TR-4979: Simplified, Self-managed Oracle in VMware Cloud on AWS with guest-mounted FSx ONTAP

NetApp database solutions

NetApp
August 11, 2025

Table of Contents

TR-4979: Simplified, Self-managed Oracle in VMware Cloud on AWS with guest-mounted FSx ONTAP

Allen Cao, Niyaz Mohamed, NetApp

This solution provides overview and details for Oracle deployment and protection in VMware Cloud in AWS with FSx ONTAP as primary database storage and Oracle database configured in standalone ReStart using asm as volume manager.

Purpose

Enterprises have been running Oracle on VMware in private data centers for decades. VMware Cloud (VMC) on AWS provides a push-button solution to bring VMware's enterprise-class Software-Defined Data Center (SDDC) software to the AWS Cloud's dedicated, elastic, bare-metal infrastructure. AWS FSx ONTAP offers premium storage to VMC SDDC and a data fabric that enables customers to run business-critical applications such as Oracle across vSphere-based private, public, and hybrid cloud environments, with optimized access to AWS services. Whether it is an existing or new Oracle workload, VMC on AWS provides a familiar, simplified, and self-managed Oracle environment on VMware with all the benefits of AWS cloud while deferring all platform management and optimization to VMware.

This documentation demonstrates the deployment and protection of an Oracle database in a VMC environment with Amazon FSx ONTAP as primary database storage. Oracle database can be deployed to VMC on FSx storage as direct VM guest-mounted LUNs or NFS-mounted VMware VMDK datastore disks. This technical report focuses on Oracle database deployment as direct guest-mounted FSx storage to VMs in the VMC cluster with the iSCSI protocol and Oracle ASM. We also demonstrate how to use the NetApp SnapCenter UI tool to backup, restore, and clone an Oracle database for dev/test or other use cases for storage-efficient database operation in the VMC on AWS.

This solution addresses the following use cases:

- Oracle database deployment in VMC on AWS with Amazon FSx ONTAP as primary database storage
- Oracle database backup and restore in VMC on AWS using NetApp SnapCenter tool
- Oracle database clone for dev/test or other use cases in VMC on AWS using NetApp SnapCenter tool

Audience

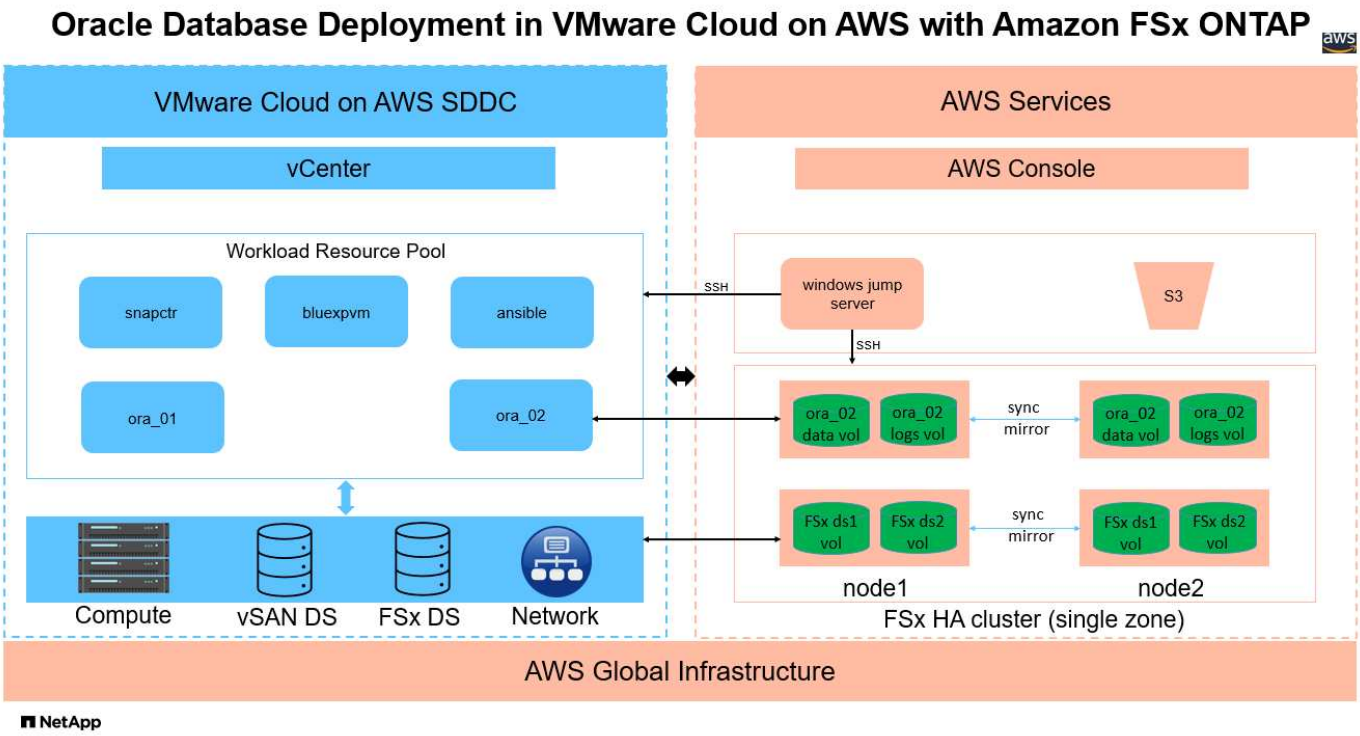
This solution is intended for the following people:

- A DBA who would like to deploy Oracle in VMC on AWS with Amazon FSx ONTAP
- A database solution architect who would like to test Oracle workloads in VMC on the AWS cloud
- A storage administrator who would like to deploy and manage an Oracle database deployed to VMC on AWS with Amazon FSx ONTAP
- An application owner who would like to stand up an Oracle database in VMC on the AWS cloud

Solution test and validation environment

The testing and validation of this solution was performed in a lab environment with VMC on AWS that might not match the final deployment environment. For more information, see the section Key factors for deployment consideration.

Architecture



Hardware and software components

Hardware		
FSx ONTAP storage	Current version offered by AWS	One FSx ONTAP HA cluster in the same VPC and availability zone as VMC
VMC SDDC cluster	Amazon EC2 i3.metal single node/Intel Xeon E5-2686 CPU,36 cores/512G RAM	10.37 TB vSAN storage
Software		
RedHat Linux	RHEL-8.6, 4.18.0-372.9.1.el8.x86_64 kernel	Deployed RedHat subscription for testing
Windows Server	2022 Standard, 10.0.20348 Build 20348	Hosting SnapCenter server
Oracle Grid Infrastructure	Version 19.18	Applied RU patch p34762026_190000_Linux-x86-64.zip

Oracle Database	Version 19.18	Applied RU patch p34765931_190000_Linux-x86-64.zip
Oracle OPatch	Version 12.2.0.1.36	Latest patch p6880880_190000_Linux-x86-64.zip
SnapCenter Server	Version 4.9P1	Workgroup deployment
BlueXP backup and recovery for VMs	Release 1.0	Deployed as an ova vSphere plugin VM
VMware vSphere	Version 8.0.1.00300	VMware Tools, Version: 11365 - Linux, 12352 - Windows
Open JDK	Version java-1.8.0-openjdk.x86_64	SnapCenter plugin requirement on DB VMs

Oracle database configuration in VMC on AWS

Server	Database	DB Storage
ora_01	cdb1(cdb1_pdb1,cdb1_pdb2,cdb1_pdb3)	VMDK datastore on FSx ONTAP
ora_01	cdb2(cdb2_pdb)	VMDK datastore on FSx ONTAP
ora_02	cdb3(cdb3_pdb1,cdb3_pdb2,cdb3_pdb3)	Direct guest mounted FSx ONTAP
ora_02	cdb4(cdb4_pdb)	Direct guest mounted FSx ONTAP

Key factors for deployment consideration

- **FSx to VMC connectivity.** When you deploy your SDDC on VMware Cloud on AWS, it is created within an AWS account and a VPC dedicated to your organization and managed by VMware. You must also connect the SDDC to an AWS account belonging to you, called the customer AWS account. This connection allows your SDDC to access AWS services belonging to your customer account. FSx ONTAP is an AWS service deployed in your customer account. Once the VMC SDDC is connected to your customer account, FSx storage is available to VMs in VMC SDDC for direct guest mount.
- **FSx storage HA clusters single- or multi-zone deployment.** In these tests and validations, we deployed an FSx HA cluster in a single AWS availability zone. NetApp also recommends deploying FSx ONTAP and VMware Cloud on AWS in the same availability zone to achieve better performance and avoid data transfer charges between availability zones.
- **FSx storage cluster sizing.** An Amazon FSx ONTAP storage file system provides up to 160,000 raw SSD IOPS, up to 4GBps throughput, and a maximum of 192TiB capacity. However, you can size the cluster in terms of provisioned IOPS, throughput, and storage limit (minimum 1,024 GiB) based on your actual requirements at the time of deployment. The capacity can be adjusted dynamically on the fly without affecting application availability.
- **Oracle data and logs layout.** In our tests and validations, we deployed two ASM disk groups for data and logs respectively. Within the +DATA asm disk group, we provisioned four LUNs in a data volume. Within the +LOGS asm disk group, we provisioned two LUNs in a log volume. In general, multiple LUNs laid out within an Amazon FSx ONTAP volume provide better performance.

- **iSCSI configuration.** The database VMs in VMC SDDC connect to FSx storage with the iSCSI protocol. It is important to gauge the Oracle database peak I/O throughput requirement by carefully analyzing the Oracle AWR report to determine the application and iSCSI traffic-throughput requirements. NetApp also recommends allocating four iSCSI connections to both FSx iSCSI endpoints with multipath properly configured.
- **Oracle ASM redundancy level to use for each Oracle ASM disk group that you create.** Because FSx ONTAP already mirrors the storage on the FSx cluster level, you should use External Redundancy, which means that the option does not allow Oracle ASM to mirror the contents of the disk group.
- **Database backup.** NetApp provides a SnapCenter software suite for database backup, restore, and cloning with a user-friendly UI interface. NetApp recommends implementing such a management tool to achieve fast (under a minute) SnapShot backup, quick (minutes) database restore, and database clone.

Solution deployment

The following sections provide step-by-step procedures for Oracle 19c deployment in VMC on AWS with directly mounted FSx ONTAP storage to DB VM in a single node Restart configuration with Oracle ASM as database volume manager.

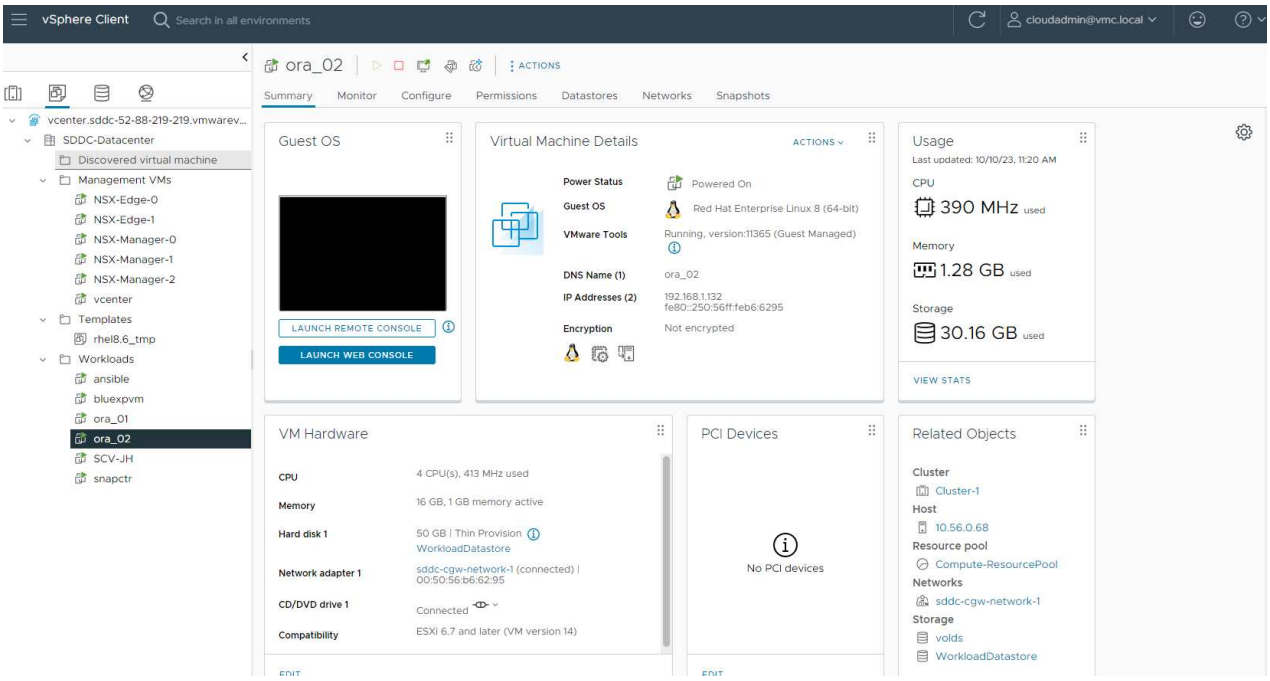
Prerequisites for deployment

Deployment requires the following prerequisites.

1. A software-defined data center (SDDC) using VMware Cloud on AWS has been created. For detailed instruction on how to create an SDDC in VMC, please refer to VMware documentation Getting Started With VMware Cloud on AWS
2. An AWS account has been set up, and the necessary VPC and network segments have been created within your AWS account. The AWS account is linked to your VMC SDDC.
3. From the AWS EC2 console, deploying an Amazon FSx ONTAP storage HA clusters to host the Oracle database volumes. If you are not familiar with the deployment of FSx storage, see the documentation Creating FSx ONTAP file systems for step-by-step instructions.
4. The above step can be performed using the following Terraform automation toolkit, which creates an EC2 instance as a jump host for SDDC in VMC access via SSH and an FSx file system. Review instructions carefully and change the variables to suit your environment before execution.

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

5. Build VMs in VMware SDDC on AWS for hosting your Oracle environment to be deployed in VMC. In our demonstration, we have built two Linux VMs as Oracle DB servers, one Windows server for the SnapCenter server, and one optional Linux server as an Ansible controller for automated Oracle installation or configuration if desired. Following is a snapshot of the lab environment for the solution validation.



6. Optionally, NetApp also provides several automation toolkits to run Oracle deployment and configuration when applicable.

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

DB VM kernel configuration

With the prerequisites provisioned, login to the Oracle VM as an admin user via SSH and sudo to the root user to configure the Linux kernel for Oracle installation. Oracle install files can be staged in an AWS S3 bucket and transferred into the VM.

1. Create a staging directory `/tmp/archive` folder and set the `777` permission.

```
mkdir /tmp/archive
```

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

2. Download and stage the Oracle binary installation files and other required rpm files to the `/tmp/archive` directory.

   See the following list of installation files to be stated in `/tmp/archive` on the DB VM.

```
[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. Install Oracle 19c preinstall RPM, which satisfies most kernel configuration requirements.

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

4. Download and install the missing `compat-libcap1` in Linux 8.

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

5. From NetApp, download and install NetApp host utilities.

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

6. Install `policycoreutils-python-utils`.

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

7. Install open JDK version 1.8.

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

8. Install iSCSI initiator utils.

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

9. Install sg3_utils.

```
yum install sg3_utils
```

10. Install device-mapper-multipath.

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

11. Disable transparent hugepages in the current system.

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

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

12. Add the following lines in `/etc/rc.local` to disable `transparent_hugepage` after reboot.

```
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. Disable selinux by changing `SELINUX=enforcing` to `SELINUX=disabled`. You must reboot the host to make the change effective.

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

14. Add the following lines to `limit.conf` to set the file descriptor limit and stack size.

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

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

15. Add swap space to DB VM if there is no swap space configured with this instruction: How do I allocate memory to work as swap space in an Amazon EC2 instance by using a swap file? The exact amount of space to add depends on the size of RAM up to 16G.
16. Change `node.session.timeo.replacement_timeout` in the `iscsi.conf` configuration file from 120 to 5 seconds.

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

17. Enable and start the iSCSI service on the EC2 instance.

```
systemctl enable iscsid
```

```
systemctl start iscsid
```

18. Retrieve the iSCSI initiator address to be used for database LUN mapping.

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

19. Add the asm groups for asm management user (oracle).

```
groupadd asmadmin
```

```
groupadd asmdba
```

```
groupadd asmoper
```

20. Modify the oracle user to add asm groups as secondary groups (the oracle user should have been created after Oracle preinstall RPM installation).

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

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

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

21. Stop and disable the Linux firewall if it is active.

```
systemctl stop firewalld
```

```
systemctl disable firewalld
```

22. Enable password-less sudo for admin user by uncommenting `# %wheel ALL=(ALL) NOPASSWD: ALL` line in /etc/sudoers file. Change the file permission to make the edit.

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

```
vi /etc/sudoers
```

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

23. Reboot the EC2 instance.

Provision and map FSx ONTAP LUNs to the DB VM

Provision three volumes from the command line by login to FSx cluster as fsxadmin user via ssh and FSx cluster management IP. Create LUNs within the volumes to host the Oracle database binary, data, and logs files.

1. Log into the FSx cluster through SSH as the fsxadmin user.

```
ssh fsxadmin@10.49.0.74
```

2. Execute the following command to create a volume for the Oracle binary.

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

3. Execute the following command to create a volume for Oracle data.

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

4. Execute the following command to create a volume for Oracle logs.

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

5. Validate the volumes created.

```
vol show ora*
```

Output from the command:

```
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. Create a binary LUN within the database binary volume.

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

7. Create data LUNs within the database data volume.

```
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. Create log LUNs within the database logs volume.

```
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. Create an igroup for the EC2 instance with the initiator retrieved from step 14 of the EC2 kernel configuration above.

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

10. Map the LUNs to the igroup created above. Increment the LUN ID sequentially for each additional LUN.

```
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. Validate the LUN mapping.

```
mapping show
```

This is expected to return:

```
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
```

DB VM storage configuration

Now, import and set up the FSx ONTAP storage for the Oracle grid infrastructure and database installation on the VMC database VM.

1. Login to the DB VM via SSH as the admin user using Putty from Windows jump server.
2. Discover the FSx iSCSI endpoints using either SVM iSCSI IP address. Change to your environment-specific portal address.

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

3. Establish iSCSI sessions by logging into each target.

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

The expected output from the command is:

```
[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. View and validate a list of active iSCSI sessions.

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

Return the iSCSI sessions.

```
[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. Verify that the LUNs were imported into the host.

```
sudo sanlun lun show
```

This will return a list of Oracle LUNs from FSx.

```
[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. Configure the `multipath.conf` file with following default and blacklist entries.

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

Add following entries:

```
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. Start the multipath service.

```
sudo systemctl start multipathd
```

Now multipath devices appear in the `/dev/mapper` directory.

```
[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. Log into the FSx ONTAP cluster as the fsxadmin user via SSH to retrieve the serial-hex number for each LUN starting with 6c574xxx…, the HEX number starts with 3600a0980, which is the AWS vendor ID.

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

and return as follow:

```
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. Update the `/dev/multipath.conf` file to add a user-friendly name for the multipath device.

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

with following entries:

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

10. Reboot the multipath service to verify that the devices under `/dev/mapper` have changed to LUN names versus serial-hex IDs.

```
sudo systemctl restart multipathd
```

Check `/dev/mapper` to return as following:

```
[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. Partition the binary LUN with a single primary partition.

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

12. Format the partitioned binary LUN with an XFS file system.

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

13. Mount the binary LUN to `/u01`.

```
sudo mkdir /u01
```

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

14. Change `/u01` mount point ownership to the oracle user and it's associated primary group.

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

15. Find the UUI of the binary LUN.

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

16. Add a mount point to `/etc/fstab`.

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

Add the following line.

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

17. As the root user, add the udev rule for Oracle devices.

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

Include following entries:

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

18. As the root user, reload the udev rules.

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

19. As the root user, trigger the udev rules.

```
udevadm trigger
```

20. As the root user, reload multipathd.

```
systemctl restart multipathd
```

21. Reboot the EC2 instance host.

Oracle grid infrastructure installation

1. Log into the DB VM as the admin user via SSH and enable password authentication by uncommenting `PasswordAuthentication yes` and then commenting out `PasswordAuthentication no`.

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

2. Restart the sshd service.

```
sudo systemctl restart sshd
```

3. Reset the Oracle user password.

```
sudo passwd oracle
```

4. Log in as the Oracle Restart software owner user (oracle). Create an Oracle directory as follows:

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

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

5. Change the directory permission setting.

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

6. Create a grid home directory and change to it.

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

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

7. Unzip the grid installation files.

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

8. From grid home, delete the `OPatch` directory.

```
rm -rf OPatch
```

9. From grid home, unzip `p6880880_190000_Linux-x86-64.zip`.

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

10. From grid home, revise `cv/admin/cvu_config`, uncomment and replace `CV_ASSUME_DISTID=OEL5` with `CV_ASSUME_DISTID=OL7`.

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

11. Prepare a `gridsetup.rsp` file for silent installation and place the rsp file in the `/tmp/archive` directory. The rsp file should cover sections A, B, and G with the following information:

```
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. Log into the EC2 instance as the root user and set `ORACLE_HOME` and `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. Initialize disk devices for use with the Oracle ASM filter driver.

```
./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. Install `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. Unset `$ORACLE_BASE`.

```
unset ORACLE_BASE
```

16. Log into the EC2 instance as the Oracle user and extract the patch in the `/tmp/archive` folder.

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

17. From grid home /u01/app/oracle/product/19.0.0/grid and as the oracle user, launch `gridSetup.sh` for grid infrastructure installation.

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

18. As root user, execute the following script(s):

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

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

19. As root user, reload the multipathd.

```
systemctl restart multipathd
```

20. As the Oracle user, execute the following command to complete the configuration:

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

21. As the Oracle user, create the LOGS disk group.

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

22. As the Oracle user, validate grid services after installation configuration.

```
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 filter driver status.

```
[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. Validate HA service status.

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

Oracle database installation

1. Log in as the Oracle user and unset `$ORACLE_HOME` and `$ORACLE_SID` if it is set.

```
unset ORACLE_HOME
```

```
unset ORACLE_SID
```

2. Create the Oracle DB home directory and change the directory to it.

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

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

3. Unzip the Oracle DB installation files.

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

4. From the DB home, delete the `OPatch` directory.

```
rm -rf OPatch
```

5. From DB home, unzip `p6880880_190000_Linux-x86-64.zip`.

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

6. From DB home, revise `cv/admin/cvu_config` and uncomment and replace `CV_ASSUME_DISTID=OEL5` with `CV_ASSUME_DISTID=OL7`.

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

7. From the `/tmp/archive` directory, unpack the DB 19.18 RU patch.

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

8. Prepare the DB silent install rsp file in `/tmp/archive/dbinstall.rsp` directory with the following values:

```
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. From cdb3 home /u01/app/oracle/product/19.0.0/cdb3, execute silent software-only DB installation.

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

10. As root user, run the `root.sh` script after software-only installation.

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

11. As oracle user, create the `dbca.rsp` file with the following entries:

```
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. As oracle user, launch DB creation with dbca.

```
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/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. Repeat the same procedures from step 2 to create a container database cdb4 in a separate ORACLE_HOME /u01/app/oracle/product/19.0.0/cdb4 with a single PDB.
2. As Oracle user, validate Oracle Restart HA services after DB creation that all databases (cdb3, cdb4) are registered with HA services.

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

   output:

   ```
   [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. Set the Oracle user `.bash_profile`.

```
vi ~/.bash_profile
```

Add following entries:

```
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. Validate the CDB/PDB created for cdb3.

```
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. Validate the CDB/PDB created for cdb4.

```
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. Login to each cdb as sysdba with sqlplus and set the DB recovery destination size to the +LOGS disk group size for both cdbs.

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

7. Login to each cdb as sysdba with sqlplus and enable archive log mode with following command sets in sequence.

```
sqlplus /as sysdba
```

```
shutdown immediate;
```

```
startup mount;
```

```
alter database archivelog;
```

```
alter database open;
```

This completes Oracle 19c version 19.18 Restart deployment on an Amazon FSx ONTAP storage and a VMC DB VM. If desired, NetApp recommends relocating the Oracle control file and online log files to the +LOGS disk group.

Oracle backup, restore, and clone with SnapCenter

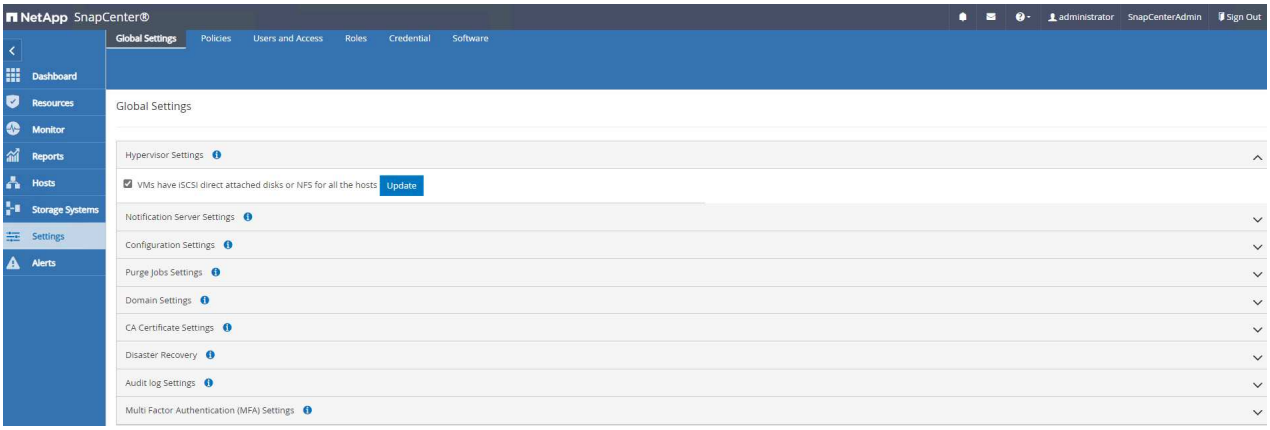
SnapCenter Setup

SnapCenter relies on a host-side plug-in on database VM to perform application-aware data protection management activities. For detailed information on NetApp SnapCenter plugin for Oracle, refer to this documentation What can you do with the Plug-in for Oracle Database. The following provides high level steps to setup SnapCenter for Oracle database backup, recovery, and clone.

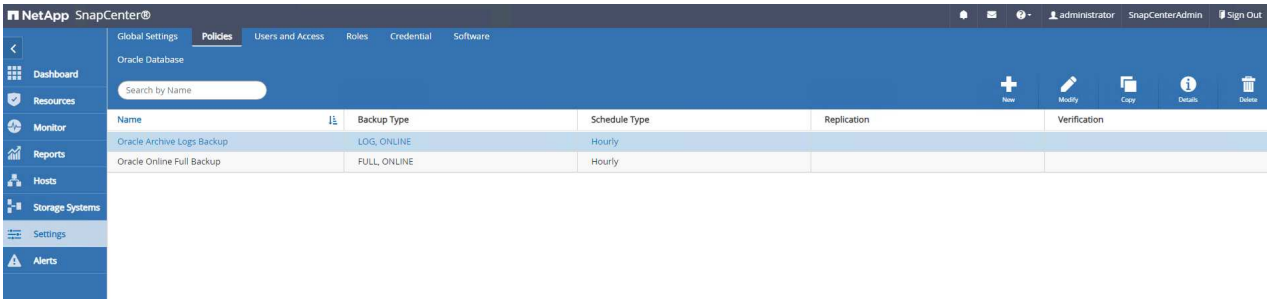
1. Download the latest version of SnapCenter software from NetApp support site: NetApp Support Downloads.
2. As administrator, install latest java JDK from Get Java for desktop applications on SnapCenter server Windows host.

> If Windows server is deployed in a domain environment, add a domain user to SnapCenter server local administrators group and run SnapCenter installation with the domain user.

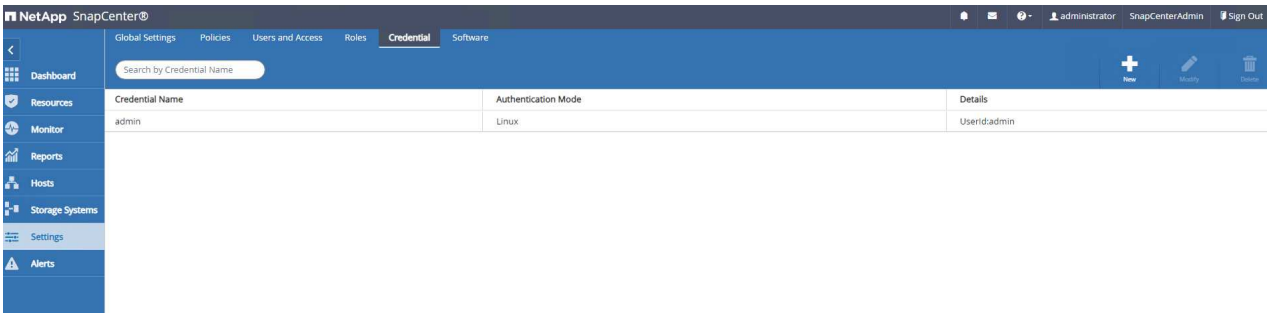
3. Login to SnapCenter UI via HTTPS port 8846 as installation user to configure SnapCenter for Oracle.
4. Update `Hypervisor Settings` in global settings.



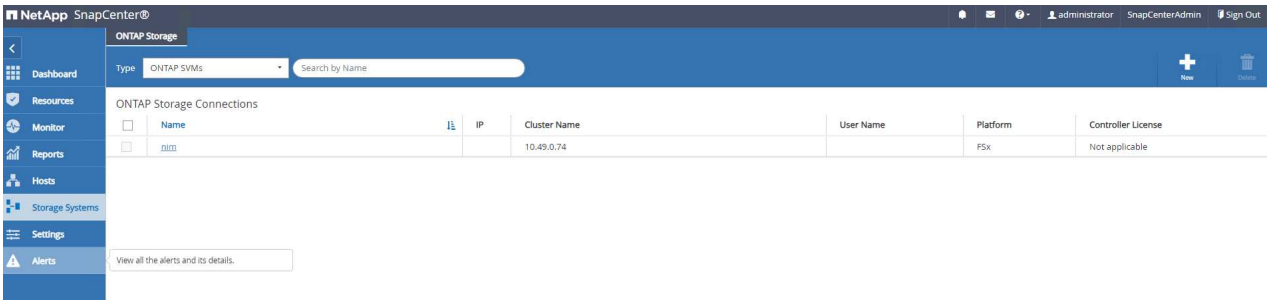
5. Create Oracle database backup policies. Ideally, create a separate archive log backup policy to allow more frequent backup interval to minimize data loss in the event of a failure.



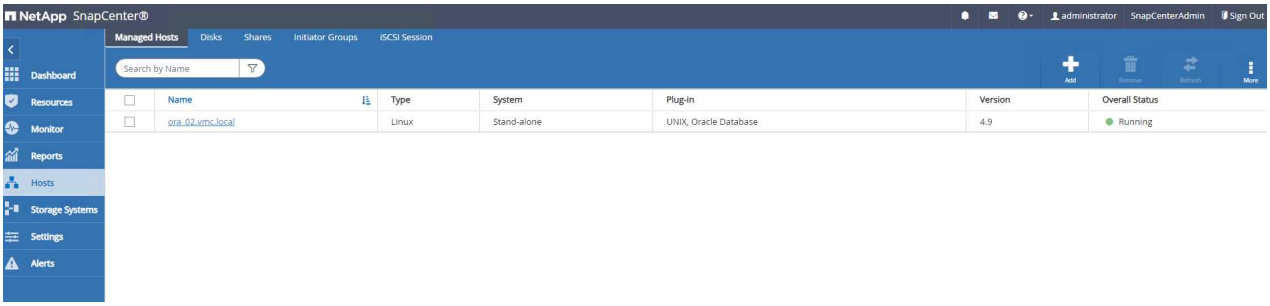
6. Add database server `Credential` for SnapCenter access to DB VM. The credential should have sudo privilege on a Linux VM or administrator privilege on a Windows VM.



7. Add FSx ONTAP storage cluster to `Storage Systems` with cluster management IP and authenticated via fsxadmin user ID.



8. Add Oracle database VM in VMC to `Hosts` with server credential created in previous step 6.

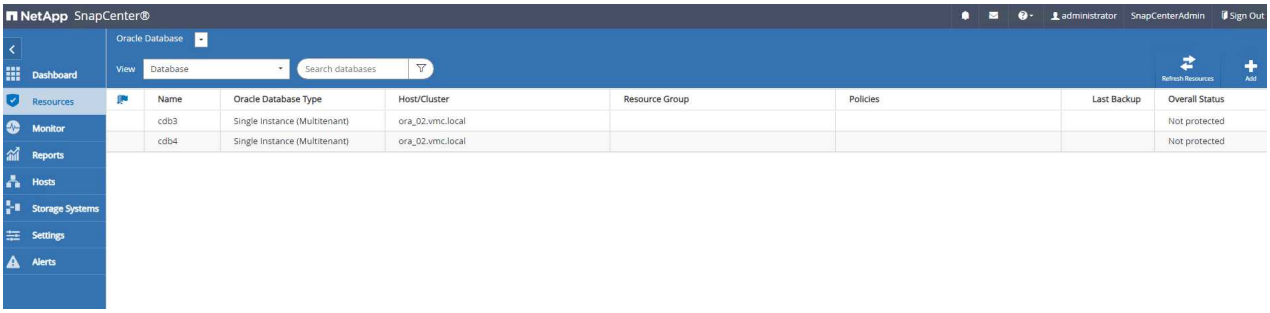


Ensure that the SnapCenter server name can be resolved to the IP address from the DB VM and DB VM name can be resolved to the IP address from the SnapCenter server.

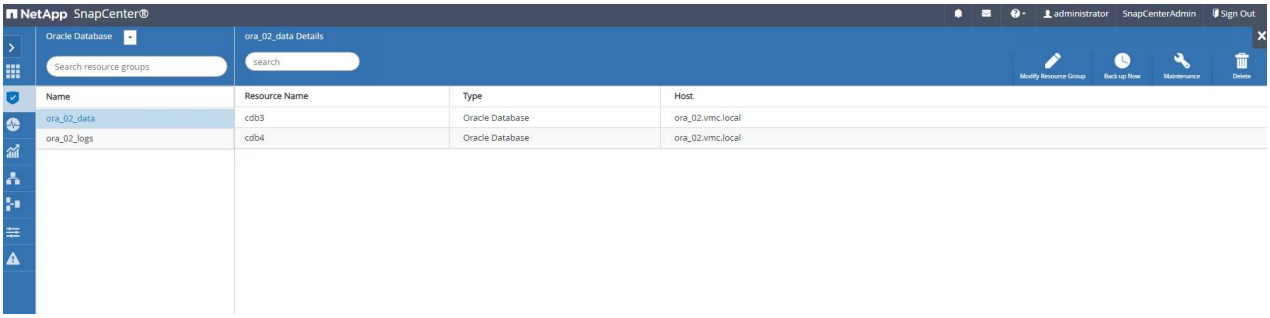
Database backup

SnapCenter leverages FSx ONTAP volume snapshot for much quicker database backup, restore, or clone compared with traditional RMAN based methodology. The snapshots are application-consistent as the database is put in Oracle backup mode before a snapshot.

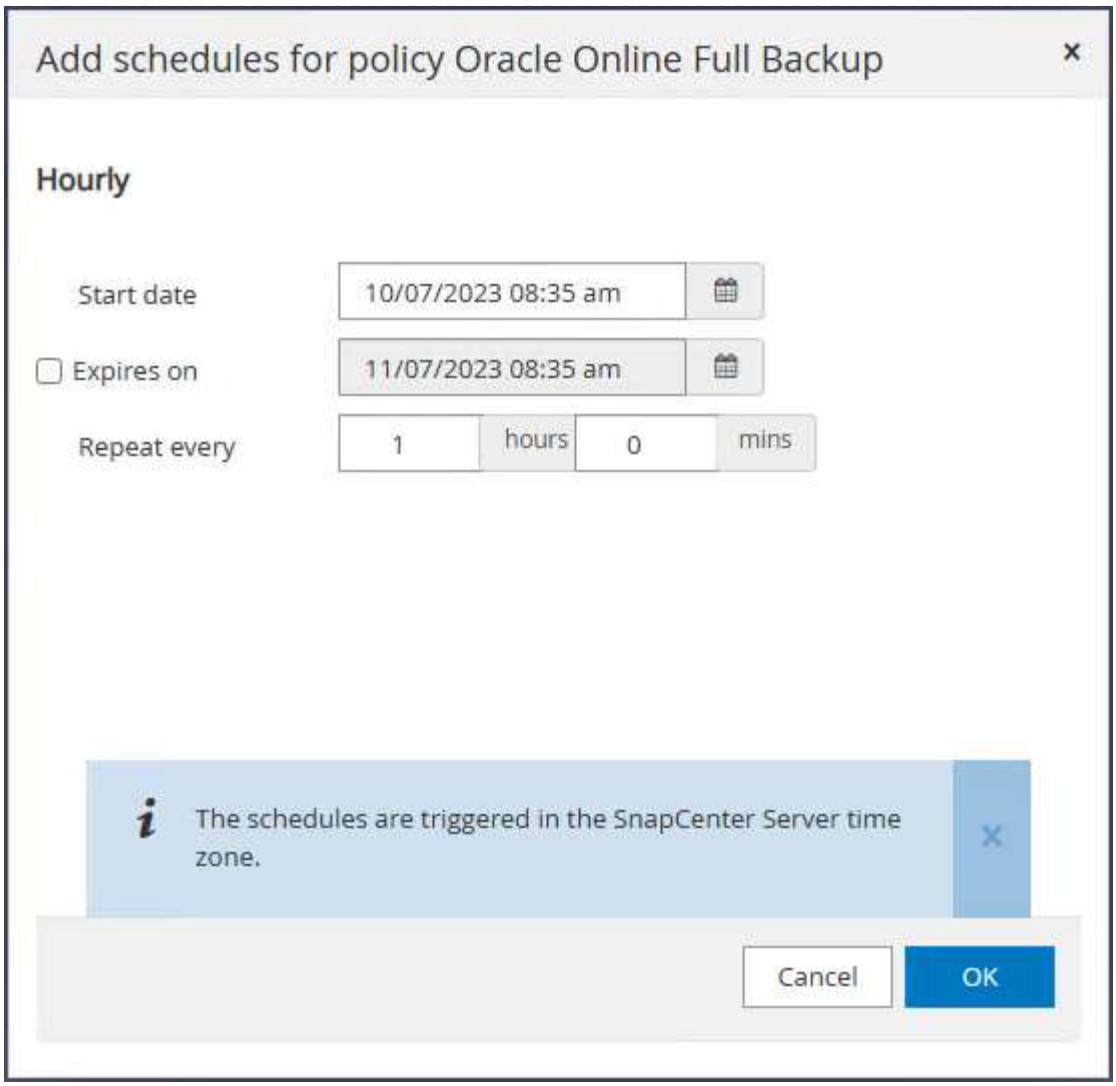
1. From the `Resources` tab, any databases on the VM are auto-discovered after the VM is added to SnapCenter. Initially, the database status shows as `Not protected`.



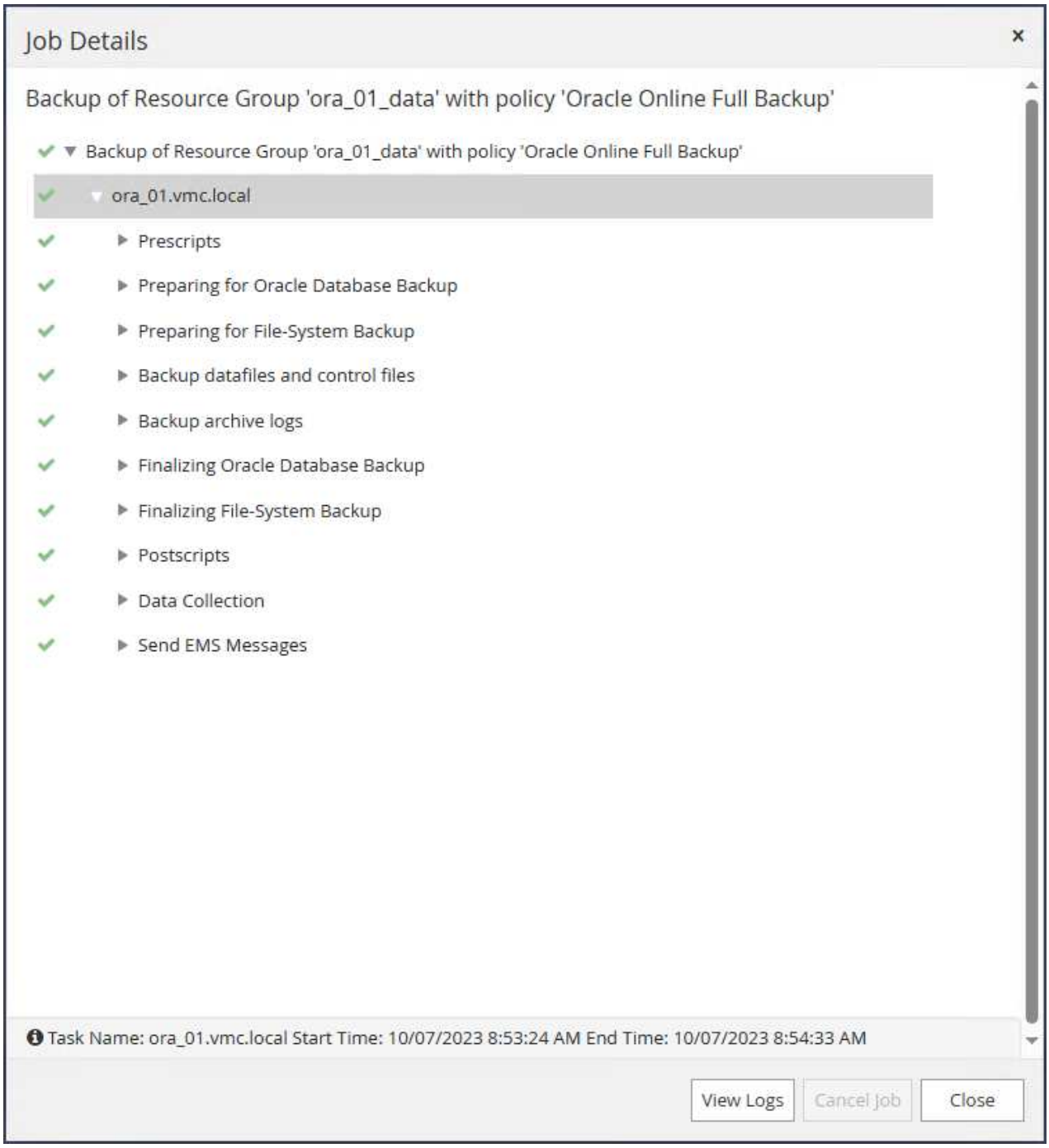
2. Create a resources group to backup the database in a logical grouping such as by DB VM etc. In this example, we created an ora_02_data group to do a full online database backup for all databases on VM ora_02. Resources group ora_02_log performs the backup of archived logs only on the VM. Creating a resources group also defines a schedule to execute the backup.



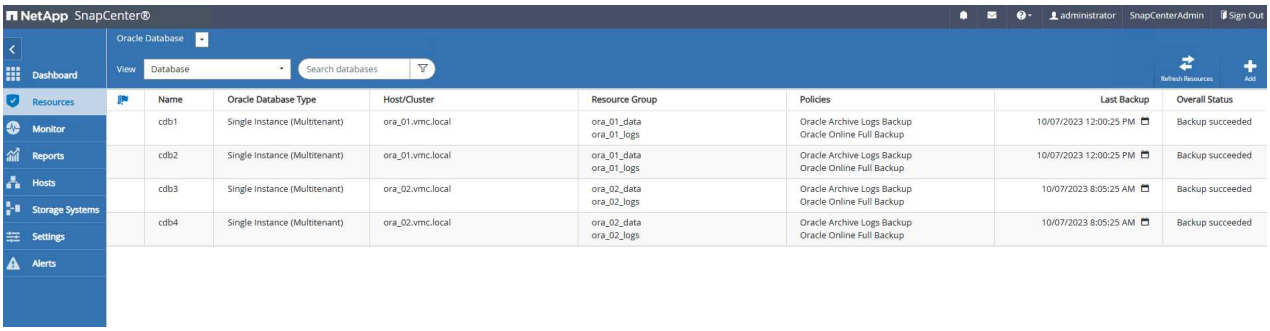
3. Resources group backup can also be triggered manually by clicking on `Back up Now` and executing the backup with the policy defined in the resources group.



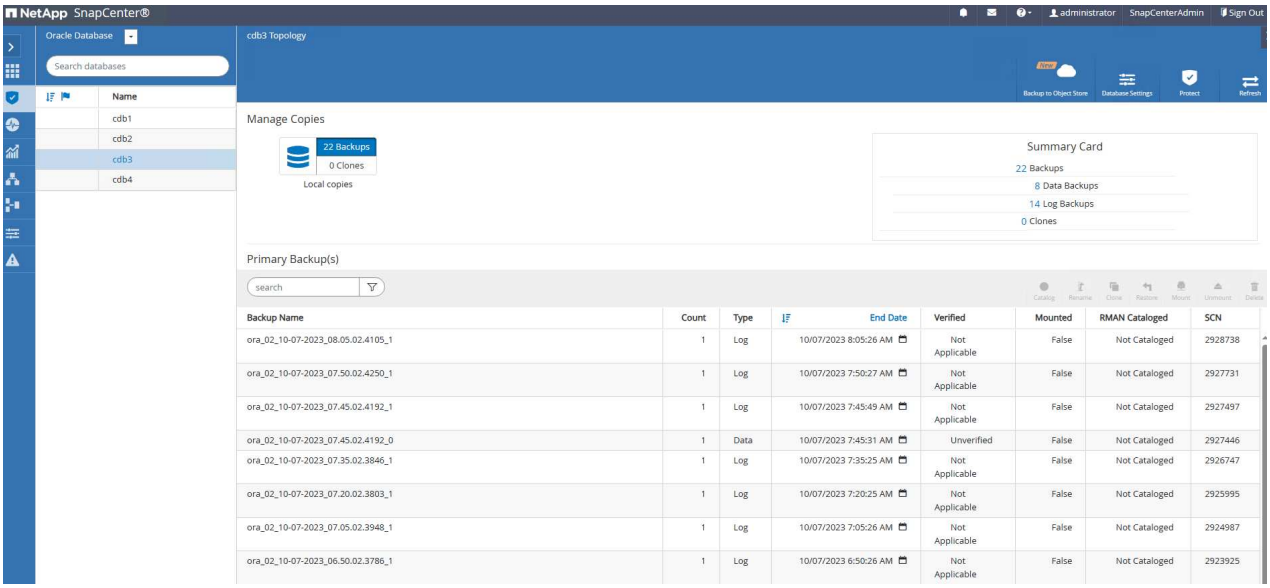
4. The backup job can be monitored at the `Monitor` tab by clicking on the running job.



5. After a successful backup, the database status shows the job status and the most recent backup time.



6. Click on database to review the backup sets for each database.



Database recovery

SnapCenter provides a number of restore and recovery options for Oracle databases from snapshot backup. In this example, we demonstrate a point in time restoration to recover a dropped table by mistake. On VM ora_02, two databases cdb3, cdb4 share the same +DATA and +LOGS disk groups. Database restoration for one database does not impact the availability of the other database.

1. First, create a test table and insert a row into table to validate a point in time recovery.

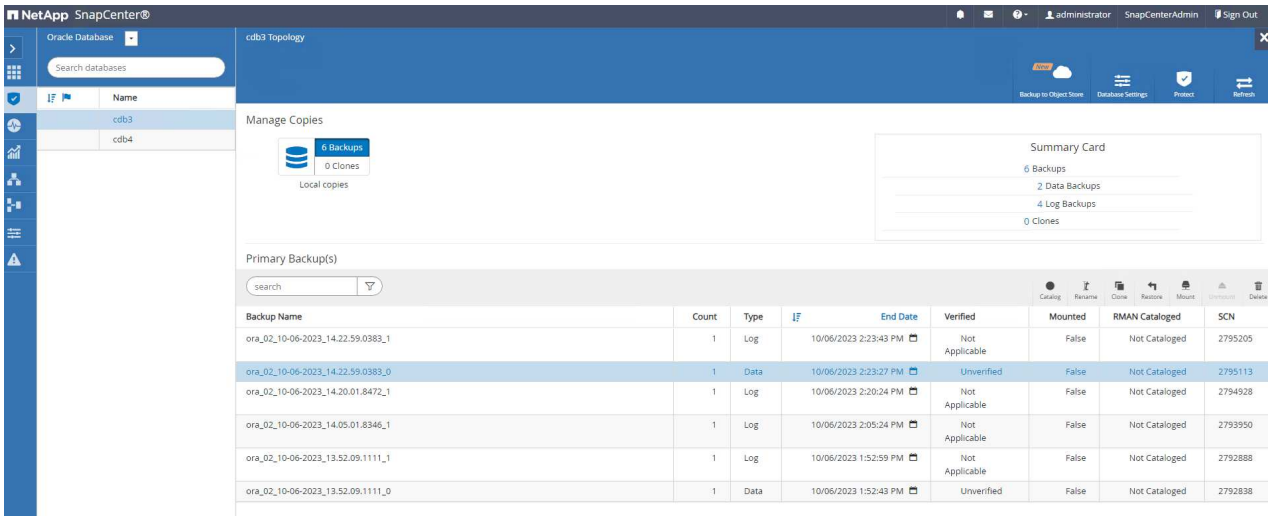
```
[oracle@ora_02 ~]$ sqlplus / as sysdba

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
```

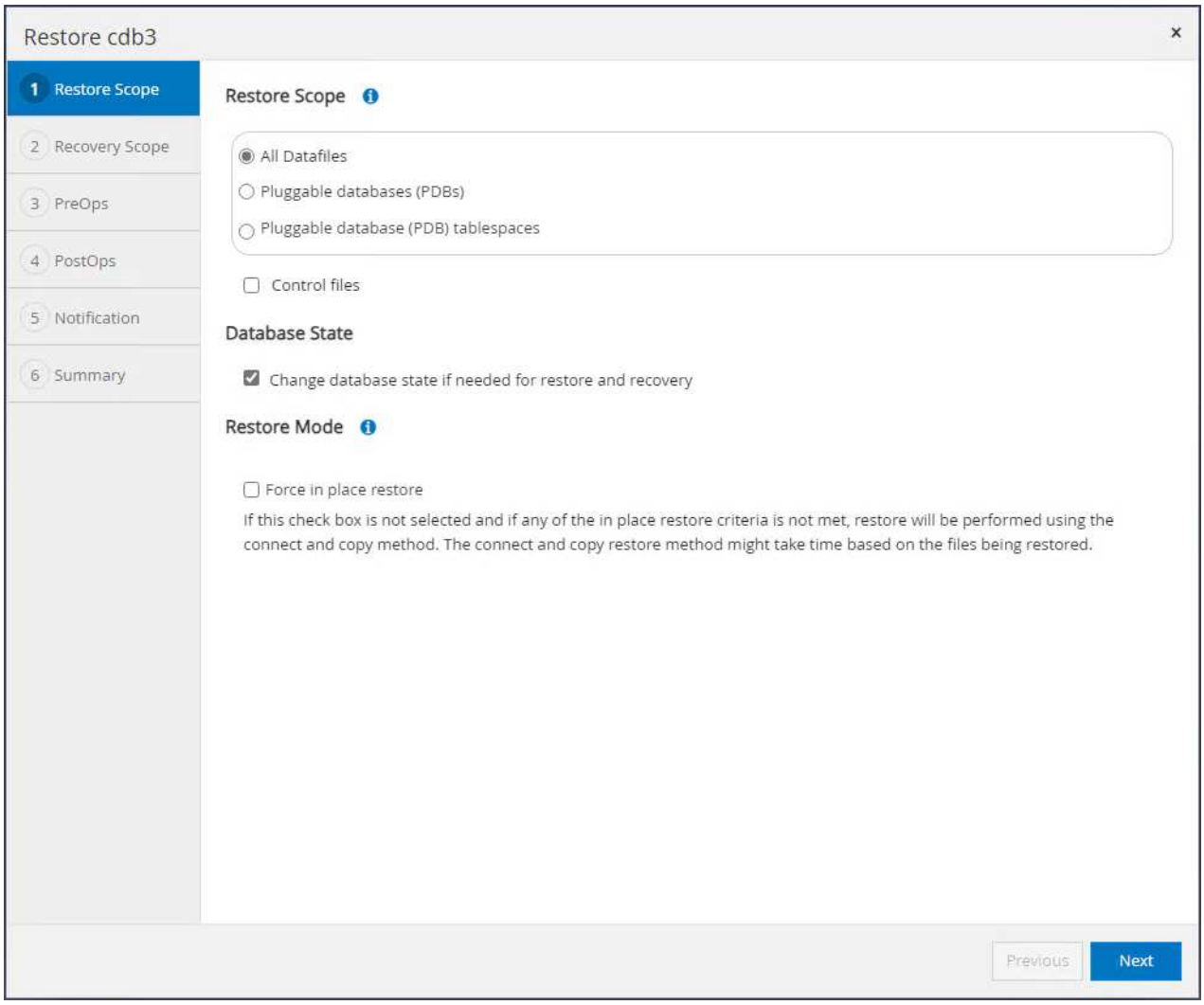
2. We run a manual snapshot backup from SnapCenter. Then drop the table.

```
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
```

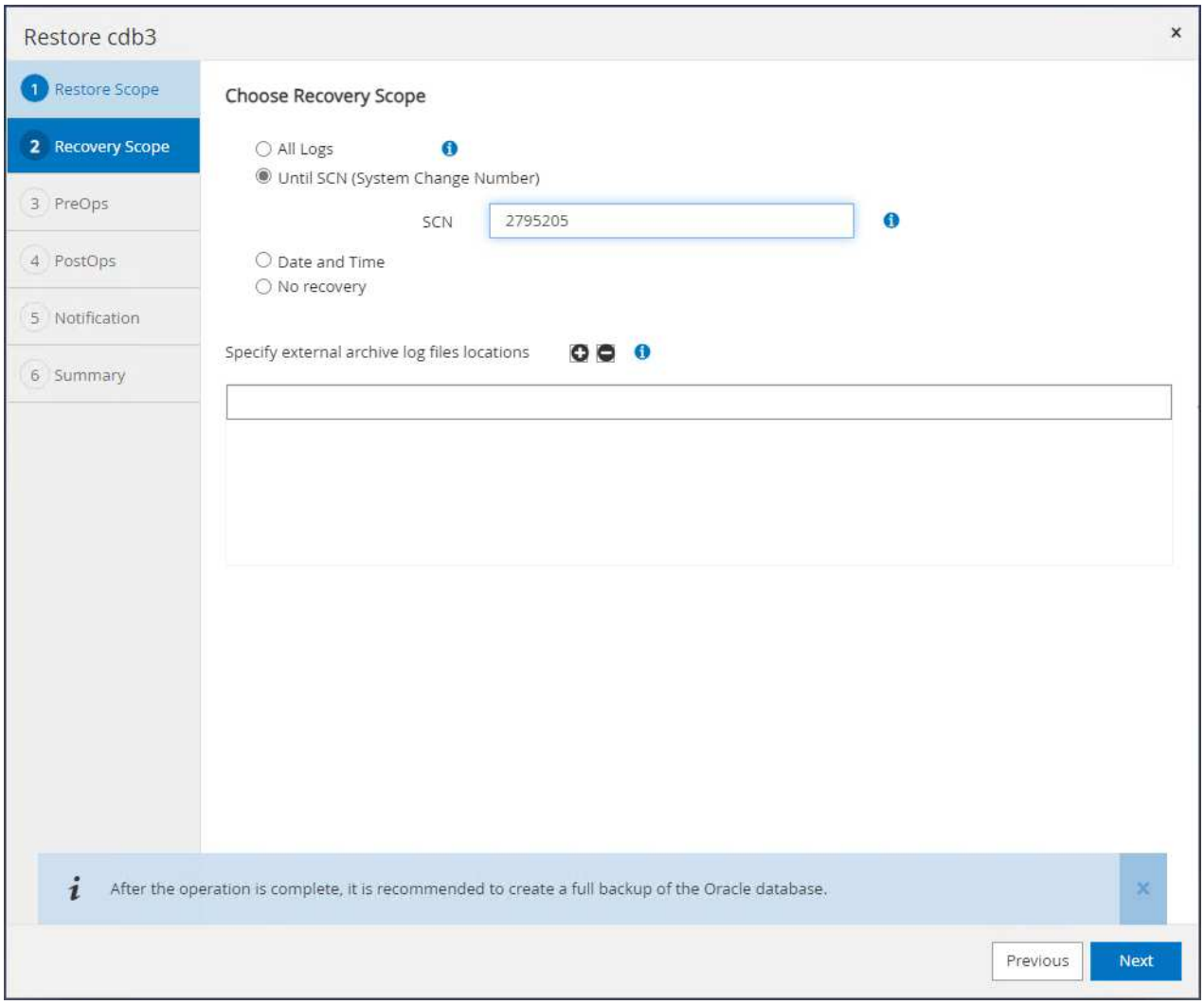
3. From backup set created from last step, take a note of the SCN number of log backup. Click on `Restore` to launch restore-recover workflow.



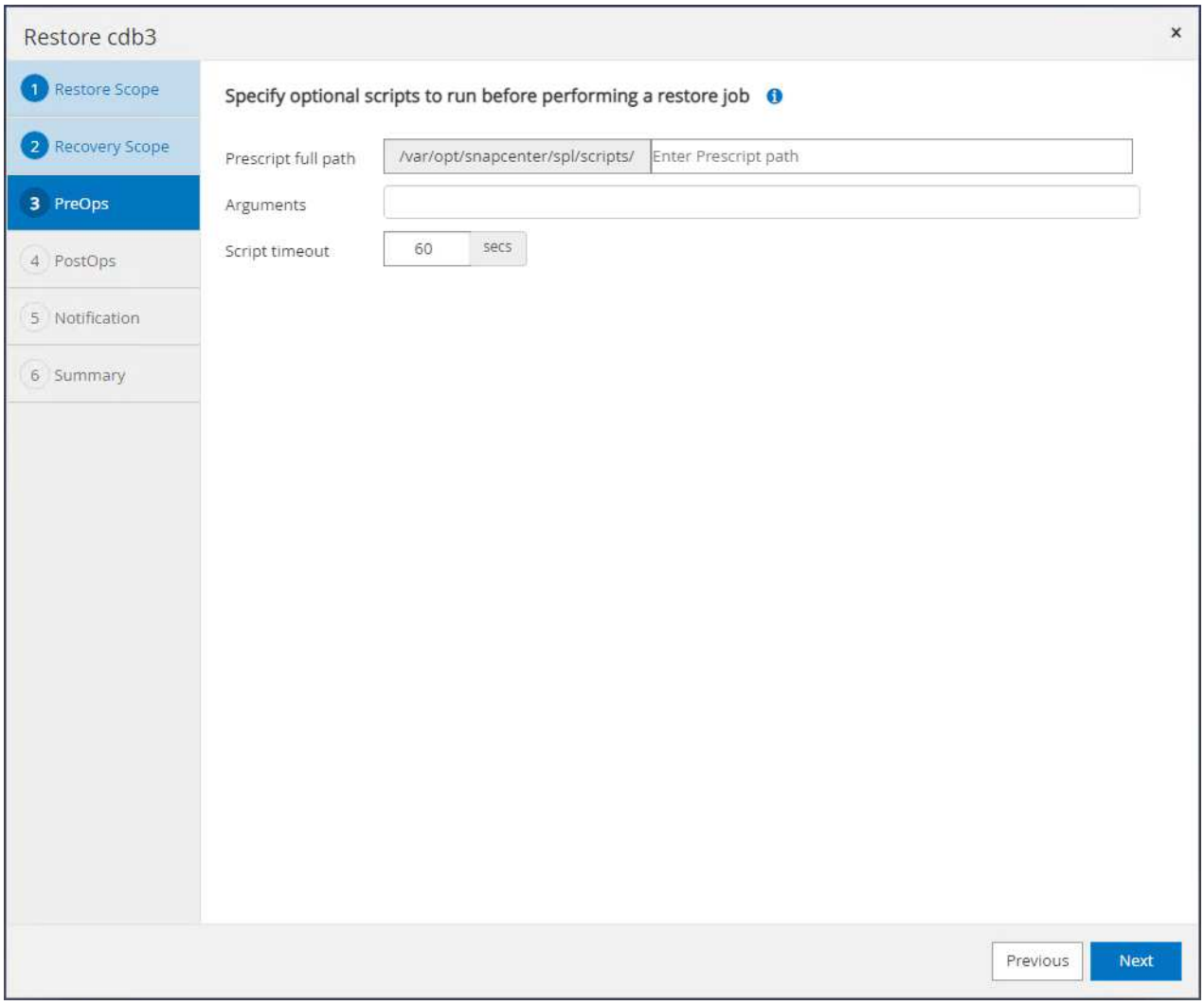
4. Choose restore scope.



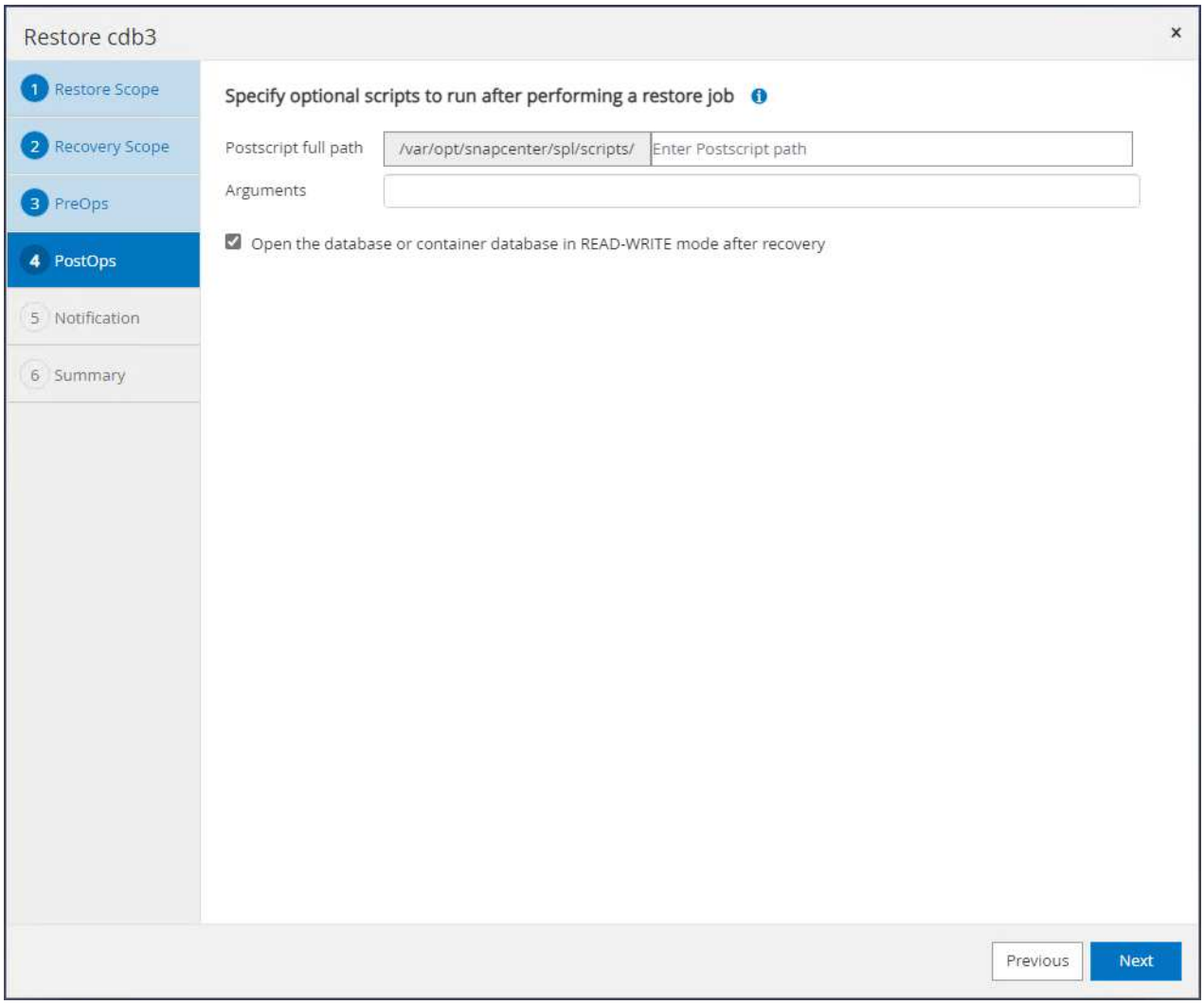
5. Choose recovery scope up to the log SCN from last full database backup.



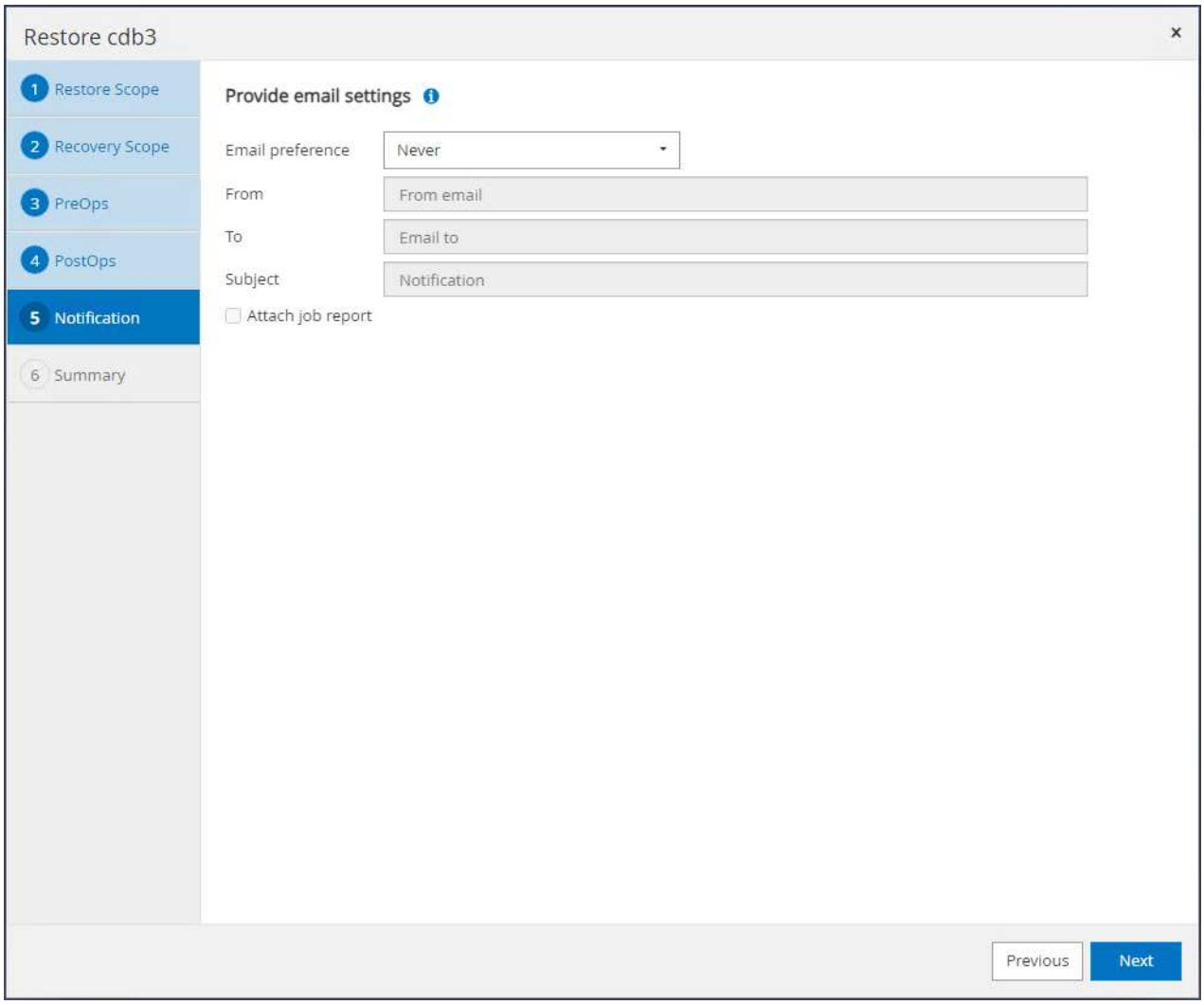
6. Specify any optional pre-scripts to run.



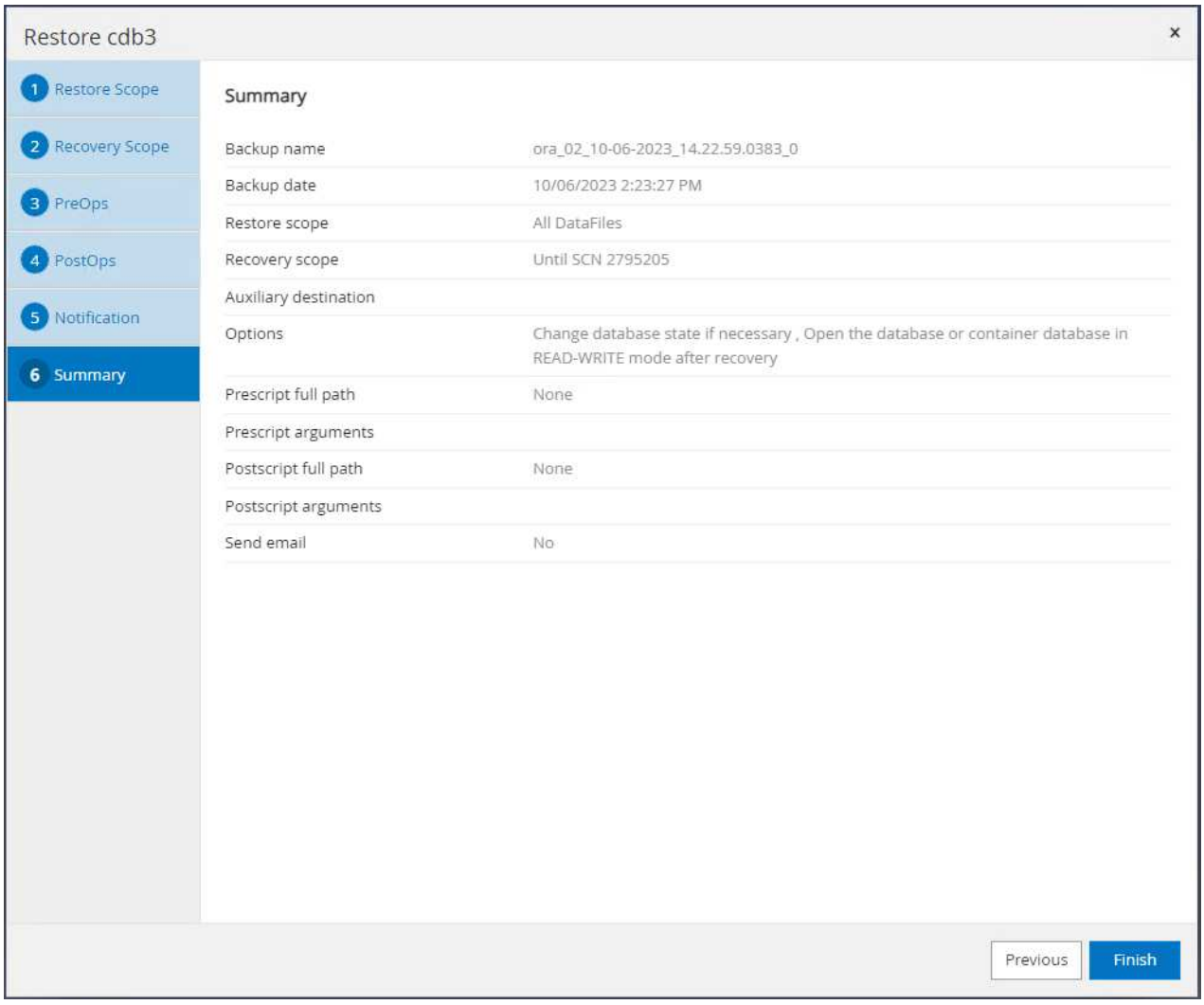
7. Specify any optional after-script to run.



8. Send a job report if desired.



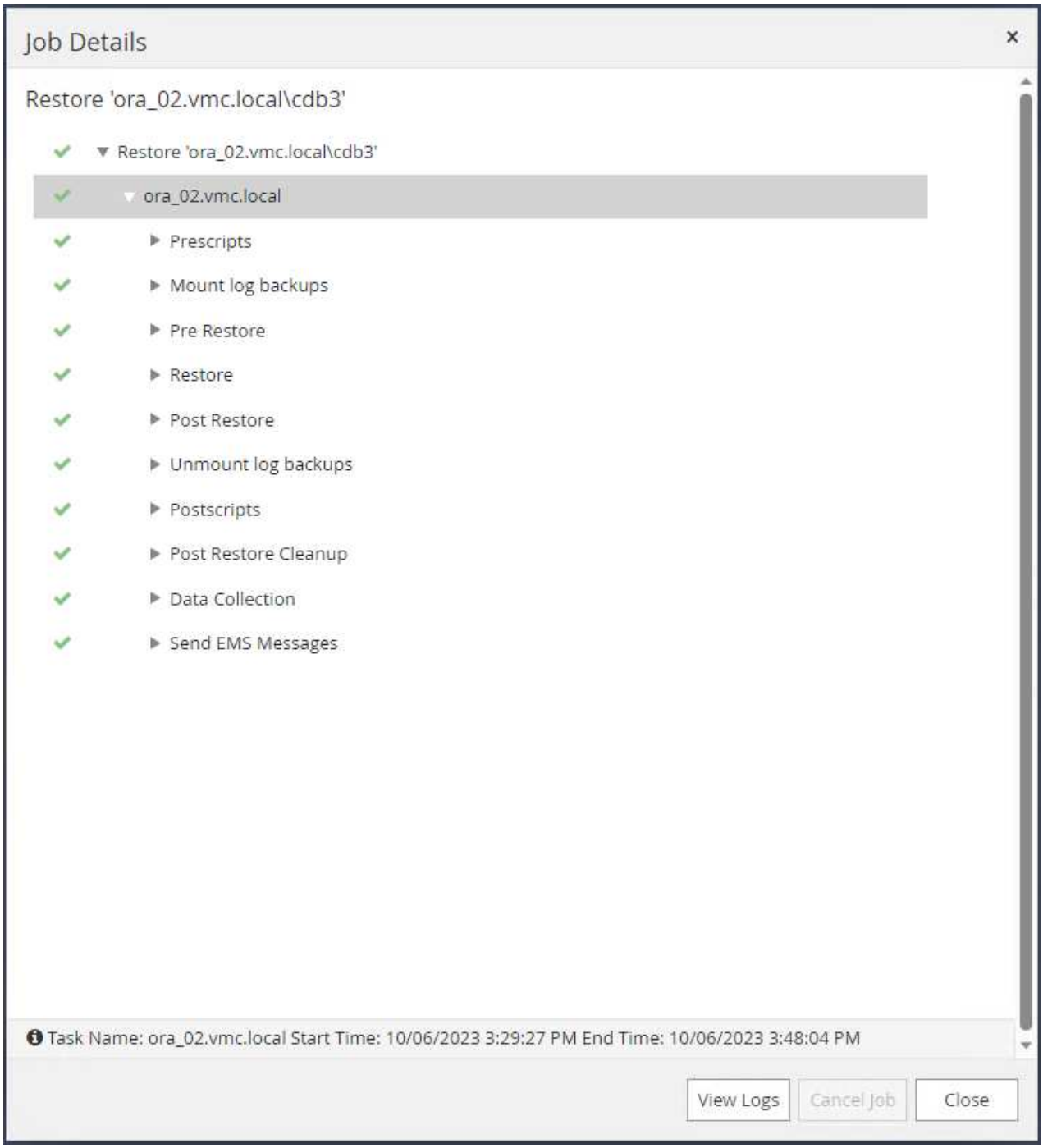
9. Review the summary and click on `Finish` to launch the restoration and recovery.



10. From Oracle Restart grid control, we observe that while cdb3 is under restoration and recovery cdb4 is online and available.

```
[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` tab, open the job to review the details.



12. From DB VM ora_02, validate the dropped table is recovered after a successful recovery.

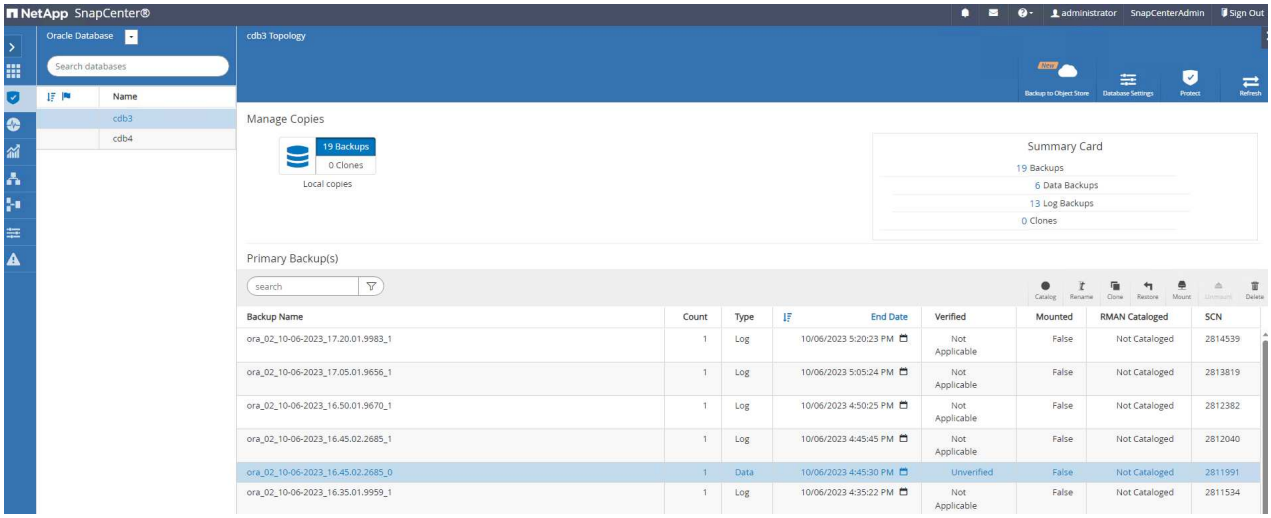
```
[oracle@ora_02 bin]$ sqlplus / as sysdba

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>
```

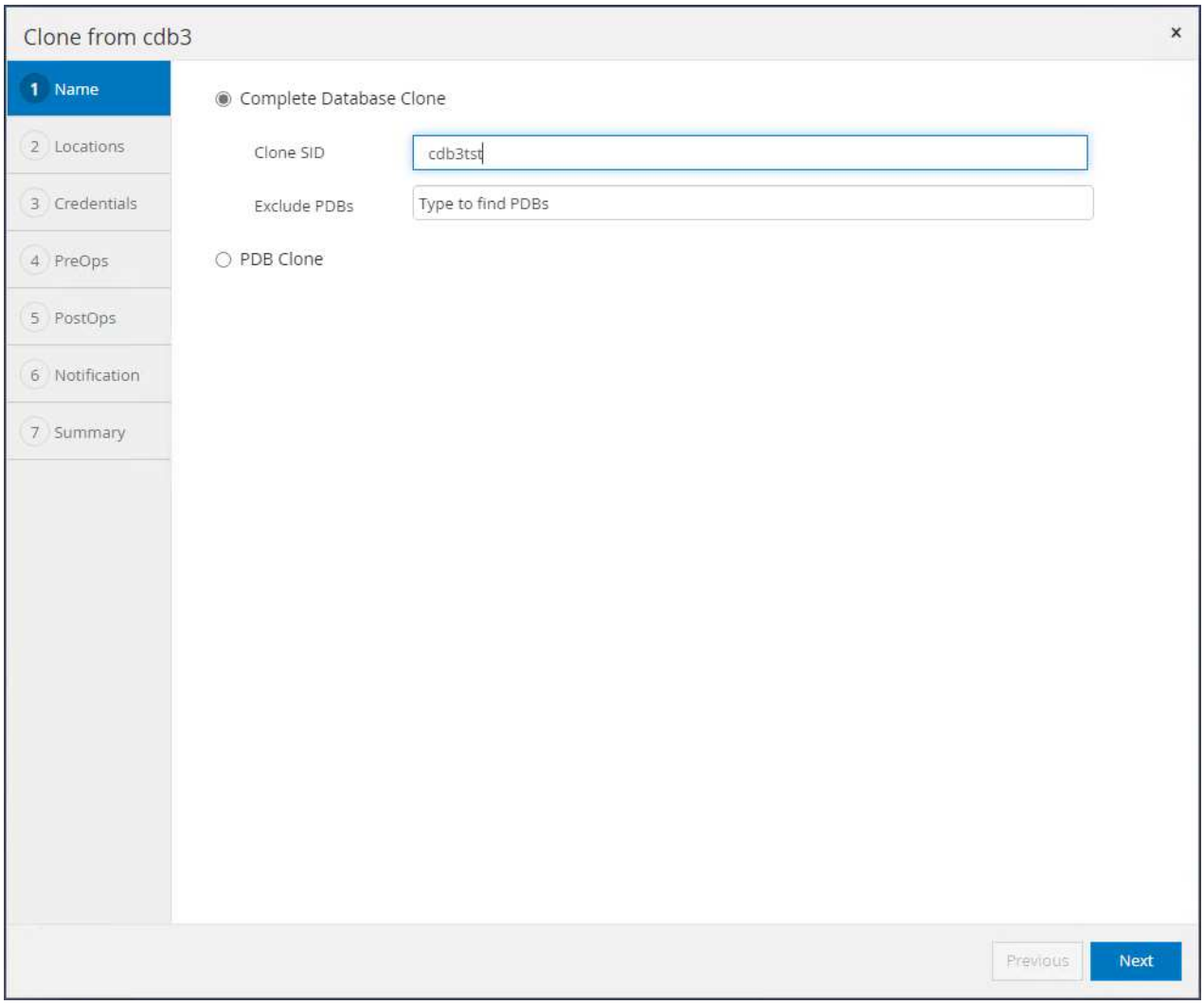
Database clone

In this example, the same backup sets is used to clone a database on the same VM in a different ORACLE_HOME. The procedures are equally applicable to clone a database from the backup to separate VM in VMC if needed.

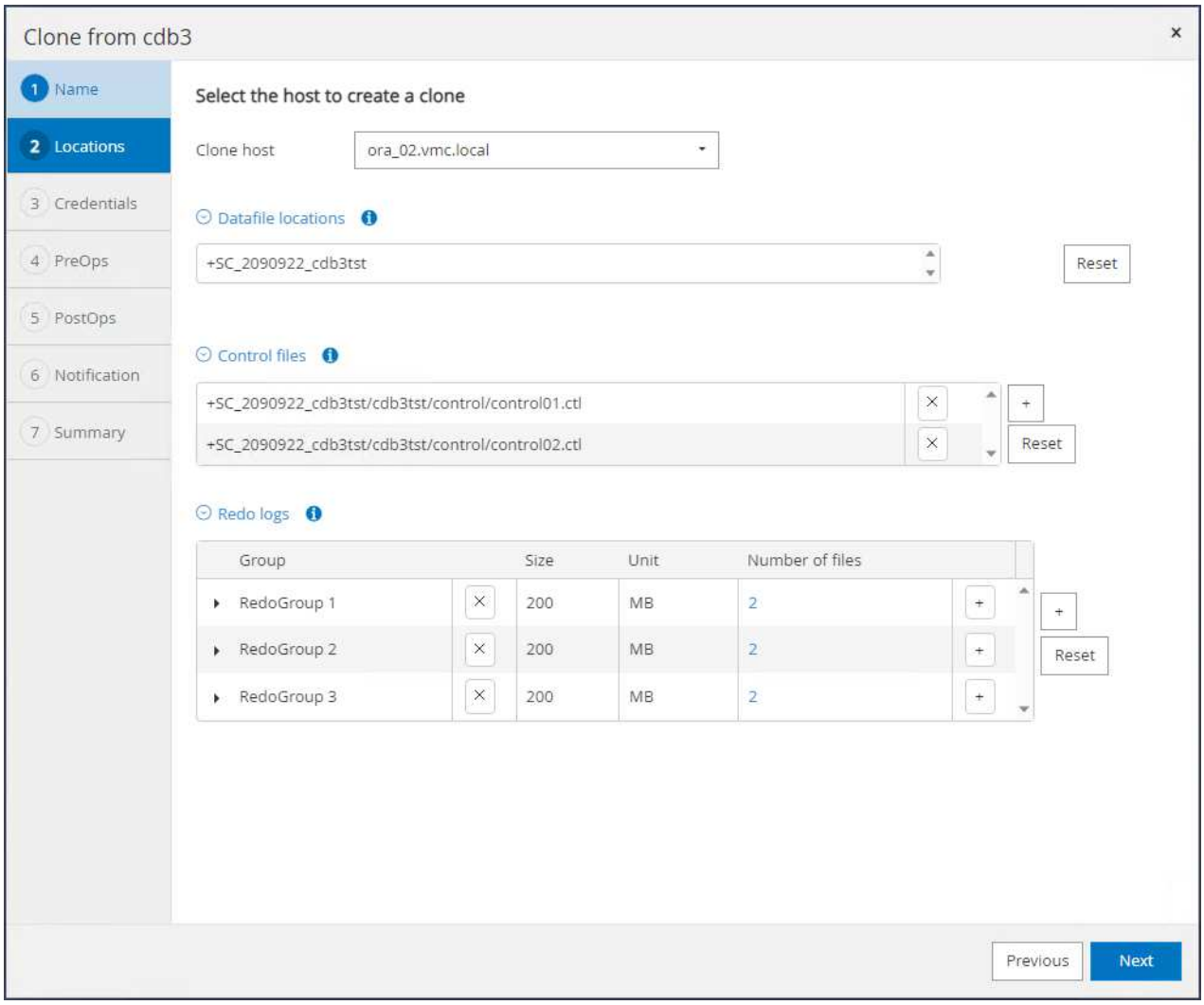
1. Open the database cdb3 backup list. From a data backup of choice, click on `Clone` button to launch database clone workflow.



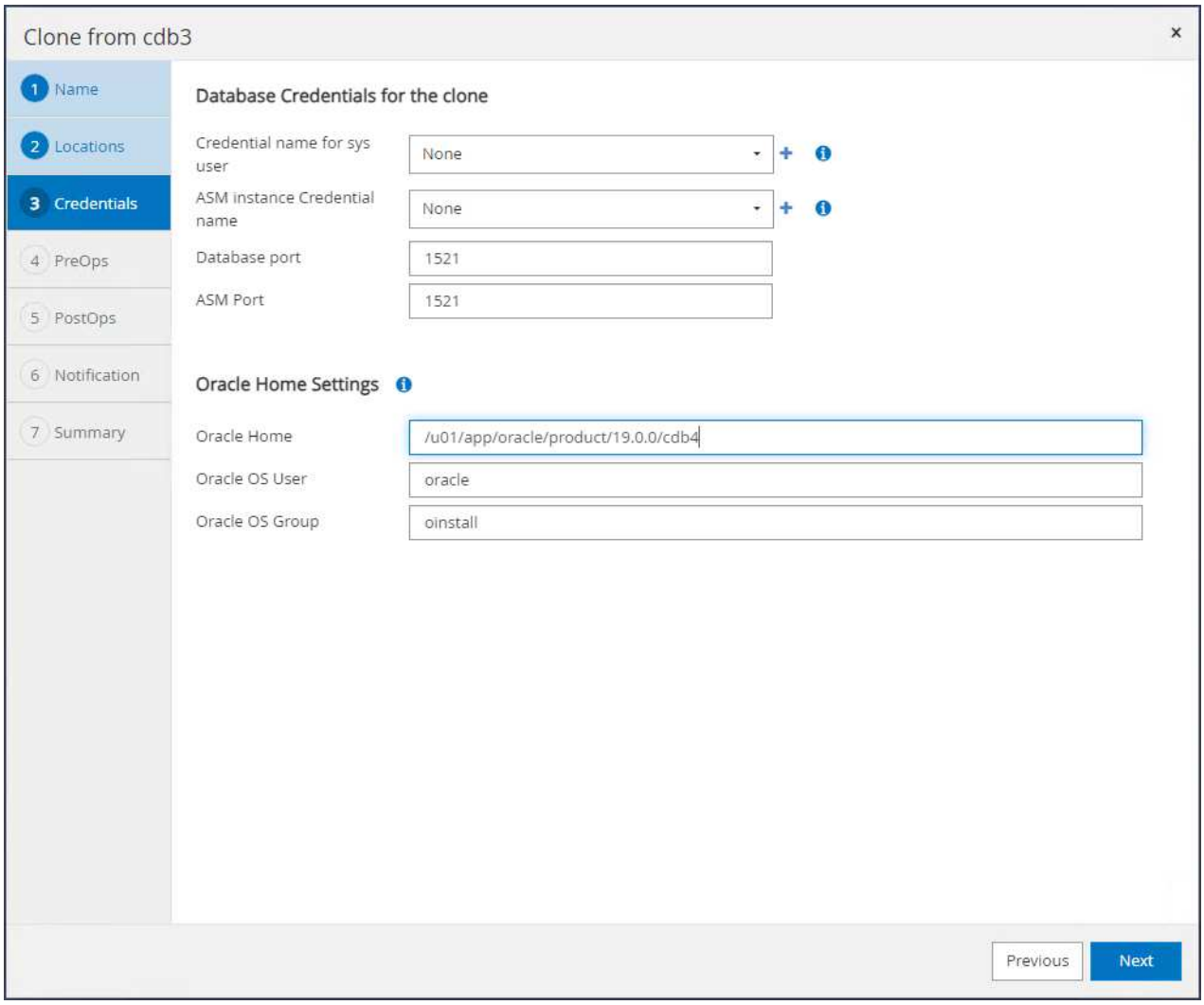
2. Name the clone database SID.



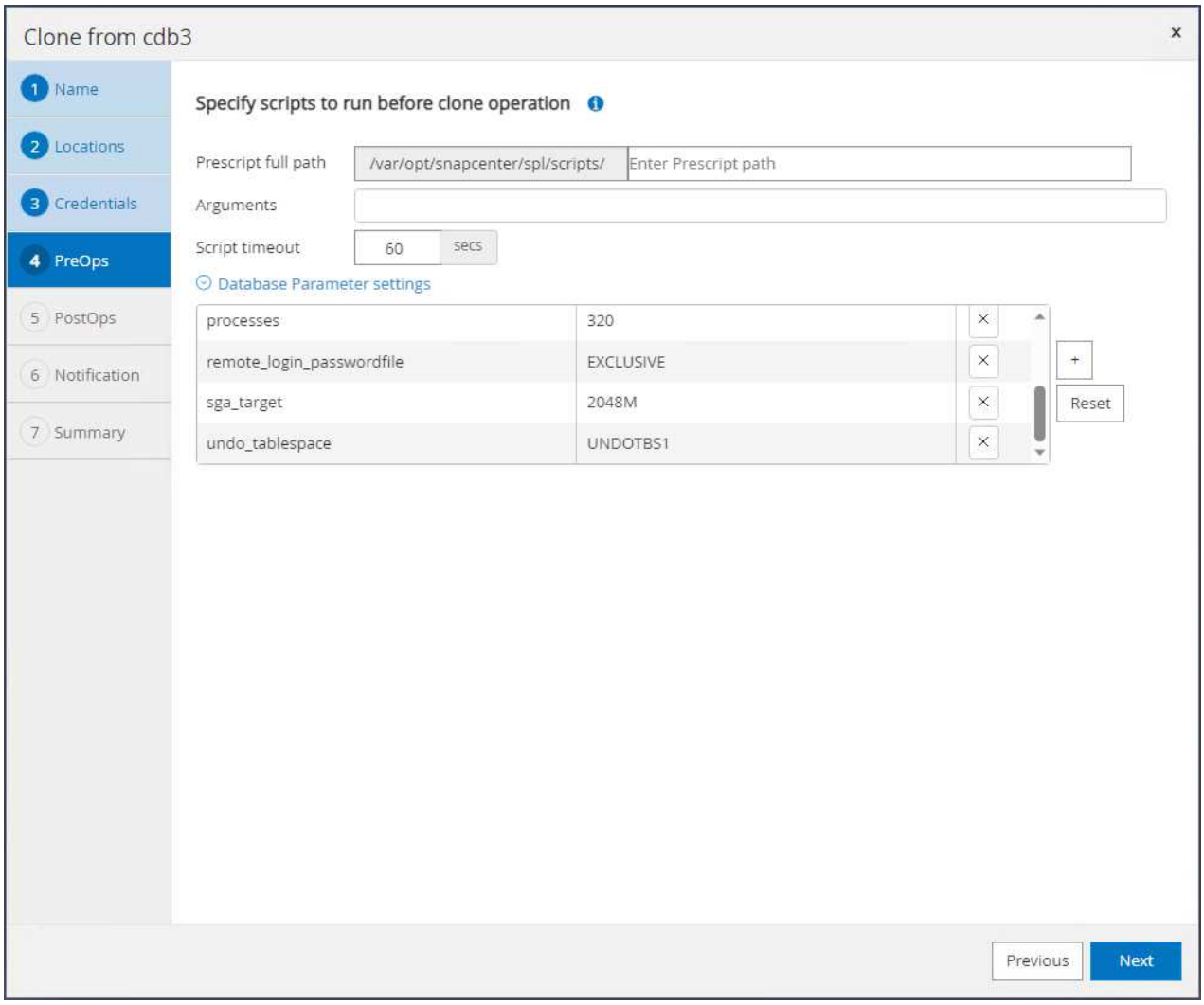
3. Select a VM in VMC as the target database host. Identical Oracle version should have been installed and configured on the host.



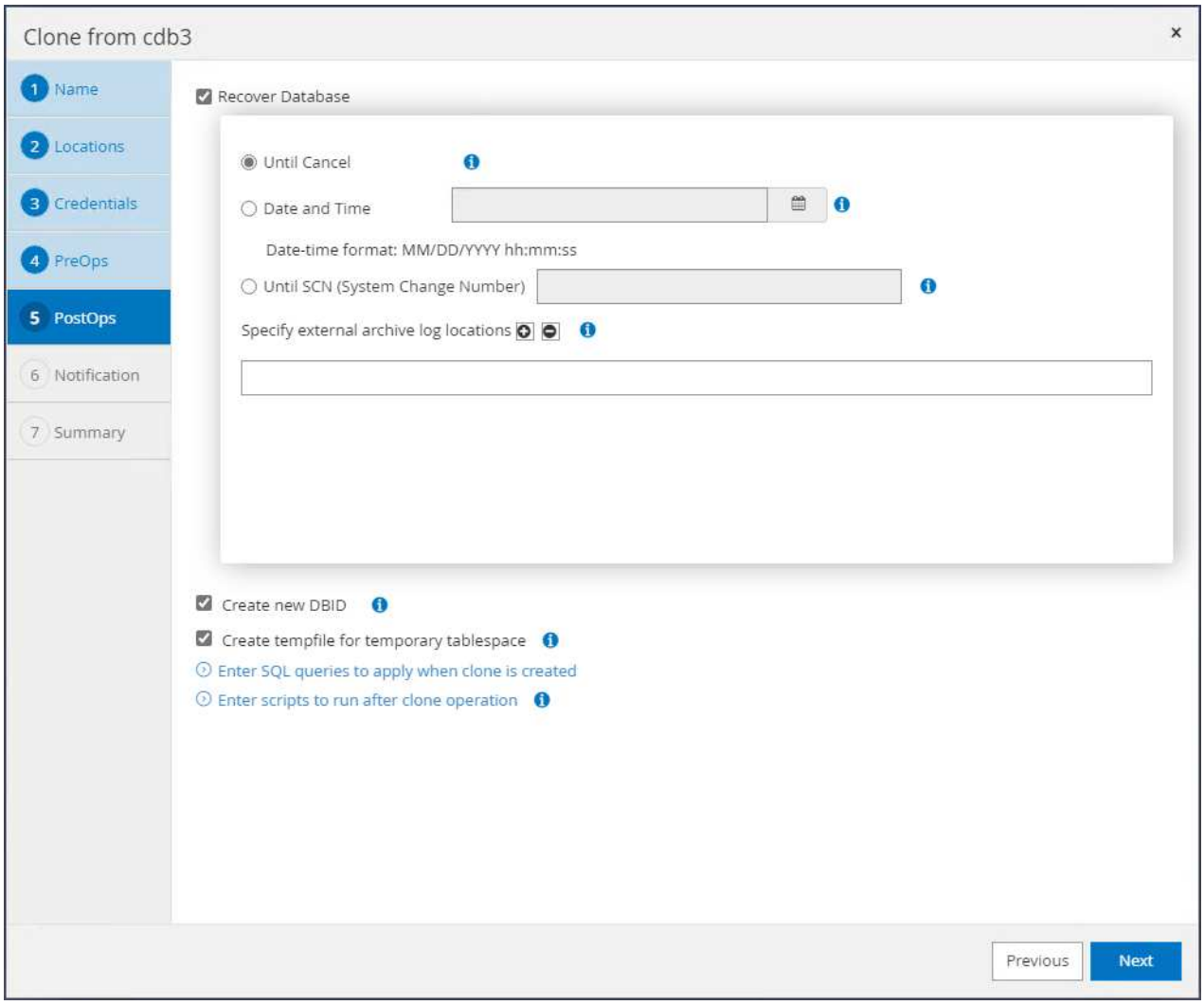
4. Select the proper ORACLE_HOME, user and group on the target host. Keep credential at default.



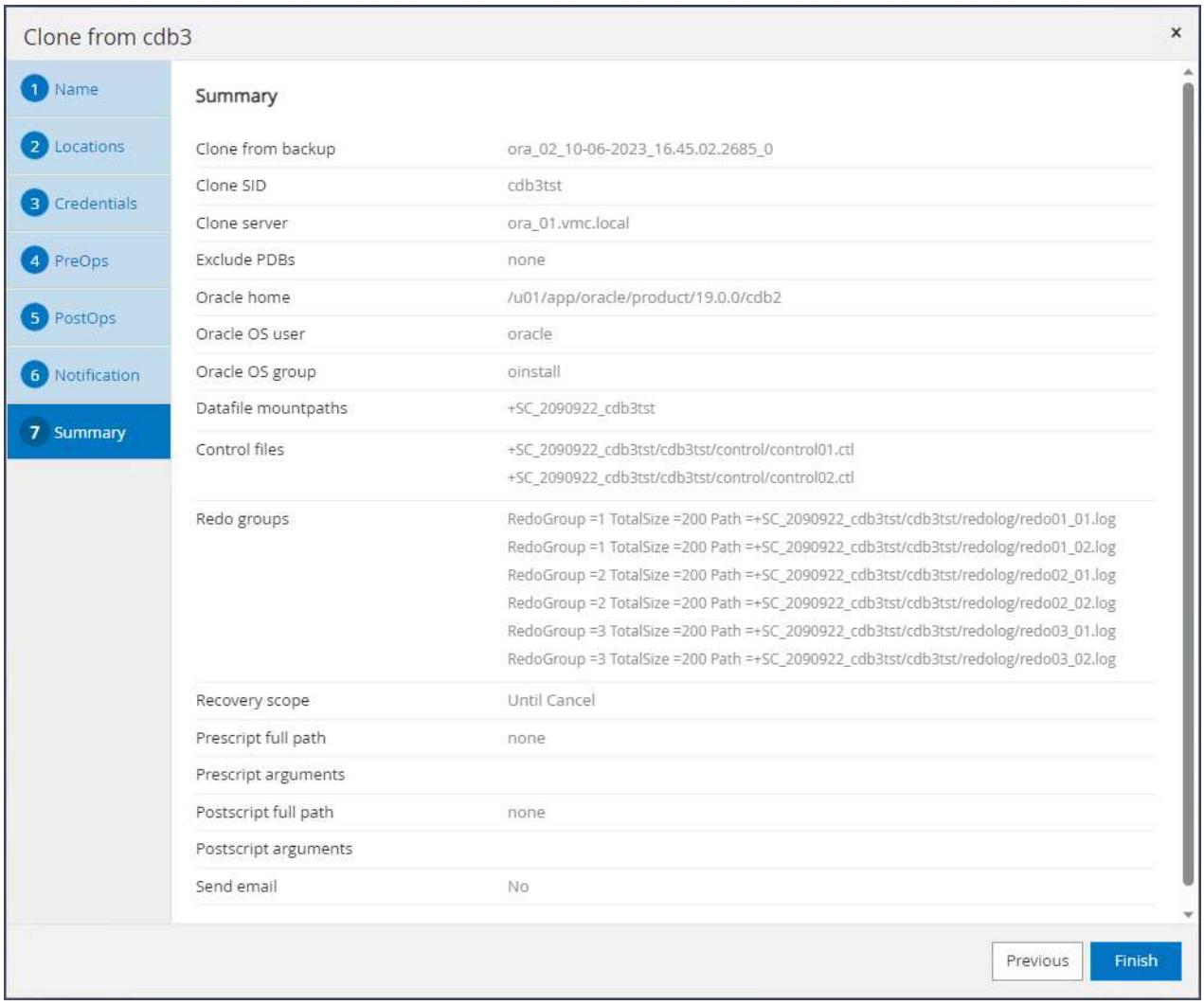
5. Change clone database parameters to meet configuration or resources requirements for the clone database.



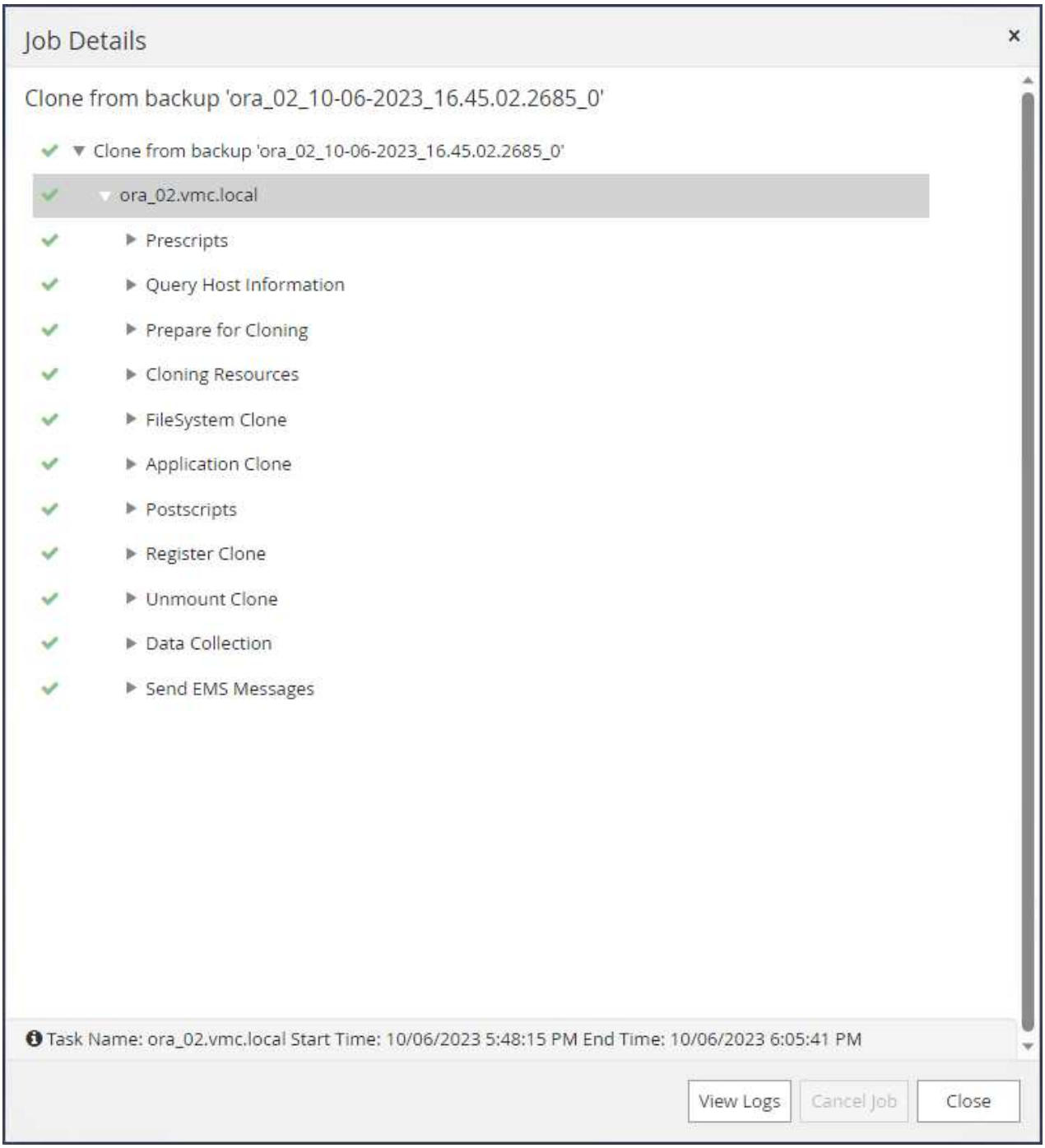
6. Choose recovery scope. `Until Cancel` recovers the clone up to last available log file in the backup set.



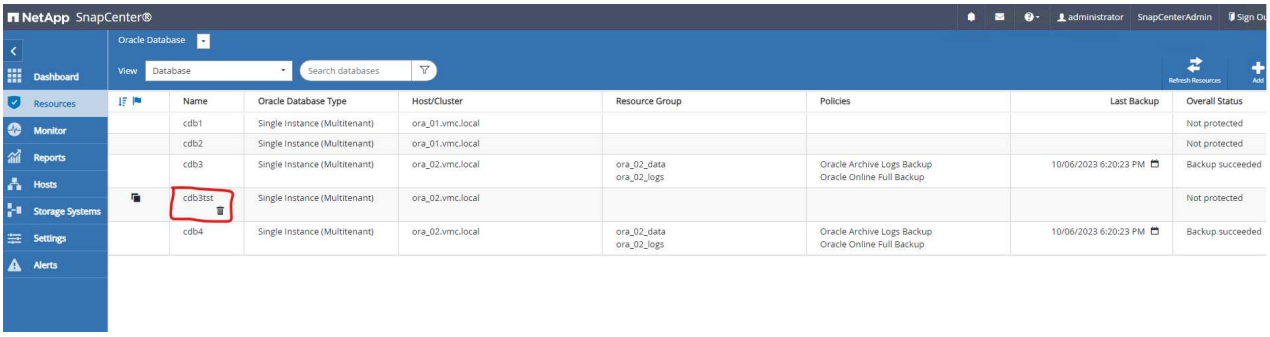
7. Review the summary and launch the clone job.



8. Monitor the clone job execution from `Monitor` tab.



9. Cloned database is immediately registered in SnapCenter.



10. From DB VM ora_02, the cloned database is also registered in Oracle Restart grid control and the dropped test table is recovered in the cloned database cdb3tst as shown below.

```
[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>
```

This completes the demonstration of SnapCenter backup, restore, and clone of Oracle database in VMC SDDC on AWS.

Where to find additional information

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

- VMware Cloud on AWS Documentation

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

- Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation

  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

- Installing and Configuring Oracle Database Using Response Files

  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

- Amazon FSx ONTAP

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

Copyright information

Trademark information