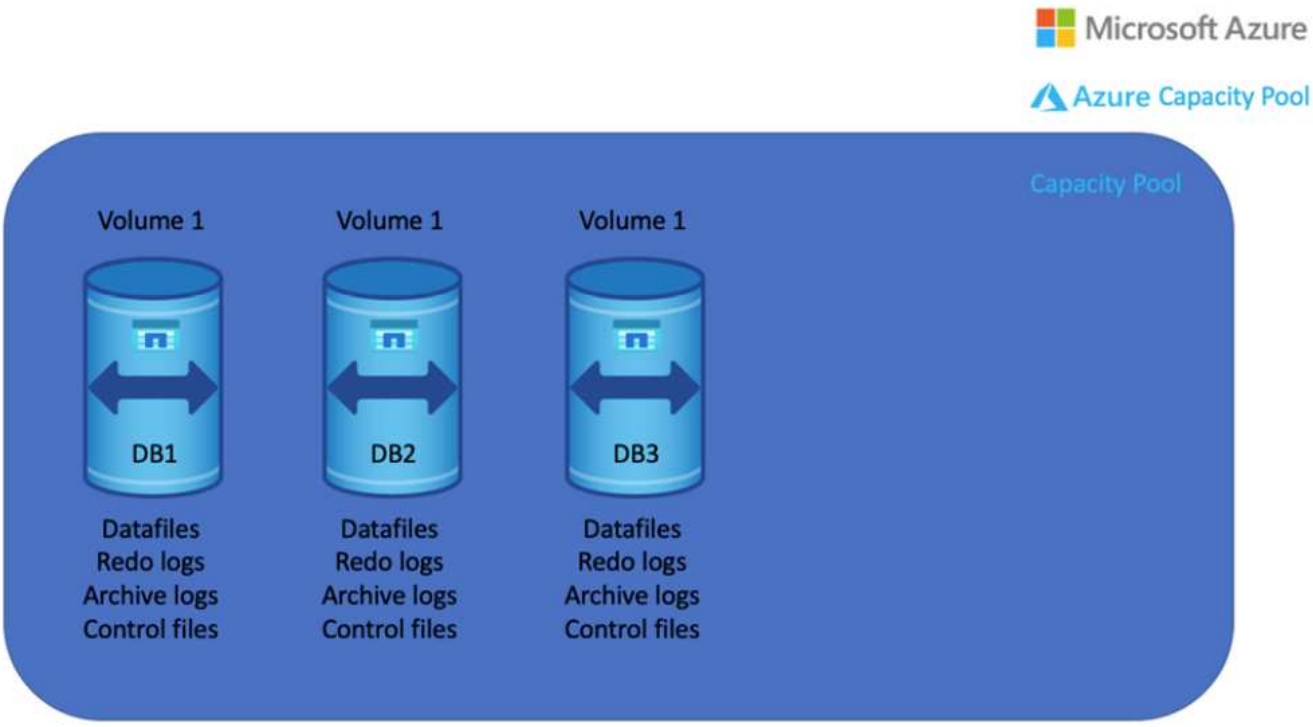
在示例2中、如果容量池中的高级存储层分配了100 GiB的配额、则该卷的吞吐量限制为6.25 MiBps (0.09765625TiB * 64 MiBps)。无论容量池大小或实际卷占用情况如何、此方案都适用。

请注意、最小卷大小为100GiB。

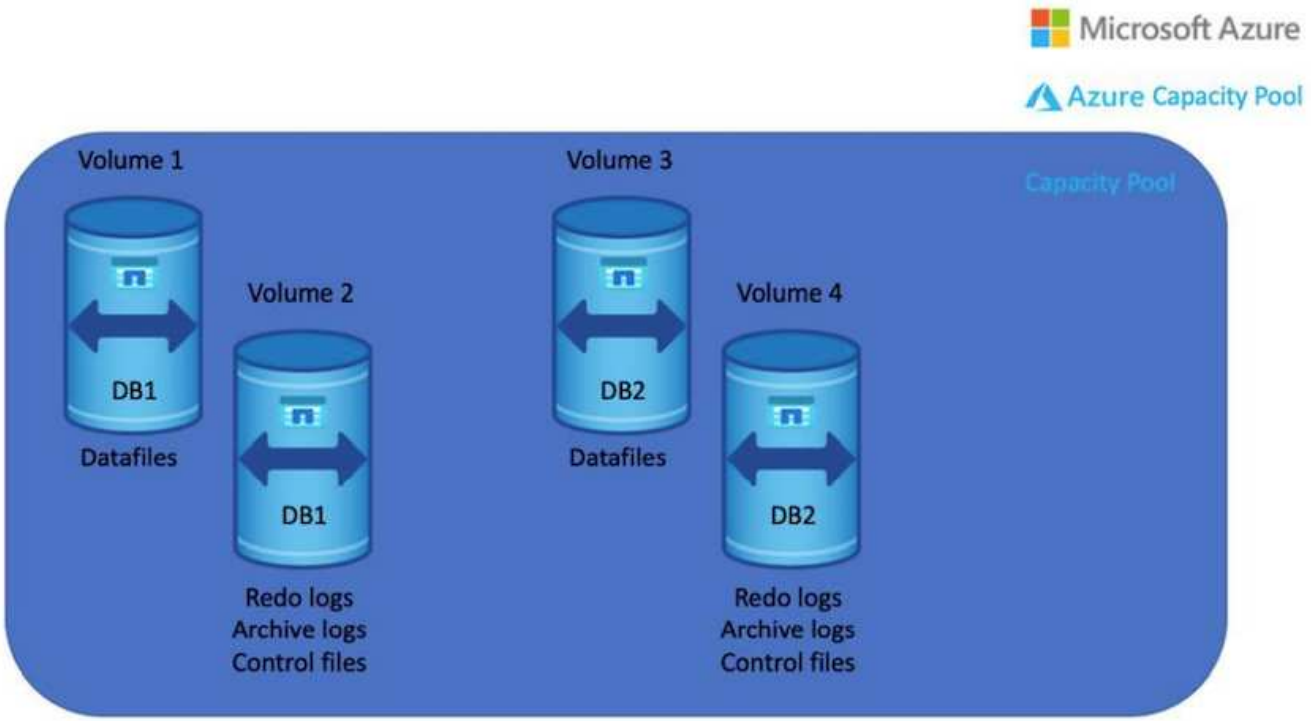
存储布局和设置

NetApp建议采用以下存储布局：

- 对于小型数据库、对所有Oracle文件使用单个卷布局。



- 对于大型数据库、建议的卷布局为多个卷：一个用于Oracle数据和一个重复的控制文件、一个用于Oracle活动日志、归档日志和控制文件。NetApp强烈建议为Oracle二进制文件而不是本地驱动器分配一个卷、以便可以将数据库重新定位到新主机并快速还原。

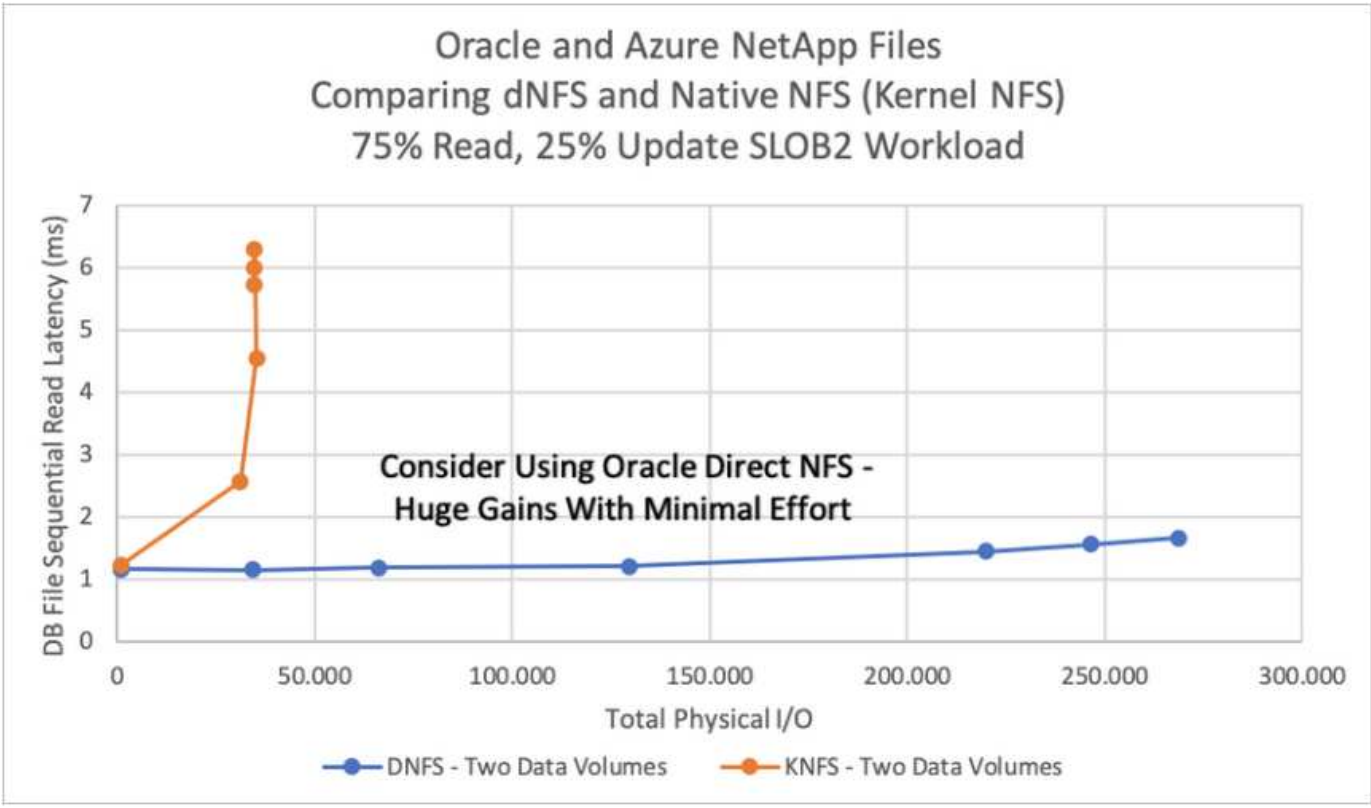


NFS 配置

最常见的操作系统Linux具有原生 NFS功能。Oracle提供了一个直接NFS (DNFS)客户端、该客户端本机集成到Oracle中。Oracle DNFS可绕过操作系统缓存并启用并行处理以提高数据库性能。Oracle支持NFSv3的时间已超过20年、而Oracle 12.1.0.2及更高版本支持NFSv4。

通过使用DNFS (自Oracle 11g起提供)、在Azure虚拟机上运行的Oracle数据库可以比本机NFS客户端驱动更多的I/O。使用NetApp自动化工具包自动部署Oracle会自动在NFSv3上配置DNFS。

下图展示了采用Oracle DNFS的Azure NetApp Files 上的SLOB基准测试。



需要考虑的其他因素：

- TCP插槽表是主机总线适配器(host-bus-adapter、HBA)队列深度的NFS等效项。这些表可控制任何时候都可以处理的NFS操作的数量。默认值通常为16、该值太低、无法实现最佳性能。在较新的Linux内核上会出现相反的问题、这会自动将TCP插槽表限制增加到使NFS服务器充满请求的级别。

  为了获得最佳性能并防止出现性能问题、请将控制TCP插槽表的内核参数调整为128。

```
sysctl -a | grep tcp.*.slot_table
```

- 下表提供了单个Linux NFSv3实例的建议NFS挂载选项。

File Type	Mount Options
• Control files • Data files • Redo logs	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536
• ORACLE_HOME • ORACLE_BASE	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsize=65536

在使用DNFS之前、请验证是否已安装Oracle文档1495104.1中所述的修补程序。NetApp针对NFSv3和NFSv4的支持列表不包括特定的操作系统。支持所有符合RFC的操作系统。在联机IMT 中搜索NFSv3或NFSv4支持时、请勿选择特定操作系统、因为不会显示任何匹配项。常规策略隐式支持所有操作系统。

Azure VM和Azure NetApp Files 上的Oracle分步部署过程

通过**Azure**门户控制台使用适用于**Oracle**的**ANF**部署**Azure**虚拟机

如果您是Azure的新用户、则首先需要设置Azure帐户环境。这包括注册您的组织以使用Azure Active Directory。以下部分总结了这些步骤。有关详细信息、请参见特定于Azure的链接文档。

创建和使用**Azure**资源

设置Azure环境并创建帐户并将其与订阅关联后、您可以使用帐户登录到Azure门户以创建运行Oracle所需的资源。

1.创建虚拟网络或**vNet**

Azure虚拟网络(vNet)是Azure中专用网络的基本组件。VNet支持Azure虚拟机(VM)等多种类型的Azure资源安全地相互通信、并与Internet和内部网络进行通信。在配置Azure VM之前、必须先配置vNet (部署VM的位置)。

请参见 "使用Azure门户创建虚拟网络" 创建vNet。

2.为**ANF**创建**NetApp**存储帐户和容量池

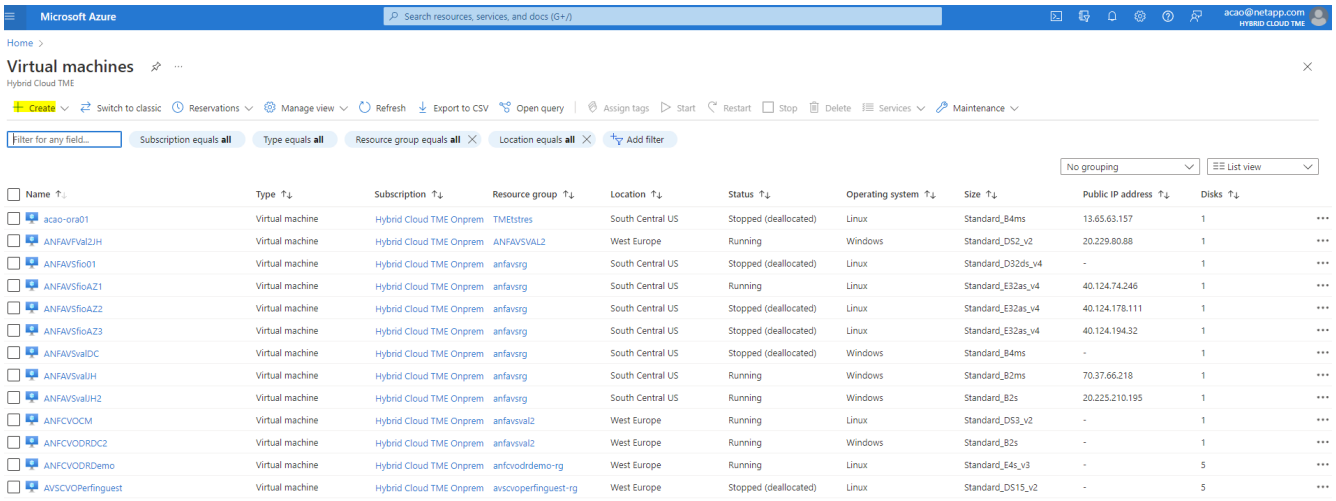
在此部署场景中、Azure VM操作系统使用常规Azure存储进行配置、但ANF卷配置为通过NFS运行Oracle数据库。首先、您需要创建NetApp存储帐户和容量池来托管存储卷。

请参见 "设置Azure NetApp Files 并创建NFS卷" 设置ANF容量池。

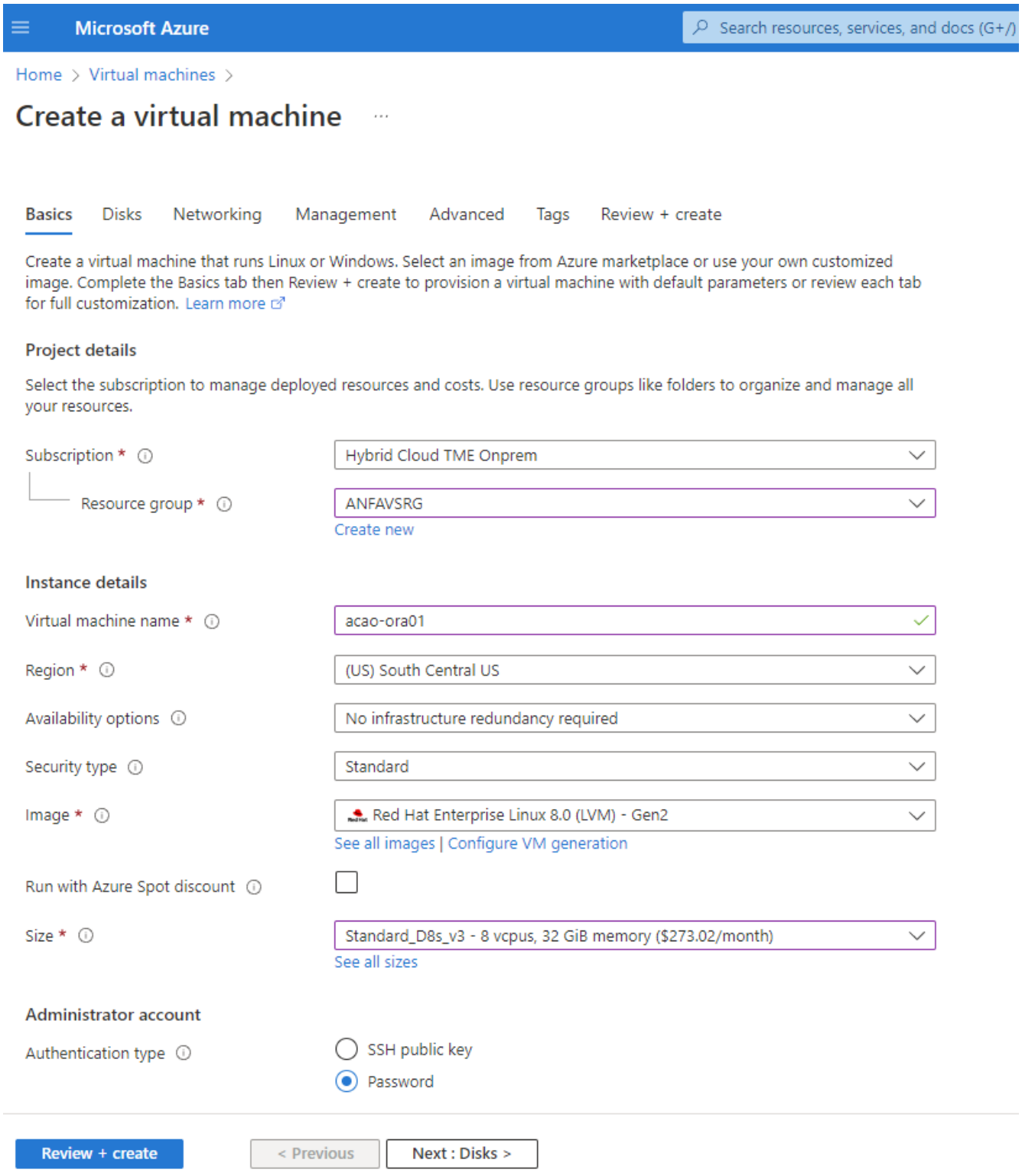
3.为**Oracle**配置**Azure VM**

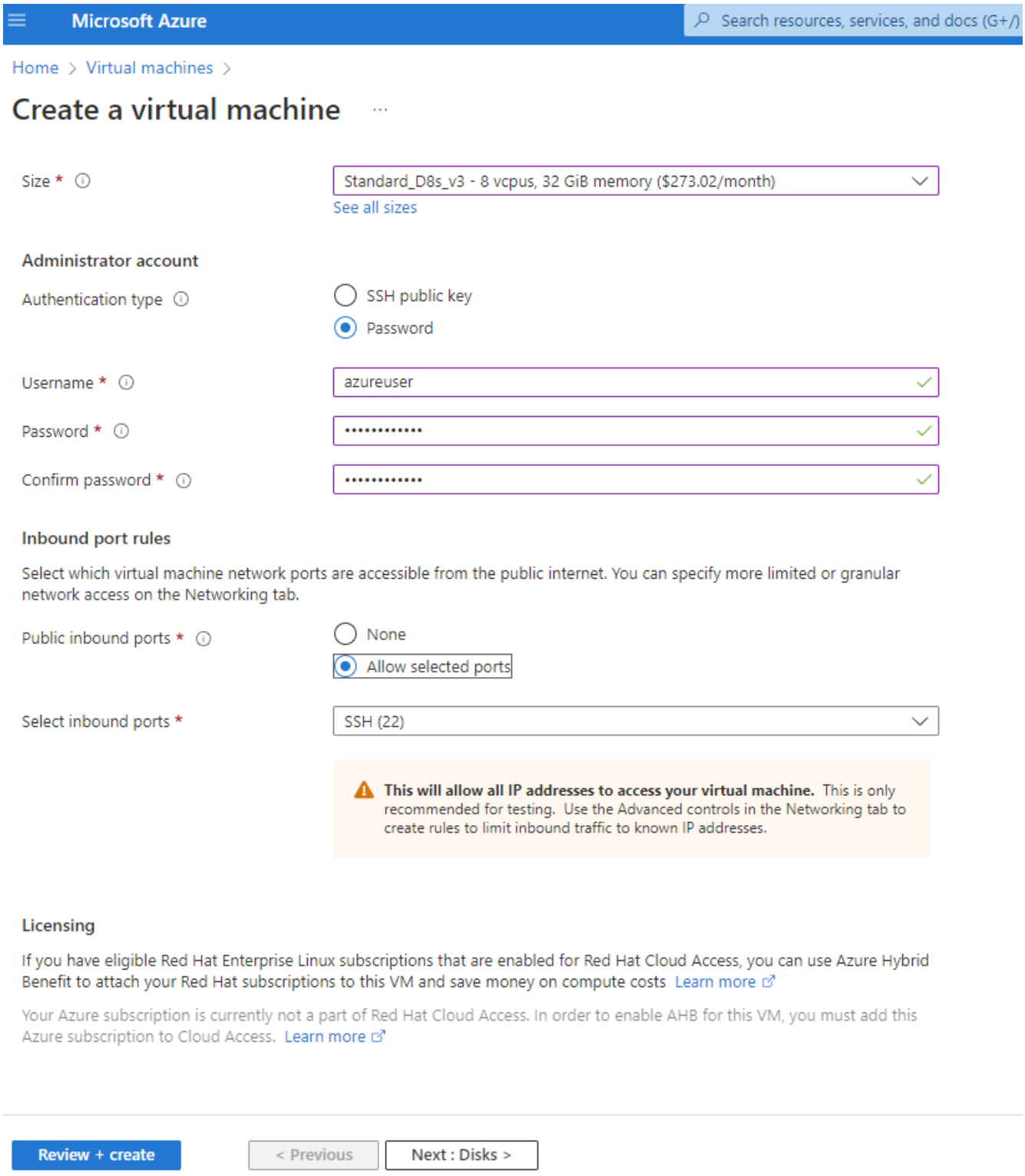
根据您的工作负载、确定您需要哪种类型的Azure VM以及要为Oracle部署的VM vCPU和RAM的大小。然后、在Azure控制台中、单击虚拟机图标以启动虚拟机部署工作流。

1. 在Azure VM页面中、单击*创建*、然后选择* Azure虚拟机*。

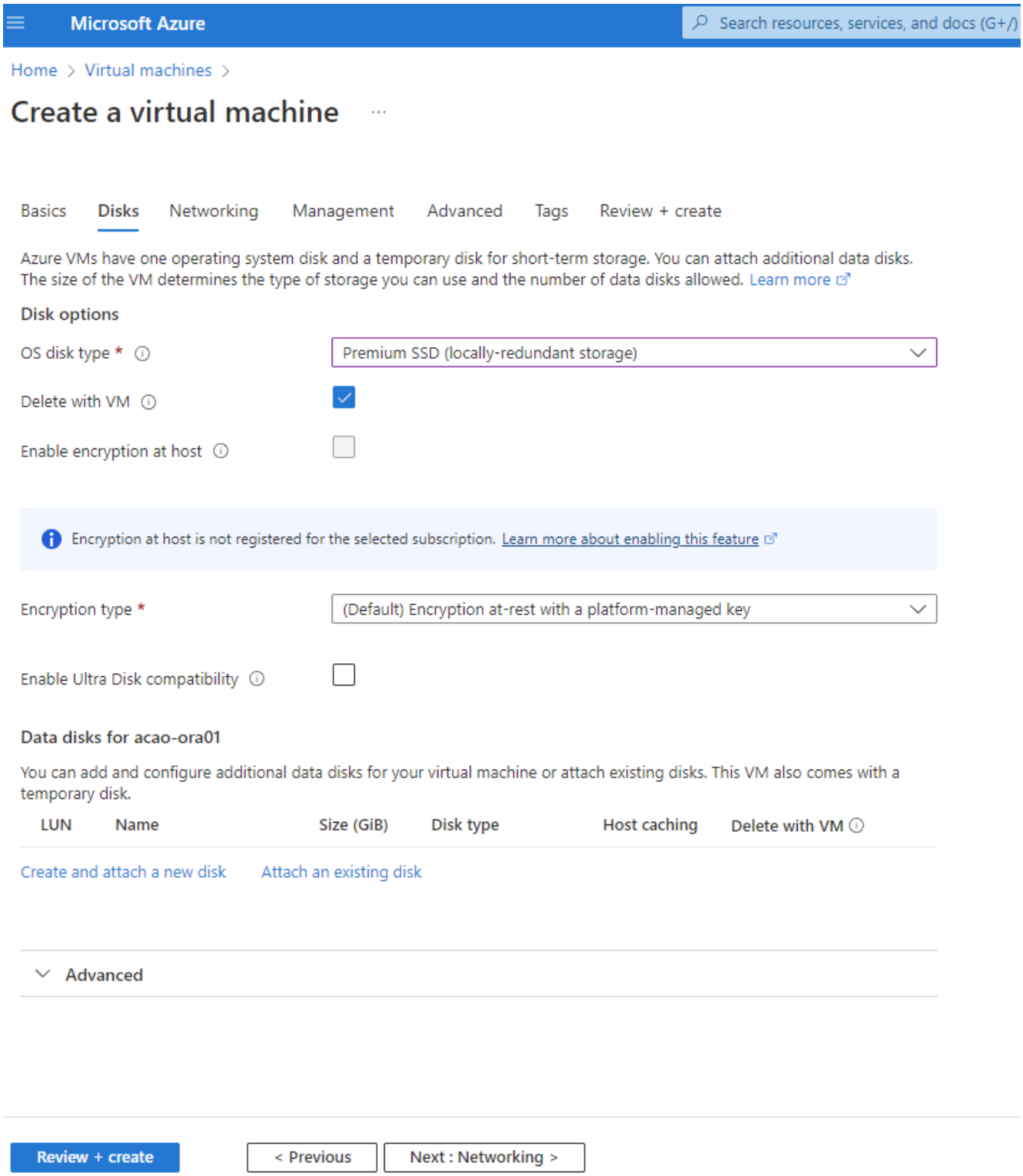


2. 选择部署的订阅ID、然后选择资源组、区域、主机名、VM映像、大小、和身份验证方法。转到磁盘页面。

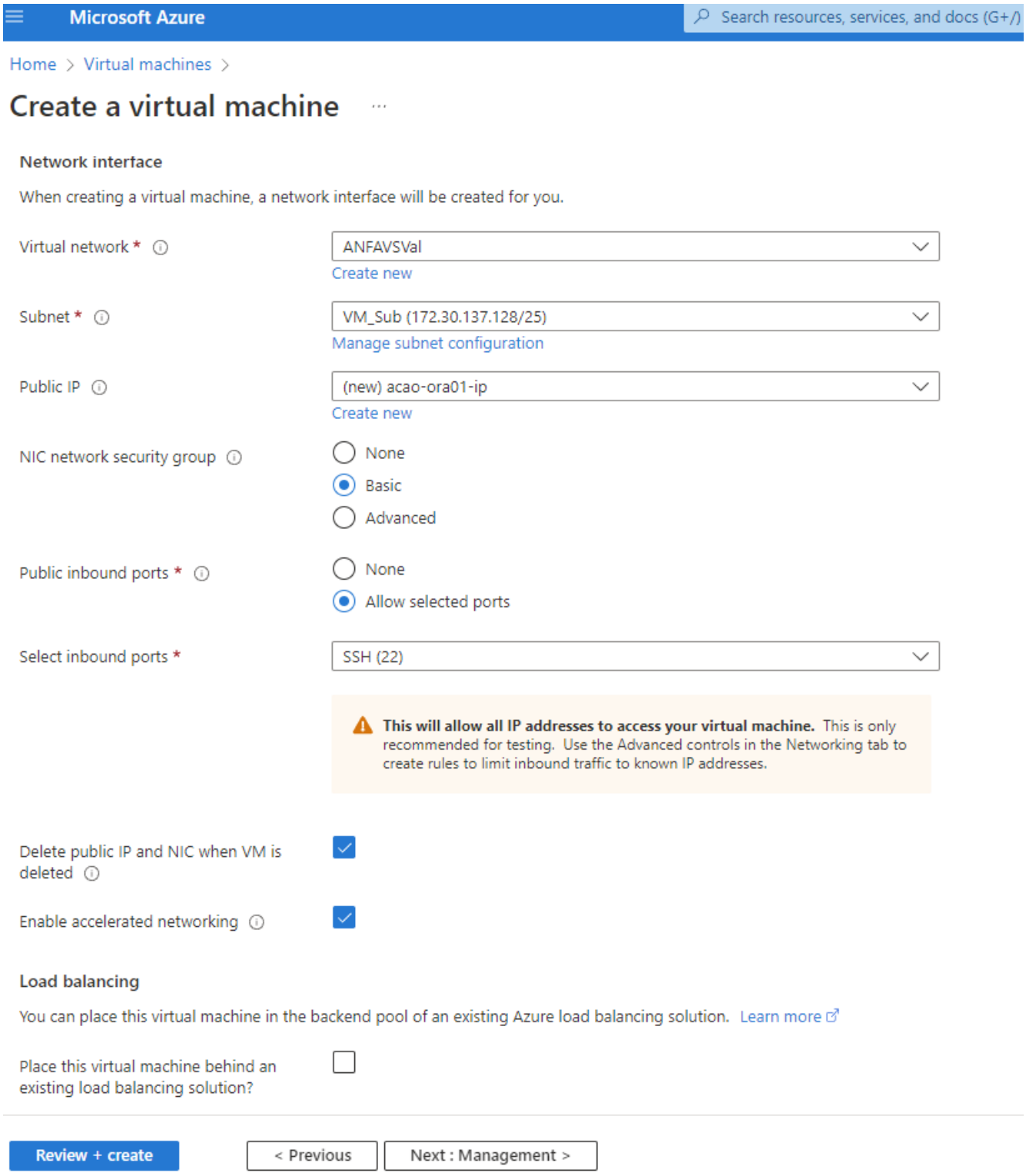

Microsoft Azure
Search resources, services, and docs (G+/)
Home > Virtual machines >
Create a virtual machine
Basics
Disks
Networking
Management
Advanced
Tags
Review + create
Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. Learn more
Project details
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.
Subscription *
Hybrid Cloud TME Onprem
Resource group *
ANFAVSRG
Create new
Instance details
Virtual machine name *
acao-ora01
Region *
(US) South Central US
Availability options
No infrastructure redundancy required
Security type
Standard
Image *
Red Hat Enterprise Linux 8.0 (LVM) - Gen2
See all images | Configure VM generation
Run with Azure Spot discount
Size *
Standard_D8s_v3 - 8 vcpus, 32 GiB memory ($273.02/month)
See all sizes
Administrator account
Authentication type
SSH public key
Password
Review + create
< Previous
Next : Disks >



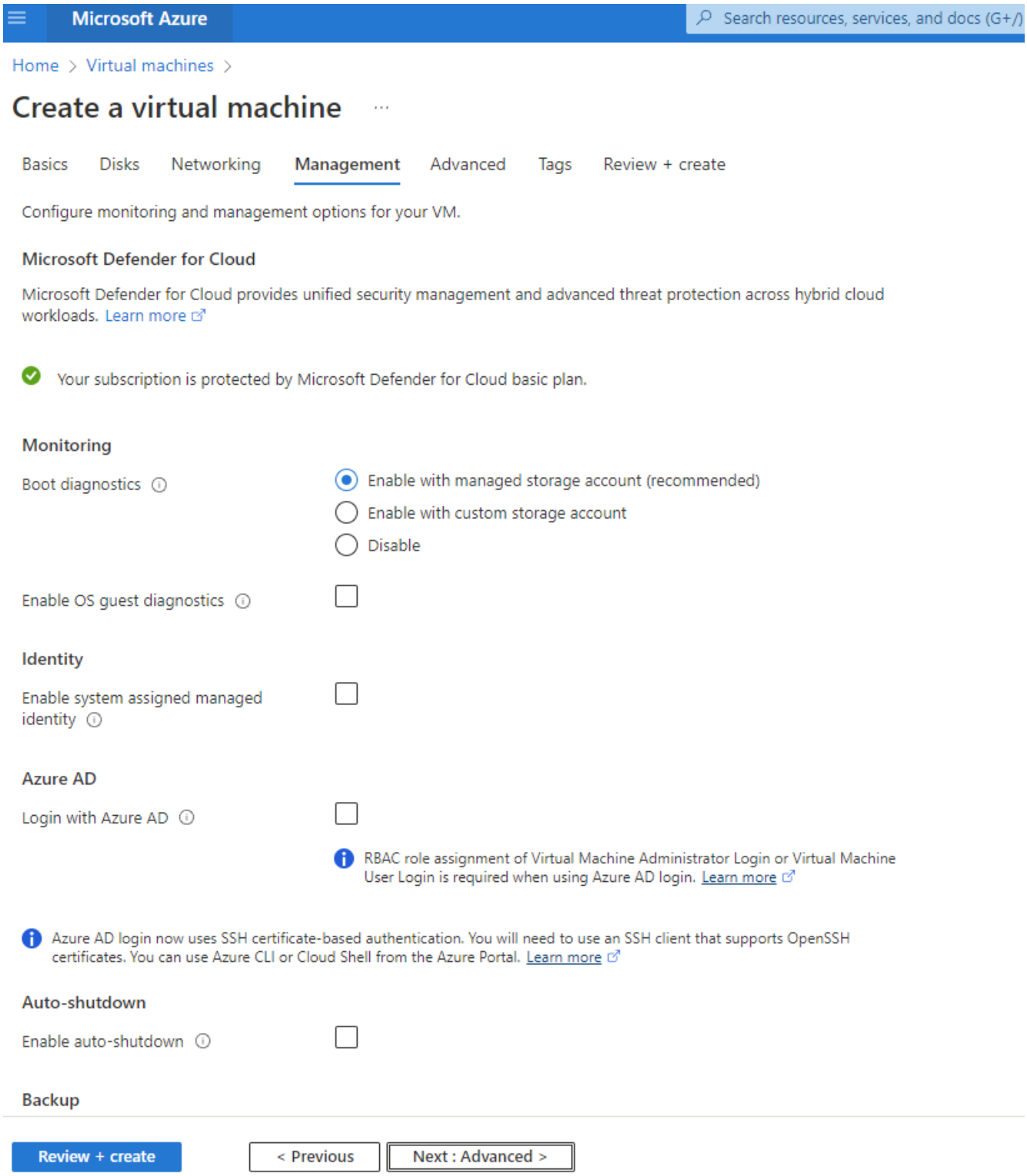
3. 选择*高级SSD*以实现操作系统本地冗余、并将数据磁盘留空、因为数据磁盘是从ANF存储挂载的。转到网络连接页面。



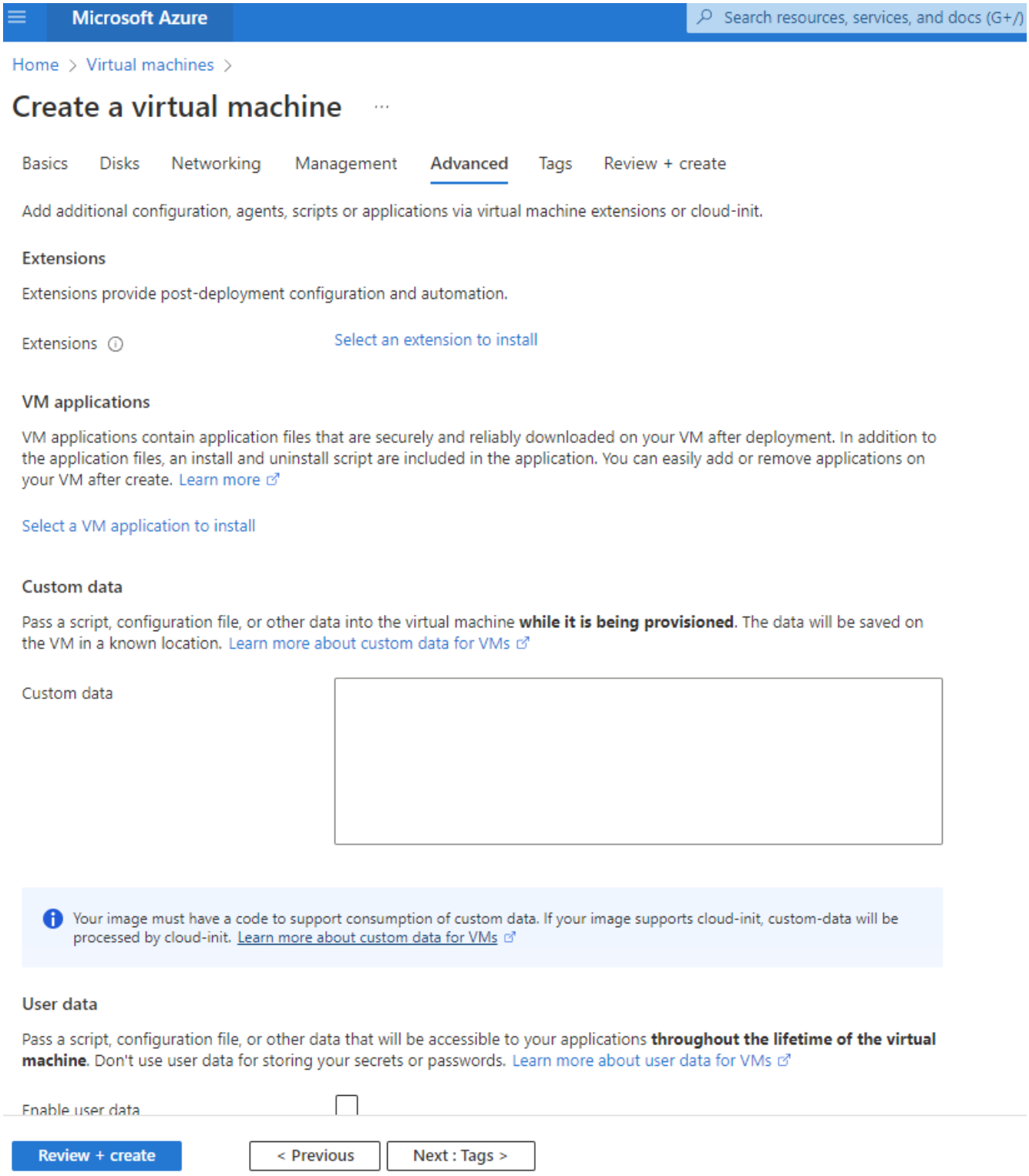
4. 选择vNet和子网。为外部VM访问分配公共IP。然后转到"管理"页面。



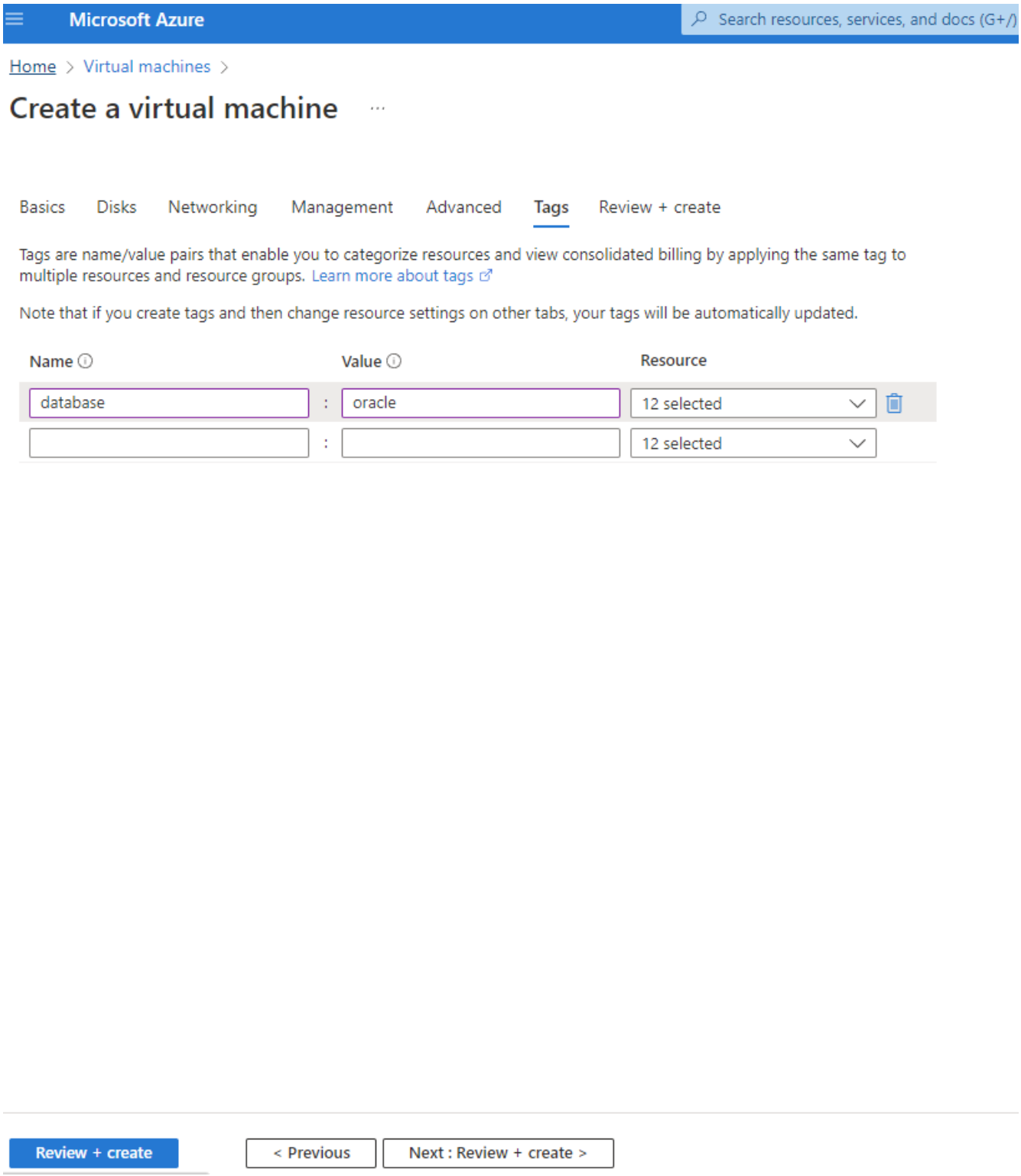
5. 保留Management的所有默认值、然后转到Advanced页面。



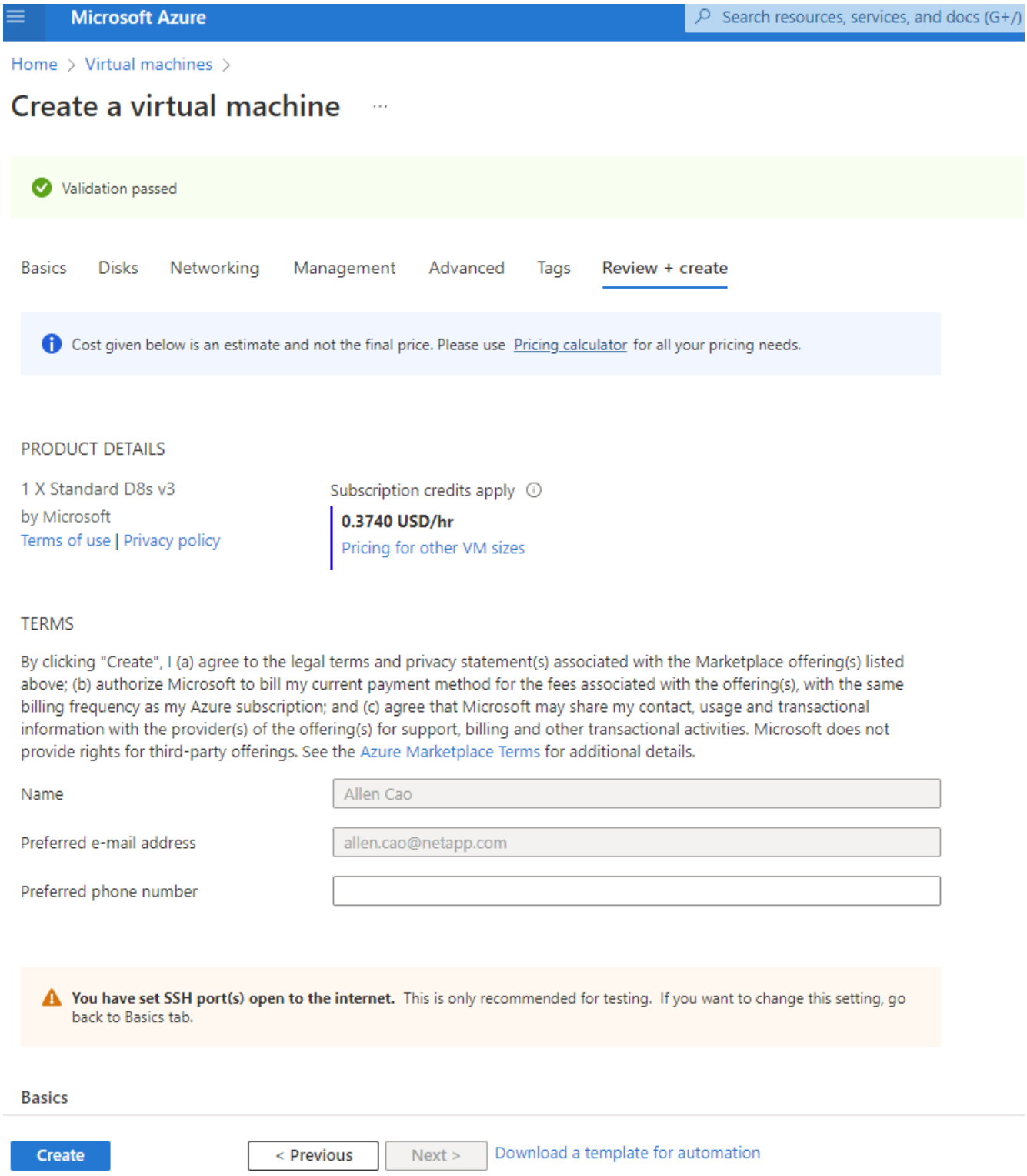
6. 保留"高级"页面的所有默认值、除非您需要在使用自定义脚本部署后自定义VM。然后转到"标记"页面。



7. 如果需要、为虚拟机添加一个标记。然后、转到Review + create页面。



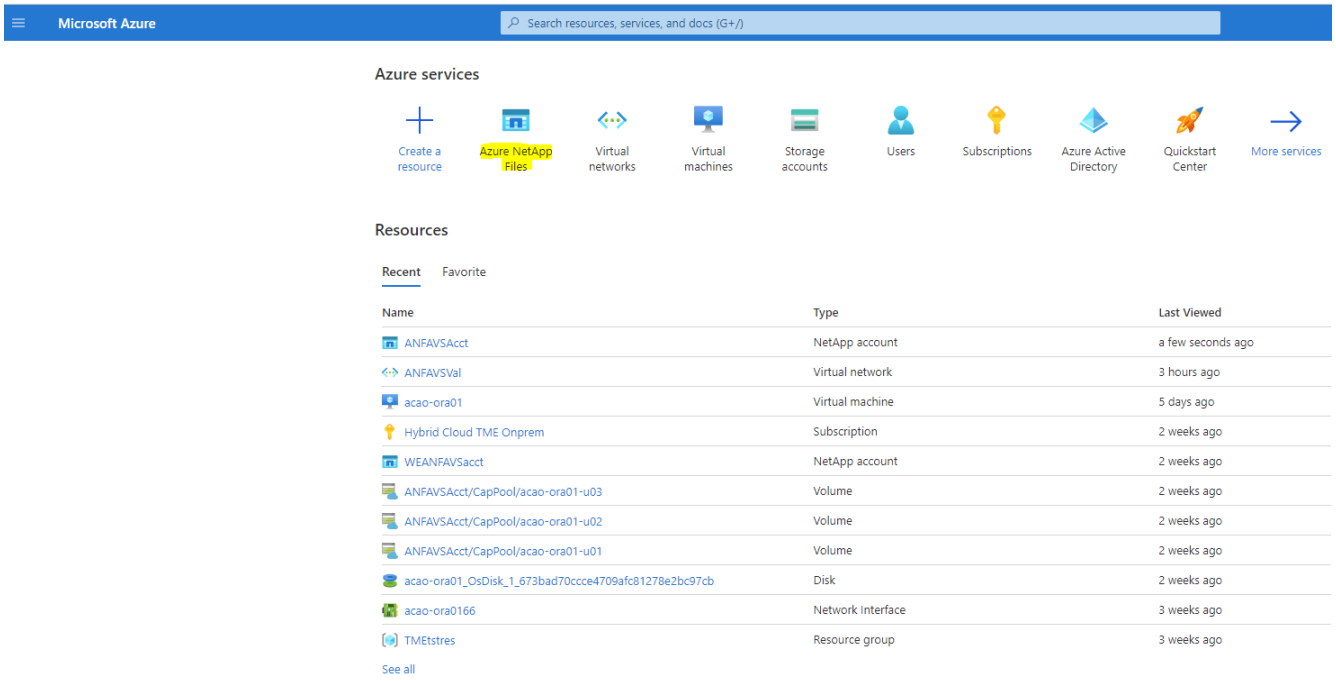
8. 部署工作流将对配置运行验证、如果验证通过、请单击*创建*以创建虚拟机。



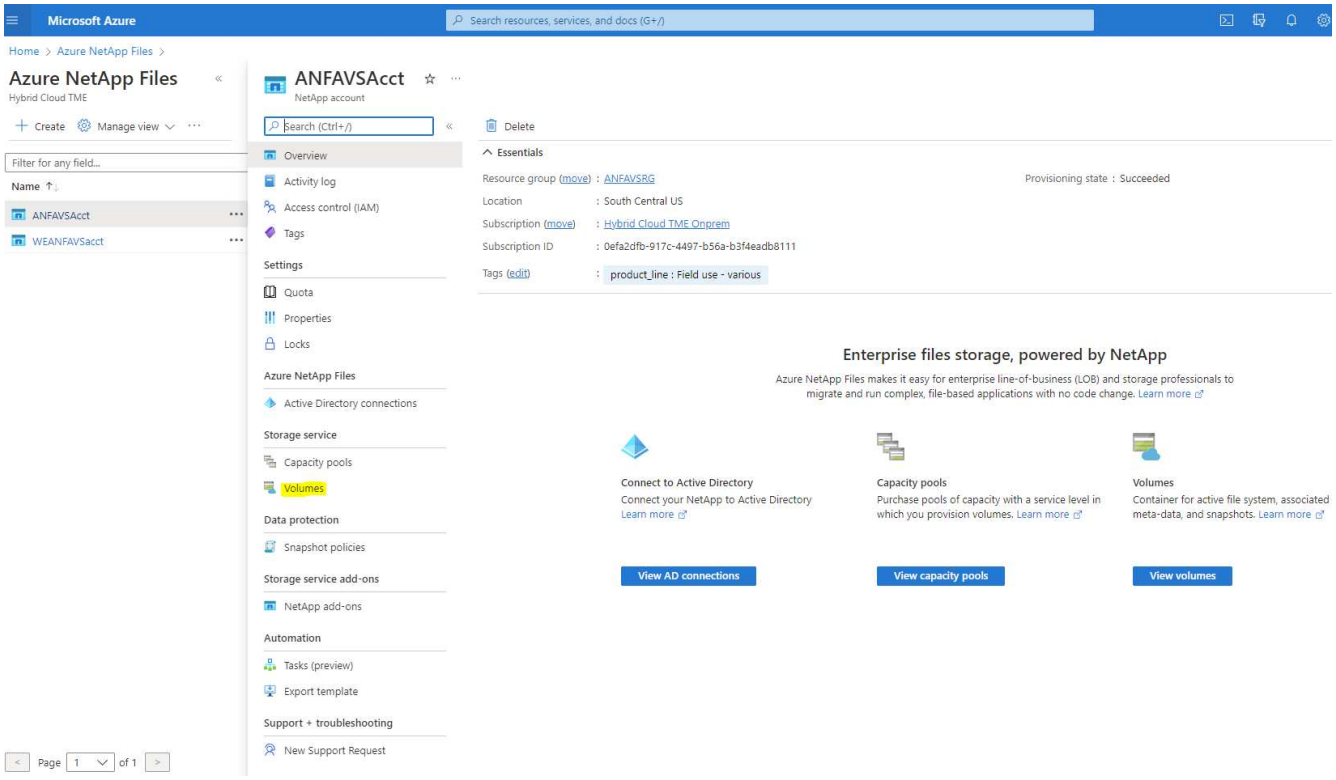
4.为Oracle配置ANF数据库卷

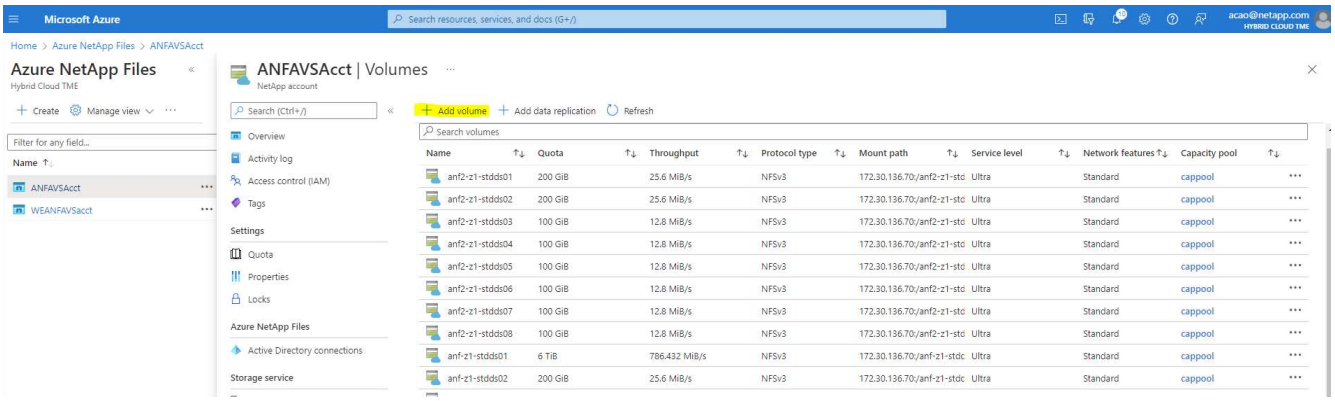
您必须分别为Oracle二进制卷、数据卷和日志卷的ANF容量池创建三个NFS卷。

1. 在Azure控制台中的Azure服务列表下、单击Azure NetApp Files 以打开卷创建工作流。如果您有多个ANF存储帐户、请单击要从中配置卷的帐户。

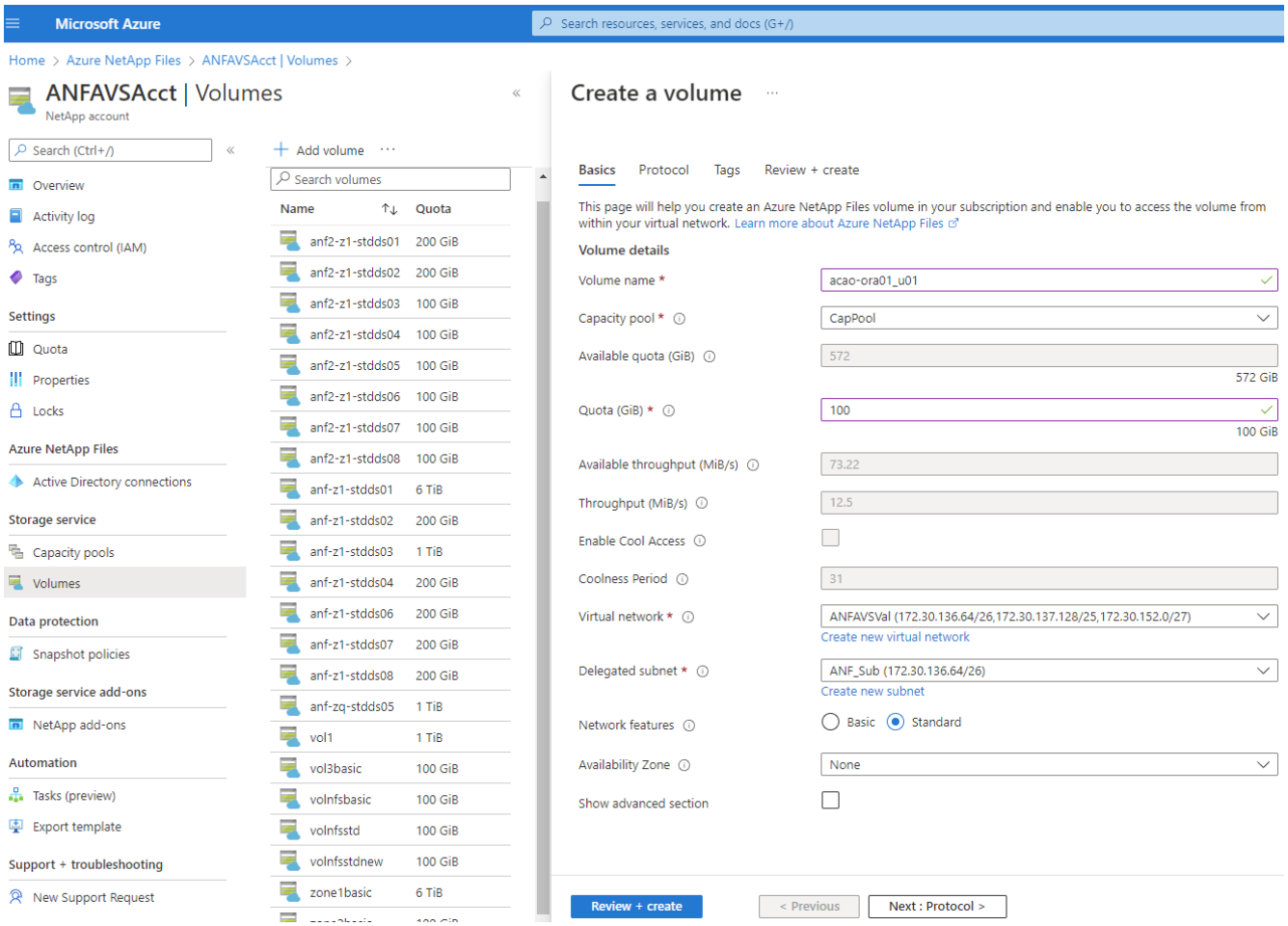


2. 在NetApp存储帐户下、单击*卷*、然后单击*添加卷*以创建新的Oracle卷。

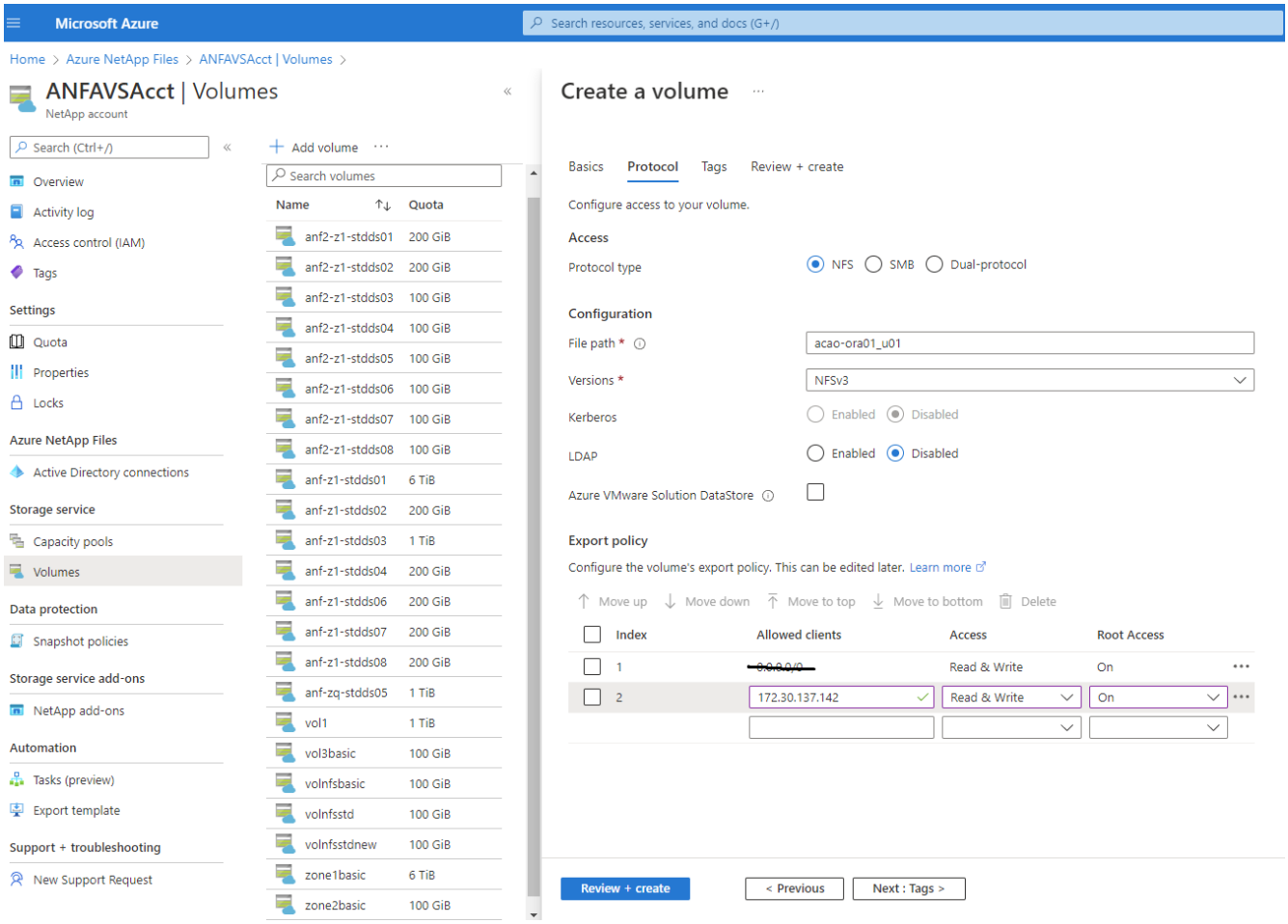




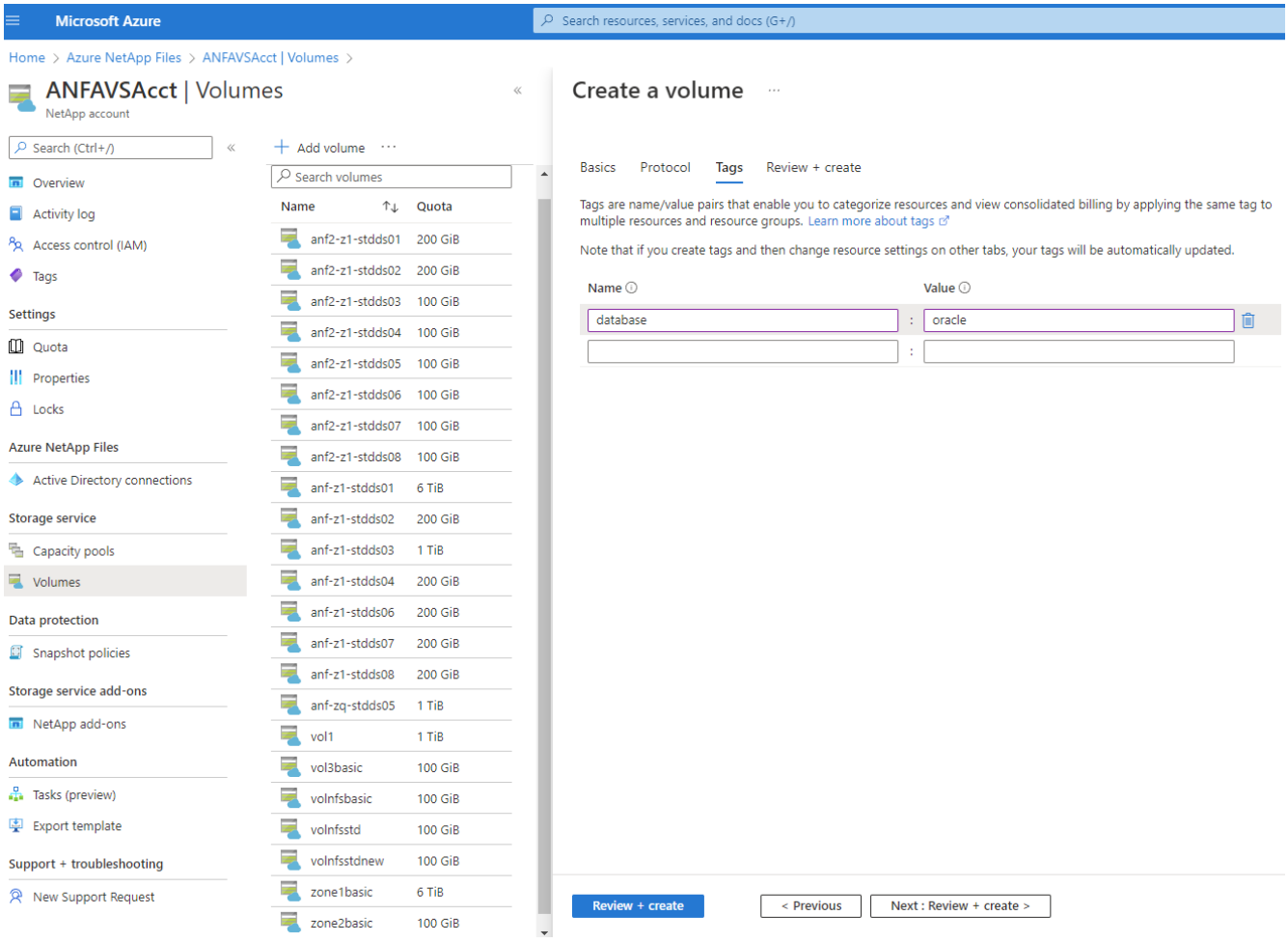
3. 作为一种良好做法、请先确定使用VM主机名作为前缀的Oracle卷、然后再确定主机上的挂载点、例如、对于Oracle二进制文件、可以使用u01表示Oracle数据、可以使用u02表示Oracle数据、也可以使用u03表示Oracle日志。为卷选择与VM相同的vNet。单击*下一步：协议>*。



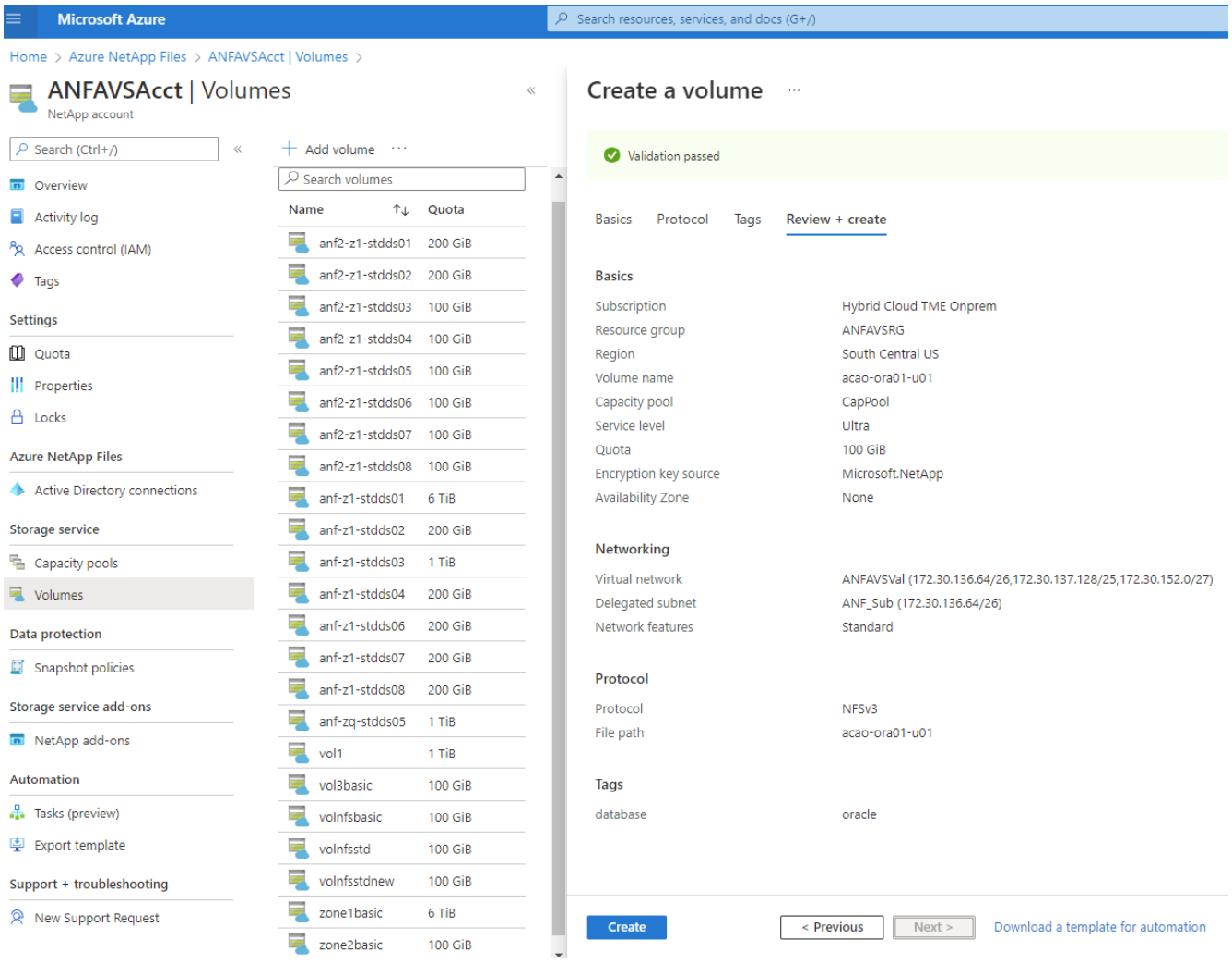
4. 选择NFS协议、将Oracle主机IP地址添加到允许的客户端、然后删除允许所有IP地址0.0.0.0/0的默认策略。然后单击*下一步：标记>*。



5. 根据需要添加卷标记。然后单击*审阅+创建>*。



6. 如果验证通过、请单击*创建*以创建卷。



使用ANF在Azure虚拟机上安装和配置Oracle

NetApp解决方案团队创建了许多基于Ansible的自动化工具包、可帮助您在Azure中顺利部署Oracle。按照以下步骤在Azure虚拟机上部署Oracle。

设置Ansible控制器

如果尚未设置Ansible控制器、请参见 "NetApp 解决方案自动化"、其中详细说明了如何设置Ansible控制器。

获取Oracle部署自动化工具包

在您的主目录中、使用您用于登录到Ansible控制器的用户ID克隆一份Oracle部署工具包副本。

```
git clone https://github.com/NetApp-Automation/na_oracle19c_deploy.git
```

根据您的配置执行工具包

请参见 "CLI 部署 Oracle 19c 数据库" 使用命令行界面执行攻略手册。从Azure控制台而非命令行界面创建数据库卷时、您可以忽略全局VARS文件中变量配置的ONTAP 部分。

该工具包默认部署Oracle 19c和RU 19.8。它可以轻松地适应任何其他修补程序级别、并对默认配置进行少量更改。此外、默认的种子数据库活动日志文件也会部署到数据卷中。如果需要日志卷上的活动日志文件、应在初始部署后重新定位。如有必要、请联系NetApp解决方案 团队以获得帮助。

为**Oracle**的应用程序一致快照设置**AzAcSnap**备份工具

Azure应用程序一致的Snapshot工具(AzAcSnap)是一个命令行工具、可通过处理在创建存储快照之前将第三方数据库置于应用程序一致状态所需的所有流程编排来为第三方数据库提供数据保护。然后、它会将这些数据库返回到运行状态。NetApp建议在数据库服务器主机上安装此工具。请参见以下安装和配置过程。

安装**AzAcSnap**工具

1. 获取的最新版本 "AzArcSnap安装程序"。
2. 将下载的自安装程序复制到目标系统。
3. 使用默认安装选项以root用户身份执行自安装程序。如有必要、请使用使文件可执行 `chmod +x *.run` 命令：

```
./azacsnap_installer_v5.0.run -I
```

配置**Oracle**连接

快照工具与Oracle数据库进行通信、并且需要具有适当权限的数据库用户来启用或禁用备份模式。

1.设置**AzAcSnap**数据库用户

以下示例显示了Oracle数据库用户的设置以及使用sqlplus与Oracle数据库进行通信。示例命令用于在Oracle数据库中设置用户(AZACSNAP)、并根据需要更改IP地址、用户名和密码。

1. 在Oracle数据库安装中、启动sqlplus以登录到数据库。

```
su - oracle
sqlplus / AS SYSDBA
```

2. 创建用户。

```
CREATE USER azacsnap IDENTIFIED BY password;
```

3. 授予用户权限。此示例设置了AZACSNAP用户启用将数据库置于备份模式的权限。

```
GRANT CREATE SESSION TO azacsnap;
GRANT SYSBACKUP TO azacsnap;
```

4. 将默认用户的密码到期时间更改为无限制。

```
ALTER PROFILE default LIMIT PASSWORD_LIFE_TIME unlimited;
```

5. 验证数据库的azacsnap连接。

```
connect azacsnap/password
quit;
```

2.使用Oracle Wallet为数据库访问配置Linux用户azacsnap

AzAcSnap默认安装会创建一个azacsnap操作系统用户。必须使用存储在Oracle Wallet中的密码为其Bash shell环境配置Oracle数据库访问。

1. 以root用户身份运行 `cat /etc/oratab` 用于标识主机上的ORACLE_HOME和ORACLE_SID变量的命令。

```
cat /etc/oratab
```

2. 将ORACLE_HOME、ORACLE_SID、TNS_admin和路径变量添加到azacsnap用户bash配置文件中。根据需要更改变量。

```
echo "export ORACLE_SID=ORATEST" >> /home/azacsnap/.bash_profile
echo "export ORACLE_HOME=/u01/app/oracle/product/19800/ORATST" >>
/home/azacsnap/.bash_profile
echo "export TNS_ADMIN=/home/azacsnap" >> /home/azacsnap/.bash_profile
echo "export PATH=\$PATH:\$ORACLE_HOME/bin" >>
/home/azacsnap/.bash_profile
```

3. 作为Linux用户azacsnap、创建电子钱包。系统将提示您输入电子邮件密码。

```
sudo su - azacsnap

mkstore -wrl $TNS_ADMIN/.oracle_wallet/ -create
```

4. 将连接字符串凭据添加到Oracle Wallet。在以下示例命令中、AZACSNAP是AzAcSnap要使用的ConnectString、azacsnap是Oracle数据库用户、AzPasswd1是Oracle用户的数据库密码。系统会再次提示您输入电子邮件密码。

```
mkstore -wrl $TNS_ADMIN/.oracle_wallet/ -createCredential AZACSNAP
azacsnap AzPasswd1
```

5. 创建 `tnsnames-ora` 文件在以下示例命令中、应将主机设置为Oracle数据库的IP地址、并将服务器SID设置为Oracle数据库SID。

```
echo "# Connection string
AZACSNAP=\"(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=172.30.137.142)(POR
T=1521))(CONNECT_DATA=(SID=ORATST)))\"
" > $TNS_ADMIN/tnsnames.ora
```

6. 创建 `sqlnet.ora` 文件

```
echo "SQLNET.WALLET_OVERRIDE = TRUE
WALLET_LOCATION=(
    SOURCE=(METHOD=FILE)
    (METHOD_DATA=(DIRECTORY=\$TNS_ADMIN/.oracle_wallet))
) " > $TNS_ADMIN/sqlnet.ora
```

7. 使用Wallet测试Oracle访问。

```
sqlplus /@AZACSNAP as SYSBACKUP
```

命令的预期输出：

```
[azacsnap@acao-ora01 ~]$ sqlplus /@AZACSNAP as SYSBACKUP

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Sep 8 18:02:07 2022
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL>
```

配置**ANF**连接

本节介绍如何启用与Azure NetApp Files (与VM)的通信。

1. 在Azure Cloud Shell会话中、确保您已登录到默认要与服务主体关联的订阅。

```
az account show
```

2. 如果订阅不正确、请使用以下命令：

```
az account set -s <subscription name or id>
```

3. 使用Azure命令行界面创建服务主体、如以下示例所示：

```
az ad sp create-for-rbac --name "AzAcSnap" --role Contributor --scopes
/subscriptions/{subscription-id} --sdk-auth
```

预期输出：

```
{
  "clientId": "00aa000a-aaaa-0000-00a0-00aa000aaa0a",
  "clientSecret": "00aa000a-aaaa-0000-00a0-00aa000aaa0a",
  "subscriptionId": "00aa000a-aaaa-0000-00a0-00aa000aaa0a",
  "tenantId": "00aa000a-aaaa-0000-00a0-00aa000aaa0a",
  "activeDirectoryEndpointUrl": "https://login.microsoftonline.com",
  "resourceManagerEndpointUrl": "https://management.azure.com/",
  "activeDirectoryGraphResourceId": "https://graph.windows.net/",
  "sqlManagementEndpointUrl":
"https://management.core.windows.net:8443/",
  "galleryEndpointUrl": "https://gallery.azure.com/",
  "managementEndpointUrl": "https://management.core.windows.net/"
}
```

4. 剪切输出内容并将其粘贴到名为的文件中 `oracle.json` 存储在Linux用户azacsnap用户箱目录中、并使用适当的系统权限保护文件。

请确保JSON文件的格式与上述格式完全相同、尤其是使用双引号(")括起的URL。

完成**AzAcSnap**工具的设置

按照以下步骤配置和测试快照工具。成功测试后、您可以执行第一个数据库一致的存储快照。

1. 更改为Snapshot用户帐户。

```
su - azacsnap
```

2. 更改命令的位置。

```
cd /home/azacsnap/bin/
```

3. 配置存储备份详细信息文件。这将创建 `azacsnap.json` 配置文件。

```
azacsnap -c configure --configuration new
```

三个Oracle卷的预期输出：

```
[azacsnap@acao-ora01 bin]$ azacsnap -c configure --configuration new
Building new config file
Add comment to config file (blank entry to exit adding comments): Oracle
snapshot bkup
Add comment to config file (blank entry to exit adding comments):
Enter the database type to add, 'hana', 'oracle', or 'exit' (for no
database): oracle

=== Add Oracle Database details ===
Oracle Database SID (e.g. CDB1): ORATST
Database Server's Address (hostname or IP address): 172.30.137.142
Oracle connect string (e.g. /@AZACSNAP): /@AZACSNAP

=== Azure NetApp Files Storage details ===
Are you using Azure NetApp Files for the database? (y/n) [n]: y
--- DATA Volumes have the Application put into a consistent state before
they are snapshot ---
Add Azure NetApp Files resource to DATA Volume section of Database
configuration? (y/n) [n]: y
Full Azure NetApp Files Storage Volume Resource ID (e.g.
/subscriptions/.../resourceGroups/.../providers/Microsoft.NetApp/netAppA
ccounts/.../capacityPools/Premium/volumes/...): /subscriptions/0efa2dfb-
917c-4497-b56a-
b3f4eadb8111/resourceGroups/ANFAVSRG/providers/Microsoft.NetApp/netAppAc
counts/ANFAVSAcct/capacityPools/CapPool/volumes/acao-ora01-u01
Service Principal Authentication filename or Azure Key Vault Resource ID
(e.g. auth-file.json or https://...): oracle.json
Add Azure NetApp Files resource to DATA Volume section of Database
configuration? (y/n) [n]: y
Full Azure NetApp Files Storage Volume Resource ID (e.g.
/subscriptions/.../resourceGroups/.../providers/Microsoft.NetApp/netAppA
ccounts/.../capacityPools/Premium/volumes/...): /subscriptions/0efa2dfb-
917c-4497-b56a-
b3f4eadb8111/resourceGroups/ANFAVSRG/providers/Microsoft.NetApp/netAppAc
counts/ANFAVSAcct/capacityPools/CapPool/volumes/acao-ora01-u02
```

```
Service Principal Authentication filename or Azure Key Vault Resource ID
(e.g. auth-file.json or https://...): oracle.json
Add Azure NetApp Files resource to DATA Volume section of Database
configuration? (y/n) [n]: n
--- OTHER Volumes are snapshot immediately without preparing any
application for snapshot ---
Add Azure NetApp Files resource to OTHER Volume section of Database
configuration? (y/n) [n]: y
Full Azure NetApp Files Storage Volume Resource ID (e.g.
/subscriptions/.../resourceGroups/.../providers/Microsoft.NetApp/netAppA
ccounts/.../capacityPools/Premium/volumes/...): /subscriptions/0efa2dfb-
917c-4497-b56a-
b3f4eadb8111/resourceGroups/ANFAVSRG/providers/Microsoft.NetApp/netAppAc
counts/ANFAVSAcct/capacityPools/CapPool/volumes/acao-ora01-u03
Service Principal Authentication filename or Azure Key Vault Resource ID
(e.g. auth-file.json or https://...): oracle.json
Add Azure NetApp Files resource to OTHER Volume section of Database
configuration? (y/n) [n]: n

=== Azure Managed Disk details ===
Are you using Azure Managed Disks for the database? (y/n) [n]: n

=== Azure Large Instance (Bare Metal) Storage details ===
Are you using Azure Large Instance (Bare Metal) for the database? (y/n)
[n]: n

Enter the database type to add, 'hana', 'oracle', or 'exit' (for no
database): exit

Editing configuration complete, writing output to 'azacsnap.json'.
```

4. 以azacsnap Linux用户身份、对Oracle备份运行azacsnap test命令。

```
cd ~/bin
azacsnap -c test --test oracle --configfile azacsnap.json
```

预期输出：

```
[azacsnap@acao-ora01 bin]$ azacsnap -c test --test oracle --configfile
azacsnap.json
BEGIN : Test process started for 'oracle'
BEGIN : Oracle DB tests
PASSED: Successful connectivity to Oracle DB version 1908000000
END   : Test process complete for 'oracle'
[azacsnap@acao-ora01 bin]$
```

5. 运行第一个快照备份。

```
azacsnap -c backup --volume data --prefix ora_test --retention=1
```

在**Azure**云中保护**Oracle**数据库

NetApp解决方案工程部的Allen Cao

本节介绍如何使用azacsnap工具以及快照备份、还原和快照分层到Azure Blob来保护Oracle数据库。

使用**AzAcSnap**工具使用**Snapshot**备份**Oracle**数据库

Azure应用程序一致的Snapshot工具(AzAcSnap)是一个命令行工具、可通过处理在创建存储快照之前将第三方数据库置于应用程序一致状态所需的所有流程编排来为其提供数据保护、之后、它会将数据库恢复为运行状态。

对于Oracle、您可以将数据库置于备份模式以创建快照、然后将数据库退出备份模式。

备份数据和日志卷

可以使用执行snapshot命令的简单shell脚本在数据库服务器主机上设置备份。然后、可以计划从crontab运行此脚本。

通常、备份频率取决于所需的RTO和RPO。频繁创建快照会占用更多存储空间。备份频率与空间占用之间存在一定的权衡。

数据卷通常比日志卷占用更多的存储空间。因此、您可以每几小时在数据卷上创建一次快照、而每15到30分钟在日志卷上创建一次更频繁的快照。

请参见以下备份脚本和计划示例。

对于数据卷快照：

```
 # /bin/sh
cd /home/azacsnap/bin
. ~/.bash_profile
azacsnap -c backup --volume data --prefix acao-ora01-data --retention 36
azacsnap -c backup --volume other --prefix acao-ora01-log --retention 250
```

对于日志卷快照：

```
 # /bin/sh
cd /home/azacsnap/bin
. ~/.bash_profile
azacsnap -c backup --volume other --prefix acao-ora01-log --retention 250
```

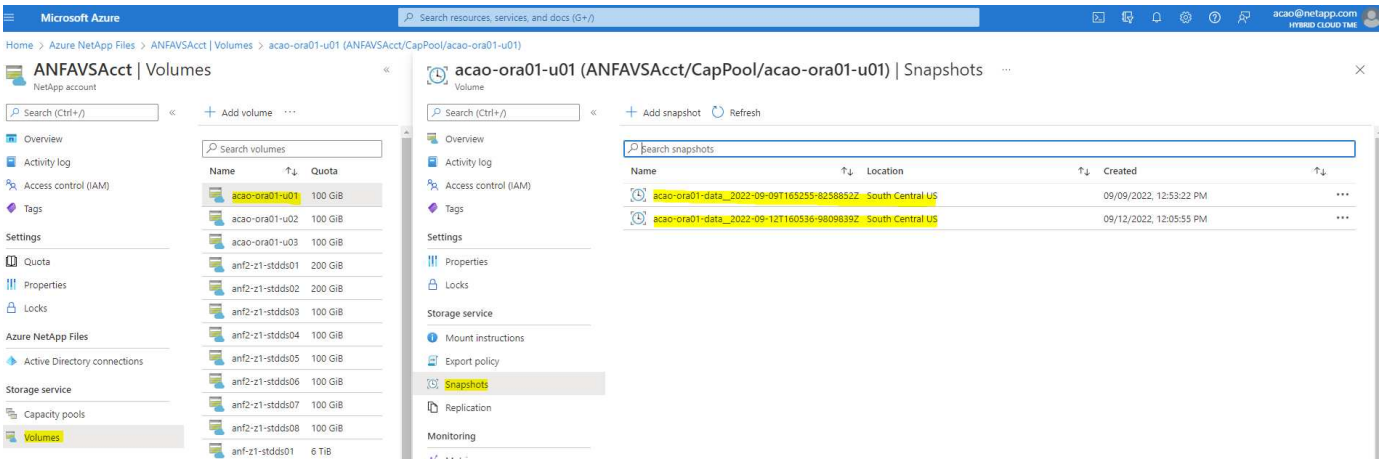
crontab计划：

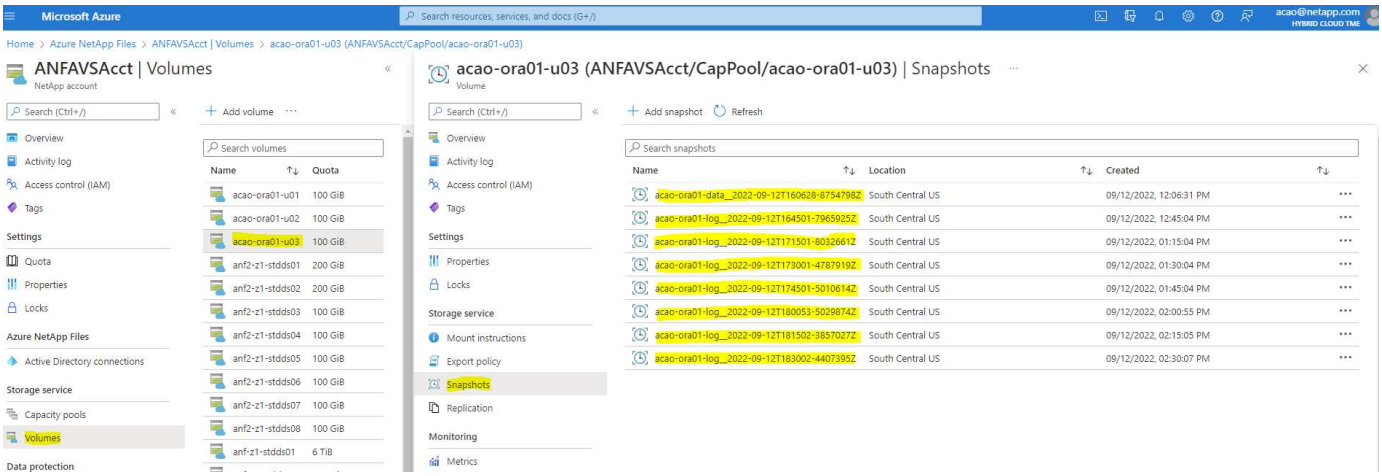
```
15,30,45 * * * * /home/azacsnap/snap_log.sh
0 */2 * * * /home/azacsnap/snap_data.sh
```

设置备份时 `azacsnap.json` 配置文件中、将所有数据卷(包括二进制卷)添加到 `dataVolume` 以及所有日志卷 `otherVolume`。快照的最大保留空间为250个副本。

验证快照

转至Azure门户> Azure NetApp文件/卷以检查是否已成功创建快照。





Oracle从本地备份还原和恢复

Snapshot备份的一个主要优势是、它与源数据库卷共存、并且主数据库卷几乎可以即时回滚。

在主服务器上还原和恢复**Oracle**

以下示例演示了如何从同一Oracle主机上的Azure信息板和CLI还原和恢复Oracle数据库。

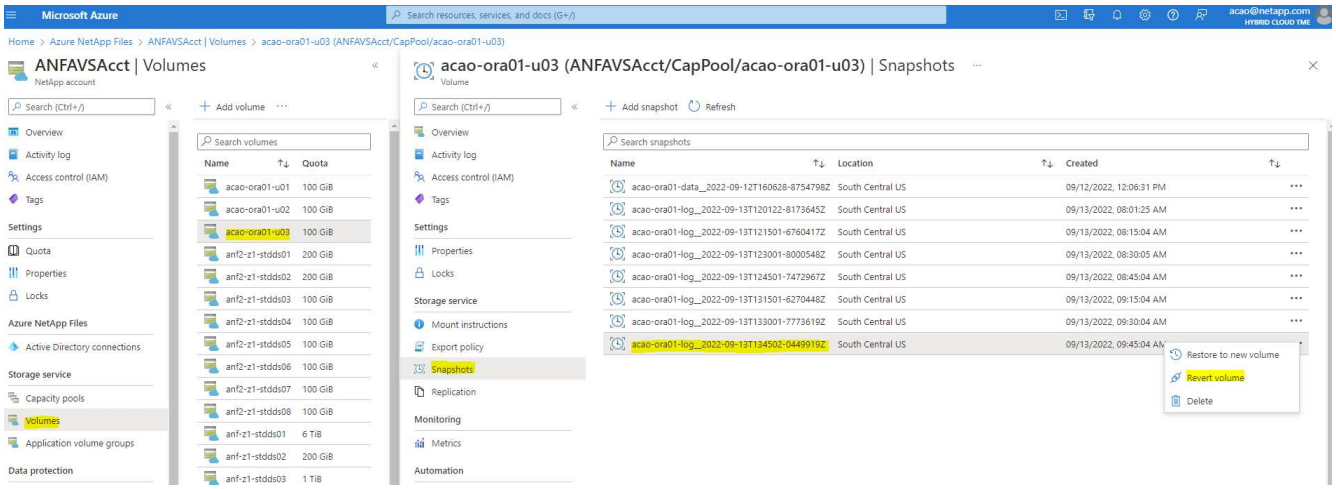
1. 在要还原的数据库中创建一个测试表。

```
[oracle@acao-ora01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Mon Sep 12 19:02:35 2022
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL> create table testsnapshot(
     id integer,
     event varchar(100),
     dt timestamp);

Table created.

SQL> insert into testsnapshot values(1,'insert a data marker to validate
snapshot restore',sysdate);

1 row created.

SQL> commit;

Commit complete.

SQL> select * from testsnapshot;

 ID
----------
EVENT
--------------------------------------------------------------------------
--------
DT
--------------------------------------------------------------------------
---
         1
insert a data marker to validate snapshot restore
12-SEP-22 07.07.35.000000 PM
```

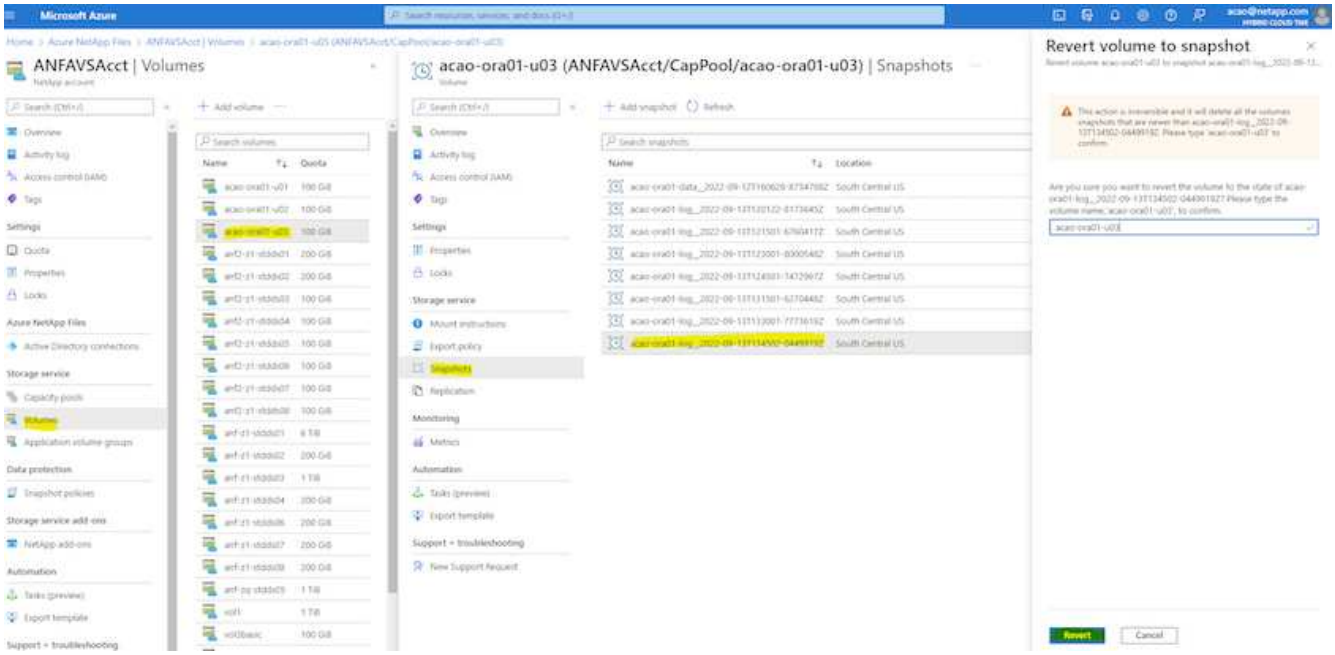
2. 将此表放到快照备份之后。

```
[oracle@acao-ora01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Tue Sep 13 14:20:22 2022
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL> drop table testsnapshot;

Table dropped.

SQL> select * from testsnapshot;
select * from testsnapshot
              *
ERROR at line 1:
ORA-00942: table or view does not exist

SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release
19.0.0.0.0 - Production
Version 19.8.0.0.0
```

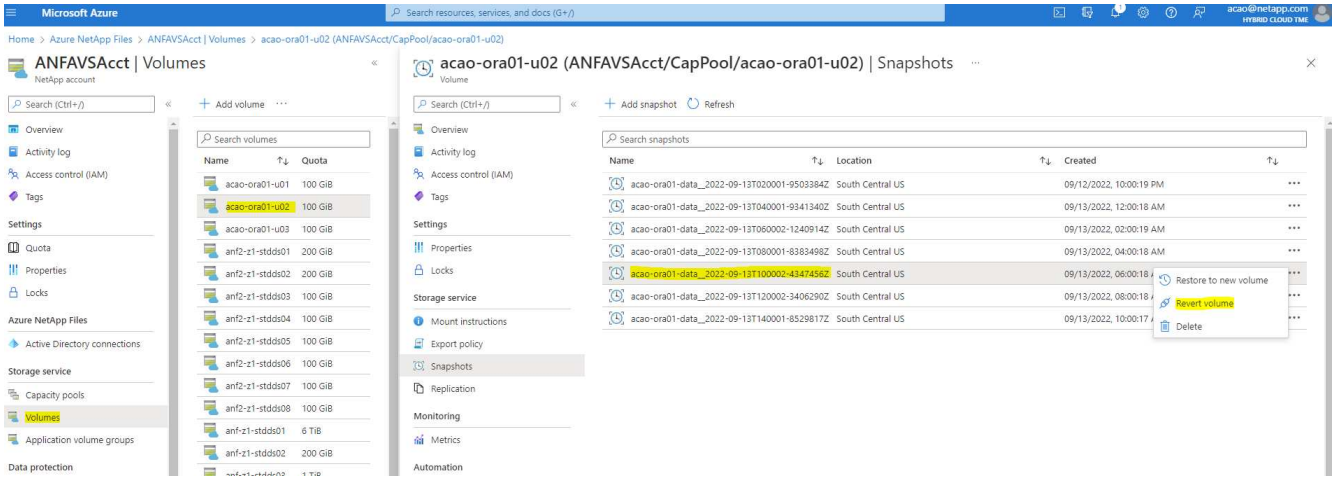
3. 从Azure NetApp Files 信息板中、将日志卷还原到最后一个可用快照。选择*还原卷*。



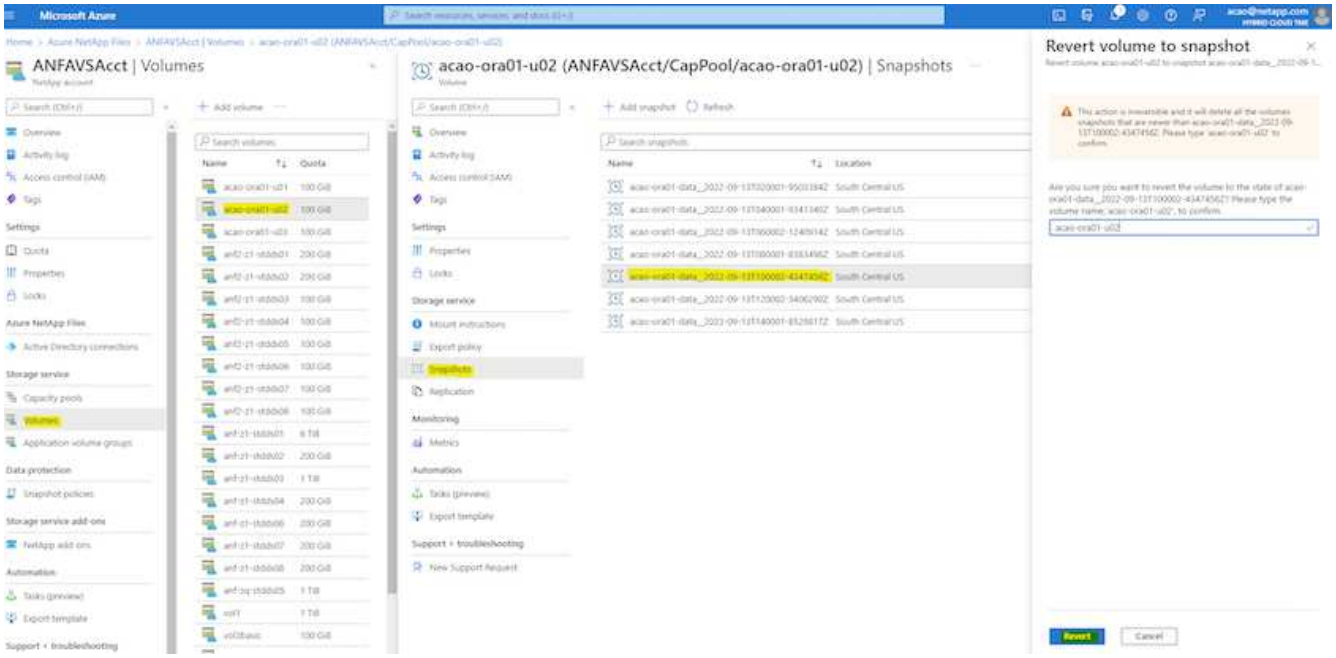
4. 确认还原卷并单击*还原*以完成卷还原到最新可用备份的过程。



5. 对数据卷重复相同的步骤、并确保备份包含要恢复的表。



6. 再次确认卷还原、然后单击"还原"。



7. 如果您有多个控制文件副本、请重新同步这些控制文件、并将旧控制文件替换为可用的最新副本。

```
[oracle@acao-ora01 ~]$ mv /u02/oradata/ORATST/control01.ctl
/u02/oradata/ORATST/control01.ctl.bk
[oracle@acao-ora01 ~]$ cp /u03/orareco/ORATST/control02.ctl
/u02/oradata/ORATST/control01.ctl
```

8. 登录到Oracle服务器VM并使用sqlplus运行数据库恢复。

```
[oracle@acao-ora01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Tue Sep 13 15:10:17 2022
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.

Connected to an idle instance.

SQL> startup mount;
ORACLE instance started.

Total System Global Area 6442448984 bytes
Fixed Size                  8910936 bytes
Variable Size            1090519040 bytes
Database Buffers         5335154688 bytes
Redo Buffers                7864320 bytes
Database mounted.
SQL> recover database using backup controlfile until cancel;
```

```
ORA-00279: change 3188523 generated at 09/13/2022 10:00:09 needed for
thread 1
ORA-00289: suggestion :
/u03/orareco/ORATST/archivelog/2022_09_13/o1_mf_1_43__22rnjq9q_.arc
ORA-00280: change 3188523 for thread 1 is in sequence #43

Specify log: {<RET>=suggested | filename | AUTO | CANCEL}

ORA-00279: change 3188862 generated at 09/13/2022 10:01:20 needed for
thread 1
ORA-00289: suggestion :
/u03/orareco/ORATST/archivelog/2022_09_13/o1_mf_1_44__29f2lgb5_.arc
ORA-00280: change 3188862 for thread 1 is in sequence #44
ORA-00278: log file
'/u03/orareco/ORATST/archivelog/2022_09_13/o1_mf_1_43__22rnjq9q_.arc' no
longer
needed for this recovery

Specify log: {<RET>=suggested | filename | AUTO | CANCEL}

ORA-00279: change 3193117 generated at 09/13/2022 12:00:08 needed for
thread 1
ORA-00289: suggestion :
/u03/orareco/ORATST/archivelog/2022_09_13/o1_mf_1_45__29h6qqyw_.arc
ORA-00280: change 3193117 for thread 1 is in sequence #45
ORA-00278: log file
'/u03/orareco/ORATST/archivelog/2022_09_13/o1_mf_1_44__29f2lgb5_.arc' no
longer
needed for this recovery

Specify log: {<RET>=suggested | filename | AUTO | CANCEL}

ORA-00279: change 3193440 generated at 09/13/2022 12:01:20 needed for
thread 1
ORA-00289: suggestion :
/u03/orareco/ORATST/archivelog/2022_09_13/o1_mf_1_46_%u_.arc
ORA-00280: change 3193440 for thread 1 is in sequence #46
ORA-00278: log file
'/u03/orareco/ORATST/archivelog/2022_09_13/o1_mf_1_45__29h6qqyw_.arc' no
longer
needed for this recovery

Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
cancel
Media recovery cancelled.
SQL> alter database open resetlogs;
```

```
Database altered.

SQL> select * from testsnapshot;

  ID
----------
EVENT
--------------------------------------------------------------------------------
--------
DT
--------------------------------------------------------------------------------
---
         1
insert a data marker to validate snapshot restore
12-SEP-22 07.07.35.000000 PM

SQL> select systimestamp from dual;

 SYSTIMESTAMP
--------------------------------------------------------------------------------
---
13-SEP-22 03.28.52.646977 PM +00:00
```

此屏幕显示已删除的表已使用本地快照备份进行恢复。

将数据库从内部迁移到**Azure**云

由于Oracle决定逐步淘汰单实例数据库、许多组织已将单实例Oracle数据库转换为多租户容器数据库。这样、可以使用最大可用性选项将一组称为PDB的容器数据库轻松地重新定位到云、从而最大限度地减少迁移期间的停机时间。

但是、如果您仍有一个Oracle数据库实例、则可以先将其转换为多租户容器数据库、然后再尝试PDB重新定位。

以下各节详细介绍了在这两种情况下将内部Oracle数据库迁移到Azure云的相关信息。

将单个实例非**CDB**转换为多租户**CDB**中的**PDB**

如果您仍有单实例Oracle数据库、则无论是否要将其迁移到云、都必须将其转换为多租户容器数据库、因为Oracle不久将停止支持单实例数据库。

以下过程会将单个实例数据库作为可插拔数据库或PDB插入容器数据库。

1. 在与单实例数据库相同的主机上单独构建Shell容器数据库 `ORACLE_HOME`。
2. 关闭单个实例数据库并在只读模式下重新启动它。
3. 运行 `DBMS_PDB.DESCRIBE` 用于生成数据库元数据的操作步骤。

```
BEGIN
  DBMS_PDB.DESCRIBE(
    pdb_descr_file => '/home/oracle/ncdb.xml');
END;
/
```

4. 关闭单实例数据库。
5. 启动容器数据库。
6. 运行 `DBMS_PDB.CHECK_PLUG_COMPATIBILITY` 用于确定非CDB是否与CDB兼容的函数。

```
SET SERVEROUTPUT ON
DECLARE
  compatible CONSTANT VARCHAR2(3) :=
    CASE DBMS_PDB.CHECK_PLUG_COMPATIBILITY(
           pdb_descr_file => '/disk1/oracle/ncdb.xml',
           pdb_name       => 'NCDB')
    WHEN TRUE THEN 'YES'
    ELSE 'NO'
END;
BEGIN
  DBMS_OUTPUT.PUT_LINE(compatible);
END;
/
```

如果输出为Yes、则表示非CDB兼容、您可以继续执行下一步。

如果输出为no、则表示非CDB不兼容、您可以检查 `PDB_PLUG_IN_VIOLATIONS` 查看其不兼容的原因。必须先更正所有违规、然后才能继续。例如、任何版本或修补程序不匹配的问题都应通过运行升级或opatch实用程序来解决。更正违规后、运行 `DBMS_PDB.CHECK_PLUG_COMPATIBILITY` 再次确认、以确保非CDB与CDB兼容。

7. 插入单个实例非CDB。

```
CREATE PLUGGABLE DATABASE ncdb USING '/home/oracle/ncdb.xml'
  COPY
  FILE_NAME_CONVERT = ('/disk1/oracle/dbs/', '/disk2/oracle/ncdb/')
;
```

如果主机上没有足够的空间、则 `NOCOPY` 选项可用于创建PDB。在这种情况下、单实例非CDB在作为PDB插入后不可用、因为PDB已使用原始数据文件。请确保在转换之前创建备份、以便在发生任何错误时可以恢复运行。

8. 如果源单实例非CDB与目标CDB之间的版本不同、则在转换后开始PDB升级。对于相同版本的转换、可以跳

过此步骤。

```
sqlplus / as sysdba;
alter session set container=ncdb
alter pluggable database open upgrade;
exit;
dbupgrade -c ncdb -l /home/oracle
```

查看中的升级日志文件 `/home/oracle` 目录。

9. 打开可插拔数据库、检查PDB插件违规、然后重新编译无效对象。

```
alter pluggable database ncdb open;
alter session set container=ncdb;
select message from pdb_plug_in_violations where type like '%ERR%' and
status <> 'RESOLVED';
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -n 1 -c
'ncdb' -e -b utlrp -d $ORACLE_HOME/rdbms/admin utlrp.sql
```

10. 执行 `noncdb_to_pdb.sql` 更新数据词典。

```
sqlplus / as sysdba
alter session set container=ncdb;
@$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql;
```

关闭并重新启动容器数据库。ndb已退出受限模式。

通过**PDB**重新定位将内部**Oracle**数据库迁移到**Azure**

使用最大可用性选项的Oracle PDB重新定位采用PDB热克隆技术、这样、在PDB复制到目标时、可以使用源PDB。切换时、用户连接会自动重定向到目标PDB。因此、无论PDB的大小如何、都可以最大限度地减少停机时间。NetApp提供了一个基于Ansible的工具包、用于自动执行迁移操作步骤。

1. 在Azure公共云中使用相同版本和修补程序级别在Azure虚拟机上创建CDB。
2. 从Ansible控制器克隆自动化工具包的副本。

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

3. 阅读README文件中的说明。
4. 为源和目标Oracle服务器配置Ansible主机变量文件、并为数据库服务器主机的配置文件配置名称解析。
5. 在Ansible控制器上安装Ansible控制器前提条件。

```
ansible-playbook -i hosts requirements.yml
ansible-galaxy collection install -r collections/requirements.yml
--force
```

6. 对内部服务器执行任何迁移前任务。

```
ansible-playbook -i hosts ora_pdb_relocate.yml -u admin -k -K -t
ora_pdb_relo_onprem
```

admin用户是内部Oracle服务器主机上具有sudo权限的管理用户。管理员用户使用密码进行身份验证。

7. 执行从内部环境到目标Azure Oracle主机的Oracle PDB重新定位。

```
ansible-playbook -i hosts ora_pdb_relocate.yml -u azureuser --private
-key db1.pem -t ora_pdb_relo_primary
```

Ansible控制器可以位于内部或Azure云中。控制器需要连接到内部Oracle服务器主机和Azure Oracle VM主机。内部Oracle服务器主机和Azure Oracle VM主机之间的Oracle数据库端口(如1521)处于打开状态。

其他**Oracle**数据库迁移选项

有关其他迁移选项、请参见Microsoft文档： "Oracle数据库迁移决策过程"。

NVA-1155：《基于FlexPod 数据中心且采用Cisco UCS和基于FC的NetApp AFF A800的Oracle 19c RAC数据库—设计和部署指南》

NetApp公司Allen Cao

本《基于Cisco UCS的FlexPod Datacenter上的Oracle 19c RAC数据库设计和部署指南》以及《基于FC的NetApp AFF A800》详细介绍了在采用Oracle Linux 8.2的最新FlexPod Datacenter基础架构上托管Oracle RAC数据库的解决方案 设计以及分步部署过程操作系统和Red Hat兼容内核。

"NVA-1155：FlexPod Datacenter上的Oracle 19c RAC数据库、采用Cisco UCS和基于FC的NetApp AFF A800"

TR-4250：SAP与基于UNIX的Oracle和NFS以及适用于SAP 3.4的NetApp集群模式Data ONTAP 和SnapManager

NetApp 公司 Nil Bauser

TR-4250解决了设计存储解决方案以使用Oracle数据库支持SAP业务套件产品所面临的挑战。本文档的主要重点是使用最新一代 SAP 解决方案的业务和 IT 主管面临的常见存储基础架构设计，部署，操作和管理挑战。本文档中的建议是通用的；它们并不特定于 SAP 应用程序或 SAP 实施的规模和范围。TR-4250假定读者已基本了解NetApp和SAP产品的技术和操作。TR-4250是在NetApp、SAP、Oracle和我们的客户的技术人员互动的基础上开发的。

"TR-4250：SAP与基于UNIX的Oracle和NFS以及适用于SAP 3.4的NetApp集群模式Data ONTAP和SnapManager"

部署 Oracle 数据库

解决方案概述

在 NFS 上自动部署适用于 ONTAP 的 Oracle19c

企业正在对其环境进行自动化，以提高效率，加快部署速度并减少手动操作。Ansible 等配置管理工具正在用于简化企业数据库操作。在此解决方案中，我们将演示如何使用 Ansible 通过 NetApp ONTAP 自动配置和配置 Oracle 19c 。通过使存储管理员，系统管理员和 DBA 能够一致快速地部署新存储，配置数据库服务器并安装 Oracle 19c 软件，您可以获得以下优势：

- 消除设计复杂性和人为错误，并实施可重复的一致部署和最佳实践
- 缩短配置存储，配置数据库主机和安装 Oracle 的时间
- 提高数据库管理员，系统和存储管理员的工作效率
- 轻松扩展存储和数据库

NetApp 为客户提供经过验证的 Ansible 模块和角色，以加快 Oracle 数据库环境的部署，配置和生命周期管理。此解决方案提供了相关说明和 Ansible 攻略手册代码，可帮助您：

- 为 Oracle 数据库创建和配置 ONTAP NFS 存储
- 在 RedHat Enterprise Linux 7/8 或 Oracle Linux 7/8 上安装 Oracle 19c
- 在 ONTAP NFS 存储上配置 Oracle 19c

有关更多详细信息或要开始，请参见下面的概述视频。

AWX/ 塔式部署

第 1 部分：入门，要求，自动化详细信息和初始 AWX/ 塔式配置

AWX部署

第 2 部分：变量和运行攻略手册

AWX操作手册运行

CLI 部署

第 1 部分：入门，要求，自动化详细信息和可变控制主机设置

CLI 部署

第 2 部分：变量和运行攻略手册

命令行界面操作手册运行

入门

此解决方案可以在 AWX/Tower 环境中运行，也可以通过命令行界面在 Ansible 控制主机上运行。

AWX/ 塔式

对于 AWX/Tower 环境，系统将引导您创建 ONTAP 集群管理和 Oracle 服务器（ IP 和主机名）清单，创建凭据，配置从 NetApp Automation Github 提取 Ansible 代码的项目以及启动自动化的作业模板。

1. 填写特定于您的环境的变量，然后将其复制并粘贴到作业模板中的额外 VAR 字段中。
2. 将额外的变量添加到作业模板后，您可以启动自动化。
3. 通过为 ontap_config ， linux_config 和 oracle_config 指定标记，作业模板将分三个阶段运行。

通过 Ansible 控制主机执行 CLI

1. 配置 Linux 主机，以便用作 Ansible 控制主机"单击此处了解详细说明"
2. 配置 Ansible 控制主机后，您可以克隆 Ansible Automation 存储库。
3. 使用 ONTAP 集群管理的 IP 和 / 或主机名以及 Oracle 服务器的管理 IP 编辑 hosts 文件。
4. 填写特定于您的环境的变量，然后将其复制并粘贴到 `vars.yml` 文件中。
5. 每个 Oracle 主机都有一个可变文件，该文件由其主机名标识，其中包含主机专用变量。
6. 完成所有可变文件后，您可以通过为 `ontap_config` ， `linux_config` 和 `oracle_config` 指定标记来分三个阶段运行此攻略手册。

要求

environment	要求
* 可逆环境 *	AWX/Tower 或 Linux 主机作为 Ansible 控制主机
	Ansible v.10 及更高版本
	Python 3.
	Python 库— netapp-lib — xmltodict — jMespath
* ONTAP *	ONTAP 版本 9.3 - 9.7
	两个数据聚合
	已创建 NFS VLAN 和 ifgrp
* Oracle 服务器 *	RHEL 7/8
	Oracle Linux 7/8
	NFS ，公有和可选管理的网络接口
	Oracle 服务器上的 Oracle 安装文件

自动化详细信息

此自动化部署采用一本 Ansible 攻略手册设计，该攻略手册包含三个不同的角色。这些角色适用于 ONTAP ，Linux 和 Oracle 配置。下表介绍了正在自动执行的任务。

Role	任务
* ontap_config*	预检查 ONTAP 环境
	为 Oracle 创建基于 NFS 的 SVM
	创建导出策略
	为 Oracle 创建卷
	创建 NFS LIF
* Linux 配置 *	创建挂载点并挂载 NFS 卷
	验证 NFS 挂载
	特定于操作系统的配置
	创建 Oracle 目录
	配置页面
	禁用 SELinux 和防火墙守护进程
	启用并启动 chronyd 服务
	增加文件描述符硬限制
	创建 pam 会话文件
* ORACLE_CONFIG *	Oracle 软件安装
	创建 Oracle 侦听器
	创建 Oracle 数据库
	Oracle 环境配置
	保存 PDB 状态
	启用实例归档模式
	启用 DNFS 客户端
	在操作系统重新启动之间启用数据库自动启动和关闭

默认参数

为了简化自动化，我们已使用默认值预设了许多必需的 Oracle 部署参数。通常，无需更改大多数部署的默认参数。更高级的用户可以谨慎地更改默认参数。默认参数位于每个角色文件夹的默认目录下。

部署说明

开始之前，请下载以下 Oracle 安装和修补程序文件，并将其放置在 ` /tmp/archive` 目录中，以便每个要部署的数据库服务器上的所有用户都能进行读取，写入和执行访问。自动化任务会在该特定目录中查找命名的安装文件，以便进行 Oracle 安装和配置。

```
LINUX.X64_193000_db_home.zip -- 19.3 base installer
p31281355_190000_Linux-x86-64.zip -- 19.8 RU patch
p6880880_190000_Linux-x86-64.zip -- opatch version 12.2.0.1.23
```

许可证

您应按照 Github 存储库中的说明读取许可证信息。访问，下载，安装或使用此存储库中的内容即表示您同意所规定的许可证条款 "此处"。

请注意，在生成和 / 或共享任何派生作品时，此存储库中的内容存在一定的限制。请务必阅读的条款 "许可证" 在使用内容之前。如果您不同意所有条款，请勿访问，下载或使用此存储库中的内容。

准备就绪后，单击 "此处可查看 AWX/ 塔式部署的详细步骤" 或 "此处用于命令行界面部署"。

分步部署操作步骤

AWX/Tower 部署 Oracle 19c 数据库

1. 为您的环境创建清单，组，主机和凭据

本节介绍如何在 AWX/Ansible 塔中设置清单，组，主机和访问凭据，以便为使用 NetApp 自动化解决方案的环境做好准备。

1. 配置清单。
    a. 导航到资源→清单→添加，然后单击添加清单。
    b. 提供名称和组织详细信息，然后单击保存。
    c. 在清单页面上，单击已创建的清单。
    d. 如果存在任何清单变量，请将其粘贴到变量字段中。
    e. 导航到组子菜单，然后单击添加。
    f. 提供 ONTAP 的组名称，粘贴组变量（如果有），然后单击保存。
    g. 对 Oracle 的另一个组重复此过程。
    h. 选择已创建的 ONTAP 组，转到主机子菜单，然后单击添加新主机。
    i. 提供 ONTAP 集群管理 IP 的 IP 地址，粘贴主机变量（如果有），然后单击保存。
    j. 必须对 Oracle 组和 Oracle 主机管理 IP/ 主机名重复此过程。
2. 创建凭据类型。对于涉及 ONTAP 的解决方案，您必须配置凭据类型以匹配用户名和密码条目。
    a. 导航到管理→凭据类型，然后单击添加。
    b. 提供名称和问题描述。
    c. 将以下内容粘贴到输入配置中：

```
fields:
  - id: username
    type: string
    label: Username
  - id: password
    type: string
    label: Password
    secret: true
  - id: vsadmin_password
    type: string
    label: vsadmin_password
    secret: true
```

1. 将以下内容粘贴到注射器配置中：

```
extra_vars:
  password: '{{ password }}'
  username: '{{ username }}'
  vsadmin_password: '{{ vsadmin_password }}'
```

1. 配置凭据。
   a. 导航到资源→凭据，然后单击添加。
   b. 输入 ONTAP 的名称和组织详细信息。
   c. 选择为 ONTAP 创建的自定义凭据类型。
   d. 在 Type Details 下，输入用户名，密码和 vsadmin_password 。
   e. 单击 Back to Credential ，然后单击 Add 。
   f. 输入 Oracle 的名称和组织详细信息。
   g. 选择计算机凭据类型。
   h. 在 Type Details 下，输入 Oracle 主机的 Username 和 Password 。
   i. 选择正确的权限升级方法，然后输入用户名和密码。

2. 创建项目

1. 转至 "Resources" → "projects" ，然后单击 "Add" 。
   a. 输入名称和组织详细信息。
   b. 在源控制凭据类型字段中选择 Git 。
   c. 输入 … <https://github.com/NetApp-Automation/na_oracle19c_deploy.git> 作为源控制 URL 。
   d. 单击保存。
   e. 当源代码发生更改时，项目可能偶尔需要同步。

3. 配置 Oracle host_vars

本节中定义的变量将应用于每个 Oracle 服务器和数据库。

1. 以以下嵌入式 Oracle hosts 变量或 host_vars 形式输入环境特定的参数。

必须根据您的环境更改蓝色项。

主机 VARS 配置

```
#################################################################################
##############      Host Variables Configuration        ##############
#################################################################################

# Add your Oracle Host
ansible_host: "10.61.180.15"

# Oracle db log archive mode: true - ARCHIVELOG or false - NOARCHIVELOG
log_archive_mode: "true"

# Number of pluggable databases per container instance identified by sid.
Pdb_name specifies the prefix for container database naming in this case
cdb2_pdb1, cdb2_pdb2, cdb2_pdb3
oracle_sid: "cdb2"
pdb_num: "3"
pdb_name: "{{ oracle_sid }}_pdb"

# CDB listener port, use different listener port for additional CDB on
same host
listener_port: "1523"

# CDB is created with SGA at 75% of memory_limit, MB. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB. The grand total SGA should not exceed 75% available RAM on node.
memory_limit: "5464"

# Set "em_configuration: DBEXPRESS" to install enterprise manager express
and choose a unique port from 5500 to 5599 for each sid on the host.
# Leave them black if em express is not installed.
em_configuration: "DBEXPRESS"
em_express_port: "5501"

# {{groups.oracle[0]}} represents first Oracle DB server as defined in
Oracle hosts group [oracle]. For concurrent multiple Oracle DB servers
deployment, [0] will be incremented for each additional DB server. For
example,  {{groups.oracle[1]}}" represents DB server 2,
```

```
"{{groups.oracle[2]}}" represents DB server 3 ... As a good practice and
the default, minimum three volumes is allocated to a DB server with
corresponding /u01, /u02, /u03 mount points, which store oracle binary,
oracle data, and oracle recovery files respectively. Additional volumes
can be added by click on "More NFS volumes" but the number of volumes
allocated to a DB server must match with what is defined in global vars
file by volumes_nfs parameter, which dictates how many volumes are to be
created for each DB server.
host_datastores_nfs:
  - {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
```

1. 填写蓝色字段中的所有变量。
2. 完成变量输入后，单击表单上的复制按钮以复制要传输到 AWX 或塔式的所有变量。
3. 导航回 AWX 或塔式，然后转到资源→主机，选择并打开 Oracle 服务器配置页面。
4. 在详细信息选项卡下，单击编辑并将从步骤 1 复制的变量粘贴到 YAML 选项卡下的变量字段。
5. 单击保存。
6. 对系统中的任何其他 Oracle 服务器重复此过程。

4. 配置全局变量

本节中定义的变量适用于所有 Oracle 主机，数据库和 ONTAP 集群。

1. 在以下嵌入式全局变量或变量表单中输入环境特定的参数。

必须根据您的环境更改蓝色项。

```
###########################################################################
###### Oracle 19c deployment global user configuration variables ######
######  Consolidate all variables from ontap, linux and oracle   ######
###########################################################################

#########################################
### Ontap env specific config variables ###
#########################################

#Inventory group name
#Default inventory group name - 'ontap'
#Change only if you are changing the group name either in inventory/hosts
file or in inventory groups in case of AWX/Tower
```

```
hosts_group: "ontap"

#CA_signed_certificates (ONLY CHANGE to 'true' IF YOU ARE USING CA SIGNED
CERTIFICATES)
ca_signed_certs: "false"

#Names of the Nodes in the ONTAP Cluster
nodes:
 - "AFF-01"
 - "AFF-02"

#Storage VLANs
#Add additional rows for vlans as necessary
storage_vlans:
   - {vlan_id: "203", name: "infra_NFS", protocol: "NFS"}
More Storage VLANsEnter Storage VLANs details

#Details of the Data Aggregates that need to be created
#If Aggregate creation takes longer, subsequent tasks of creating volumes
may fail.
#There should be enough disks already zeroed in the cluster, otherwise
aggregate create will zero the disks and will take long time
data_aggregates:
  - {aggr_name: "aggr01_node01"}
  - {aggr_name: "aggr01_node02"}

#SVM name
svm_name: "ora_svm"

# SVM Management LIF Details
svm_mgmt_details:
  - {address: "172.21.91.100", netmask: "255.255.255.0", home_port: "e0M"}

# NFS storage parameters when data_protocol set to NFS. Volume named after
Oracle hosts name identified by mount point as follow for oracle DB server
1. Each mount point dedicates to a particular Oracle files: u01 - Oracle
binary, u02 - Oracle data, u03 - Oracle redo. Add additional volumes by
click on "More NFS volumes" and also add the volumes list to corresponding
host_vars as host_datastores_nfs variable. For multiple DB server
deployment, additional volumes sets needs to be added for additional DB
server. Input variable "{{groups.oracle[1]}}_u01",
"{{groups.oracle[1]}}_u02", and "{{groups.oracle[1]}}_u03" as vol_name for
second DB server. Place volumes for multiple DB servers alternatingly
between controllers for balanced IO performance, e.g. DB server 1 on
controller node1, DB server 2 on controller node2 etc. Make sure match lif
address with controller node.
```

```
volumes_nfs:
  - {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}

#NFS LIFs IP address and netmask

nfs_lifs_details:
  - address: "172.21.94.200" #for node-1
    netmask: "255.255.255.0"
  - address: "172.21.94.201" #for node-2
    netmask: "255.255.255.0"

#NFS client match

client_match: "172.21.94.0/24"

##############################################
### Linux env specific config variables ###
##############################################

#NFS Mount points for Oracle DB volumes

mount_points:
  - "/u01"
  - "/u02"
  - "/u03"

# Up to 75% of node memory size divided by 2mb. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB.
# Leave it blank if hugepage is not configured on the host.

hugepages_nr: "1234"

# RedHat subscription username and password

redhat_sub_username: "xxx"
redhat_sub_password: "xxx"

##################################################
### DB env specific install and config variables ###
##################################################
```

```
db_domain: "your.domain.com"

# Set initial password for all required Oracle passwords. Change them
after installation.

initial_pwd_all: "netapp123"
```

1. 在蓝色字段中填写所有变量。
2. 完成变量输入后，单击表单上的复制按钮，将要传输到 AWX 或塔式的所有变量复制到以下作业模板中。

5. 配置并启动作业模板。

1. 创建作业模板。
   a. 导航到资源→模板→添加，然后单击添加作业模板。
   b. 输入名称和问题描述
   c. 选择作业类型；运行将根据攻略手册配置系统，而检查将在不实际配置系统的情况下执行攻略手册的试运行。
   d. 为攻略手册选择相应的清单，项目，攻略手册和凭据。
   e. 选择 all_playbook.yml 作为要执行的默认攻略手册。
   f. 将从步骤 4 复制的全局变量粘贴到 YAML 选项卡下的模板变量字段中。
   g. 选中作业标记字段中的启动时提示框。
   h. 单击保存。
2. 启动作业模板。
   a. 导航到资源→模板。
   b. 单击所需模板，然后单击启动。
   c. 在启动作业标记时系统提示时，键入 requirements_config 。您可能需要单击 requirements_config 下方的 Create Job Tag 行以输入作业标记。

requirements_config 可确保您有正确的库来运行其他角色。

1. 单击下一步，然后单击启动以启动作业。
2. 单击查看→作业以监控作业输出和进度。
3. 在启动作业标记时，系统提示您键入 ontap_config 。您可能需要单击 ontap_config 下方的 Create "Job Tag（创建作业标记） " 行以输入作业标记。
4. 单击下一步，然后单击启动以启动作业。
5. 单击查看→作业以监控作业输出和进度
6. 完成 ontap_config 角色后，对 linux_config 再次运行此过程。
7. 导航到资源→模板。
8. 选择所需模板，然后单击启动。

9. 在 Linux_config 中启动作业标记类型时，系统提示您选择 Linux_config 下方的创建 " 作业标记 " 行以输入作业标记。
10. 单击下一步，然后单击启动以启动作业。
11. 选择 View → Jobs 以监控作业输出和进度。
12. 完成 Linux_config 角色后，对 ORACLE_CONFIG 再次运行此过程。
13. 转至资源→模板。
14. 选择所需模板，然后单击启动。
15. 在启动作业标记时，系统提示您键入 ORACLE_CONFIG 。您可能需要选择 ORACLE_CONFIG 下方的 "Create Job Tag （创建作业标记） " 行以输入作业标记。
16. 单击下一步，然后单击启动以启动作业。
17. 选择 View → Jobs 以监控作业输出和进度。

6. 在同一 **Oracle** 主机上部署其他数据库

此攻略手册的 Oracle 部分会每次在 Oracle 服务器上创建一个 Oracle 容器数据库。要在同一服务器上创建其他容器数据库，请完成以下步骤。

1. 修改 host_vars 变量。
   a. 返回到步骤 2 —配置 Oracle host_vars 。
   b. 将 Oracle SID 更改为其他命名字符串。
   c. 将侦听器端口更改为其他编号。
   d. 如果要安装 EM Express ，请将 EM Express 端口更改为其他编号。
   e. 将修订后的主机变量复制并粘贴到主机配置详细信息选项卡中的 Oracle 主机变量字段中。
2. 启动仅包含 ORACLE_CONFIG 标记的部署作业模板。
3. 以Oracle用户身份登录到Oracle服务器并执行以下命令：

```
ps -ef | grep ora
```

   如果安装按预期完成且 Oracle 数据库已启动，则此列表将列出 Oracle 进程

4. 登录到数据库以检查数据库配置设置以及使用以下命令集创建的PDB。

```
[oracle@localhost ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 12:52:51 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL>

SQL> select name, log_mode from v$database;
NAME      LOG_MODE
--------- ------------
CDB2      ARCHIVELOG

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 CDB2_PDB1                      READ WRITE NO
         4 CDB2_PDB2                      READ WRITE NO
         5 CDB2_PDB3                      READ WRITE NO

col svrname form a30
col dirname form a30
select svrname, dirname, nfsversion from v$dnfs_servers;

SQL> col svrname form a30
SQL> col dirname form a30
SQL> select svrname, dirname, nfsversion from v$dnfs_servers;

SVRNAME                       DIRNAME                       NFSVERSION
----------------------------- ------------------------------
----------------
172.21.126.200                /rhelora03_u02                NFSv3.0
172.21.126.200                /rhelora03_u03                NFSv3.0
172.21.126.200                /rhelora03_u01                NFSv3.0
```

这可确认DNFS运行正常。

5. 使用以下命令通过侦听器连接到数据库以检查Oracle侦听器配置。更改为相应的侦听器端口和数据库服务名称。

```
[oracle@localhost ~]$ sqlplus
system@//localhost:1523/cdb2_pdb1.cie.netapp.com

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 13:19:57 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.

Enter password:
Last Successful login time: Wed May 05 2021 17:11:11 -04:00

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL> show user
USER is "SYSTEM"
SQL> show con_name
CON_NAME
CDB2_PDB1
```

这样可以确认Oracle侦听器工作正常。

如何获取帮助?

如果您需要有关该工具包的帮助，请加入 "NetApp 解决方案自动化社区支持 Slack 通道" 并寻找解决方案自动化渠道来发布您的问题或询问。

分步部署操作步骤

本文档详细介绍了如何使用自动化命令行界面(CLI)部署Oracle 19c。

CLI 部署 Oracle 19c 数据库

本节介绍使用 CLI 准备和部署 Oracle19c 数据库所需的步骤。确保您已查看 "入门和要求部分" 并相应地准备好您的环境。

下载 Oracle19c repo

1. 从您的Ans负责人 控制器中、运行以下命令：

```
git clone https://github.com/NetApp-Automation/na_oracle19c_deploy.git
```

2. 下载存储库后、将目录更改为na_oracle19c_Deploy <cd na_oracle19c_deploy>。

编辑 **hosts** 文件

在部署之前完成以下操作：

1. 编辑 hosts 文件 na_oracle19c_deploy 目录。
2. 在 ONTAP 下，将 IP 地址更改为集群管理 IP 。
3. 在 Oracle 组下，添加 Oracle 主机名称。主机名必须通过 DNS 或 hosts 文件解析为其 IP 地址，或者必须在主机中指定。
4. 完成这些步骤后，请保存所做的任何更改。

以下示例说明了一个主机文件：

```
#ONTAP Host

[ontap]

"10.61.184.183"

#Oracle hosts

[oracle]

"rtpora01"

"rtpora02"
```

此示例将执行该攻略手册，并在两个 Oracle DB 服务器上同时部署 Oracle 19c 。您也可以仅使用一个数据库服务器进行测试。在这种情况下，您只需要配置一个主机变量文件。

无论您部署多少 Oracle 主机和数据库，本攻略手册的执行方式都是相同的。

编辑 **host_vars** 下的 **host_name.yml** 文件

每个 Oracle 主机都有其主机变量文件，该文件由包含主机专用变量的主机名标识。您可以为主机指定任何名称。从 Host VARS Config 部分编辑并复制 `host_vars` ，然后将其粘贴到所需的 `host_name.yml` 文件中。

必须根据您的环境更改蓝色项。

主机 **VARS** 配置

```
#################################################################################
##############       Host Variables Configuration        ##############
#################################################################################

# Add your Oracle Host
```

```
ansible_host: "10.61.180.15"

# Oracle db log archive mode: true - ARCHIVELOG or false - NOARCHIVELOG
log_archive_mode: "true"

# Number of pluggable databases per container instance identified by sid.
Pdb_name specifies the prefix for container database naming in this case
cdb2_pdb1, cdb2_pdb2, cdb2_pdb3
oracle_sid: "cdb2"
pdb_num: "3"
pdb_name: "{{ oracle_sid }}_pdb"

# CDB listener port, use different listener port for additional CDB on
same host
listener_port: "1523"

# CDB is created with SGA at 75% of memory_limit, MB. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB. The grand total SGA should not exceed 75% available RAM on node.
memory_limit: "5464"

# Set "em_configuration: DBEXPRESS" to install enterprise manager express
and choose a unique port from 5500 to 5599 for each sid on the host.
# Leave them black if em express is not installed.
em_configuration: "DBEXPRESS"
em_express_port: "5501"

# {{groups.oracle[0]}} represents first Oracle DB server as defined in
Oracle hosts group [oracle]. For concurrent multiple Oracle DB servers
deployment, [0] will be incremented for each additional DB server. For
example,  {{groups.oracle[1]}}" represents DB server 2,
"{{groups.oracle[2]}}" represents DB server 3 ... As a good practice and
the default, minimum three volumes is allocated to a DB server with
corresponding /u01, /u02, /u03 mount points, which store oracle binary,
oracle data, and oracle recovery files respectively. Additional volumes
can be added by click on "More NFS volumes" but the number of volumes
allocated to a DB server must match with what is defined in global vars
file by volumes_nfs parameter, which dictates how many volumes are to be
created for each DB server.
host_datastores_nfs:
  - {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
```

编辑 **vars.yml** 文件

`vars.yml` 文件整合了所有环境特定的变量（ ONTAP ， Linux 或 Oracle ），用于 Oracle 部署。

1. 编辑并复制 VARS 部分中的变量，然后将这些变量粘贴到 `vars.yml` 文件中。

```
###########################################################################
###### Oracle 19c deployment global user configuration variables ######
######  Consolidate all variables from ontap, linux and oracle   ######
###########################################################################

#########################################
### Ontap env specific config variables ###
#########################################

#Inventory group name
#Default inventory group name - 'ontap'
#Change only if you are changing the group name either in inventory/hosts
file or in inventory groups in case of AWX/Tower
hosts_group: "ontap"

#CA_signed_certificates (ONLY CHANGE to 'true' IF YOU ARE USING CA SIGNED
CERTIFICATES)
ca_signed_certs: "false"

#Names of the Nodes in the ONTAP Cluster
nodes:
 - "AFF-01"
 - "AFF-02"

#Storage VLANs
#Add additional rows for vlans as necessary
storage_vlans:
   - {vlan_id: "203", name: "infra_NFS", protocol: "NFS"}
More Storage VLANsEnter Storage VLANs details

#Details of the Data Aggregates that need to be created
#If Aggregate creation takes longer, subsequent tasks of creating volumes
may fail.
#There should be enough disks already zeroed in the cluster, otherwise
aggregate create will zero the disks and will take long time
data_aggregates:
  - {aggr_name: "aggr01_node01"}
  - {aggr_name: "aggr01_node02"}

#SVM name
```

```
svm_name: "ora_svm"

# SVM Management LIF Details
svm_mgmt_details:
  - {address: "172.21.91.100", netmask: "255.255.255.0", home_port: "e0M"}

# NFS storage parameters when data_protocol set to NFS. Volume named after
Oracle hosts name identified by mount point as follow for oracle DB server
1. Each mount point dedicates to a particular Oracle files: u01 - Oracle
binary, u02 - Oracle data, u03 - Oracle redo. Add additional volumes by
click on "More NFS volumes" and also add the volumes list to corresponding
host_vars as host_datastores_nfs variable. For multiple DB server
deployment, additional volumes sets needs to be added for additional DB
server. Input variable "{{groups.oracle[1]}}_u01",
"{{groups.oracle[1]}}_u02", and "{{groups.oracle[1]}}_u03" as vol_name for
second DB server. Place volumes for multiple DB servers alternatingly
between controllers for balanced IO performance, e.g. DB server 1 on
controller node1, DB server 2 on controller node2 etc. Make sure match lif
address with controller node.

volumes_nfs:
  - {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}

#NFS LIFs IP address and netmask

nfs_lifs_details:
  - address: "172.21.94.200" #for node-1
    netmask: "255.255.255.0"
  - address: "172.21.94.201" #for node-2
    netmask: "255.255.255.0"

#NFS client match

client_match: "172.21.94.0/24"

###########################################
### Linux env specific config variables ###
###########################################

#NFS Mount points for Oracle DB volumes

mount_points:
```

```
  - "/u01"
  - "/u02"
  - "/u03"

# Up to 75% of node memory size divided by 2mb. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB.
# Leave it blank if hugepage is not configured on the host.

hugepages_nr: "1234"

# RedHat subscription username and password

redhat_sub_username: "xxx"
redhat_sub_password: "xxx"

#####################################################
### DB env specific install and config variables ###
#####################################################

db_domain: "your.domain.com"

# Set initial password for all required Oracle passwords. Change them
after installation.

initial_pwd_all: "netapp123"
```

运行攻略手册

完成所需的环境前提条件并将变量复制到 `vars.yml` 和 `yor_host.yml` 中后，您便可部署攻略手册了。

必须根据您的环境更改 < 用户名 > 。

1. 通过传递正确的标记和ONTAP集群用户名来运行ONTAP操作手册。根据提示填写ONTAP集群的密码、然后填写vsadmin。

```
ansible-playbook -i hosts all_playbook.yml -u username -k -K -t
ontap_config -e @vars/vars.yml
```

2. 运行Linux操作手册以执行Linux部署部分。输入admin ssh密码和sudo密码。

```
ansible-playbook -i hosts all_playbook.yml -u username -k -K -t
linux_config -e @vars/vars.yml
```

3. 运行Oracle操作手册以执行Oracle部署部分。输入admin ssh密码和sudo密码。

```
ansible-playbook -i hosts all_playbook.yml -u username -k -K -t
oracle_config -e @vars/vars.yml
```

在同一 **Oracle** 主机上部署其他数据库

此攻略手册的 Oracle 部分会每次在 Oracle 服务器上创建一个 Oracle 容器数据库。要在同一服务器上创建其他容器数据库，请完成以下步骤：

1. 修改 host_vars 变量。
   a. 返回到步骤 3 - 编辑 `host_vars` 下的 `host_name.yml` 文件。
   b. 将 Oracle SID 更改为其他命名字符串。
   c. 将侦听器端口更改为其他编号。
   d. 如果已安装 EM Express ，请将 EM Express 端口更改为其他编号。
   e. 将修订后的主机变量复制并粘贴到 `host_vars` 下的 Oracle 主机变量文件中。
2. 使用 `ORACLE_CONFIG` 标记执行攻略手册，如中所示 [运行攻略手册]。

验证 **Oracle** 安装

1. 以Oracle用户身份登录到Oracle服务器并执行以下命令：

```
ps -ef | grep ora
```

如果安装按预期完成且 Oracle 数据库已启动，则此列表将列出 Oracle 进程

2. 登录到数据库以检查数据库配置设置以及使用以下命令集创建的PDB。

```
[oracle@localhost ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 12:52:51 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL>

SQL> select name, log_mode from v$database;
NAME      LOG_MODE
--------- ------------
CDB2      ARCHIVELOG

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 CDB2_PDB1                      READ WRITE NO
         4 CDB2_PDB2                      READ WRITE NO
         5 CDB2_PDB3                      READ WRITE NO

col svrname form a30
col dirname form a30
select svrname, dirname, nfsversion from v$dnfs_servers;

SQL> col svrname form a30
SQL> col dirname form a30
SQL> select svrname, dirname, nfsversion from v$dnfs_servers;

SVRNAME                       DIRNAME                       NFSVERSION
----------------------------- ------------------------------
----------------
172.21.126.200                /rhelora03_u02                NFSv3.0
172.21.126.200                /rhelora03_u03                NFSv3.0
172.21.126.200                /rhelora03_u01                NFSv3.0
```

这可确认DNFS运行正常。

3. 使用以下命令通过侦听器连接到数据库以检查Oracle侦听器配置。更改为相应的侦听器端口和数据库服务名称。

```
[oracle@localhost ~]$ sqlplus
system@//localhost:1523/cdb2_pdb1.cie.netapp.com

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 13:19:57 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.

Enter password:
Last Successful login time: Wed May 05 2021 17:11:11 -04:00

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL> show user
USER is "SYSTEM"
SQL> show con_name
CON_NAME
CDB2_PDB1
```

这样可以确认Oracle侦听器工作正常。

如何获取帮助?

如果您需要有关该工具包的帮助，请加入 "NetApp 解决方案自动化社区支持 Slack 通道" 并寻找解决方案自动化渠道来发布您的问题或询问。

解决方案概述

为 Oracle 数据库提供自动化数据保护

企业正在对其环境进行自动化，以提高效率，加快部署速度并减少手动操作。Ansible 等配置管理工具正在用于简化企业数据库操作。在本解决方案中，我们演示了如何使用 Ansible 通过 NetApp ONTAP 自动保护 Oracle 数据。通过使存储管理员，系统管理员和 DBA 能够一致快速地将数据复制到异地数据中心或公有云，您可以获得以下优势：

- 消除设计复杂性和人为错误，并实施可重复的一致部署和最佳实践
- 缩短配置集群间复制， CVO 实例化和恢复 Oracle 数据库的时间
- 提高数据库管理员，系统和存储管理员的工作效率
- 提供数据库恢复工作流，以便于测试灾难恢复场景。

NetApp 为客户提供经过验证的 Ansible 模块和角色，以加快 Oracle 数据库环境的部署，配置和生命周期管理。此解决方案提供了相关说明和 Ansible 攻略手册代码，可帮助您：

在内部复制到内部复制

- 在源和目标上创建集群间 LIF
- 建立集群和 SVM 对等关系
- 创建并初始化 Oracle 卷的 SnapMirror
- 通过 AWX/Tower 为 Oracle 二进制文件，数据库和日志创建复制计划
- 还原目标上的 Oracle 数据库，并使数据库联机

在内部迁移到 **AWS** 中的 **CVO**

- 创建 AWS 连接器
- 在 AWS 中创建 CVO 实例
- 将内部集群添加到 Cloud Manager
- 在源上创建集群间 LIF
- 建立集群和 SVM 对等关系
- 创建并初始化 Oracle 卷的 SnapMirror
- 通过 AWX/Tower 为 Oracle 二进制文件，数据库和日志创建复制计划
- 还原目标上的 Oracle 数据库，并使数据库联机

准备就绪后，单击 "请单击此处了解解决方案入门"。

入门

此解决方案设计为在 AWX/ 塔式环境中运行。

AWX/ 塔式

对于 AWX/Tower 环境，系统将引导您创建 ONTAP 集群管理和 Oracle 服务器（ IP 和主机名）清单，创建凭据，配置从 NetApp Automation Github 提取 Ansible 代码的项目以及启动自动化的作业模板。

1. 解决方案设计用于在私有云环境（内部到内部）和混合云（内部到公有 Cloud Cloud Volumes ONTAP （ CVO ））中运行
2. 填写特定于您的环境的变量，然后将其复制并粘贴到作业模板中的额外 VAR 字段中。
3. 将额外的变量添加到作业模板后，您可以启动自动化。
4. 此自动化设置为运行三个阶段（安装， Oracle 二进制文件复制计划，数据库，日志和仅适用于日志的复制计划），以及第四个阶段以在灾难恢复站点恢复数据库。
5. 有关获取 CVO 数据保护所需密钥和令牌的详细说明，请访问 "收集 CVO 和连接器部署的前提条件"

要求

<Strong class="bi"> 内部部署 </Strong> <Strong>|</Strong>

environment	要求
* 可逆环境 *	AWX/ 塔式
	Ansible v.10 及更高版本
	Python 3.
	Python 库— netapp-lib — xmltodict — jMespath
* ONTAP *	ONTAP 版本 9.8 及更高版本
	两个数据聚合
	已创建 NFS VLAN 和 ifgrp
* Oracle 服务器 *	RHEL 7/8
	Oracle Linux 7/8
	NFS ，公有和可选管理的网络接口
	源上的现有 Oracle 环境以及目标上的等效 Linux 操作系统（灾难恢复站点或公有云）

CVO</Strong>

environment	要求
* 可逆环境 *	AWX/ 塔式
	Ansible v.10 及更高版本
	Python 3.
	Python 库— netapp-lib — xmltodict — jMespath
* ONTAP *	ONTAP 版本 9.8 及更高版本
	两个数据聚合
	已创建 NFS VLAN 和 ifgrp
* Oracle 服务器 *	
	RHEL 7/8
	Oracle Linux 7/8
	NFS ，公有和可选管理的网络接口
	源上的现有 Oracle 环境以及目标上的等效 Linux 操作系统（灾难恢复站点或公有云）
	在 Oracle EC2 实例上设置适当的交换空间，默认情况下，某些 EC2 实例使用 0 交换部署
* Cloud Manager/AWS*	AWS 访问 / 机密密钥
	NetApp Cloud Manager 帐户
	NetApp Cloud Manager 刷新令牌

<Strong class="bi"> 内部部署 **</Strong> <Strong>|</Strong>**

此自动化部署采用一本 Ansible 攻略手册设计，该攻略手册包含三个不同的角色。这些角色适用于 ONTAP ， Linux 和 Oracle 配置。下表介绍了正在自动执行的任务。

攻略手册	任务
* ontap_setup*	预检查 ONTAP 环境
	在源集群上创建集群间 LIF （可选）
	在目标集群上创建集群间 LIF （可选）
	创建集群和 SVM 对等关系
	创建目标 SnapMirror 并初始化指定的 Oracle 卷
* 复制 _cg	在 /etc/oratab 中为每个数据库启用备份模式
	为 Oracle 二进制卷和数据库卷创建的快照
	已更新 SnapMirror
	关闭 /etc/oratab 中每个数据库的备份模式
* 。 ora_replication ； log*	切换 /etc/oratab 中每个数据库的当前日志
	为 Oracle 日志卷创建的快照
	已更新 SnapMirror
* ora_recovery*	中断 SnapMirror
	在目标上启用 NFS 并为 Oracle 卷创建接合路径
	配置 DR Oracle 主机
	挂载并验证 Oracle 卷
	恢复并启动 Oracle 数据库

CVO</Strong>

此自动化部署采用一本 Ansible 攻略手册设计，该攻略手册包含三个不同的角色。这些角色适用于 ONTAP ， Linux 和 Oracle 配置。下表介绍了正在自动执行的任务。

攻略手册	任务
* CVO_setup*	对环境进行预检查
	AWS 配置 /AWS 访问密钥 ID/ 机密密钥 / 默认区域
	创建 AWS 角色
	在 AWS 中创建 NetApp Cloud Manager Connector 实例
	在 AWS 中创建 Cloud Volumes ONTAP （ CVO ）实例
	将内部源 ONTAP 集群添加到 NetApp Cloud Manager 中
	创建目标 SnapMirror 并初始化指定的 Oracle 卷

攻略手册	任务
* 复制 _cg	在 /etc/oratab 中为每个数据库启用备份模式
	为 Oracle 二进制卷和数据库卷创建的快照
	已更新 SnapMirror
	关闭 /etc/oratab 中每个数据库的备份模式
* 。 ora_replication ； log*	切换 /etc/oratab 中每个数据库的当前日志
	为 Oracle 日志卷创建的快照
	已更新 SnapMirror
* ora_recovery*	中断 SnapMirror
	在目标 CVO 上启用 NFS 并为 Oracle 卷创建接合路径
	配置 DR Oracle 主机
	挂载并验证 Oracle 卷
	恢复并启动 Oracle 数据库

默认参数

为了简化自动化，我们已使用默认值预设了许多必需的 Oracle 参数。通常，无需更改大多数部署的默认参数。更高级的用户可以谨慎地更改默认参数。默认参数位于每个角色文件夹的默认目录下。

许可证

您应按照 Github 存储库中的说明读取许可证信息。访问，下载，安装或使用此存储库中的内容即表示您同意所规定的许可证条款 "此处"。

请注意，在生成和 / 或共享任何派生作品时，此存储库中的内容存在一定的限制。请务必阅读的条款 "许可证" 在使用内容之前。如果您不同意所有条款，请勿访问，下载或使用此存储库中的内容。

准备就绪后，单击 "此处可查看 AWX/ 塔式服务器的详细流程"。

分步部署操作步骤

AWX/ 塔式 **Oracle** 数据保护

为您的环境创建清单、组、主机和凭据

本节介绍如何在 AWX/Ansible 塔中设置清单，组，主机和访问凭据，以便为使用 NetApp 自动化解决方案的环境做好准备。

1. 配置清单。
    a. 导航到资源→清单→添加，然后单击添加清单。
    b. 提供名称和组织详细信息，然后单击保存。
    c. 在清单页面上，单击已创建的清单。

d. 导航到组子菜单，然后单击添加。

e. 提供第一个组的名称 oracle ，然后单击 Save 。

f. 对名为 dr_oracle 的第二个组重复此过程。

g. 选择已创建的 Oracle 组，转至主机子菜单，然后单击添加新主机。

h. 提供源 Oracle 主机管理 IP 的 IP 地址，然后单击保存。

i. 必须对 dr_oracle 组重复此过程，并添加 DR/Destination Oracle 主机的管理 IP/ 主机名。

下面介绍了如何使用 ONTAP 为内部部署或 AWS 上的 CVO 创建凭据类型和凭据。

内部

1. 配置凭据。
2. 创建凭据类型。对于涉及 ONTAP 的解决方案，您必须配置凭据类型以匹配用户名和密码条目。
   a. 导航到管理→凭据类型，然后单击添加。
   b. 提供名称和问题描述。
   c. 将以下内容粘贴到输入配置中：

```
fields:
  - id: dst_cluster_username
    type: string
    label: Destination Cluster Username
  - id: dst_cluster_password
    type: string
    label: Destination Cluster Password
    secret: true
  - id: src_cluster_username
    type: string
    label: Source Cluster Username
  - id: src_cluster_password
    type: string
    label: Source Cluster Password
    secret: true
```

   d. 将以下内容粘贴到 " 注入器配置 " 中，然后单击保存：

```
extra_vars:
  dst_cluster_username: '{{ dst_cluster_username }}'
  dst_cluster_password: '{{ dst_cluster_password }}'
  src_cluster_username: '{{ src_cluster_username }}'
  src_cluster_password: '{{ src_cluster_password }}'
```

3. 为 ONTAP 创建凭据
   a. 导航到资源→凭据，然后单击添加。
   b. 输入 ONTAP 凭据的名称和组织详细信息
   c. 选择上一步中创建的凭据类型。
   d. 在 Type Details 下，输入源集群和目标集群的用户名和密码。
   e. 单击保存。
4. 为 Oracle 创建凭据
   a. 导航到资源→凭据，然后单击添加。

b. 输入 Oracle 的名称和组织详细信息。

c. 选择计算机凭据类型。

d. 在 Type Details 下，输入 Oracle 主机的 Username 和 Password 。

e. 选择正确的权限升级方法，然后输入用户名和密码。

f. 单击保存。

g. 如果需要为 dr_oracle 主机配置其他凭据，请重复此过程。

CVO

1. 配置凭据。
2. 创建凭据类型。对于涉及 ONTAP 的解决方案，您必须配置凭据类型以匹配用户名和密码条目，我们还会为 Cloud Central 和 AWS 添加条目。
   a. 导航到管理→凭据类型，然后单击添加。
   b. 提供名称和问题描述。
   c. 将以下内容粘贴到输入配置中：

```
fields:
  - id: dst_cluster_username
    type: string
    label: CVO Username
  - id: dst_cluster_password
    type: string
    label: CVO Password
    secret: true
  - id: cvo_svm_password
    type: string
    label: CVO SVM Password
    secret: true
  - id: src_cluster_username
    type: string
    label: Source Cluster Username
  - id: src_cluster_password
    type: string
    label: Source Cluster Password
    secret: true
  - id: regular_id
    type: string
    label: Cloud Central ID
    secret: true
  - id: email_id
    type: string
    label: Cloud Manager Email
    secret: true
  - id: cm_password
    type: string
    label: Cloud Manager Password
    secret: true
  - id: access_key
    type: string
    label: AWS Access Key
    secret: true
  - id: secret_key
    type: string
    label: AWS Secret Key
    secret: true
  - id: token
    type: string
    label: Cloud Central Refresh Token
    secret: true
```

d. 将以下内容粘贴到 "Injector Configuration" 中，然后单击 "Save：

```
extra_vars:
  dst_cluster_username: '{{ dst_cluster_username }}'
  dst_cluster_password: '{{ dst_cluster_password }}'
  cvo_svm_password: '{{ cvo_svm_password }}'
  src_cluster_username: '{{ src_cluster_username }}'
  src_cluster_password: '{{ src_cluster_password }}'
  regular_id: '{{ regular_id }}'
  email_id: '{{ email_id }}'
  cm_password: '{{ cm_password }}'
  access_key: '{{ access_key }}'
  secret_key: '{{ secret_key }}'
  token: '{{ token }}'
```

3. 为 ontap/CVO/AWS 创建凭据
   a. 导航到资源→凭据，然后单击添加。
   b. 输入 ONTAP 凭据的名称和组织详细信息
   c. 选择上一步中创建的凭据类型。
   d. 在 Type Details 下，输入源集群和 CVO 集群， Cloud Central/Manager ， AWS 访问 / 机密密钥和 Cloud Central 刷新令牌的用户名和密码。
   e. 单击保存。
4. 为 Oracle 创建凭据（源）
   a. 导航到资源→凭据，然后单击添加。
   b. 输入 Oracle 主机的名称和组织详细信息
   c. 选择计算机凭据类型。
   d. 在 Type Details 下，输入 Oracle 主机的 Username 和 Password 。
   e. 选择正确的权限升级方法，然后输入用户名和密码。
   f. 单击保存。
5. 为 Oracle 目标创建凭据
   a. 导航到资源→凭据，然后单击添加。
   b. 输入 DR Oracle 主机的名称和组织详细信息
   c. 选择计算机凭据类型。
   d. 在 Type Details 下，输入 Username （ EC2-user ，或者如果您已对其进行了默认更改，请输入该用户名）和 SSH 私钥
   e. 选择正确的权限升级方法（ sudo ），然后根据需要输入用户名和密码。
   f. 单击保存。

创建项目

1. 转至 "Resources" → "projects" ，然后单击 "Add" 。
   a. 输入名称和组织详细信息。
   b. 在源控制凭据类型字段中选择 Git 。
   c. 输入 … <https://github.com/NetApp-Automation/na_oracle19c_data_protection.git> 作为源控制 URL 。
   d. 单击保存。
   e. 当源代码发生更改时，项目可能偶尔需要同步。

配置全局变量

本节中定义的变量适用于所有 Oracle 主机，数据库和 ONTAP 集群。

1. 在以下嵌入式全局变量或变量表单中输入环境特定的参数。

   必须根据您的环境更改蓝色项。

内部

```
# Oracle Data Protection global user configuration variables
# Ontap env specific config variables
hosts_group: "ontap"
ca_signed_certs: "false"

# Inter-cluster LIF details
src_nodes:
  - "AFF-01"
  - "AFF-02"

dst_nodes:
  - "DR-AFF-01"
  - "DR-AFF-02"

create_source_intercluster_lifs: "yes"

source_intercluster_network_port_details:
  using_dedicated_ports: "yes"
  using_ifgrp: "yes"
  using_vlans: "yes"
  failover_for_shared_individual_ports: "yes"
  ifgrp_name: "a0a"
  vlan_id: "10"
  ports:
    - "e0b"
    - "e0g"
  broadcast_domain: "NFS"
  ipspace: "Default"
  failover_group_name: "iclifs"

source_intercluster_lif_details:
  - name: "icl_1"
    address: "10.0.0.1"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "AFF-01"
  - name: "icl_2"
    address: "10.0.0.2"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "AFF-02"

create_destination_intercluster_lifs: "yes"
```

```
destination_intercluster_network_port_details:
  using_dedicated_ports: "yes"
  using_ifgrp: "yes"
  using_vlans: "yes"
  failover_for_shared_individual_ports: "yes"
  ifgrp_name: "a0a"
  vlan_id: "10"
  ports:
    - "e0b"
    - "e0g"
  broadcast_domain: "NFS"
  ipspace: "Default"
  failover_group_name: "iclifs"

destination_intercluster_lif_details:
  - name: "icl_1"
    address: "10.0.0.3"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "DR-AFF-01"
  - name: "icl_2"
    address: "10.0.0.4"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "DR-AFF-02"

# Variables for SnapMirror Peering
passphrase: "your-passphrase"

# Source & Destination List
dst_cluster_name: "dst-cluster-name"
dst_cluster_ip: "dst-cluster-ip"
dst_vserver: "dst-vserver"
dst_nfs_lif: "dst-nfs-lif"
src_cluster_name: "src-cluster-name"
src_cluster_ip: "src-cluster-ip"
src_vserver: "src-vserver"

# Variable for Oracle Volumes and SnapMirror Details
cg_snapshot_name_prefix: "oracle"
src_orabinary_vols:
  - "binary_vol"
src_db_vols:
  - "db_vol"
src_archivelog_vols:
  - "log_vol"
```

```
snapmirror_policy: "async_policy_oracle"

# Export Policy Details
export_policy_details:
  name: "nfs_export_policy"
  client_match: "0.0.0.0/0"
  ro_rule: "sys"
  rw_rule: "sys"

# Linux env specific config variables
mount_points:
  - "/u01"
  - "/u02"
  - "/u03"
hugepages_nr: "1234"
redhat_sub_username: "xxx"
redhat_sub_password: "xxx"

# DB env specific install and config variables
recovery_type: "scn"
control_files:
  - "/u02/oradata/CDB2/control01.ctl"
  - "/u03/orareco/CDB2/control02.ctl"
```

CVO

```
############################################
### Ontap env specific config variables ###
############################################

#Inventory group name
#Default inventory group name - "ontap"
#Change only if you are changing the group name either in
inventory/hosts file or in inventory groups in case of AWX/Tower
hosts_group: "ontap"

#CA_signed_certificates (ONLY CHANGE to "true" IF YOU ARE USING CA
SIGNED CERTIFICATES)
ca_signed_certs: "false"

#Names of the Nodes in the Source ONTAP Cluster
src_nodes:
  - "AFF-01"
  - "AFF-02"

#Names of the Nodes in the Destination CVO Cluster
```

```
dst_nodes:
  - "DR-AFF-01"
  - "DR-AFF-02"

#Define whether or not to create intercluster lifs on source cluster
(ONLY CHANGE to "No" IF YOU HAVE ALREADY CREATED THE INTERCLUSTER LIFS)
create_source_intercluster_lifs: "yes"

source_intercluster_network_port_details:
  using_dedicated_ports: "yes"
  using_ifgrp: "yes"
  using_vlans: "yes"
  failover_for_shared_individual_ports: "yes"
  ifgrp_name: "a0a"
  vlan_id: "10"
  ports:
    - "e0b"
    - "e0g"
  broadcast_domain: "NFS"
  ipspace: "Default"
  failover_group_name: "iclifs"

source_intercluster_lif_details:
  - name: "icl_1"
    address: "10.0.0.1"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "AFF-01"
  - name: "icl_2"
    address: "10.0.0.2"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "AFF-02"

##########################################
### CVO Deployment Variables ###
##########################################

####### Access Keys Variables ######

# Region where your CVO will be deployed.
region_deploy: "us-east-1"

########### CVO and Connector Vars ########

# AWS Managed Policy required to give permission for IAM role creation.
```

```
aws_policy: "arn:aws:iam::1234567:policy/OCCM"

# Specify your aws role name, a new role is created if one already does
not exist.
aws_role_name: "arn:aws:iam::1234567:policy/OCCM"

# Name your connector.
connector_name: "awx_connector"

# Name of the key pair generated in AWS.
key_pair: "key_pair"

# Name of the Subnet that has the range of IP addresses in your VPC.
subnet: "subnet-12345"

# ID of your AWS secuirty group that allows access to on-prem
resources.
security_group: "sg-123123123"

# You Cloud Manager Account ID.
account: "account-A23123A"

# Name of the your CVO instance
cvo_name: "test_cvo"

# ID of the VPC in AWS.
vpc: "vpc-123123123"

#########################################################################
############################
# Variables for - Add on-prem ONTAP to Connector in Cloud Manager
#########################################################################
############################

# For Federated users, Client ID from API Authentication Section of
Cloud Central to generate access token.
sso_id: "123123123123123123123"

# For regular access with username and password, please specify "pass"
as the connector_access. For SSO users, use "refresh_token" as the
variable.
connector_access: "pass"

#########################################################################
############################
# Variables for SnapMirror Peering
#########################################################################
```

```
##############################
passphrase: "your-passphrase"

#################################################################################
##############################
# Source & Destination List
#################################################################################
##############################
#Please Enter Destination Cluster Name
dst_cluster_name: "dst-cluster-name"

#Please Enter Destination Cluster (Once CVO is Created Add this
Variable to all templates)
dst_cluster_ip: "dst-cluster-ip"

#Please Enter Destination SVM to create mirror relationship
dst_vserver: "dst-vserver"

#Please Enter NFS Lif for dst vserver (Once CVO is Created Add this
Variable to all templates)
dst_nfs_lif: "dst-nfs-lif"

#Please Enter Source Cluster Name
src_cluster_name: "src-cluster-name"

#Please Enter Source Cluster
src_cluster_ip: "src-cluster-ip"

#Please Enter Source SVM
src_vserver: "src-vserver"

#################################################################################
##############################
# Variable for Oracle Volumes and SnapMirror Details
#################################################################################
##############################
#Please Enter Source Snapshot Prefix Name
cg_snapshot_name_prefix: "oracle"

#Please Enter Source Oracle Binary Volume(s)
src_orabinary_vols:
  - "binary_vol"
#Please Enter Source Database Volume(s)
src_db_vols:
  - "db_vol"
#Please Enter Source Archive Volume(s)
```

```
src_archivelog_vols:
  - "log_vol"
#Please Enter Destination Snapmirror Policy
snapmirror_policy: "async_policy_oracle"

#################################################################################
###############################
# Export Policy Details
#################################################################################
###############################
#Enter the destination export policy details (Once CVO is Created Add
this Variable to all templates)
export_policy_details:
  name: "nfs_export_policy"
  client_match: "0.0.0.0/0"
  ro_rule: "sys"
  rw_rule: "sys"

#################################################################################
###############################
### Linux env specific config variables ###
#################################################################################
###############################

#NFS Mount points for Oracle DB volumes
mount_points:
  - "/u01"
  - "/u02"
  - "/u03"

# Up to 75% of node memory size divided by 2mb. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB.
# Leave it blank if hugepage is not configured on the host.
hugepages_nr: "1234"

# RedHat subscription username and password
redhat_sub_username: "xxx"
redhat_sub_password: "xxx"

#####################################################
### DB env specific install and config variables ###
#####################################################
#Recovery Type (leave as scn)
recovery_type: "scn"
```

```
#Oracle Control Files
control_files:
  - "/u02/oradata/CDB2/control01.ctl"
  - "/u03/orareco/CDB2/control02.ctl"
```

自动化攻略手册

需要运行四本单独的攻略手册。

1. 用于设置环境的攻略手册，内部部署或 CVO 。
2. 用于按计划复制 Oracle 二进制文件和数据库的攻略手册
3. 用于按计划复制 Oracle 日志的攻略手册
4. 用于在目标主机上恢复数据库的攻略手册

ONTAP/CVO 设置

ONTAP 和 CVO 设置

配置并启动作业模板。

1. 创建作业模板。
   a. 导航到资源→模板→添加，然后单击添加作业模板。
   b. 输入名称 ontap/CVO 设置
   c. 选择作业类型；运行将根据攻略手册配置系统。
   d. 为攻略手册选择相应的清单，项目，攻略手册和凭据。
   e. 为内部环境选择 ontap_setup.yml 攻略手册，或者选择 CVO_setup.yml 以复制到 CVO 实例。
   f. 将从步骤 4 复制的全局变量粘贴到 YAML 选项卡下的模板变量字段中。
   g. 单击保存。
2. 启动作业模板。
   a. 导航到资源→模板。
   b. 单击所需模板，然后单击启动。

   我们将使用此模板并将其复制到其他攻略手册中。

二进制卷和数据库卷的复制

计划二进制文件和数据库复制攻略手册

配置并启动作业模板。

1. 复制先前创建的作业模板。
   a. 导航到资源→模板。
   b. 找到 ontap/CVO 设置模板，然后在最右侧单击复制模板
   c. 单击复制的模板上的编辑模板，然后将名称更改为二进制和数据库复制攻略手册。
   d. 保留该模板的相同清单，项目和凭据。
   e. 选择 ora_replication cc.yml 作为要执行的攻略手册。
   f. 这些变量将保持不变，但需要在变量 dst_cluster_ip 中设置 CVO 集群 IP 。
   g. 单击保存。
2. 计划作业模板。
   a. 导航到资源→模板。
   b. 单击二进制和数据库复制攻略手册模板，然后单击顶部选项集的计划。
   c. 单击添加，为二进制和数据库复制添加名称计划，在小时开始时选择开始日期 / 时间，选择您的本地时区以及运行频率。运行频率通常会更新 SnapMirror 复制。

系统将为日志卷复制创建一个单独的计划，以便可以更频繁地进行复制。

复制日志卷

计划日志复制攻略手册

配置并启动作业模板。

1. 复制先前创建的作业模板。
   a. 导航到资源→模板。
   b. 找到 ontap/CVO 设置模板，然后在最右侧单击复制模板
   c. 单击复制的模板上的 Edit Template ，然后将名称更改为 Log Replication 攻略手册。
   d. 保留该模板的相同清单，项目和凭据。
   e. 选择 ora_replication logs.yml 作为要执行的攻略手册。
   f. 这些变量将保持不变，但需要在变量 dst_cluster_ip 中设置 CVO 集群 IP 。
   g. 单击保存。
2. 计划作业模板。
   a. 导航到资源→模板。
   b. 单击日志复制攻略手册模板，然后单击顶部选项集的计划。
   c. 单击添加，为日志复制添加名称计划，选择开始日期 / 时间，选择本地时区和运行频率。运行频率通常会更新 SnapMirror 复制。

   建议将日志计划设置为每小时更新一次，以确保恢复到上一个每小时更新。

还原和恢复数据库

计划日志复制攻略手册

配置并启动作业模板。

1. 复制先前创建的作业模板。
   a. 导航到资源→模板。
   b. 找到 ontap/CVO 设置模板，然后在最右侧单击复制模板
   c. 单击复制的模板上的 Edit Template ，然后将名称更改为 Restore and Recovery Playbook 。
   d. 保留该模板的相同清单，项目和凭据。
   e. 选择 ora_recovery.yml 作为要执行的攻略手册。
   f. 这些变量将保持不变，但需要在变量 dst_cluster_ip 中设置 CVO 集群 IP 。
   g. 单击保存。

   只有在准备好在远程站点还原数据库后，才会运行此攻略手册。

1. 内部生产 Oracle 数据库数据卷通过 NetApp SnapMirror 复制到二级数据中心的冗余 ONTAP 集群或公有云中的 Cloud Volume ONTAP 进行保护。在完全配置的灾难恢复环境中，二级数据中心或公有云中的恢复计算实例处于备用状态，可以在发生灾难时恢复生产数据库。备用计算实例通过在操作系统内核修补程序上运行准面更新或在一个锁定步骤中进行升级，与内部实例保持同步。
2. 在此解决方案演示中， Oracle 二进制卷会复制到目标并挂载到目标实例，以启动 Oracle 软件堆栈。与灾难发生时的最后一分钟全新安装 Oracle 相比，这种恢复 Oracle 的方法更有优势。它可以保证 Oracle 安装与当前内部生产软件安装和修补程序级别等完全同步。但是，根据 Oracle 软件许可的结构，这可能会对恢复站点上复制的 Oracle 二进制卷产生额外的软件许可影响，也可能不会产生额外的软件许可影响。建议用户在决定使用相同方法之前，先咨询其软件许可人员，以评估潜在的 Oracle 许可要求。
3. 目标位置的备用 Oracle 主机已配置 Oracle 前提条件配置。
4. SnapMirrors 已损坏，卷可写并挂载到备用 Oracle 主机。
5. 在备用计算实例上挂载所有数据库卷后， Oracle 恢复模块将在恢复站点执行以下恢复和启动 Oracle 任务。
    a. 同步控制文件：我们在不同的数据库卷上部署了重复的 Oracle 控制文件，以保护关键数据库控制文件。一个位于数据卷上，另一个位于日志卷上。由于数据卷和日志卷的复制频率不同，因此在恢复时它们将不同步。
    b. 重新链接 Oracle 二进制文件：由于 Oracle 二进制文件已重新定位到新主机，因此需要重新链接。
    c. 恢复 Oracle 数据库：恢复机制从控制文件中检索 Oracle 日志卷中最后一个可用归档日志中的最后一个系统更改编号，并恢复 Oracle 数据库以重新处理在发生故障时能够复制到灾难恢复站点的所有业务事务。然后，数据库将在新的形式中启动，以在恢复站点执行用户连接和业务事务。

> 运行恢复攻略手册之前，请确保您具备以下条件：确保它通过 /etc/oratab 和 /etc/oraInst.oc 从源 Oracle 主机复制到目标主机

TR-4794：《基于NetApp EF系列的Oracle数据库》

NetApp公司Ebin Kadavy的Mitch Blackburn

TR-4794旨在帮助存储管理员和数据库管理员在NetApp EF系列存储上成功部署Oracle。

"TR-4794：《基于NetApp EF系列的Oracle数据库》"

版权信息

版权所有 © 2024 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。 美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 http://www.netapp.com/TM 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。