



AWS 클라우드 NetApp Solutions

NetApp
May 03, 2024

목차

AWS 클라우드	1
TR-4986: iSCSI를 지원하는 Amazon FSx ONTAP에서 간편하고 자동화된 Oracle 배포	1
TR-4979: 게스트 마운트 FSx ONTAP를 사용하는 AWS 기반 VMware Cloud의 단순화된 자가 관리 Oracle	18
TR-4981: Amazon FSx ONTAP을 사용한 Oracle Active Data Guard 비용 절감	86
TR-4973: AWS FSx ONTAP에서 증분 병합을 사용하여 Oracle VLDB의 빠른 복구 및 복제	121
TR-4974: NFS/ASM을 사용하는 AWS FSx/EC2에서 독립 실행형 재가동 시 Oracle 19c	202
TR-4965: iSCSI/ASM을 사용하는 AWS FSx/EC2에서 Oracle 데이터베이스 구축 및 보호	228
AWS EC2 및 FSx Best Practices에 Oracle Database 구축	259

AWS 클라우드

TR-4986: iSCSI를 지원하는 Amazon FSx ONTAP에서 간편하고 자동화된 Oracle 배포

Allen Cao, Niyaz Mohamed, NetApp

목적

Amazon FSx for NetApp ONTAP은 AWS 클라우드에서 완전 관리형 NetApp ONTAP 파일 시스템을 시작하고 실행할 수 있는 스토리지 서비스입니다. NetApp 파일 시스템의 익숙한 특징, 성능, 기능 및 API와 함께 완전 관리형 AWS 서비스의 민첩성, 확장성 및 단순성을 제공합니다. Oracle과 같이 가장 까다로운 데이터베이스 워크로드를 AWS 클라우드에서 안심하고 실행할 수 있습니다.

이 문서에서는 Ansible 자동화를 사용하여 Amazon FSx ONTAP 파일 시스템에 Oracle 데이터베이스를 간편하게 구축하는 방법을 설명합니다. Oracle 데이터베이스는 데이터 액세스를 위한 iSCSI 프로토콜 및 데이터베이스 스토리지 디스크 관리를 위한 Oracle ASM이 포함된 독립 실행형 재시작 구성으로 구축됩니다. 또한 AWS Cloud에서 스토리지 효율적인 데이터베이스 작업을 위해 NetApp SnapCenter UI 툴을 사용하여 Oracle 데이터베이스 백업, 복원 및 클론에 대한 정보를 제공합니다.

이 솔루션은 다음과 같은 사용 사례를 해결합니다.

- Amazon FSx ONTAP 파일 시스템에서 자동화된 Oracle 데이터베이스 구축
- NetApp SnapCenter 툴을 사용하여 Amazon FSx ONTAP 파일 시스템에서 Oracle 데이터베이스 백업 및 복원
- NetApp SnapCenter 툴을 사용하여 Amazon FSx ONTAP 파일 시스템에서 개발/테스트용 Oracle 데이터베이스 클론 또는 기타 사용 사례입니다

대상

이 솔루션은 다음과 같은 사용자를 대상으로 합니다.

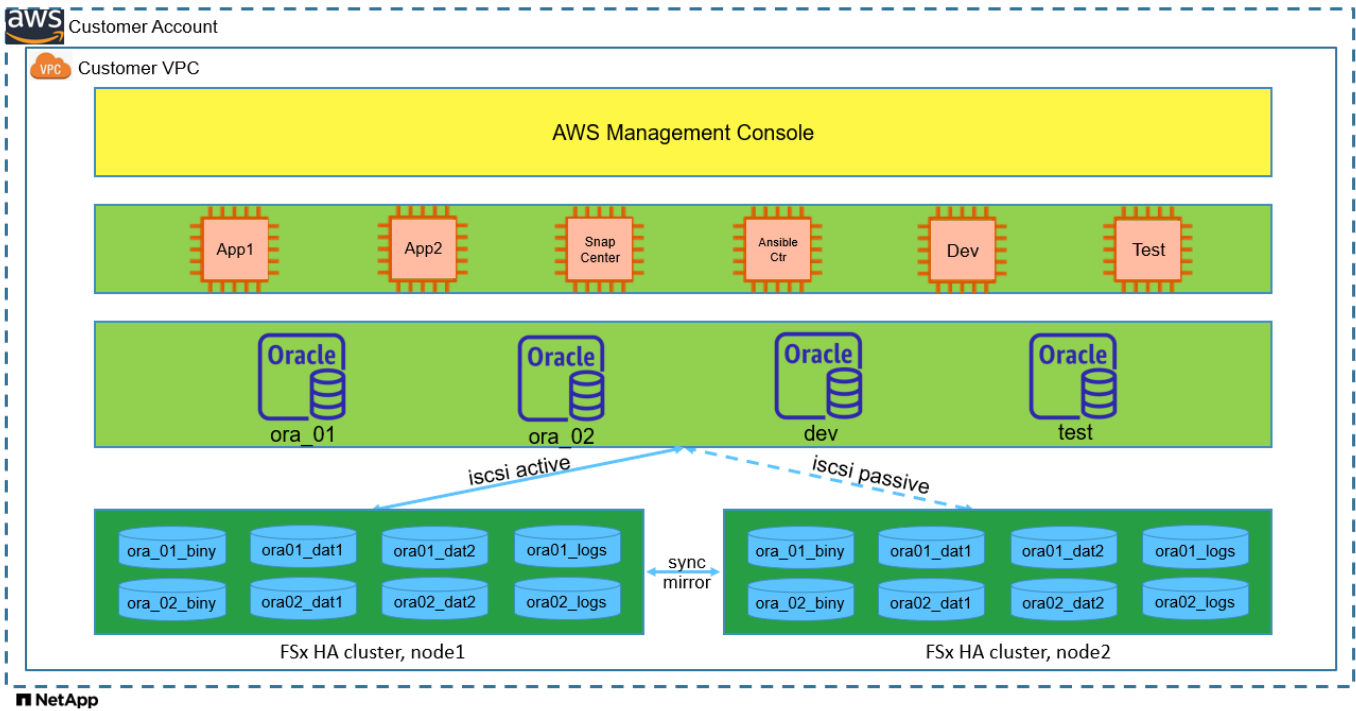
- Amazon FSx ONTAP 파일 시스템에 Oracle을 구축하고자 하는 DBA
- Amazon FSx ONTAP 파일 시스템에서 Oracle 워크로드를 테스트하려는 데이터베이스 솔루션 설계자
- Amazon FSx ONTAP 파일 시스템에서 Oracle 데이터베이스를 구축하고 관리하고자 하는 스토리지 관리자
- Amazon FSx ONTAP 파일 시스템에서 Oracle 데이터베이스를 설치하려는 애플리케이션 소유자

솔루션 테스트 및 검증 환경

이 솔루션의 테스트 및 검증은 최종 구축 환경과 일치하지 않을 수 있는 랩 설정에서 수행되었습니다. 섹션을 참조하십시오 [\[Key Factors for Deployment Consideration\]](#) 를 참조하십시오.

있습니다

Simplified, automated Oracle deployment on Amazon FSx ONTAP with iSCSI



하드웨어 및 소프트웨어 구성 요소

* 하드웨어 *		
Amazon FSx ONTAP 스토리지	AWS에서 제공하는 현재 버전입니다	동일한 VPC 및 가용성 존에 FSx HA 클러스터 1개
컴퓨팅용 EC2 인스턴스	T2.xLarge/4vCPU/16G	동시 배포를 위한 EC2 T2 xlarge EC2 인스턴스 2개
* 소프트웨어 *		
RedHat Linux	RHEL-8.6, 4.18.0-372.9.1.el8.x86_64 커널	테스트를 위해 RedHat 서브스크립션을 배포했습니다
Windows Server를 선택합니다	2022 표준, 10.0.20348 빌드 20348	호스팅 SnapCenter 서버
Oracle Grid Infrastructure	버전 19.18	RU 패치 p34762026_190000_Linux-x86-64.zip 를 적용했습니다
Oracle 데이터베이스	버전 19.18	RU 패치 p34765931_190000_Linux-x86-64.zip 를 적용했습니다
Oracle OPatch	버전 12.2.0.1.36	최신 패치 p6880880_190000_Linux-x86-64.zip
SnapCenter 서버	버전 4.9P1	작업 그룹 배포
JDK를 엽니다	버전 java-1.8.0-openjdk.x86_64	DB VM에 대한 SnapCenter 플러그인 요구 사항

실습 환경의 Oracle 데이터베이스 구성

* 서버 *	* 데이터베이스 *	* DB 스토리지 *
오라_01	NTAP1(NTAP1_PDB1, NTAP1_PDB2, NTAP1_PDB3)	Amazon FSx ONTAP 파일 시스템의 iSCSI LUN
오라_02	NTAP2(NTAP2_PDB1, NTAP2_PDB2, NTAP2_PDB3)	Amazon FSx ONTAP 파일 시스템의 iSCSI LUN

구축 시 고려해야 할 주요 요소

- * Oracle 데이터베이스 스토리지 레이아웃. * 이 자동화된 Oracle 배포에서는 기본적으로 Oracle 바이너리, 데이터 및 로그를 호스팅하기 위해 4개의 데이터베이스 볼륨을 프로비저닝합니다. 볼륨의 단일 LUN이 Oracle 바이너리에 할당합니다. 그런 다음 데이터 및 로그 LUN에서 2개의 ASM 디스크 그룹을 생성합니다. 데이터 ASM 디스크 그룹 내에서 볼륨 하나에 두 개의 LUN이 포함된 두 개의 데이터 볼륨을 프로비저닝합니다. logs ASM 디스크 그룹 내에서 로그 볼륨에 2개의 LUN을 생성합니다. ONTAP 볼륨 내에 여러 LUN이 배치되므로 일반적으로 성능이 향상됩니다.
- * 여러 DB 서버 배포. * 자동화 솔루션은 하나의 Ansible 플레이북 실행에서 여러 DB 서버에 Oracle 컨테이너 데이터베이스를 구축할 수 있습니다. DB 서버 수에 관계없이 Playbook 실행은 동일하게 유지됩니다. 여러 컨테이너 데이터베이스를 서로 다른 데이터베이스 인스턴스 ID(Oracle SID)를 사용하는 단일 EC2 인스턴스에 구축할 수 있습니다. 그러나 호스트에 구축된 데이터베이스를 지원할 충분한 메모리가 있는지 확인하십시오.
- * iSCSI 구성. * EC2 인스턴스 데이터베이스 서버는 iSCSI 프로토콜을 사용하여 FSx 스토리지에 연결합니다. EC2 인스턴스는 일반적으로 단일 네트워크 인터페이스 또는 ENI로 구축합니다. 단일 NIC 인터페이스는 iSCSI 및 애플리케이션 트래픽을 모두 전달합니다. 애플리케이션 및 iSCSI 트래픽-처리량 요구 사항을 모두 충족하는 올바른 EC2 컴퓨팅 인스턴스를 선택하려면 Oracle AWR 보고서를 신중하게 분석하여 Oracle 데이터베이스의 최대 I/O 처리량 요구 사항을 측정하는 것이 중요합니다. 또한 AWS EC2는 일반적으로 각 TCP 흐름을 5Gbps로 제한합니다. 각 iSCSI 경로는 5Gbps(625Mbps)의 대역폭을 제공하며 더 높은 처리량 요구 사항을 지원하려면 여러 iSCSI 연결이 필요할 수 있습니다.
- * 생성하는 각 Oracle ASM 디스크 그룹에 사용할 Oracle ASM 이중화 수준. * Amazon FSx ONTAP는 클러스터 디스크 수준에서 데이터 보호를 위해 HA를 사용할 수 있으므로 이를 사용해야 합니다. 'External Redundancy' 다시 말해, 이 옵션이 Oracle ASM이 디스크 그룹의 콘텐츠를 미러링하도록 허용하지 않습니다.
- * 데이터베이스 백업. * NetApp는 사용자에게 친숙한 UI 인터페이스를 통해 데이터베이스 백업, 복원 및 복제를 위한 SnapCenter 소프트웨어 제품군을 제공합니다. NetApp은 이와 같은 관리 툴을 구현하여 1분 이내에 신속하게 스냅샷 백업, 신속한(분) 데이터베이스 복원 및 데이터베이스 복제를 수행할 것을 권장합니다.

솔루션 구축

다음 섹션에서는 Oracle ASM을 데이터베이스 볼륨 관리자로 사용하는 단일 노드 재시작 구성에서 iSCSI를 통해 EC2 인스턴스 VM에 직접 마운트된 데이터베이스 LUN이 있는 Amazon FSx ONTAP 파일 시스템에서 자동화된 Oracle 19c 배포 및 보호를 위한 단계별 절차를 제공합니다.

배포를 위한 사전 요구 사항

배포에는 다음과 같은 사전 요구 사항이 필요합니다.

1. AWS 계정이 설정되었으며 AWS 계정 내에 필요한 VPC 및 네트워크 세그먼트가 생성되었습니다.
2. AWS EC2 콘솔에서 EC2 Linux 인스턴스를 Oracle DB 서버로 배포합니다. EC2-user에 대해 SSH 개인 /공개 키 인증을 사용합니다. 환경 설정에 대한 자세한 내용은 이전 섹션의 아키텍처 다이어그램을 참조하십시오. 또한 를 검토합니다 "[Linux 인스턴스에 대한 사용자 가이드](#)" 를 참조하십시오.
3. AWS FSx 콘솔에서 요구사항을 충족하는 Amazon FSx ONTAP 파일 시스템을 프로비저닝합니다. 문서를 검토합니다 "[ONTAP 파일 시스템용 FSx 생성](#)" 을 참조하십시오.
4. 2단계와 3단계는 라는 EC2 인스턴스를 만드는 다음 Terraform 자동화 툴킷을 사용하여 수행할 수 있습니다 ora_01 이라는 FSx 파일 시스템이 있습니다 fsx_01. 실행 전에 지침을 주의 깊게 검토하고 환경에 맞게 변수를 변경하십시오. 사용자 고유의 배포 요구 사항에 맞게 템플릿을 쉽게 수정할 수 있습니다.

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

5. 최신 버전의 Ansible 및 Git가 설치된 Ansible 컨트롤러 노드로 EC2 Linux 인스턴스를 프로비저닝합니다. 자세한 내용은 다음 링크를 참조하십시오. "[NetApp 솔루션 자동화 시작하기](#)" 섹션 -
Setup the Ansible Control Node for CLI deployments on RHEL / CentOS 또는
Setup the Ansible Control Node for CLI deployments on Ubuntu / Debian.
6. NetApp SnapCenter UI 도구를 최신 버전으로 실행할 Windows 서버를 프로비저닝합니다. 자세한 내용은 다음 링크를 참조하십시오. "[SnapCenter 서버를 설치합니다](#)"
7. NetApp Oracle Deployment Automation Toolkit for iSCSI의 복제본을 복제합니다.

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

8. EC2 인스턴스/tmp/archive 디렉토리에 Oracle 19c 설치 파일 다음에 스테이징

```
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"
```



Oracle 설치 파일을 스테이징할 수 있는 충분한 공간을 확보하기 위해 Oracle VM 루트 볼륨에 50G 이상을 할당했는지 확인합니다.

9. 다음 동영상을 시청하십시오.

[iSCSI를 사용하여 Amazon FSx ONTAP에서 Oracle을 간편하고 자동으로 배포합니다](#)

자동화 매개 변수 파일

Ansible 플레이북은 사전 정의된 매개 변수를 사용하여 데이터베이스 설치 및 구성 작업을 실행합니다. 이 Oracle 자동화 솔루션의 경우 플레이북을 실행하기 전에 사용자가 입력해야 하는 세 가지 사용자 정의 매개 변수 파일이 있습니다.

- 호스트 - 자동화 플레이북이 실행되는 타겟을 정의합니다.
- vars/vars.yml - 모든 대상에 적용되는 변수를 정의하는 전역 변수 파일입니다.
- host_vars/host_name.yml - 명명된 대상에만 적용되는 변수를 정의하는 지역 변수 파일입니다. 본 사용 사례에서는 Oracle DB 서버가 해당됩니다.

이러한 사용자 정의 변수 파일 외에도 필요한 경우가 아니면 변경할 필요가 없는 기본 매개 변수가 포함된 여러 기본 변수 파일이 있습니다. 다음 섹션에서는 사용자 정의 변수 파일을 구성하는 방법을 보여 줍니다.

매개 변수 파일 구성

1. Ansible 대상 hosts 파일 구성:

```
# 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. 글로벌 vars/vars.yml 파일 구성

```
#####
#####
#####          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: "xxxxxxx"

#####
#####
###           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. 로컬 DB 서버 host_vars/host_name.yml ora_01.yml, ora_02.yml와 같은 구성...

```
# 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
```

플레이북 실행

자동화 툴킷에는 총 6개의 플레이북이 있습니다. 각 작업 블록은 서로 다른 작업 블록을 수행하며 서로 다른 용도로 사용됩니다.

```
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.
```

다음 명령을 사용하여 플레이북을 실행할 수 있는 세 가지 옵션이 있습니다.

1. 모든 구현 플레이북을 하나의 실행 방식으로 실행합니다.

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

2. 1-4의 숫자 순서에 따라 한 번에 하나씩 플레이북을 실행합니다.

```
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. 태그를 사용하여 0-ALL_Playbook.yml을 실행합니다.

```
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. 환경을 실행 취소하십시오

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

사후 실행 검증

플레이북을 실행한 후 Oracle DB 서버에 Oracle 사용자로 로그인하여 Oracle 그리드 인프라 및 데이터베이스가 성공적으로 생성되었는지 확인합니다. 다음은 호스트 ora_01에서 Oracle 데이터베이스 검증의 예입니다.

1. EC2 인스턴스에서 Oracle 컨테이너 데이터베이스를 검증합니다

```
[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. Oracle Listener를 검증합니다.

```
[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 "0c03aafa7c6fd2e5e063280f1eacfb0.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. 생성된 그리드 인프라 및 리소스를 검증합니다.

```
[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        ip-172-30-15-40  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. Oracle ASM을 검증합니다.

```
[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. Oracle Enterprise Manager Express에 로그인하여 데이터베이스를 검증합니다.

Not secure | <https://172.30.15.40:5500/em/login>

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://172.30.15.40:5500/em/shell>

ORACLE Enterprise Manager Database Express

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

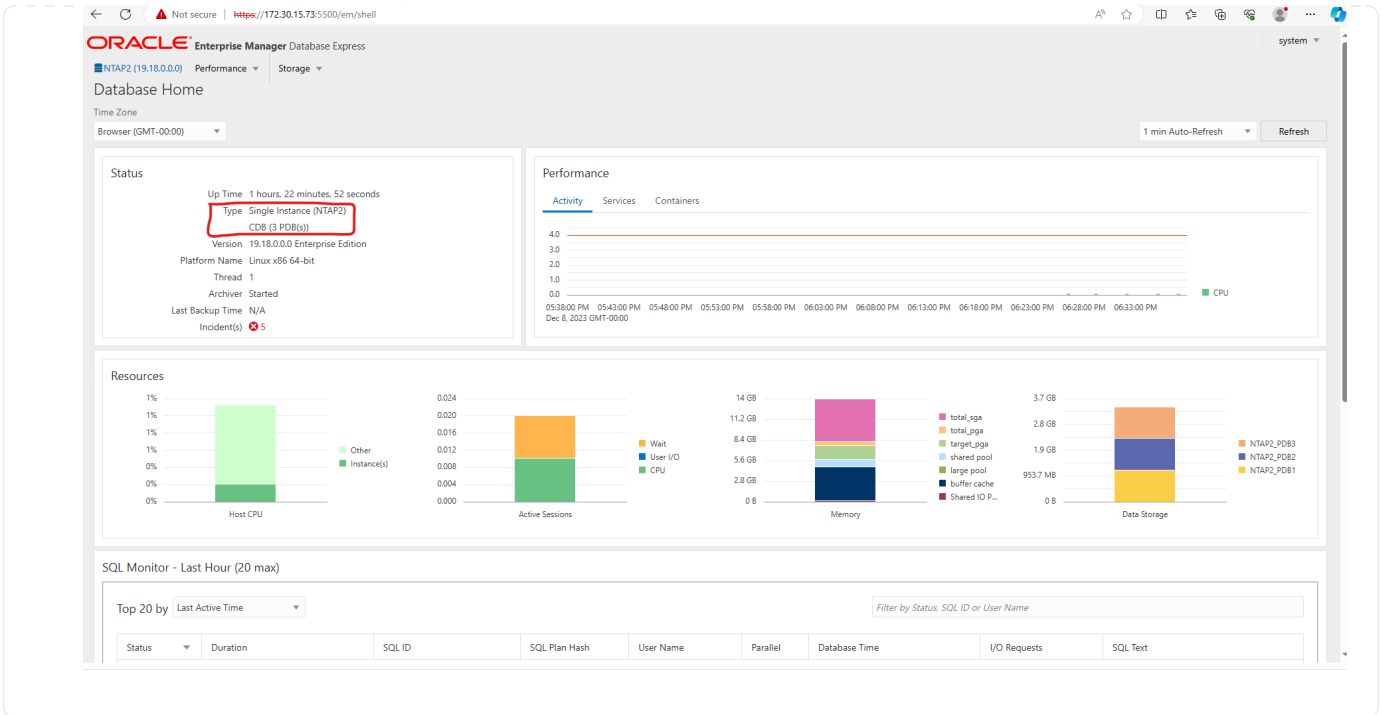
Activity Services Containers

Resources

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
--------	----------	--------	---------------	-----------	----------	---------------	--------------	----------



SnapCenter를 사용하여 Oracle 백업, 복원 및 클론 복제를 수행합니다

TR-4979를 참조하십시오 "게스트 마운트 FSx ONTAP를 사용하여 AWS 기반 VMware Cloud에서 간편하게 자가 관리 가능한 Oracle" 섹션을 참조하십시오 Oracle backup, restore, and clone with SnapCenter SnapCenter 설정 및 데이터베이스 백업, 복원 및 클론 워크플로우의 실행에 대한 자세한 내용을 참조하십시오.

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 문서 및/또는 웹 사이트를 참조하십시오.

- NetApp ONTAP용 Amazon FSx

["https://aws.amazon.com/fsx/netapp-ontap/"](https://aws.amazon.com/fsx/netapp-ontap/)

- Amazon EC2

https://aws.amazon.com/pm/ec2/?trk=36c6da98-7b20-48fa-8225-4784bced9843&sc_channel=ps&s_kwcid=AL!4422!3!467723097970!e!!g!!aws%20ec2&ef_id=Cj0KCQiA54KfBhCKARIsAJzSrdqwQrghn6i71jiWzSeaT9Uh1-vY-VfhJixF-xnv5rWwn2S7RqZOTQ0aAh7eEALw_wcB:G:s&s_kwcid=AL!4422!3!467723097970!e!!g!!aws%20ec2

- 새 데이터베이스 설치를 통해 독립 실행형 서버용 Oracle Grid Infrastructure 설치

["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"](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)

- 응답 파일을 사용하여 Oracle 데이터베이스 설치 및 구성

["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"](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)

- ONTAP와 함께 Red Hat Enterprise Linux 8.2를 사용하십시오

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

TR-4979: 게스트 마운트 FSx ONTAP를 사용하는 AWS 기반 VMware Cloud의 단순화된 자가 관리 Oracle

Allen Cao, Niyaz Mohamed, NetApp

목적

기업들은 수십 년 전부터 프라이빗 데이터 센터에서 VMware 기반 Oracle을 실행하고 있습니다. AWS의 VMC(VMware Cloud)는 푸시 버튼 방식의 솔루션을 통해 VMware의 엔터프라이즈급 SDDC(소프트웨어 정의 데이터 센터) 소프트웨어를 AWS Cloud의 탄력적인 전용 베어 메탈 인프라에 통합합니다. AWS FSx ONTAP는 VMC SDDC에 대한 프리미엄 스토리지와 Data Fabric을 제공하여 고객이 AWS 서비스에 최적화된 액세스를 통해 vSphere® 기반 프라이빗, 퍼블릭 및 하이브리드 클라우드 환경에서 Oracle과 같은 비즈니스 크리티컬 애플리케이션을 실행할 수 있도록 지원합니다. 기존 또는 신규 Oracle 작업이든 상관없이 AWS의 VMC는 친숙하고 간편하며 자가 관리형 Oracle 환경을 VMware에서 제공하는 동시에 AWS 클라우드의 모든 이점을 활용하여 모든 플랫폼 관리 및 최적화를 VMware로 연기합니다.

이 문서에서는 Amazon FSx ONTAP를 기본 데이터베이스 스토리지로 사용하는 VMC 환경에서 Oracle 데이터베이스를 구축하고 보호하는 방법을 설명합니다. Oracle 데이터베이스는 FSx 스토리지의 VMC에 직접 VM 게스트 마운트 LUN 또는 NFS 마운트 VMware VMDK 데이터 저장소 디스크로 구축할 수 있습니다. 이 기술 보고서에서는 Oracle 데이터베이스를 iSCSI 프로토콜 및 Oracle ASM이 포함된 VMC 클러스터의 VM에 게스트 마운트 FSx 스토리지로 직접 구축하는 방법을 중점적으로 설명합니다. 또한 NetApp SnapCenter UI 툴을 사용하여 개발/테스트를 위해 Oracle 데이터베이스를 백업, 복원 및 복제하는 방법과 AWS의 VMC에서 스토리지 효율적인 데이터베이스 작업을 위한 기타 사용 사례를 보여줍니다.

이 솔루션은 다음과 같은 사용 사례를 해결합니다.

- Amazon FSx ONTAP을 기본 데이터베이스 스토리지로 사용하여 AWS의 VMC에 Oracle 데이터베이스를 구현합니다
- NetApp SnapCenter 툴을 사용하여 AWS 기반 VMC에서 Oracle 데이터베이스 백업 및 복원
- NetApp SnapCenter 툴을 사용하여 AWS 기반 VMC에서 개발/테스트용 Oracle 데이터베이스 클론 복제 또는 기타 사용 사례입니다

대상

이 솔루션은 다음과 같은 사용자를 대상으로 합니다.

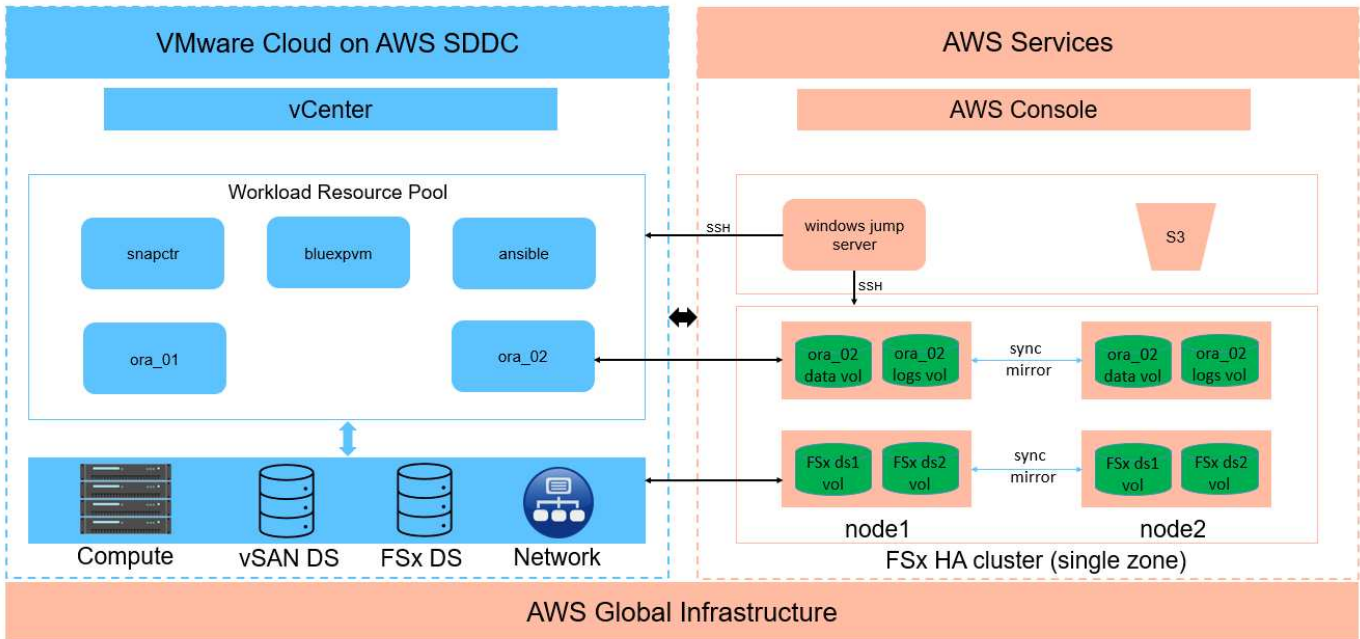
- Amazon FSx ONTAP를 통해 AWS 기반 VMC에서 Oracle을 구축하고자 하는 DBA
- AWS 클라우드에서 VMC에서 Oracle 워크로드를 테스트하려는 데이터베이스 솔루션 설계자
- Amazon FSx ONTAP를 통해 AWS의 VMC에 배포된 Oracle 데이터베이스를 구축하고 관리하고자 하는 스토리지 관리자
- AWS 클라우드의 VMC에서 Oracle 데이터베이스를 구축하려는 애플리케이션 소유자

솔루션 테스트 및 검증 환경

이 솔루션의 테스트 및 검증은 AWS 기반 VMC가 있는 연구소 환경에서 수행되었으며, 최종 구축 환경과 일치하지 않을 수 있습니다. 자세한 내용은 섹션을 참조하십시오 [\[Key Factors for Deployment Consideration\]](#).

있습니다

Oracle Database Deployment in VMware Cloud on AWS SDDC with Amazon FSx ONTAP



NetApp

하드웨어 및 소프트웨어 구성 요소

* 하드웨어 *		
FSX ONTAP 저장소	AWS에서 제공하는 현재 버전입니다	VMC와 동일한 VPC 및 가용 영역에 FSx ONTAP HA 클러스터 1개
VMC SDDC 클러스터	Amazon EC2 i3.metal 단일 노드/Intel Xeon E5-2686 CPU, 36코어/512G RAM	10.37TB vSAN 스토리지
* 소프트웨어 *		
RedHat Linux	RHEL-8.6, 4.18.0-372.9.1.el8.x86_64 커널	테스트를 위해 RedHat 서브스크립션을 배포했습니다
Windows Server를 선택합니다	2022 표준, 10.0.20348 빌드 20348	호스팅 SnapCenter 서버
Oracle Grid Infrastructure	버전 19.18	RU 패치 p34762026_190000_Linux-x86-64.zip 를 적용했습니다
Oracle 데이터베이스	버전 19.18	RU 패치 p34765931_190000_Linux-x86-64.zip 를 적용했습니다

Oracle OPatch	버전 12.2.0.1.36	최신 패치 p6880880_190000_Linux-x86-64.zip
SnapCenter 서버	버전 4.9P1	작업 그룹 배포
VM용 BlueXP 백업 및 복구	릴리즈 1.0	ova vSphere 플러그인 VM으로 구축됩니다
VMware vSphere를 참조하십시오	버전 8.0.1.00300	VMware Tools, 버전: 11365 - Linux, 12352 - Windows
JDK를 엽니다	버전 java-1.8.0-openjdk.x86_64	DB VM에 대한 SnapCenter 플러그인 요구 사항

AWS 기반 VMC의 Oracle 데이터베이스 구성

* 서버 *	* 데이터베이스 *	* DB 스토리지 *
오라_01	cdb1(cdb1_pdb1,cdb1_pdb2,cdb1_pdb3)	FSx ONTAP의 VMDK 데이터 저장소
오라_01	cdb2(cdb2_pdb)	FSx ONTAP의 VMDK 데이터 저장소
오라_02	cdb3(cdb3_pdb1, cdb3_pdb2, cdb3_pdb3)	직접 게스트가 FSx ONTAP를 마운트합니다
오라_02	cdb4(cdb4_pdb)	직접 게스트가 FSx ONTAP를 마운트합니다

구축 시 고려해야 할 주요 요소

- * FSx to VMC 연결. * AWS 기반 VMware Cloud에 SDDC를 배포하면 AWS 계정 내에서 생성되고 VMware에서 관리하는 VPC에서 생성됩니다. 또한 SDDC를 고객 AWS 계정이라고 하는 귀사에 속한 AWS 계정도 연결해야 합니다. 이 연결을 통해 SDDC는 고객 계정에 속한 AWS 서비스에 액세스할 수 있습니다. FSx for ONTAP은 고객 계정에 구축된 AWS 서비스입니다. VMC SDDC가 고객 계정에 연결되면 VMC SDDC의 VM에서 직접 게스트 마운트를 위해 FSx 스토리지를 사용할 수 있습니다.
- * FSx 스토리지 HA 클러스터 단일 또는 다중 영역 배포. * 이러한 테스트 및 검증에서는 단일 AWS 가용성 영역에 FSx HA 클러스터를 구축했습니다. 또한 NetApp은 성능을 높이고 가용 영역 간 데이터 전송 비용을 피하기 위해 동일한 가용성 영역에 FSx for NetApp ONTAP 및 VMware Cloud를 AWS에 구축할 것을 권장합니다.
- * FSx 스토리지 클러스터 크기 조정 * ONTAP 스토리지 파일 시스템용 Amazon FSx는 최대 160,000개의 원시 SSD IOPS, 최대 4Gbps 처리량 및 최대 192TiB 용량을 제공합니다. 하지만 구축 시점의 실제 요구사항에 따라 프로비저닝된 IOPS, 처리량 및 스토리지 제한(최소 1,024GiB)을 기준으로 클러스터 크기를 조정할 수 있습니다. 애플리케이션 가용성에 영향을 주지 않고 용량을 즉각적으로 동적으로 조정할 수 있습니다.
- * Oracle 데이터 및 로그 레이아웃 * 테스트 및 검증에서 각각 데이터 및 로그용 ASM 디스크 그룹 2개를 구축했습니다. DATA ASM 디스크 그룹 내에서 데이터 볼륨에 4개의 LUN을 프로비저닝했습니다. logs ASM 디스크 그룹 내에서 로그 볼륨에 두 개의 LUN을 프로비저닝했습니다. 일반적으로 Amazon FSx for ONTAP 볼륨 내에 여러 LUN이 배치되므로 성능이 향상됩니다.
- * iSCSI 구성. * VMC SDDC의 데이터베이스 VM은 iSCSI 프로토콜을 사용하여 FSx 스토리지에 연결됩니다. Oracle AWR 보고서를 신중하게 분석하여 애플리케이션 및 iSCSI 트래픽-처리량 요구사항을 결정함으로써 Oracle 데이터베이스의 최대 I/O 처리량 요구사항을 측정하는 것이 중요합니다. 또한 다중 경로가 올바르게 구성된 두 FSx iSCSI 엔드포인트 모두에 4개의 iSCSI 연결을 할당하는 것이 좋습니다.
- * 귀하가 생성하는 각 Oracle ASM 디스크 그룹에 사용할 Oracle ASM 이중화 수준. * FSx ONTAP는 이미 FSx

클러스터 수준의 스토리지를 미러링하므로 외부 이중화를 사용해야 합니다. 즉, Oracle ASM이 디스크 그룹의 내용을 미러링할 수 없습니다.

- * 데이터베이스 백업. * NetApp는 사용자에게 친숙한 UI 인터페이스를 통해 데이터베이스 백업, 복원 및 복제를 위한 SnapCenter 소프트웨어 제품군을 제공합니다. NetApp은 이와 같은 관리 툴을 구현하여 1분 이내에 신속하게 스냅샷 백업, 신속한(분) 데이터베이스 복원 및 데이터베이스 복제를 수행할 것을 권장합니다.

솔루션 구축

다음 섹션에서는 Oracle ASM을 데이터베이스 볼륨 관리자로 사용하는 단일 노드 재시작 구성에서 FSx ONTAP 스토리지를 DB VM에 직접 마운트하여 AWS의 VMC에서 Oracle 19c 구축을 위한 단계별 절차를 제공합니다.

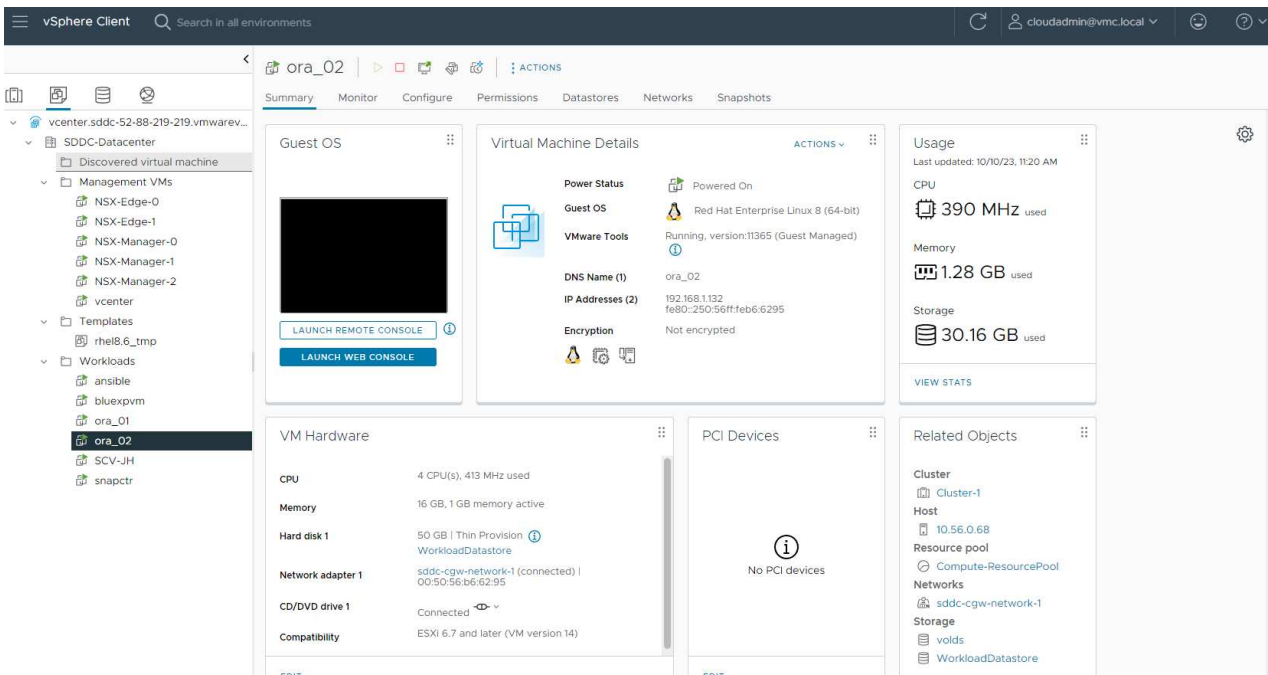
배포를 위한 사전 요구 사항

배포에는 다음과 같은 사전 요구 사항이 필요합니다.

1. VMware Cloud on AWS를 사용하는 SDDC(소프트웨어 정의 데이터 센터)가 생성되었습니다. VMC에서 SDDC를 생성하는 방법에 대한 자세한 지침은 VMware 설명서를 참조하십시오 ["AWS 기반 VMware Cloud 시작하기"](#)
2. AWS 계정이 설정되었으며 AWS 계정 내에 필요한 VPC 및 네트워크 세그먼트가 생성되었습니다. AWS 계정이 VMC SDDC에 연결되어 있습니다.
3. AWS EC2 콘솔에서 Amazon FSx for ONTAP 스토리지 HA 클러스터를 구축하여 Oracle 데이터베이스 볼륨을 호스팅합니다. FSx 저장소 배포에 익숙하지 않은 경우 설명서를 참조하십시오 ["ONTAP 파일 시스템용 FSx 생성"](#) 을 참조하십시오.
4. 위의 단계는 SSH 및 FSx 파일 시스템을 통한 VMC 액세스에서 SDDC의 점프 호스트로 EC2 인스턴스를 생성하는 Terraform 자동화 툴킷을 사용하여 수행할 수 있습니다. 실행 전에 지침을 주의 깊게 검토하고, 환경에 맞게 변수를 변경하십시오.

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

5. VMC에 구축할 Oracle 환경을 호스팅할 수 있도록 AWS에서 VMware SDDC에 VM을 구축합니다. 이 데모에서는 Oracle DB 서버로 2개의 Linux VM, SnapCenter 서버용 1개의 Windows 서버, 자동화된 Oracle 설치 또는 구성을 위한 Ansible 컨트롤러로 선택적 Linux 서버 1개를 구축했습니다. 다음은 솔루션 검증에 위한 실습 환경의 스냅샷입니다.



6. 선택적으로 NetApp는 필요한 경우 Oracle 배포 및 구성을 실행할 수 있는 몇 가지 자동화 툴킷을 제공합니다. 을 참조하십시오 ["DB Automation 툴킷"](#) 를 참조하십시오.



Oracle 설치 파일을 스테이징할 수 있는 충분한 공간을 확보하려면 Oracle VM 루트 볼륨에 50G 이상을 할당해야 합니다.

사전 요구 사항이 프로비저닝되면 SSH를 통해 Oracle VM에 관리자 사용자로 로그인하고 루트 사용자에게 sudo를 사용하여 Oracle 설치를 위한 Linux 커널을 구성합니다. Oracle 설치 파일은 AWS S3 버킷에서 스테이징된 후 VM으로 전송할 수 있습니다.

1. 스테이징 디렉토리를 만듭니다 /tmp/archive 폴더를 지정하고 777 권한을 설정합니다.

```
mkdir /tmp/archive
```

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

2. Oracle 바이너리 설치 파일 및 기타 필요한 rpm 파일을 에 다운로드하고 스테이징합니다 /tmp/archive 디렉토리.

에 명시된 설치 파일의 다음 목록을 참조하십시오 /tmp/archive DB VM에 있습니다.

```
[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. 대부분의 커널 구성 요구 사항을 충족하는 Oracle 19c 사전 설치 RPM을 설치합니다.

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

4. 누락된 을 다운로드하고 설치합니다 compat-libcap1 Linux 8에서

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

5. NetApp에서 NetApp 호스트 유틸리티를 다운로드하고 설치합니다.

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

6. 설치합니다 polycycoreutils-python-utils.

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

7. 열려 있는 JDK 버전 1.8을 설치합니다.

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

8. iSCSI 초기자 유틸리티를 설치합니다.

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

9. sg3_utils를 설치합니다.

```
yum install sg3_utils
```

10. device-mapper-multipath를 설치합니다.

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

11. 현재 시스템에서 투명 HugePages를 비활성화합니다.

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

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

12. 에 다음 행을 추가합니다 /etc/rc.local 를 눌러 비활성화합니다 transparent_hugepage 재부팅 후

```
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. SELinux를 변경하여 해제합니다 SELINUX=enforcing 를 선택합니다 SELINUX=disabled. 변경 사항을 적용하려면 호스트를 재부팅해야 합니다.

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

14. 에 다음 행을 추가합니다 limit.conf 파일 설명자 제한과 스택 크기를 설정합니다.

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

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

15. 다음 명령으로 구성된 스왑 공간이 없는 경우 DB VM에 스왑 공간을 추가합니다. ["스왑 파일을 사용하여 Amazon EC2 인스턴스에서 스왑 공간으로 사용할 메모리를 어떻게 할당합니까?"](#) 정확한 추가 공간은 최대 16G RAM의 크기에 따라 달라집니다.
16. 변경 node.session.timeo.replacement_timeout 에 있습니다 iscsi.conf 120 ~ 5초 사이의 구성 파일.

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

17. EC2 인스턴스에서 iSCSI 서비스를 설정 및 시작합니다.

```
systemctl enable iscsid
```

```
systemctl start iscsid
```

18. 데이터베이스 LUN 매핑에 사용할 iSCSI 이니시에이터 주소를 검색합니다.

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

19. ASM 관리 사용자(Oracle)에 대한 ASM 그룹을 추가합니다.

```
groupadd asmadmin
```

```
groupadd asmdba
```

```
groupadd asmoper
```

20. Oracle 사용자를 수정하여 ASM 그룹을 보조 그룹으로 추가합니다(Oracle 사전 설치 RPM 설치 후 Oracle 사용자가 생성되어야 함).

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

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

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

21. Linux 방화벽이 활성화된 경우 중지하고 비활성화합니다.

```
systemctl stop firewalld
```

```
systemctl disable firewalld
```

22. 관리자 사용자에게 대해 주석 처리를 해제하여 암호 없는 sudo를 활성화합니다 # %wheel ALL=(ALL) NOPASSWD: ALL /etc/sudoers 파일에 줄을 입력합니다. 파일 권한을 변경하여 편집합니다.

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

```
vi /etc/sudoers
```

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

23. EC2 인스턴스를 재부팅합니다.

FSx ONTAP LUN을 DB VM에 프로비저닝하고 매핑합니다

SSH 및 FSx 클러스터 관리 IP를 통해 FSx 클러스터에 fsxadmin 사용자로 로그인하여 명령줄에서 세 개의 볼륨을 프로비저닝합니다. Oracle 데이터베이스 바이너리, 데이터 및 로그 파일을 호스팅할 볼륨 내에 LUN을 생성합니다.

1. SSH를 통해 FSx 클러스터에 fsxadmin 사용자로 로그인합니다.

```
ssh fsxadmin@10.49.0.74
```

2. 다음 명령을 실행하여 Oracle 바이너리에 대한 볼륨을 생성합니다.

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

3. 다음 명령을 실행하여 Oracle 데이터용 볼륨을 생성합니다.

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

4. 다음 명령을 실행하여 Oracle 로그용 볼륨을 생성합니다.

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

5. 생성된 볼륨을 확인합니다.

```
vol show ora*
```

명령의 출력:

```
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. 데이터베이스 바이너리 볼륨 내에 바이너리 LUN을 생성합니다.

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

7. 데이터베이스 데이터 볼륨 내에 데이터 LUN을 생성합니다.

```
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. 데이터베이스 로그 볼륨 내에 로그 LUN을 생성합니다.

```
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. 위의 EC2 커널 구성의 14단계에서 검색된 이니시에이터를 사용하여 EC2 인스턴스에 대한 igroup을 생성합니다.

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

10. LUN을 위에서 생성한 igroup에 매핑합니다. 각각의 추가 LUN에 대해 LUN ID를 순차적으로 증분합니다.


```

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. LUN 매핑을 확인합니다.

```
mapping show
```

이 문제는 다음 항목을 반환해야 합니다.

```

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

```


이제 VMC 데이터베이스 VM에 Oracle 그리드 인프라 및 데이터베이스 설치용 FSx ONTAP 스토리지를 가져오고 설정합니다.

1. Windows 점프 서버에서 Putty를 사용하여 관리자 권한으로 SSH를 통해 DB VM에 로그인합니다.
2. SVM iSCSI IP 주소를 사용하여 FSx iSCSI 엔드포인트를 검색합니다. 환경별 포털 주소로 변경합니다.

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

3. 각 타겟에 로그인하여 iSCSI 세션을 설정합니다.

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

명령의 예상 출력은 다음과 같습니다.

```
[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. 활성 iSCSI 세션 목록을 보고 확인합니다.

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

iSCSI 세션을 반환합니다.

```
[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. LUN을 호스트로 가져왔는지 확인합니다.

```
sudo sanlun lun show
```

그러면 FSx의 Oracle LUN 목록이 반환됩니다.

```
[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. 를 구성합니다 multipath.conf 다음 기본 항목과 블랙리스트 항목이 있는 파일입니다.

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

다음 항목 추가:

```
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. 다중 경로 서비스를 시작합니다.

```
sudo systemctl start multipathd
```

이제 다중 경로 장치가 에 나타납니다 /dev/mapper 디렉토리.

```
[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. SSH를 통해 fsxadmin 사용자로 FSx ONTAP 클러스터에 로그인하여 6c574xxx...로 시작하는 각 LUN의 일련-16진수 번호를 검색합니다. 16진수 번호는 AWS 공급업체 ID인 3600a0980으로 시작합니다.

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

그리고 다음과 같이 돌아옵니다.

```
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. 를 업데이트합니다 /dev/multipath.conf 파일 - 다중 경로 장치에 대한 사용자 친화적인 이름을 추가합니다.

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

다음 항목이 있는 경우:

```

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. 다중 경로 서비스를 재부팅하여 에서 장치가 에 있는지 확인합니다 /dev/mapper LUN 이름 대 직렬 16진수 ID로 변경되었습니다.

```
sudo systemctl restart multipathd
```

확인합니다 /dev/mapper 다음과 같이 돌아가려면:

```
[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. 단일 주 파티션으로 바이너리 LUN을 파티셔닝합니다.

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

12. XFS 파일 시스템을 사용하여 분할된 바이너리 LUN을 포맷합니다.

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

13. 에 바이너리 LUN을 마운트합니다 /u01.

```
sudo mkdir /u01
```

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

14. 변경 /u01 Oracle 사용자 및 연결된 기본 그룹에 지점 소유권을 마운트합니다.

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

15. 바이너리 LUN의 UUI를 찾습니다.


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

16. 에 마운트 지점을 추가합니다 /etc/fstab.

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

다음 줄을 추가합니다.

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

17. 루트 사용자로 Oracle 디바이스에 대한 udev 규칙을 추가합니다.

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

다음 항목 포함:

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

18. 루트 사용자로 udev 규칙을 다시 로드합니다.

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

19. 루트 사용자로 udev 규칙을 트리거합니다.

```
udevadm trigger
```

20. 루트 사용자로 multipathd를 다시 로드합니다.

```
systemctl restart multipathd
```

21. EC2 인스턴스 호스트를 재부팅합니다.

Oracle 그리드 인프라 설치

1. SSH를 통해 관리자 사용자로 DB VM에 로그인하고 주석 처리를 해제하여 암호 인증을 활성화합니다
PasswordAuthentication yes 그런 다음 의견을 남기기 바랍니다 PasswordAuthentication no.

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

2. sshd 서비스를 다시 시작합니다.

```
sudo systemctl restart sshd
```

3. Oracle 사용자 암호를 재설정합니다.

```
sudo passwd oracle
```

4. Oracle Restart 소프트웨어 소유자 사용자(Oracle)로 로그인합니다. 다음과 같이 Oracle 디렉토리를 생성합니다.

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

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

5. 디렉터리 권한 설정을 변경합니다.

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

6. 그리드 홈 디렉토리를 만들고 변경합니다.

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

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

7. 그리드 설치 파일의 압축을 풉니다.

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

8. 그리드 홈에서 을 삭제합니다 OPatch 디렉토리.

```
rm -rf OPatch
```

9. 그리드 홈에서 압축을 풉니다 p6880880_190000_Linux-x86-64.zip.

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

10. 그리드 홈에서 수정합니다 cv/admin/cvu_config, 주식 취소 및 바꾸기 CV_ASSUME_DISTID=OEL5 와 함께 CV_ASSUME_DISTID=OL7.

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

11. 를 준비합니다 gridsetup.rsp 자동 설치용 파일 및 RSP 파일을 에 배치합니다 /tmp/archive 디렉토리. RSP 파일은 섹션 A, B 및 G에 다음 정보를 포함해야 합니다.

```
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/mapper/ora_02_data_02,/dev/mapper/ora_02_data_03,/dev/mapper/ora_02_data_04
oracle.install.asm.diskGroup.diskDiscoveryString=/dev/mapper/*
oracle.install.asm.monitorPassword="SetPWD"
oracle.install.asm.configureAFD=true
```

12. EC2 인스턴스에 루트 사용자로 로그인하여 설정합니다 ORACLE_HOME 및 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. Oracle ASM 필터 드라이버와 함께 사용할 디스크 디바이스를 초기화합니다.

```
./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. 설치합니다 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. 설정 해제 \$ORACLE_BASE.

```
unset ORACLE_BASE
```

16. EC2 인스턴스에 Oracle 사용자로 로그인하여 에서 패치를 추출합니다 /tmp/archive 폴더.

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

17. 그리드 홈 /u01/app/oracle/product/19.0.0/GRID에서 Oracle 사용자로 를 실행합니다 gridSetup.sh 그리드 인프라 설치용.

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

18. 루트 사용자로 다음 스크립트를 실행합니다.

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

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

19. 루트 사용자로 multipathd를 다시 로드합니다.

```
systemctl restart multipathd
```

20. Oracle 사용자는 다음 명령을 실행하여 구성을 완료합니다.

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

21. Oracle 사용자로 로그 디스크 그룹을 생성합니다.

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

22. Oracle 사용자는 설치 구성 후 그리드 서비스를 검증합니다.

```
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 필터 드라이버 상태입니다.

```

[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
ASMCMDB> 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/
ASMCMDB> afd_state
ASMCMDB-9526: The AFD state is 'LOADED' and filtering is 'ENABLED' on
host 'ora_02'
ASMCMDB> exit
[oracle@ora_02 grid]$

```

24. HA 서비스 상태를 검증합니다.

```

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

```

Oracle 데이터베이스 설치

1. Oracle 사용자로 로그인하고 설정을 해제합니다 \$ORACLE_HOME 및 \$ORACLE_SID 설정되어 있는지 확인합니다.

```
unset ORACLE_HOME
```

```
unset ORACLE_SID
```

2. Oracle DB 홈 디렉토리를 생성하고 디렉토리를 이 디렉토리로 변경합니다.

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

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

3. Oracle DB 설치 파일의 압축을 풉니다.

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

4. DB 홈에서 를 삭제합니다 OPatch 디렉토리.

```
rm -rf OPatch
```

5. DB HOME에서 압축을 풉니다 p6880880_190000_Linux-x86-64.zip.

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

6. DB 홈에서 수정한다 cv/admin/cvu_config 주석을 해제하고 바꿉니다 CV_ASSUME_DISTID=OEL5 와 함께 CV_ASSUME_DISTID=OL7.

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

7. 에서 /tmp/archive 디렉토리에서 DB 19.18 RU 패치의 포장을 풉니다.

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

8. 에서 DB 자동 설치 RSP 파일을 준비합니다 /tmp/archive/dbinstall.rsp 다음 값이 있는 디렉터리:


```
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. cdb3 home/u01/app/oracle/product/19.0.0/cdb3에서 자동 소프트웨어 전용 DB 설치를 실행합니다.

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

10. 루트 사용자로 를 실행합니다 root.sh 소프트웨어 전용 설치 후 스크립트.

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

11. Oracle 사용자로 를 생성합니다 dbca.rsp 다음 항목이 있는 파일:

```
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. Oracle 사용자로 dbca를 사용하여 DB 생성을 시작합니다.

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

출력:

```

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. 2단계부터 동일한 절차를 반복하여 단일 PDB로 별도의 ORACLE_HOME/u01/APP/ORACLE/PRODUCT/19.0.0/cdb4에 컨테이너 데이터베이스 cdb4를 생성합니다.
2. Oracle 사용자로서 DB 생성 후 Oracle Restart HA 서비스를 검증하여 모든 데이터베이스(cdb3, cdb4)가 HA 서비스에 등록되었는지 확인합니다.

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

출력:

```
[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. Oracle 사용자를 설정합니다 .bash_profile.

```
vi ~/.bash_profile
```

다음 항목 추가:

```
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. cdb3에 대해 생성된 CDB/PDB를 확인합니다.

```
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. cdb4에 대해 생성된 CDB/PDB를 확인합니다.

```

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. sqlplus를 사용하여 각 CDB에 sysdba로 로그인하고 두 CDB에 대해 DB 복구 대상 크기를 +logs 디스크 그룹 크기로 설정합니다.

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

7. sqlplus를 사용하여 각 CDB에 sysdba로 로그인하고 다음 명령 세트를 순서대로 사용하여 아카이브 로그 모드를 활성화합니다.

```
sqlplus /as sysdba
```

```
shutdown immediate;
```



```
startup mount;
```

```
alter database archivelog;
```

```
alter database open;
```

이로써 Oracle 19c 버전 19.18 Amazon FSx for ONTAP 스토리지 및 VMC DB VM에서 구축을 다시 시작합니다. 필요한 경우 Oracle 제어 파일 및 온라인 로그 파일을 + 로그 디스크 그룹으로 재배포하는 것이 좋습니다.

SnapCenter를 사용하여 **Oracle** 백업, 복원 및 클론 복제를 수행합니다

SnapCenter 설정

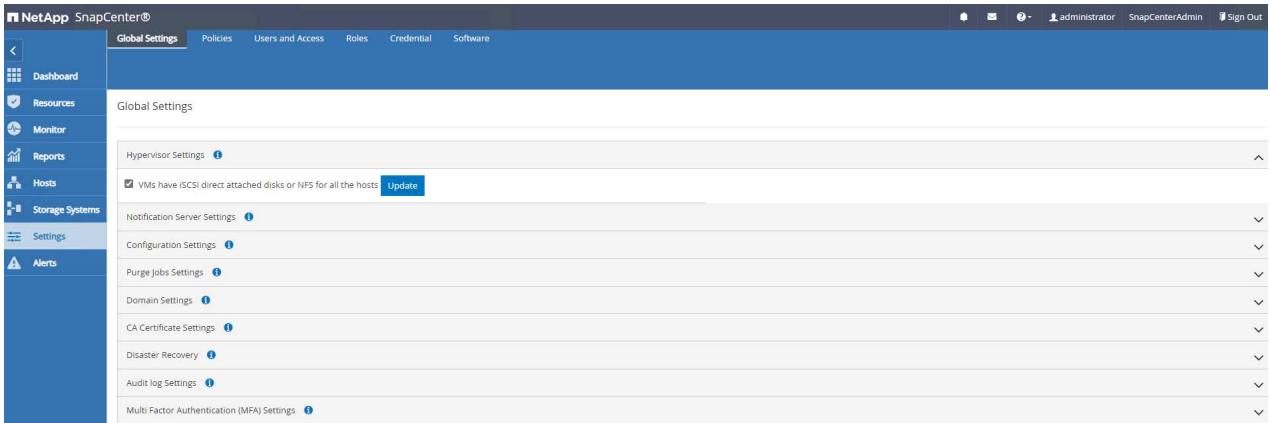
SnapCenter는 호스트 측 플러그인을 데이터베이스 VM에 사용하여 애플리케이션 인식 데이터 보호 관리 작업을 수행합니다. Oracle용 NetApp SnapCenter 플러그인에 대한 자세한 내용은 이 설명서를 참조하십시오 **"Oracle 데이터베이스용 플러그인을 사용하여 수행할 수 있는 작업"**. 다음은 Oracle 데이터베이스 백업, 복구 및 클론 생성을 위해 SnapCenter를 설정하는 상위 단계입니다.

1. NetApp Support 사이트에서 최신 버전의 SnapCenter 소프트웨어를 다운로드합니다. **"NetApp 지원 다운로드"**.
2. 관리자 권한으로 에서 최신 Java JDK를 설치합니다 **"데스크톱 응용 프로그램용 Java를 다운로드하십시오"** SnapCenter 서버 Windows 호스트

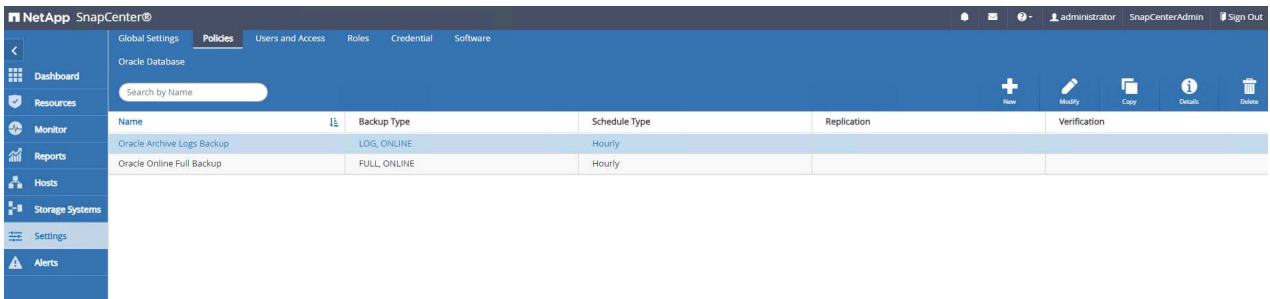


도메인 환경에 Windows 서버가 배포되는 경우 도메인 사용자를 SnapCenter 서버 로컬 관리자 그룹에 추가하고 도메인 사용자와 함께 SnapCenter 설치를 실행합니다.

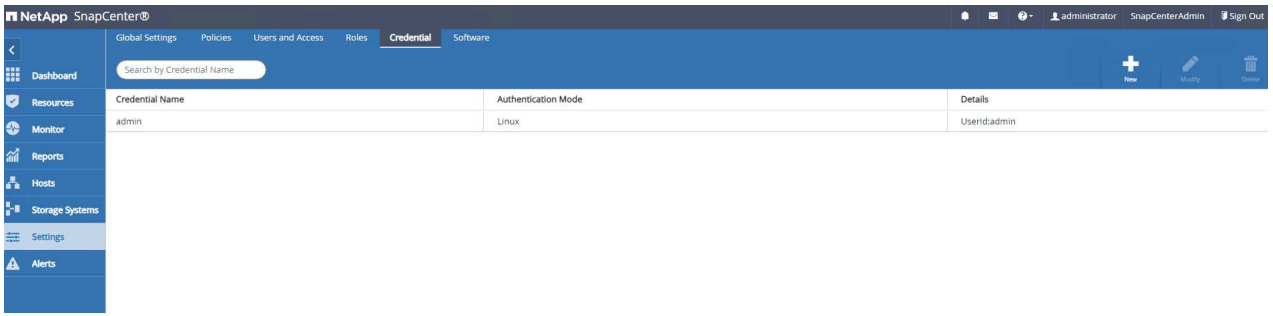
3. 설치 사용자로 HTTPS 포트 8846을 통해 SnapCenter UI에 로그인하여 SnapCenter for Oracle을 구성합니다.
4. 업데이트 Hypervisor Settings 전체 설정에서.



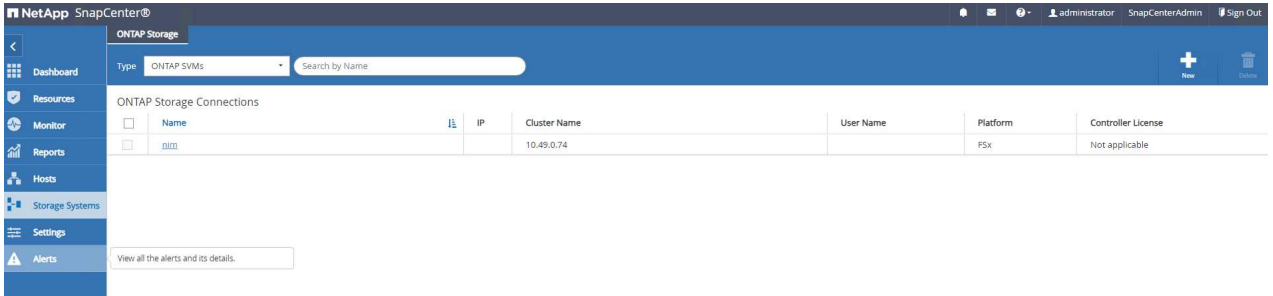
5. Oracle 데이터베이스 백업 정책을 생성합니다. 장애 발생 시 데이터 손실을 최소화하기 위해 보다 빈번한 백업 간격을 허용하는 별도의 아카이브 로그 백업 정책을 생성하는 것이 가장 좋습니다.



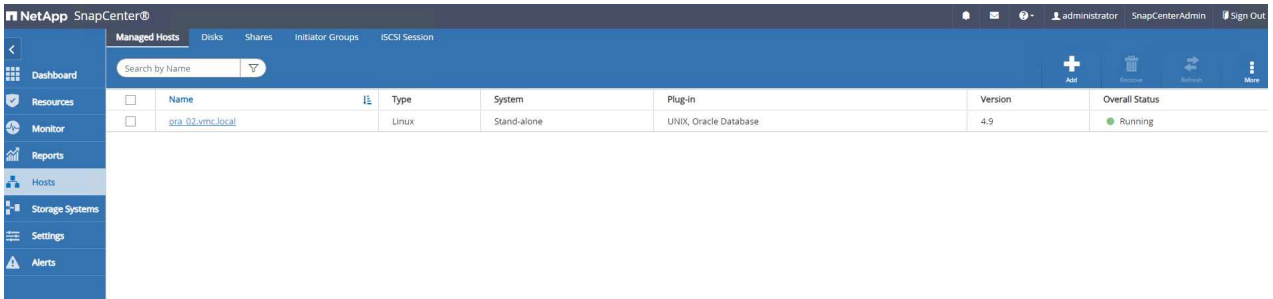
6. 데이터베이스 서버를 추가합니다 Credential DB VM에 대한 SnapCenter 액세스용 자격 증명에는 Linux VM에 대한 sudo 권한 또는 Windows VM에 대한 관리자 권한이 있어야 합니다.



7. FSx ONTAP 스토리지 클러스터를 에 추가합니다 Storage Systems 클러스터 관리 IP를 사용하여 fsxadmin 사용자 ID를 통해 인증됩니다.



8. VMC의 Oracle 데이터베이스 VM을 에 추가합니다 Hosts 이전 6단계에서 만든 서버 자격 증명



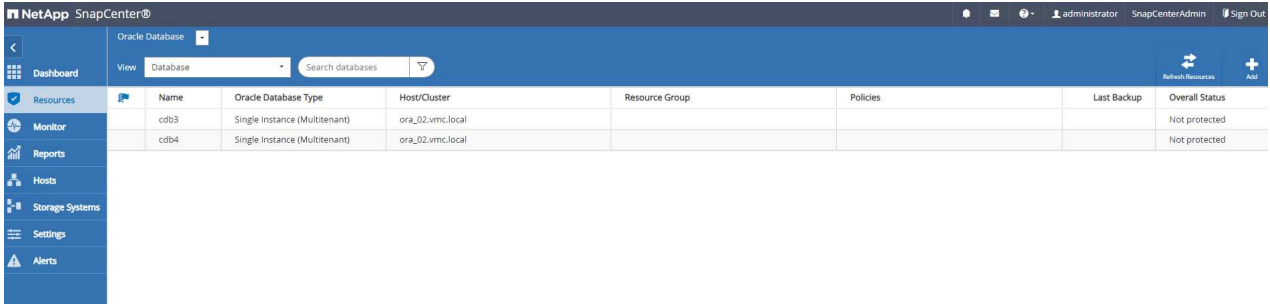
SnapCenter 서버 이름을 DB VM의 IP 주소로 확인할 수 있고 DB VM 이름을 SnapCenter 서버의 IP 주소로 확인할 수 있는지 확인합니다.

데이터베이스 백업

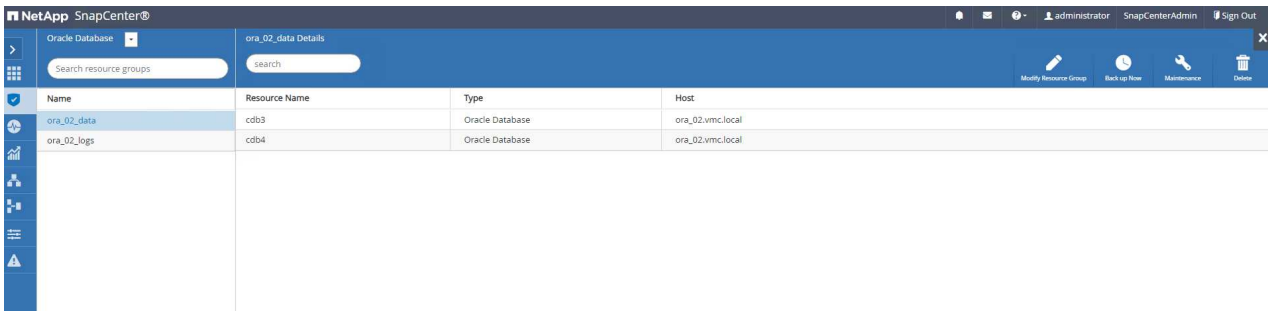


SnapCenter는 FSx ONTAP 볼륨 스냅샷을 활용하여 기존의 RMAN 기반 방법론에 비해 훨씬 더 빠른 데이터베이스 백업, 복원 또는 복제를 수행합니다. 스냅샷은 데이터베이스가 스냅샷 전에 Oracle 백업 모드로 전환되므로 애플리케이션 정합성이 보장됩니다.

1. 에서 Resources 탭에서 VM이 SnapCenter에 추가된 후 VM의 모든 데이터베이스가 자동으로 검색됩니다. 처음에는 데이터베이스 상태가 로 표시됩니다 Not protected.




2. 리소스 그룹을 생성하여 데이터베이스를 DB VM별 등의 논리적 그룹으로 백업합니다 이 예에서는 VM ora_02의 모든 데이터베이스에 대해 전체 온라인 데이터베이스 백업을 수행하기 위해 ora_02_data 그룹을 만들었습니다. 리소스 그룹 ora_02_log는 VM에서 아카이빙된 로그만 백업합니다. 리소스 그룹을 생성하면 백업을 실행할 스케줄도 정의됩니다.




3. 를 클릭하여 리소스 그룹 백업을 수동으로 트리거할 수도 있습니다 Back up Now 리소스 그룹에 정의된 정책으로 백업을 실행합니다.

Add schedules for policy Oracle Online Full Backup ✕

Hourly

Start date 

Expires on 

Repeat every hours mins

i The schedules are triggered in the SnapCenter Server time zone. ✕

4. 에서 백업 작업을 모니터링할 수 있습니다 Monitor 탭을 클릭하여 실행 중인 작업을 클릭합니다.

Job Details

Backup of Resource Group 'ora_01_data' with policy 'Oracle Online Full Backup'

- ✓ ▾ Backup of Resource Group 'ora_01_data' with policy 'Oracle Online Full Backup'
 - ✓ ▾ ora_01.vmc.local
 - ✓ ▶ Prescripts
 - ✓ ▶ Preparing for Oracle Database Backup
 - ✓ ▶ Preparing for File-System Backup
 - ✓ ▶ Backup datafiles and control files
 - ✓ ▶ Backup archive logs
 - ✓ ▶ Finalizing Oracle Database Backup
 - ✓ ▶ Finalizing File-System Backup
 - ✓ ▶ Postscripts
 - ✓ ▶ Data Collection
 - ✓ ▶ Send EMS Messages

Task Name: ora_01.vmc.local Start Time: 10/07/2023 8:53:24 AM End Time: 10/07/2023 8:54:33 AM

View Logs Cancel job Close

5. 백업이 성공한 후 데이터베이스 상태는 작업 상태와 가장 최근 백업 시간을 표시합니다.

NetApp SnapCenter

Oracle Database

View Database Search databases

Resources	Name	Oracle Database Type	Host/Cluster	Resource Group	Policies	Last Backup	Overall Status
Monitor	cdb1	Single Instance (Multitenant)	ora_01.vmc.local	ora_01_data ora_01_logs	Oracle Archive Logs Backup Oracle Online Full Backup	10/07/2023 12:00:25 PM	Backup succeeded
Reports	cdb2	Single Instance (Multitenant)	ora_01.vmc.local	ora_01_data ora_01_logs	Oracle Archive Logs Backup Oracle Online Full Backup	10/07/2023 12:00:25 PM	Backup succeeded
Hosts	cdb3	Single Instance (Multitenant)	ora_02.vmc.local	ora_02_data ora_02_logs	Oracle Archive Logs Backup Oracle Online Full Backup	10/07/2023 8:05:25 AM	Backup succeeded
Storage Systems	cdb4	Single Instance (Multitenant)	ora_02.vmc.local	ora_02_data ora_02_logs	Oracle Archive Logs Backup Oracle Online Full Backup	10/07/2023 8:05:25 AM	Backup succeeded

6. 데이터베이스를 클릭하여 각 데이터베이스의 백업 세트를 검토합니다.

The screenshot shows the NetApp SnapCenter interface for Oracle Database management. The main content area displays the 'Primary Backup(s)' table for the 'cdb3' topology. The table lists various backup entries with columns for Backup Name, Count, Type, JF, End Date, Verified, Mounted, RMAN Cataloged, and SCN. A summary card on the right indicates 22 Backups, 8 Data Backups, 14 Log Backups, and 0 Clones.

Backup Name	Count	Type	JF	End Date	Verified	Mounted	RMAN Cataloged	SCN
ora_02_10-07-2023_08.05.02.4105_1	1	Log		10/07/2023 8:05:26 AM	Not Applicable	False	Not Cataloged	2928738
ora_02_10-07-2023_07.50.02.4250_1	1	Log		10/07/2023 7:50:27 AM	Not Applicable	False	Not Cataloged	2927731
ora_02_10-07-2023_07.45.02.4192_1	1	Log		10/07/2023 7:45:49 AM	Not Applicable	False	Not Cataloged	2927497
ora_02_10-07-2023_07.45.02.4192_0	1	Data		10/07/2023 7:45:31 AM	Unverified	False	Not Cataloged	2927446
ora_02_10-07-2023_07.35.02.3846_1	1	Log		10/07/2023 7:35:25 AM	Not Applicable	False	Not Cataloged	2926747
ora_02_10-07-2023_07.20.02.3803_1	1	Log		10/07/2023 7:20:25 AM	Not Applicable	False	Not Cataloged	2925995
ora_02_10-07-2023_07.05.02.3948_1	1	Log		10/07/2023 7:05:26 AM	Not Applicable	False	Not Cataloged	2924987
ora_02_10-07-2023_06.50.02.3786_1	1	Log		10/07/2023 6:50:26 AM	Not Applicable	False	Not Cataloged	2923925

데이터베이스 복구

SnapCenter는 스냅샷 백업을 통해 Oracle 데이터베이스를 위한 다양한 복원 및 복구 옵션을 제공합니다. 이 예에서는 삭제된 테이블을 실수로 복구하는 시점 복원을 보여 줍니다. VM ora_02에서 두 개의 데이터베이스 cdb3, cdb4는 동일한 + 데이터 및 + 로그 디스크 그룹을 공유합니다. 한 데이터베이스에 대한 데이터베이스 복원은 다른 데이터베이스의 가용성에 영향을 주지 않습니다.

1. 먼저 테스트 테이블을 만들고 테이블에 행을 삽입하여 시점 복구를 확인합니다.

```
[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. 우리는 SnapCenter에서 수동 스냅샷 백업을 실행합니다. 그런 다음 테이블을 놓습니다.

```

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. 마지막 단계에서 생성된 백업 세트에서 로그 백업의 SCN 번호를 기록합니다. 을 클릭합니다 Restore 복원-복구 워크플로우를 시작합니다.

The screenshot shows the NetApp SnapCenter interface for an Oracle database. The main area displays a table of primary backups with columns for Backup Name, Count, Type, IF, End Date, Verified, Mounted, RMAN Cataloged, and SCN. A summary card on the right shows 6 Backups, 2 Data Backups, 4 Log Backups, and 0 Clones.

Backup Name	Count	Type	IF	End Date	Verified	Mounted	RMAN Cataloged	SCN
ora_02_10-06-2023_14.22.59.0383_1	1	Log		10/06/2023 2:23:43 PM	Not Applicable	False	Not Cataloged	2795205
ora_02_10-06-2023_14.22.59.0383_0	1	Data		10/06/2023 2:23:27 PM	Unverified	False	Not Cataloged	2795113
ora_02_10-06-2023_14.20.01.8472_1	1	Log		10/06/2023 2:20:24 PM	Not Applicable	False	Not Cataloged	2794928
ora_02_10-06-2023_14.05.01.8346_1	1	Log		10/06/2023 2:05:04 PM	Not Applicable	False	Not Cataloged	2793950
ora_02_10-06-2023_13.52.09.1111_1	1	Log		10/06/2023 1:52:59 PM	Not Applicable	False	Not Cataloged	2792888
ora_02_10-06-2023_13.52.09.1111_0	1	Data		10/06/2023 1:52:43 PM	Unverified	False	Not Cataloged	2792838

4. 복구 범위를 선택합니다.

Restore cdb3

1 Restore Scope

2 Recovery Scope

3 PreOps

4 PostOps

5 Notification

6 Summary

Restore Scope ⓘ

All Datafiles

Pluggable databases (PDBs)

Pluggable database (PDB) tablespaces

Control files

Database State

Change database state if needed for restore and recovery

Restore Mode ⓘ

Force in place restore

If this check box is not selected and if any of the in place restore criteria is not met, restore will be performed using the connect and copy method. The connect and copy restore method might take time based on the files being restored.

Previous Next

5. 마지막 전체 데이터베이스 백업에서 로그 SCN까지 복구 범위를 선택합니다.

Restore cdb3

1 Restore Scope

2 Recovery Scope

3 PreOps

4 PostOps

5 Notification

6 Summary

Choose Recovery Scope

All Logs

Until SCN (System Change Number)

SCN

Date and Time

No recovery

Specify external archive log files locations

i After the operation is complete, it is recommended to create a full backup of the Oracle database.

Previous Next

6. 실행할 사전 스크립트를 지정합니다.

Restore cdb3 x

1 Restore Scope

2 Recovery Scope

3 PreOps

4 PostOps

5 Notification

6 Summary

Specify optional scripts to run before performing a restore job ⓘ

Prescript full path

Arguments

Script timeout

7. 실행할 선택적 스크립트 후 를 지정합니다.

Restore cdb3 x

1 Restore Scope

2 Recovery Scope

3 PreOps

4 PostOps

5 Notification

6 Summary

Specify optional scripts to run after performing a restore job ⓘ

Postscript full path

Arguments

Open the database or container database in READ-WRITE mode after recovery

8. 필요한 경우 작업 보고서를 전송합니다.

Restore cdb3 ×

- 1 Restore Scope
- 2 Recovery Scope
- 3 PreOps
- 4 PostOps
- 5 Notification**
- 6 Summary

Provide email settings ⓘ

Email preference:

From:

To:

Subject:

Attach job report

9. 요약을 검토하고 **Finish** 를 클릭합니다 **Finish** 를 눌러 복원 및 복구를 시작합니다.

Restore cdb3
x

- 1 Restore Scope
- 2 Recovery Scope
- 3 PreOps
- 4 PostOps
- 5 Notification
- 6 Summary

Summary

Backup name	ora_02_10-06-2023_14.22.59.0383_0
Backup date	10/06/2023 2:23:27 PM
Restore scope	All DataFiles
Recovery scope	Until SCN 2795205
Auxiliary destination	
Options	Change database state if necessary , Open the database or container database in READ-WRITE mode after recovery
Prescript full path	None
Prescript arguments	
Postscript full path	None
Postscript arguments	
Send email	No

Previous
Finish

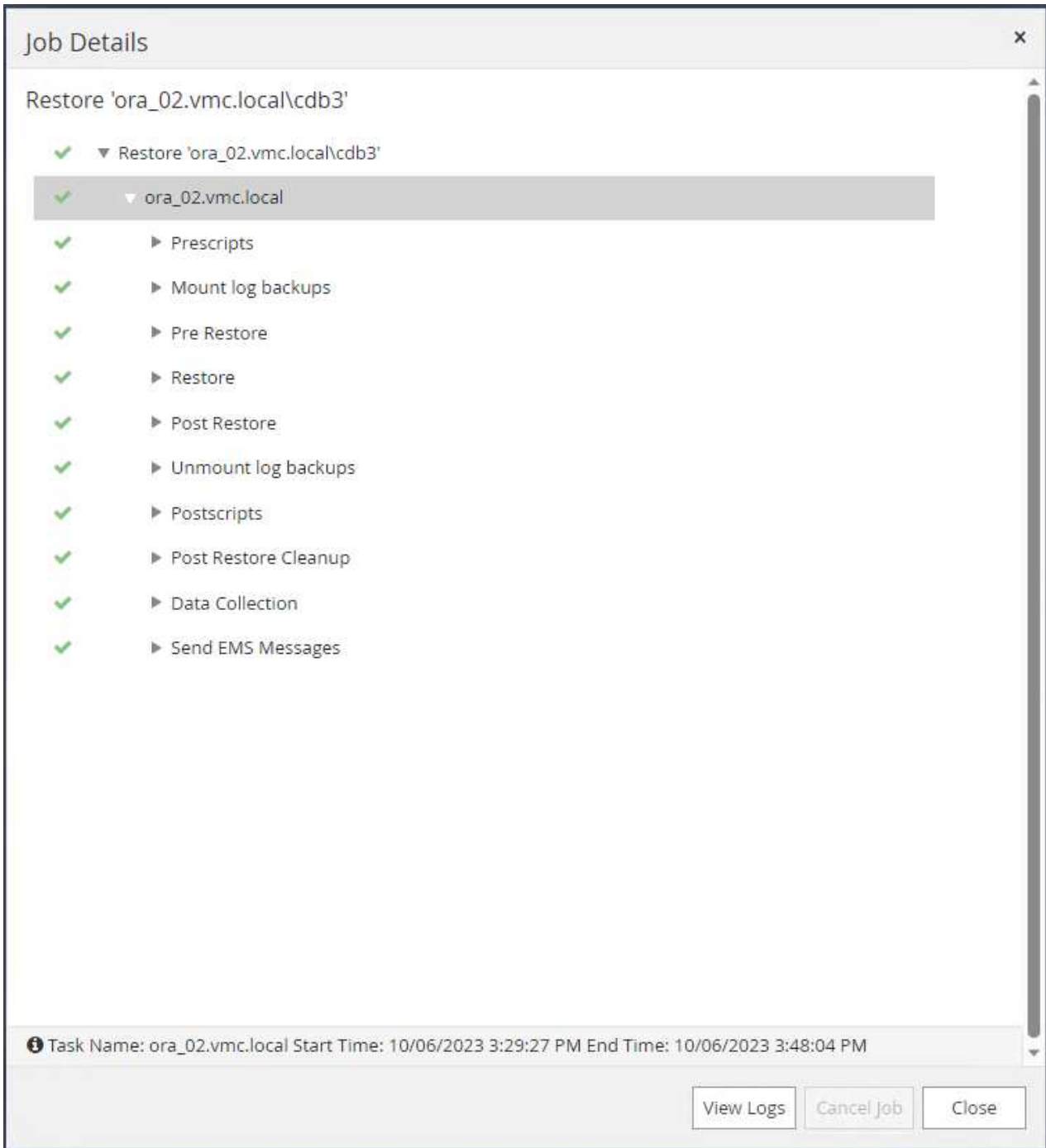
10. Oracle Restart 그리드 제어에서 cdb3이 복원 중이며 복구 cdb4가 온라인 상태이며 사용 가능한 것으로 관찰됩니다.


```

[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
                    registered, 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. 보낸 사람 Monitor 탭에서 작업을 열어 세부 정보를 검토합니다.



12. DB VM ora_02에서 삭제된 테이블이 성공적으로 복구되었는지 확인합니다.

```
[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
1

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

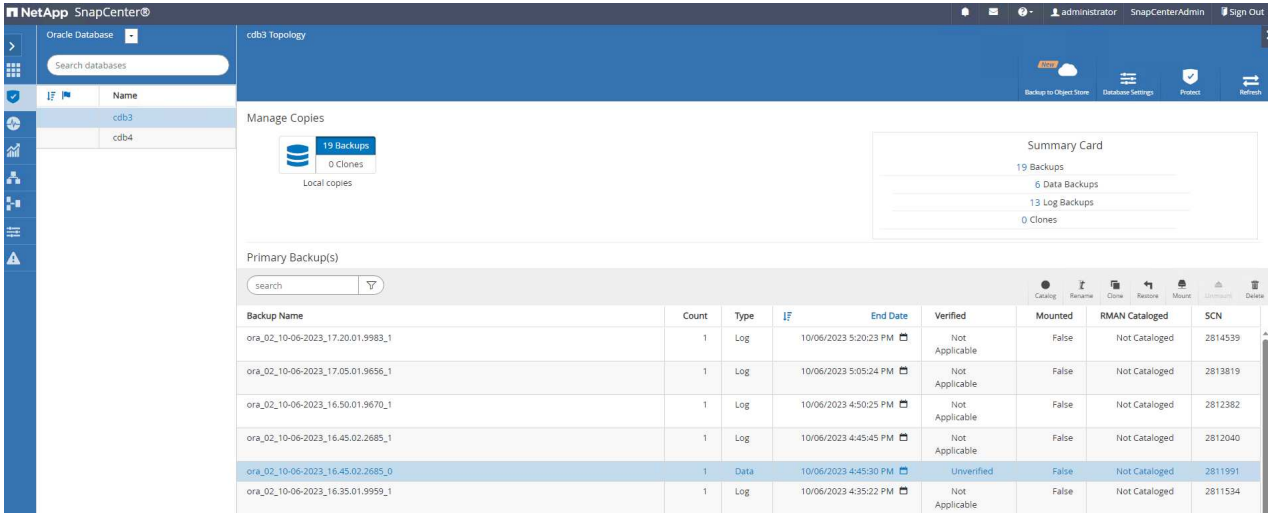
SQL> select current_timestamp from dual;

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

SQL>

이 예에서는 동일한 백업 세트를 사용하여 다른 ORACLE_HOME의 동일한 VM에 있는 데이터베이스를 복제합니다. 이 절차는 필요한 경우 백업에서 VMC의 개별 VM으로 데이터베이스를 복제하는 경우에도 동일하게 적용됩니다.

1. 데이터베이스 cdb3 백업 목록을 엽니다. 선택한 데이터 백업에서 를 클릭합니다 Clone 데이터베이스 복제 워크플로우를 시작하는 버튼



2. 클론 데이터베이스 SID의 이름을 지정합니다.

Clone from cdb3

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

7 Summary

Complete Database Clone

Clone SID:

Exclude PDBs:

PDB Clone

Previous Next

3. VMC의 VM을 타겟 데이터베이스 호스트로 선택합니다. 호스트에 동일한 Oracle 버전이 설치 및 구성되어 있어야 합니다.

x
Clone from cdb3

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

7 Summary

Select the host to create a clone

Clone host

Datafile locations ⓘ

+SC_2090922_cdb3tst

Control files ⓘ

+SC_2090922_cdb3tst/cdb3tst/control/control01.ctl

+SC_2090922_cdb3tst/cdb3tst/control/control02.ctl

Redo logs ⓘ

Group	Size	Unit	Number of files
▶ RedoGroup 1	200	MB	2
▶ RedoGroup 2	200	MB	2
▶ RedoGroup 3	200	MB	2

4. 타겟 호스트에서 적절한 ORACLE_HOME, 사용자 및 그룹을 선택합니다. 자격 증명을 기본값으로 유지합니다.

Clone from cdb3

1 Name

2 Locations

3 **Credentials**

4 PreOps

5 PostOps

6 Notification

7 Summary

Database Credentials for the clone

Credential name for sys user: None + ⓘ

ASM instance Credential name: None + ⓘ

Database port: 1521

ASM Port: 1521

Oracle Home Settings ⓘ

Oracle Home: /u01/app/oracle/product/19.0.0/cdb4

Oracle OS User: oracle

Oracle OS Group: oinstall

Previous Next

5. 클론 데이터베이스의 구성 또는 리소스 요구 사항을 충족하도록 클론 데이터베이스 매개 변수를 변경합니다.

Clone from cdb3
✕

- 1 Name
- 2 Locations
- 3 Credentials
- 4 PreOps
- 5 PostOps
- 6 Notification
- 7 Summary

Specify scripts to run before clone operation ?

Prescript full path

Arguments

Script timeout secs

Database Parameter settings

processes	320	✕	▲
remote_login_passwordfile	EXCLUSIVE	✕	+
sga_target	2048M	✕	▼
undo_tablespace	UNDOTBS1	✕	▼

6. 복구 범위를 선택합니다. Until Cancel 백업 세트에서 사용 가능한 마지막 로그 파일까지 클론을 복구합니다.

Clone from cdb3
✕

- 1 Name
- 2 Locations
- 3 Credentials
- 4 PreOps
- 5 PostOps
- 6 Notification
- 7 Summary

Recover Database

Until Cancel ?

Date and Time ?

Date-time format: MM/DD/YYYY hh:mm:ss

 Until SCN (System Change Number) ?

Specify external archive log locations ?

Create new DBID ?

Create tempfile for temporary tablespace ?

Enter SQL queries to apply when clone is created

Enter scripts to run after clone operation ?

7. 요약을 검토하고 클론 작업을 시작합니다.

Clone from cdb3

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

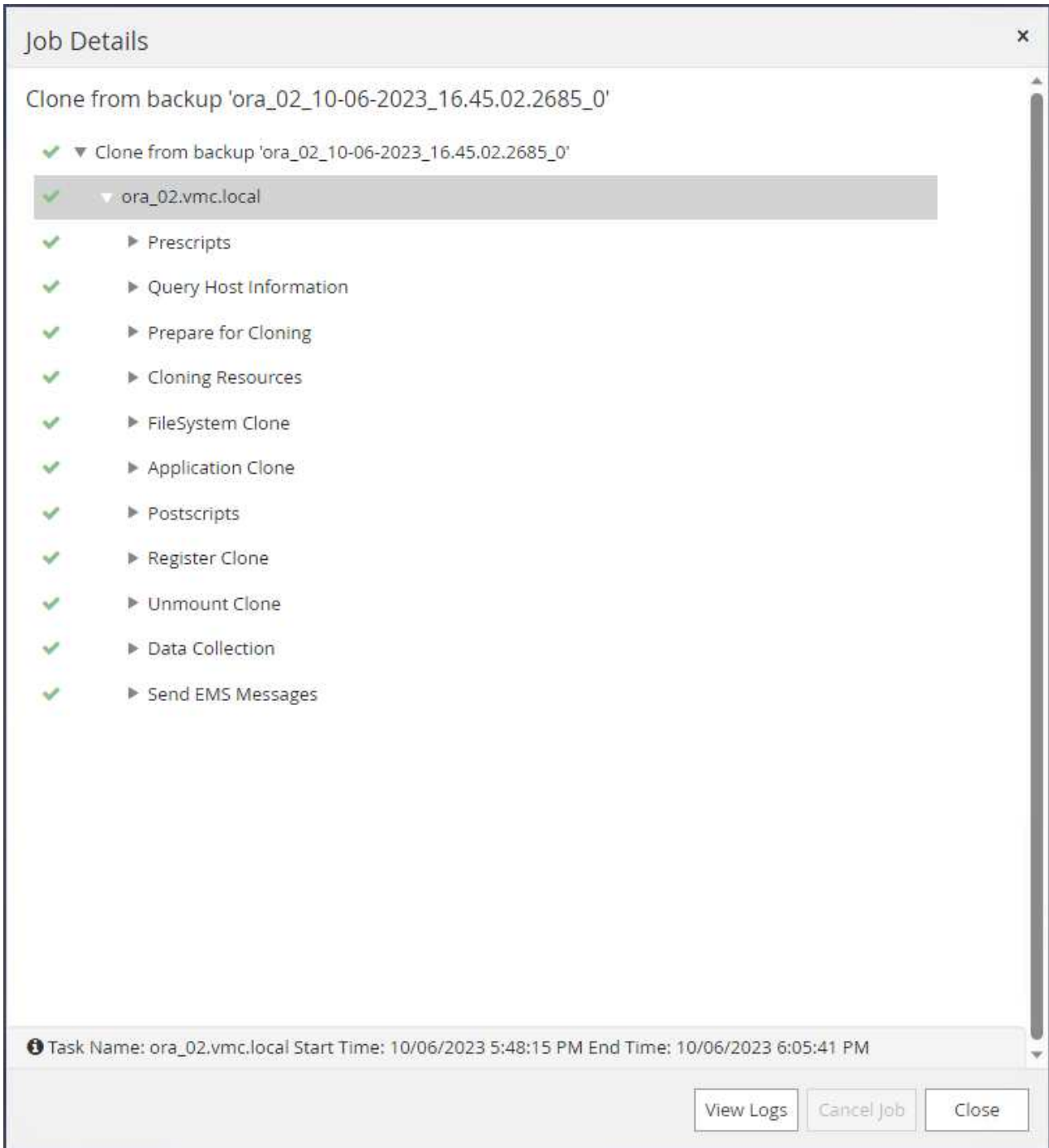
7 Summary

Summary

Clone from backup	ora_02_10-06-2023_16.45.02.2685_0
Clone SID	cdb3tst
Clone server	ora_01.vmc.local
Exclude PDBs	none
Oracle home	/u01/app/oracle/product/19.0.0/cdb2
Oracle OS user	oracle
Oracle OS group	oinstall
Datafile mountpaths	+SC_2090922_cdb3tst
Control files	+SC_2090922_cdb3tst/cdb3tst/control/control01.ctl +SC_2090922_cdb3tst/cdb3tst/control/control02.ctl
Redo groups	RedoGroup =1 TotalSize =200 Path =+SC_2090922_cdb3tst/cdb3tst/redo01_01.log RedoGroup =1 TotalSize =200 Path =+SC_2090922_cdb3tst/cdb3tst/redo01_02.log RedoGroup =2 TotalSize =200 Path =+SC_2090922_cdb3tst/cdb3tst/redo02_01.log RedoGroup =2 TotalSize =200 Path =+SC_2090922_cdb3tst/cdb3tst/redo02_02.log RedoGroup =3 TotalSize =200 Path =+SC_2090922_cdb3tst/cdb3tst/redo03_01.log RedoGroup =3 TotalSize =200 Path =+SC_2090922_cdb3tst/cdb3tst/redo03_02.log
Recovery scope	Until Cancel
Prescript full path	none
Prescript arguments	
Postscript full path	none
Postscript arguments	
Send email	No

Previous Finish

8. 에서 클론 작업 실행을 모니터링합니다 Monitor 탭을 클릭합니다.



9. 복제된 데이터베이스는 즉시 SnapCenter에 등록됩니다.

Name	Oracle Database Type	Host/Cluster	Resource Group	Policies	Last Backup	Overall Status
cdb1	Single Instance (Multitenant)	ora_01.vmc.local				Not protected
cdb2	Single Instance (Multitenant)	ora_01.vmc.local				Not protected
cdb3	Single Instance (Multitenant)	ora_02.vmc.local	ora_02_data ora_02_logs	Oracle Archive Logs Backup Oracle Online Full Backup	10/06/2023 6:20:23 PM	Backup succeeded
cdb3st	Single Instance (Multitenant)	ora_02.vmc.local				Not protected
cdb4	Single Instance (Multitenant)	ora_02.vmc.local	ora_02_data ora_02_logs	Oracle Archive Logs Backup Oracle Online Full Backup	10/06/2023 6:20:23 PM	Backup succeeded

10. DB VM ora_02에서 복제된 데이터베이스도 Oracle Restart grid control에 등록되고 삭제된 테스트 테이블은 아래와 같이 복제된 데이터베이스 cdb3tst에 복구됩니다.

```
[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
```

```
oracle/product/19.0.0
```

```
/cdb4,STABLE
```

```
ora.cdb4.db
```

```
1 ONLINE ONLINE ora_02  
Open,HOME=/u01/app/o
```

```
oracle/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>
```

이것으로 AWS의 VMC SDDC에서 Oracle 데이터베이스의 SnapCenter 백업, 복구 및 클론 복제에 대한 데모를 마치겠습니다.

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 문서 및/또는 웹 사이트를 참조하십시오.

- VMware Cloud on AWS 설명서 를 참조하십시오

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

- 새 데이터베이스 설치를 통해 독립 실행형 서버용 Oracle Grid Infrastructure 설치

["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"](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)

- 응답 파일을 사용하여 Oracle 데이터베이스 설치 및 구성

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

- NetApp ONTAP용 Amazon FSx

["https://aws.amazon.com/fsx/netapp-ontap/"](https://aws.amazon.com/fsx/netapp-ontap/)

TR-4981: Amazon FSx ONTAP을 사용한 Oracle Active Data Guard 비용 절감

Allen Cao, Niyaz Mohamed, NetApp

목적

Oracle Data Guard는 기본 데이터베이스 및 대기 데이터베이스 복제 구성의 엔터프라이즈 데이터에 대한 고가용성, 데이터 보호 및 재해 복구를 보장합니다. Oracle Active Data Guard는 기본 데이터베이스에서 대기 데이터베이스로 데이터 복제가 활성 상태인 동안 사용자가 대기 데이터베이스에 액세스할 수 있도록 합니다. Data Guard는 Oracle Database Enterprise Edition의 기능입니다. 별도의 라이선스가 필요하지 않습니다. 반면 Active Data Guard는 Oracle Database Enterprise Edition 옵션이므로 별도의 라이선스가 필요합니다. 여러 대기 데이터베이스가 Active Data Guard 설정의 기본 데이터베이스로부터 데이터 복제를 수신할 수 있습니다. 그러나 각각의 추가 대기 데이터베이스에는 활성 Data Guard 라이선스와 기본 데이터베이스의 크기로 추가 스토리지가 필요합니다. 운영 비용이 빠르게 증가합니다.

Oracle 데이터베이스 작업의 비용 절감에 관심이 있고 AWS에서 Active Data Guard를 설정할 계획이라면 대안을 고려해야 합니다. Active Data Guard 대신 Data Guard를 사용하여 기본 데이터베이스에서 Amazon FSx ONTAP 스토리지의 단일 물리적 대기 데이터베이스로 복제합니다. 그런 다음 이 대기 데이터베이스의 여러 복사본을 복제해 읽기/쓰기 액세스를 위해 열어 보고, 개발, 테스트 등과 같은 다른 여러 활용 사례를 지원할 수 있습니다. 그 결과 Active Data Guard의 기능을 효과적으로 제공하는 동시에 추가 대기 데이터베이스당 Active Data Guard 라이선스 및 추가 스토리지 비용을 절감할 수 있습니다. 이 문서에서는 AWS의 기존 기본 데이터베이스로 Oracle Data Guard를 설정하고 Amazon FSx ONTAP 스토리지에 물리적 대기 데이터베이스를 배치하는 방법을 설명합니다. 대기 데이터베이스는 스냅샷을 통해 백업되고 필요에 따라 활용 사례를 위해 읽기/쓰기 액세스를 위해 클론됩니다.

이 솔루션은 다음과 같은 사용 사례를 해결합니다.

- Oracle Data Guard는 AWS의 모든 스토리지에 있는 기본 데이터베이스와 Amazon FSx ONTAP 스토리지의 대기 데이터베이스 사이에 있습니다.
- 보고, 개발, 테스트 등과 같은 활용 사례를 제공하기 위해 데이터 복제를 위해 닫혀 있는 동안 대기 데이터베이스의 클론을 생성합니다

대상

이 솔루션은 다음과 같은 사용자를 대상으로 합니다.

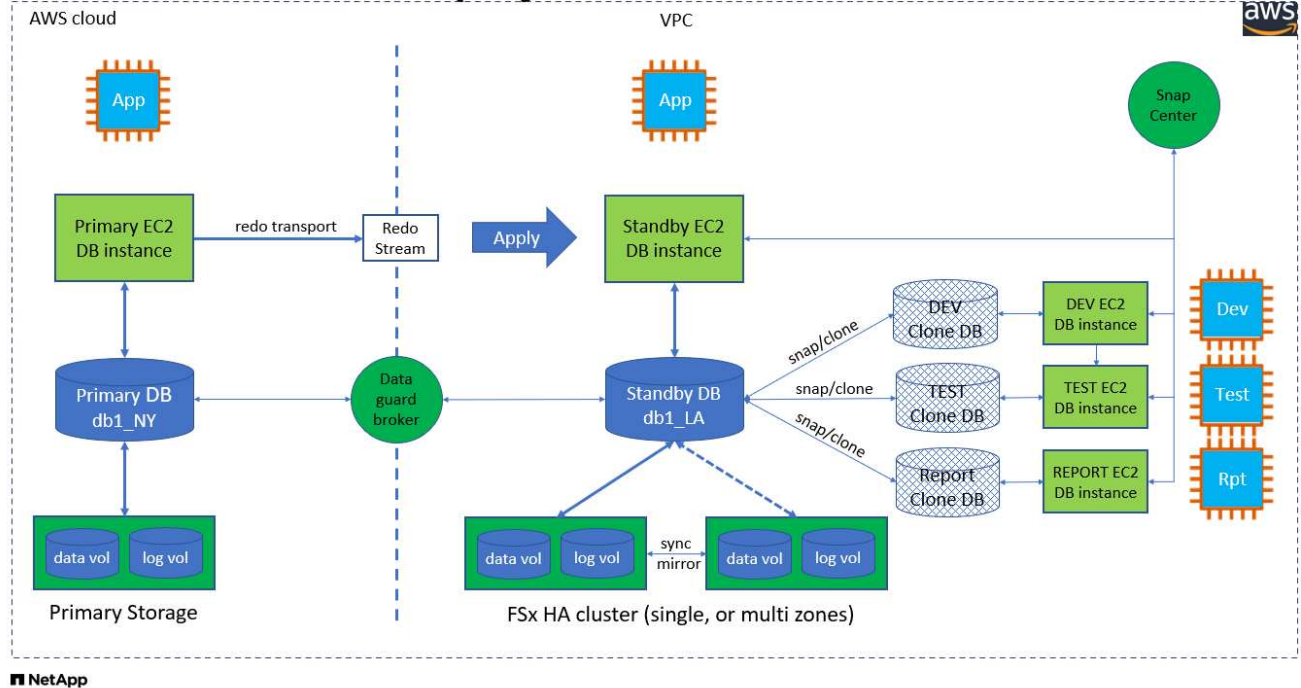
- 고가용성, 데이터 보호 및 재해 복구를 위해 AWS에서 Oracle Active Data Guard를 설정한 DBA
- AWS 클라우드에서 Oracle Active Data Guard 구성에 관심이 있는 데이터베이스 솔루션 설계자
- Oracle Data Guard를 지원하는 AWS FSx ONTAP 스토리지를 관리하는 스토리지 관리자
- AWS FSx/EC2 환경에서 Oracle Data Guard를 실행하려는 애플리케이션 소유자.

솔루션 테스트 및 검증 환경

이 솔루션의 테스트 및 검증은 최종 구축 환경과 일치하지 않을 수 있는 AWS FSx ONTAP 및 EC2 연구소 환경에서 수행되었습니다. 자세한 내용은 섹션을 참조하십시오 [\[Key Factors for Deployment Consideration\]](#).

있습니다

Oracle Data Guard Deployment with Amazon FSx for ONTAP



하드웨어 및 소프트웨어 구성 요소

* 하드웨어 *		
FSX ONTAP 저장소	AWS에서 제공하는 현재 버전입니다	동일한 VPC 및 가용성 존에 FSx HA 클러스터 1개
컴퓨팅용 EC2 인스턴스	T2.xLarge/4vCPU/16G	EC2 T2 xlarge EC2 인스턴스 3개, 하나는 운영 DB 서버로, 다른 하나는 대기 DB 서버로, 다른 하나는 클론 DB 서버로 사용됩니다
* 소프트웨어 *		
RedHat Linux	RHEL-8.6.0_HVM-20220503-x86_64-2-Hourly2-GP2	테스트를 위해 RedHat 서브스크립션을 배포했습니다
Oracle Grid Infrastructure	버전 19.18	RU 패치 p34762026_190000_Linux-x86-64.zip 를 적용했습니다
Oracle 데이터베이스	버전 19.18	RU 패치 p34765931_190000_Linux-x86-64.zip 를 적용했습니다

Oracle OPatch	버전 12.2.0.1.36	최신 패치 p6880880_190000_Linux-x86-64.zip
---------------	----------------	--

가상 NY to LA DR 설정을 사용한 Oracle Data Guard 구성

* 데이터베이스 *	* DB_UNIQUE_NAME *	* Oracle Net 서비스 이름 *
기본	DB1_NY입니다	db1_NY.demo.netapp.com
물리적 대기	DB1_LA를 클릭합니다	db1_LA.demo.netapp.com

구축 시 고려해야 할 주요 요소

- * Oracle 대기 데이터베이스 FlexClone 작동 방식 * AWS FSx ONTAP FlexClone은 쓰기 가능한 동일한 대기 데이터베이스 볼륨의 공유 복사본을 제공합니다. 볼륨의 복사본은 클론에서 새 쓰기가 시작될 때까지 원래 데이터 블록에 다시 연결되는 포인터입니다. 그런 다음 ONTAP은 새 쓰기에 대해 새 스토리지 블록을 할당합니다. 모든 읽기 IO는 활성 복제 상태의 원본 데이터 블록에 의해 처리됩니다. 따라서 클론은 스토리지 효율성이 매우 뛰어나 새로운 쓰기 입출력에 대해 최소한의 증분 새 스토리지 할당으로 다른 많은 활용 사례에서 사용할 수 있습니다. 따라서 Active Data Guard 스토리지 설치 공간을 크게 줄여 스토리지 비용을 크게 절감할 수 있습니다. 데이터베이스가 운영 스토리지에서 대기 FSx 스토리지로 전환될 때 Oracle 성능을 최고 수준으로 유지하기 위해 FlexClone 활동을 최소화하는 것이 좋습니다 NetApp.
- * Oracle 소프트웨어 요구 사항. * 일반적으로 물리적 대기 데이터베이스는 패치 세트 예외(PSE), 중요 패치 업데이트(CPU), 및 패치 세트 업데이트(PSU)(의 My Oracle Support Note 1265700.1 에 설명된 대로 Oracle Data Guard Standby-First Patch Apply 프로세스가 진행 중인 경우 제외 ["support.oracle.com"](https://support.oracle.com))
- * 대기 데이터베이스 디렉터리 구조 고려 사항. 가능한 경우 기본 및 대기 시스템의 데이터 파일, 로그 파일 및 제어 파일은 이름과 경로 이름이 같아야 하며 OFA(Optimal Flexible Architecture) 명명 규칙을 사용해야 합니다. 대기 데이터베이스의 보관 디렉터리도 크기 및 구조를 포함하여 사이트 간에 동일해야 합니다. 이 전략을 통해 백업, 스위치오버 및 페일오버 같은 다른 작업에서 동일한 단계를 실행할 수 있으므로 유지 관리의 복잡성이 줄어듭니다.
- * 강제 로깅 모드. * 대기 데이터베이스로 전파할 수 없는 기본 데이터베이스에서 로깅되지 않은 직접 쓰기로부터 보호하려면 대기 생성을 위한 데이터 파일 백업을 수행하기 전에 기본 데이터베이스에서 강제 로깅을 켜십시오.
- * 데이터베이스 스토리지 관리. * 운영을 단순화하기 위해 Oracle Data Guard 구성에서 Oracle 자동 스토리지 관리(Oracle ASM) 및 Oracle 관리 파일(OMF)을 설정할 때 기본 및 대기 데이터베이스에 대칭적으로 설정할 것을 권장합니다.
- * EC2 컴퓨팅 인스턴스 * 이러한 테스트 및 검증에서는 AWS EC2 T2.xLarge 인스턴스를 Oracle 데이터베이스 컴퓨팅 인스턴스로 사용했습니다. NetApp은 운영 구축 시 데이터베이스 워크로드에 최적화되어 있는 M5 유형 EC2 인스턴스를 Oracle의 컴퓨팅 인스턴스로 사용할 것을 권장합니다. 실제 워크로드 요구 사항에 따라 vCPU 수와 RAM 양에 따라 EC2 인스턴스를 적절하게 사이징해야 합니다.
- * FSx 스토리지 HA 클러스터 단일 또는 다중 영역 배포. * 이러한 테스트 및 검증에서는 단일 AWS 가용성 영역에 FSx HA 클러스터를 구축했습니다. 프로덕션 배포를 위해 FSx HA 쌍을 두 가지 가용성 영역에 배포하는 것이 좋습니다. FSx 클러스터는 스토리지 레벨 이중화를 제공하기 위해 액티브-패시브 파일 시스템 쌍으로 동기화되는 HA 쌍으로 프로비저닝됩니다. 다중 영역 구축을 통해 단일 AWS 영역에서 장애가 발생할 경우 고가용성을 더욱 강화할 수 있습니다.
- * FSx 스토리지 클러스터 크기 조정 * ONTAP 스토리지 파일 시스템용 Amazon FSx는 최대 160,000개의 원시 SSD IOPS, 최대 4Gbps 처리량 및 최대 192TiB 용량을 제공합니다. 그러나 구현 시 실제 요구 사항에 따라 프로비저닝된 IOPS, 처리량 및 스토리지 제한(최소 1,024GiB)을 기준으로 클러스터 크기를 조정할 수 있습니다. 애플리케이션 가용성에 영향을 주지 않고 용량을 즉각적으로 동적으로 조정할 수 있습니다.

솔루션 구축

Data Guard를 설정하기 위한 시작점으로 VPC 내의 AWS EC2 환경에 기본 Oracle 데이터베이스가 이미 구축되어 있다고 가정합니다. 기본 데이터베이스는 스토리지 관리를 위해 Oracle ASM을 사용하여 구축됩니다. Oracle 데이터 파일, 로그 파일 및 제어 파일 등을 위해 2개의 ASM 디스크 그룹 - + 데이터 및 + 로그가 생성됩니다. AWS에서 ASM을 사용한 Oracle 배포에 대한 자세한 내용은 다음 기술 보고서를 참조하십시오.

- ["EC2 및 FSx Best Practices에 Oracle Database 구축"](#)
- ["iSCSI/ASM이 포함된 AWS FSx/EC2에서 Oracle 데이터베이스 구축 및 보호"](#)
- ["NFS/ASM을 사용하는 AWS FSx/EC2에서 독립 실행형 재시작 시 Oracle 19c"](#)

기본 Oracle 데이터베이스는 FSx ONTAP 또는 AWS EC2 에코시스템 내의 선택한 다른 스토리지에서 실행될 수 있습니다. 다음 섹션에서는 ASM 스토리지가 있는 기본 EC2 DB 인스턴스와 ASM 스토리지가 있는 대기 EC2 DB 인스턴스 간에 Oracle Data Guard를 설정하는 단계별 구축 절차를 제공합니다.

배포를 위한 사전 요구 사항

배포에는 다음과 같은 사전 요구 사항이 필요합니다.

1. AWS 계정이 설정되었으며 AWS 계정 내에 필요한 VPC 및 네트워크 세그먼트가 생성되었습니다.
2. AWS EC2 콘솔에서 최소 3개의 EC2 Linux 인스턴스를 구축해야 합니다. 하나는 운영 Oracle DB 인스턴스로, 하나는 대기 Oracle DB 인스턴스로, 또 다른 하나는 보고, 개발, 테스트를 위한 클론 타겟 DB 인스턴스로 구축됩니다. 환경 설정에 대한 자세한 내용은 이전 섹션의 아키텍처 다이어그램을 참조하십시오. 또한 AWS에 대해서도 검토합니다 ["Linux 인스턴스에 대한 사용자 가이드"](#) 를 참조하십시오.
3. AWS EC2 콘솔에서 Amazon FSx for ONTAP 스토리지 HA 클러스터를 구축하여 Oracle 대기 데이터베이스를 저장하는 Oracle 볼륨을 호스팅합니다. FSx 저장소 배포에 익숙하지 않은 경우 설명서를 참조하십시오 ["ONTAP 파일 시스템용 FSx 생성"](#) 을 참조하십시오.
4. 2단계와 3단계는 라는 EC2 인스턴스를 만드는 다음 Terraform 자동화 툴킷을 사용하여 수행할 수 있습니다 ora_01 이라는 FSx 파일 시스템이 있습니다 fsx_01. 실행 전에 지침을 주의 깊게 검토하고 환경에 맞게 변수를 변경하십시오. 사용자 고유의 배포 요구 사항에 맞게 템플릿을 쉽게 수정할 수 있습니다.

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



Oracle 설치 파일을 스테이징할 충분한 공간을 확보하기 위해 EC2 인스턴스 루트 볼륨에 50G 이상을 할당했는지 확인합니다.

Data Guard용 기본 데이터베이스를 준비합니다

이 데모에서는 기본 EC2 DB 인스턴스에 db1이라는 기본 Oracle 데이터베이스를 설정했습니다. 독립 실행형 Restart 구성의 ASM 디스크 그룹 2개와 ASM 디스크 그룹 + 데이터 및 ASM 디스크 그룹 + 로그의 플래시 복구 영역에 있는 데이터 파일이 포함된 ASM 디스크 그룹을 사용했습니다. 다음은 Data Guard용 기본 데이터베이스를 설정하는 자세한 절차를 보여 줍니다. 모든 단계는 데이터베이스 소유자인 Oracle 사용자로 실행해야 합니다.

1. 운영 EC2 DB 인스턴스 IP-172-30-15-45의 기본 데이터베이스 db1 구성 ASM 디스크 그룹은 EC2 에코시스템 내 모든 유형의 스토리지에 있을 수 있습니다.

```
[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.dbf.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
ora.driver.afd
      1      ONLINE  ONLINE      ip-172-30-15-45      STABLE
ora.evmd
      1      ONLINE  ONLINE      ip-172-30-15-45      STABLE
-----

```

2. sqlplus에서 기본 로그온을 활성화합니다.

```
alter database force logging;
```

3. sqlplus에서 primary에서 플래시백을 활성화합니다. 플래시백은 장애 조치 후 기본 데이터베이스를 대기 데이터베이스로 쉽게 복원할 수 있도록 합니다.

```
alter database flashback on;
```

- Oracle 암호 파일을 사용하여 redo 전송 인증 구성 - 설정하지 않은 경우 orapwd 유틸리티를 사용하여 기본 데이터베이스에 pwd 파일을 생성하고 대기 데이터베이스 \$oracle_home/dbs 디렉토리로 복사합니다.
- 현재 온라인 로그 파일과 크기가 같은 기본 DB에 대기 redo 로그를 생성합니다. 로그 그룹은 하나 이상의 온라인 로그 파일 그룹입니다. 그러면 기본 데이터베이스가 신속하게 대기 역할로 전환되고 필요한 경우 다시 실행 데이터 수신을 시작할 수 있습니다.

```
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.

- sqlplus 에서 편집할 spfile 에서 pfile 을 만듭니다.

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

- pfile을 수정하고 다음 매개 변수를 추가합니다.

```
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. sqlplus에서 /home/oracle 디렉토리의 수정된 pfile에서 asm+ data 디렉토리에 spfile을 생성합니다.

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

9. data disk group(필요한 경우 asmcmd 유틸리티 사용)에서 새로 생성된 spfile을 찾습니다. srvctl을 사용하여 아래 그림과 같이 새 spfile에서 데이터베이스를 시작하도록 그리드를 수정합니다.

```

[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. 이름 확인을 위해 db_unique_name을 추가하려면 tnsnames.ora를 수정하십시오.


```
# 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. 기본 데이터베이스에 대한 데이터 가드 서비스 이름 db1_NY_DGMGRL.demo.netapp 을 listener.ora 파일에 추가합니다.

```
#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. `srvctl`을 사용하여 데이터베이스를 종료하고 다시 시작하고 데이터 가드 매개변수가 현재 활성 상태인지 확인합니다.

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

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

이것으로 Data Guard에 대한 기본 데이터베이스 설정이 완료되었습니다.

대기 데이터베이스를 준비하고 **Data Guard**를 활성화합니다

Oracle Data Guard를 사용하려면 운영 EC2 DB 인스턴스와 일치하도록 대기 EC2 DB 인스턴스의 패치 세트를 포함한 Oracle 소프트웨어 스택과 OS 커널 구성이 필요합니다. 순쉬운 관리와 간소화를 위해 대기 EC2 DB 인스턴스 데이터베이스 스토리지 구성은 ASM 디스크 그룹의 이름, 수 및 크기 등 기본 EC2 DB 인스턴스와 이상적으로 일치해야 합니다. 다음은 Data Guard용 Standby EC2 DB 인스턴스를 설정하는 세부 절차입니다. 모든 명령은 Oracle 소유자 사용자 ID로 실행해야 합니다.

1. 먼저 운영 EC2 인스턴스에서 운영 데이터베이스의 구성을 검토합니다. 이 데모에서는 독립 실행형 재시작 구성에서 두 개의 ASM 디스크 그룹 + 데이터 및 + 로그를 사용하여 운영 EC2 DB 인스턴스에 db1이라는 기본 Oracle 데이터베이스를 설정했습니다. 기본 ASM 디스크 그룹은 EC2 에코시스템 내의 모든 유형의 스토리지에 있을 수 있습니다.
2. 문서의 절차를 따릅니다 "[TR-4965: iSCSI/ASM을 사용하는 AWS FSx/EC2에서 Oracle 데이터베이스 구축 및 보호](#)" 기본 데이터베이스와 일치하도록 대기 EC2 DB 인스턴스에 그리드 및 Oracle을 설치하고 구성합니다. 데이터베이스 스토리지는 스토리지 용량이 운영 EC2 DB 인스턴스와 동일한 FSx ONTAP의 대기 EC2 DB 인스턴스에 프로비저닝하고 할당해야 합니다.



의 10단계에서 멈춥니다 Oracle database installation 섹션을 참조하십시오. 대기 데이터베이스는 dbca 데이터베이스 복제 기능을 사용하여 기본 데이터베이스에서 인스턴스화됩니다.

3. Oracle 소프트웨어가 설치 및 구성되면 스탠바이 \$ORACLE_HOME DBS 디렉토리에서 기본 데이터베이스에서 Oracle 암호를 복사합니다.

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

4. 다음 항목을 사용하여 tnsnames.ora 파일을 만듭니다.

```
# 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. listener.ora 파일에 DB 데이터 가드 서비스 이름을 추가합니다.

```

#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. Oracle 홈 및 경로를 설정합니다.

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

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

7. dbca를 사용하여 운영 데이터베이스 db1에서 대기 데이터베이스를 인스턴스화합니다.

```

[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. 중복된 대기 데이터베이스를 검증합니다. 새로 복제된 대기 데이터베이스는 처음에 읽기 전용 모드로 열립니다.

```

[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. 에서 대기 데이터베이스를 다시 시작합니다 mount 대기 데이터베이스 관리 복구를 활성화하려면 다음 명령을 준비하고 실행합니다.

```
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. 대기 데이터베이스 복구 상태를 확인합니다. 를 확인합니다 recovery logmerger 인치 APPLYING_LOG 조치.

```
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>
```

이렇게 하면 관리되는 대기 복구가 활성화된 상태에서 db1에 대한 Data Guard 보호 설정이 기본에서 스탠바이로 완료됩니다.

Data Guard Broker를 설정합니다

Oracle Data Guard Broker는 Oracle Data Guard 구성의 생성, 유지 관리 및 모니터링을 자동화하고 중앙 집중화하는 분산 관리 프레임워크입니다. 다음 섹션에서는 Data Guard Broker를 설정하여 Data Guard 환경을 관리하는 방법을 보여 줍니다.

1. sqlplus 를 통해 다음 명령을 사용하여 기본 데이터베이스와 대기 데이터베이스 모두에서 데이터 가드 브로커를 시작합니다.

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

2. 기본 데이터베이스에서 Data Guard Broker에 SYSDBA로 연결합니다.

```
[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. Data Guard Broker 구성을 만들고 활성화합니다.

```
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. Data Guard Broker 관리 프레임워크 내에서 데이터베이스 상태를 검증합니다.

```
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>
```

오류가 발생할 경우 Data Guard Broker를 사용하여 기본 데이터베이스를 즉시 스탠바이로 페일오버할 수 있습니다.

다른 활용 사례에 대한 복제 스탠바이 데이터베이스

Data Guard의 AWS FSx ONTAP에서 대기 데이터베이스를 스테이징할 때의 주요 이점은 최소한의 추가 스토리지 투자로 다른 많은 사용 사례를 지원할 수 있도록 FlexCloned를 사용할 수 있다는 것입니다. 다음 섹션에서는 개발, 테스트, 보고서 등과 같은 다른 목적으로 FSx ONTAP에서 마운트된 복구 대기 데이터베이스 볼륨을 스냅샷하고 복제하는 방법을 보여 줍니다. NetApp SnapCenter 도구를 사용합니다.

다음은 SnapCenter를 사용하여 Data Guard의 관리되는 물리적 대기 데이터베이스에서 읽기/쓰기 데이터베이스를 복제하는 고급 절차입니다. SnapCenter 설정 및 구성 방법에 대한 자세한 내용은 을 참조하십시오 ["SnapCenter을 사용한 하이브리드 클라우드 데이터베이스 솔루션"](#) Oracle 섹션 재지정.

1. 먼저 테스트 테이블을 만들고 기본 데이터베이스의 테스트 테이블에 행을 삽입합니다. 그런 다음 트랜잭션이 대기 상태로 전환되는지 확인하고 마지막으로 클론을 생성합니다.

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

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

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

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

SQL> alter session set container=db1_pdb1;

Session altered.

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

Table created.

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

1 row created.

SQL> commit;

Commit complete.
```

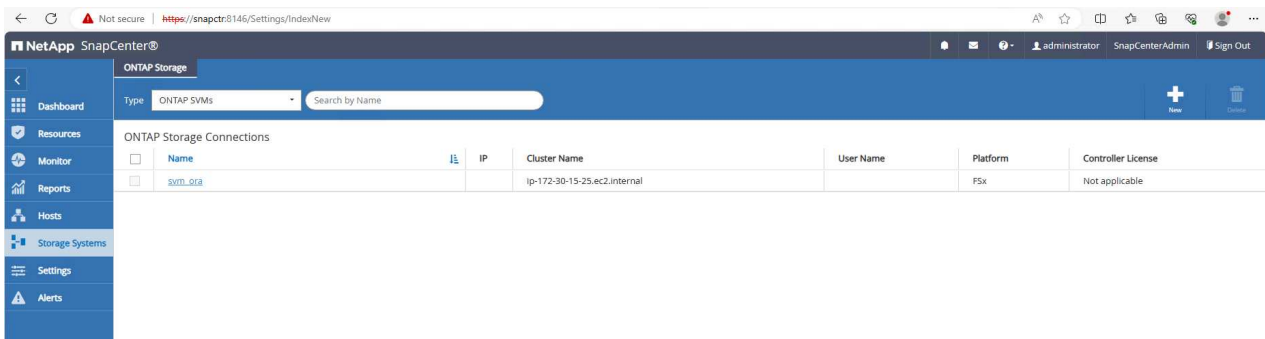
```
SQL> select * from test;
```

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

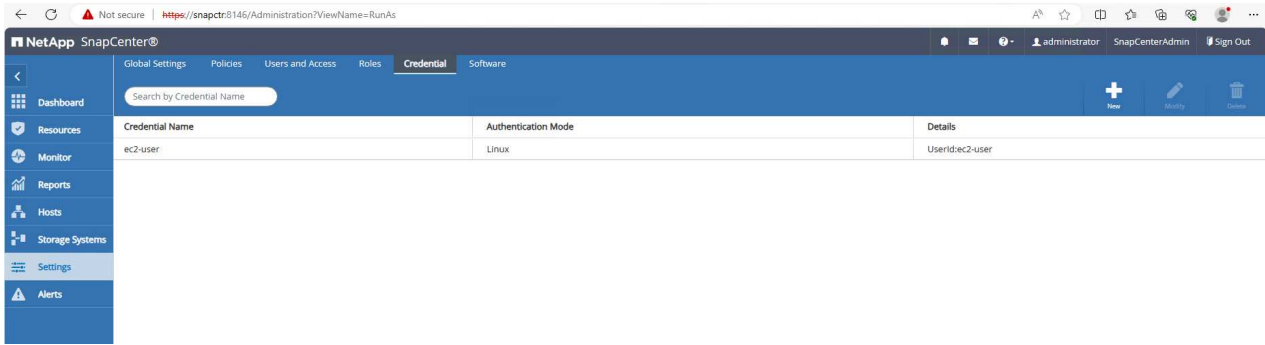
```
SQL> select instance_name, host_name from v$instance;
```

```
INSTANCE_NAME
-----
HOST_NAME
-----
db1
ip-172-30-15-45.ec2.internal
```

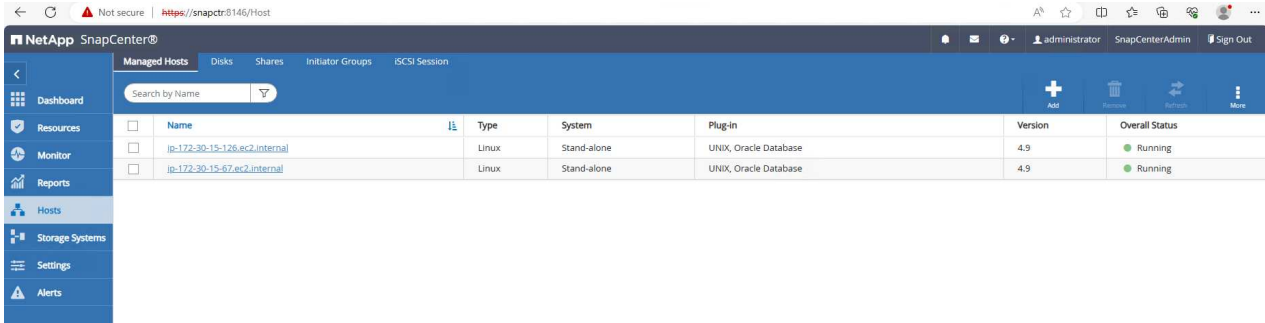
2. FSx 스토리지 클러스터를 에 추가합니다 Storage Systems SnapCenter에서 FSx 클러스터 관리 IP 및 fsxadmin 자격 증명을 사용합니다.



3. AWS EC2 - 사용자를 에 추가합니다 Credential 인치 Settings.

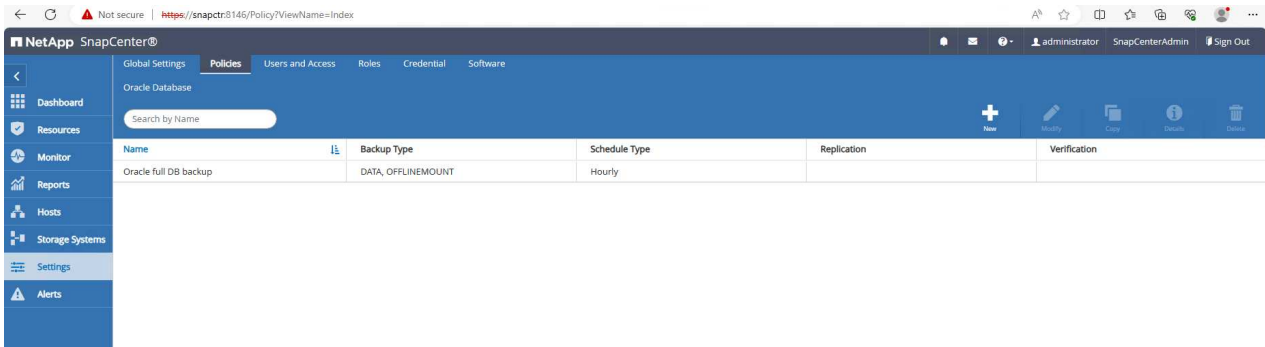


4. 대기 EC2 DB 인스턴스를 추가하고 EC2 DB 인스턴스에 복제합니다 Hosts.

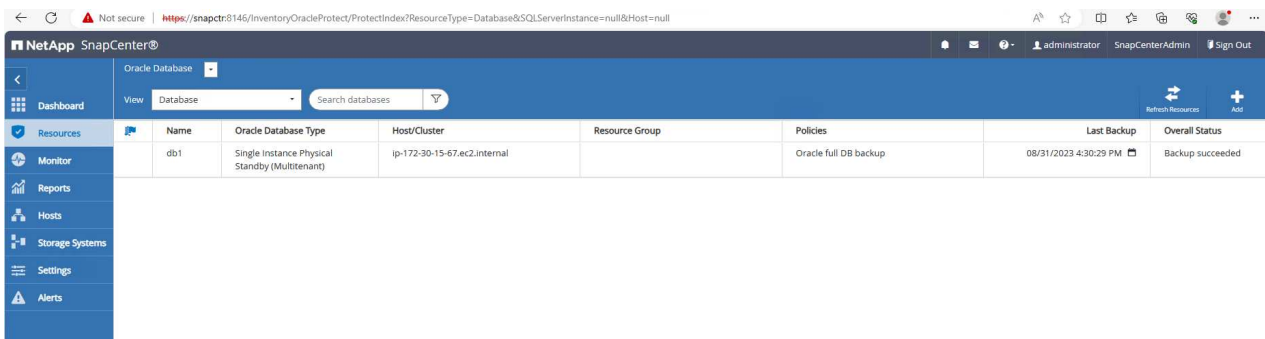


클론 EC2 DB 인스턴스에는 유사한 Oracle 소프트웨어 스택이 설치 및 구성되어 있어야 합니다. 이 테스트 사례에서는 그리드 인프라스트럭처와 Oracle 19C가 설치 및 구성되었지만 데이터베이스가 생성되지 않았습니다.

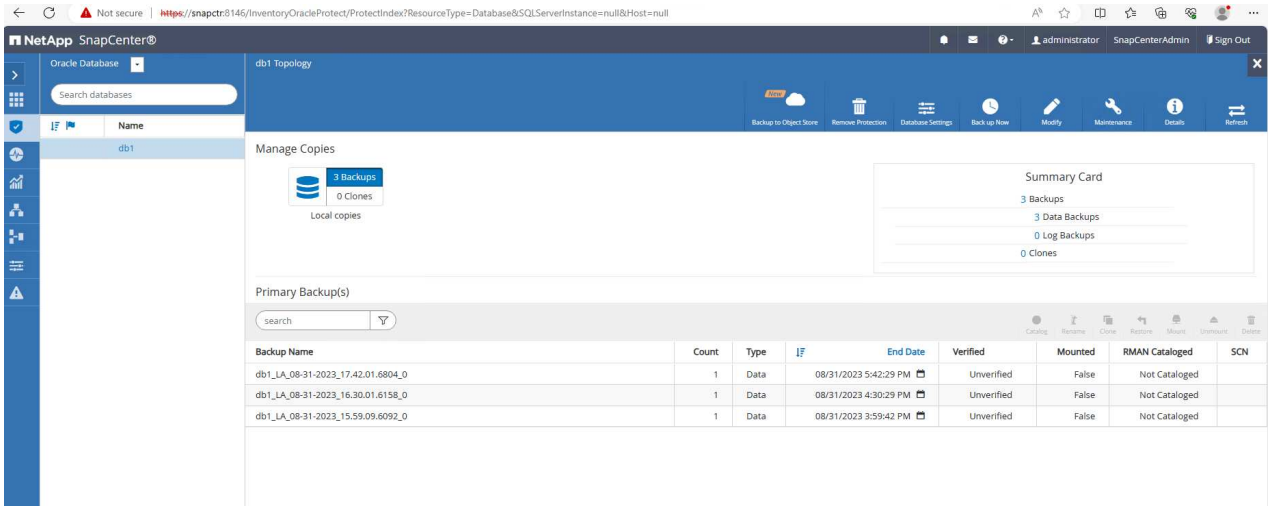
5. 오프라인/전체 데이터베이스 백업에 맞게 조정된 백업 정책을 생성합니다.



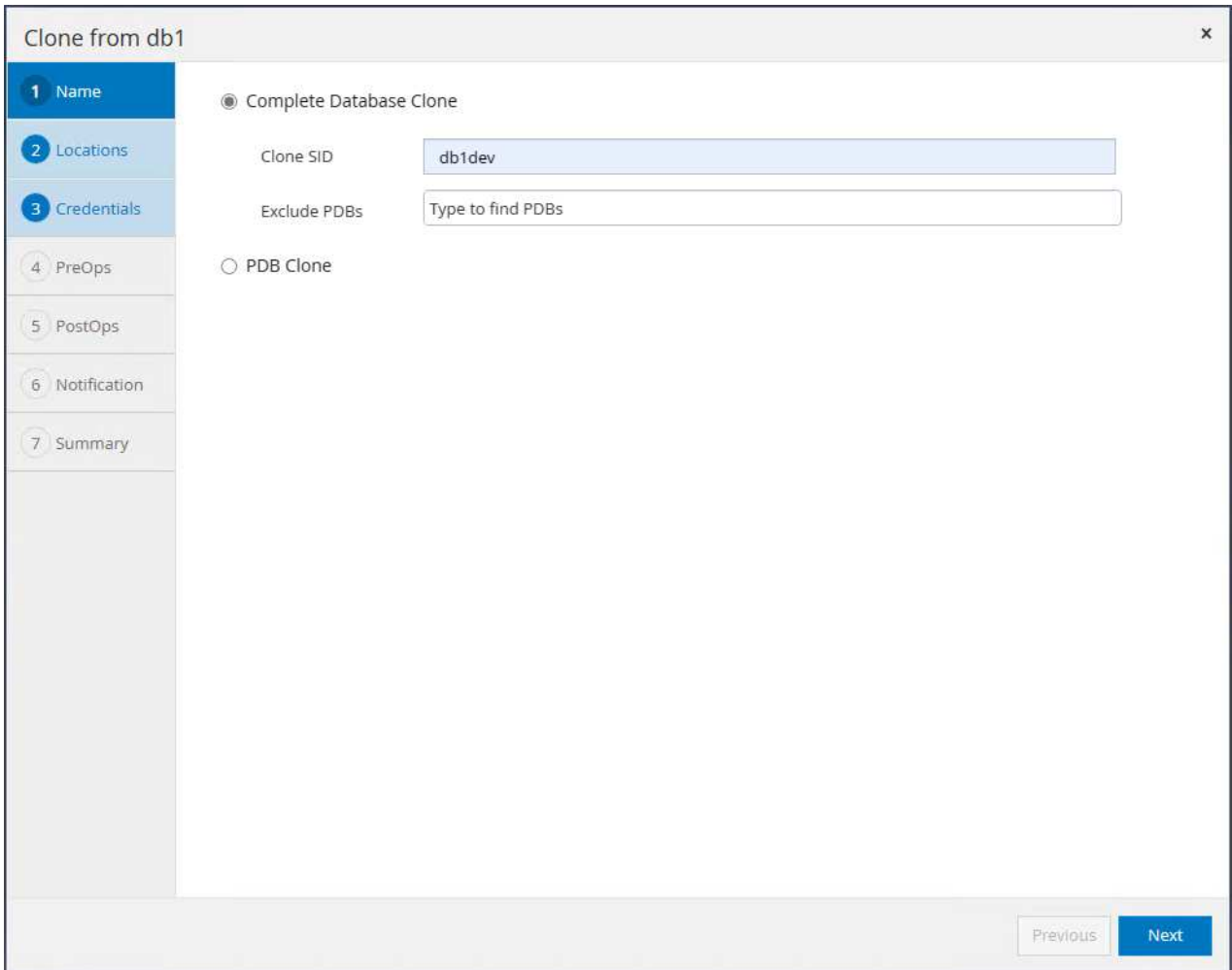
6. 에서 대기 데이터베이스를 보호하기 위한 백업 정책을 적용합니다 Resources 탭을 클릭합니다.



7. 데이터베이스 이름을 클릭하여 데이터베이스 백업 페이지를 엽니다. 데이터베이스 클론에 사용할 백업을 선택하고 을 클릭합니다 Clone 버튼을 클릭하여 클론 워크플로우를 시작합니다.



8. 를 선택합니다 Complete Database Clone 클론 인스턴스 SID의 이름을 지정합니다.



9. 클론 생성 데이터베이스를 대기 DB에서 호스팅하는 클론 호스트를 선택합니다. 데이터 파일, 제어 파일 및 재실행 로그에 대한 기본값을 적용합니다. 대기 데이터베이스의 디스크 그룹에 해당하는 클론 호스트에 ASM 디스크 그룹 두 개가 생성됩니다.

x
Clone from db1

- 1 Name
- 2 Locations
- 3 Credentials
- 4 PreOps
- 5 PostOps
- 6 Notification
- 7 Summary

Select the host to create a clone

Clone host

Datafile locations ?

+SC_2090922_db1dev
▲

+SC_2342319_db1dev
▼

Control files ?

+SC_2090922_db1dev/db1dev/control/control01.ctl
×
▲

+SC_2090922_db1dev/db1dev/control/control02.ctl
×
▼

Redo logs ?

Group	Size	Unit	Number of files
▶ RedoGroup 1	200	MB	2
▶ RedoGroup 2	200	MB	2
▶ RedoGroup 3	200	MB	2

10. OS 기반 인증에 데이터베이스 자격 증명 필요하지 않습니다. Oracle 홈 설정과 클론 EC2 데이터베이스 인스턴스에 구성된 설정을 일치시킵니다.

x
Clone from db1

- 1 Name
- 2 Locations
- 3 Credentials
- 4 PreOps
- 5 PostOps
- 6 Notification
- 7 Summary

Database Credentials for the clone

Credential name for sys user + ⓘ

ASM instance Credential name + ⓘ

Database port

ASM Port

Oracle Home Settings ⓘ

Oracle Home

Oracle OS User

Oracle OS Group

11. 필요한 경우 클론 데이터베이스 매개 변수를 변경하고 필요한 경우 은폐 전에 실행할 스크립트를 지정합니다.

Clone from db1
✕

- 1 Name
- 2 Locations
- 3 Credentials
- 4 PreOps
- 5 PostOps
- 6 Notification
- 7 Summary

Specify scripts to run before clone operation ?

Prescript full path

Arguments

Script timeout secs

⊖ Database Parameter settings

audit_file_dest	/u01/app/oracle/admin/db1dev_LA/adump	✕	<input type="button" value="+"/> <input type="button" value="Reset"/>
audit_trail	DB	✕	
open_cursors	300	✕	
pga_aggregate_target	2684354560	✕	

12. SQL을 입력하여 클론 후에 실행합니다. 데모에서는 개발/테스트/보고서 데이터베이스에 대한 데이터베이스 아카이브 모드를 해제하는 명령을 실행했습니다.

x
Clone from db1

- 1 Name
- 2 Locations
- 3 Credentials
- 4 PreOps
- 5 PostOps
- 6 Notification
- 7 Summary

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

Create new DBID i

Create tempfile for temporary tablespace i

⊖ Enter SQL queries to apply when clone is created

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

Reset

⊕ Enter scripts to run after clone operation i

Previous
Next

13. 원하는 경우 이메일 알림을 구성합니다.

Clone from db1 ×

- 1 Name
- 2 Locations
- 3 Credentials
- 4 PreOps
- 5 PostOps
- 6 Notification**
- 7 Summary

Provide email settings ⓘ

Email preference

From

To

Subject

Attach job report

14. 요약을 검토하고 **Finish** 를 눌러 클론을 시작합니다.

x
Clone from db1

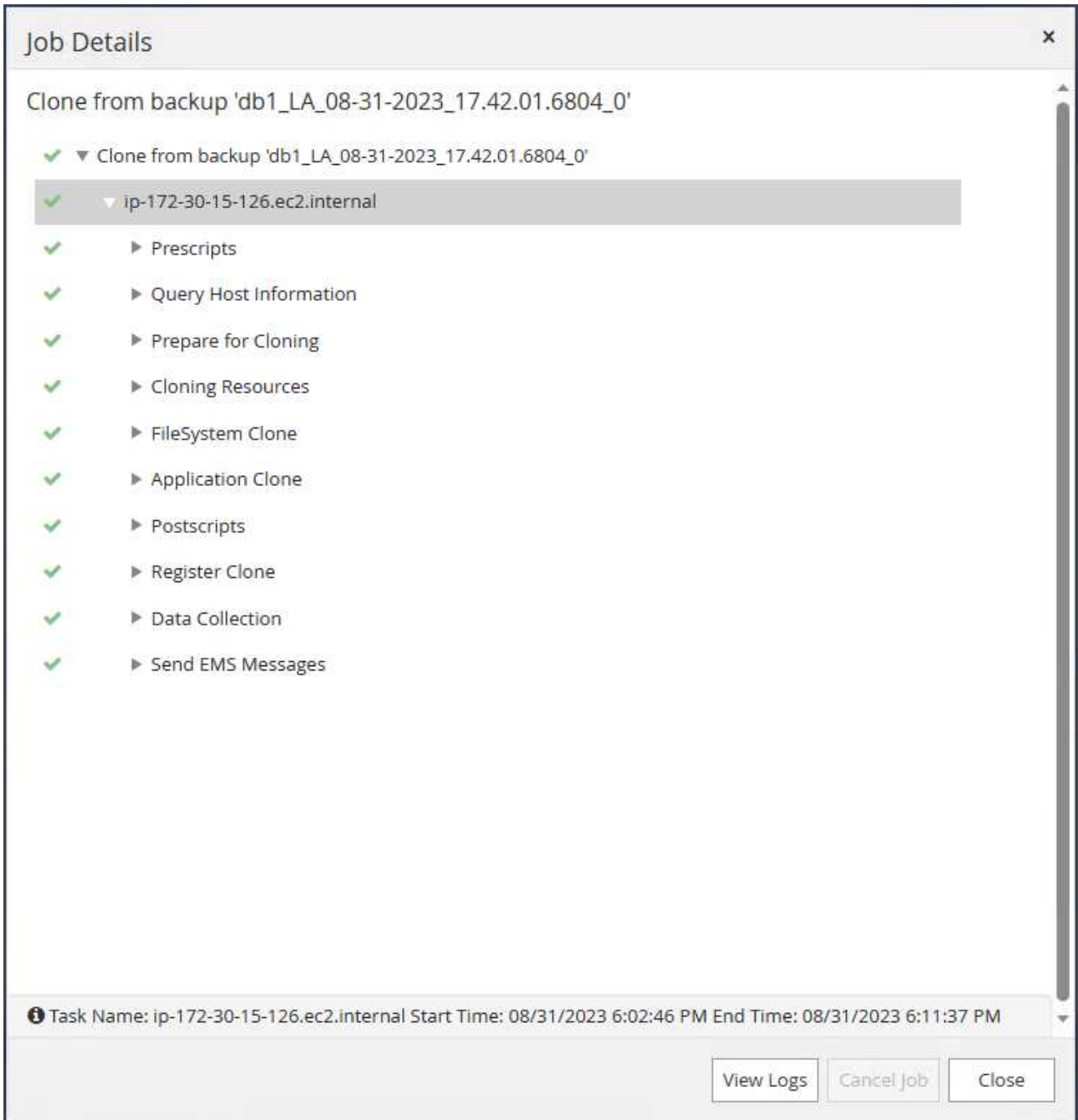
- 1 Name
- 2 Locations
- 3 Credentials
- 4 PreOps
- 5 PostOps
- 6 Notification
- 7 Summary

Summary

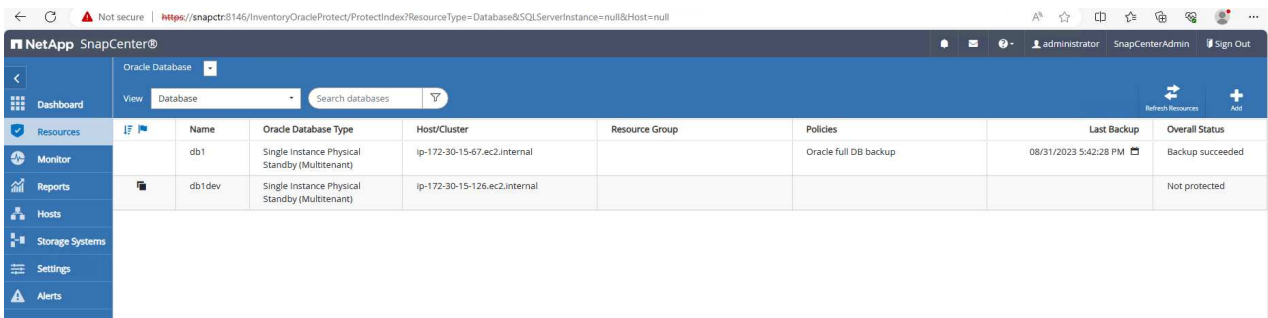
Clone from backup	db1_LA_08-31-2023_17.42.01.6804_0
Clone SID	db1 dev
Clone server	ip-172-30-15-126.ec2.internal
Exclude PDBs	none
Oracle home	/u01/app/oracle/product/19.0.0/dev
Oracle OS user	oracle
Oracle OS group	oinstall
Datafile mountpaths	+SC_2090922_db1 dev +SC_2342319_db1 dev
Control files	+SC_2090922_db1 dev/db1 dev/control/control01.ctl +SC_2090922_db1 dev/db1 dev/control/control02.ctl
Redo groups	RedoGroup =1 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo01_01.log RedoGroup =1 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo01_02.log RedoGroup =2 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo02_01.log RedoGroup =2 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo02_02.log RedoGroup =3 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo03_01.log RedoGroup =3 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo03_02.log RedoGroup =4 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo04_01.log RedoGroup =4 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo04_02.log RedoGroup =5 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo05_01.log RedoGroup =5 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo05_02.log RedoGroup =6 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo06_01.log RedoGroup =6 TotalSize =200 Path =+SC_2090922_db1 dev/db1 dev/redo06_02.log

Previous
Finish

15. 에서 클론 작업을 모니터링합니다 Monitor 탭을 클릭합니다. 데이터베이스 볼륨 크기로 약 300GB의 데이터베이스를 복제하는 데 8분 정도 걸렸습니다.



16. 에 즉시 등록되는 SnapCenter에서 클론 데이터베이스를 검증합니다 Resources 탭을 클릭합니다.



17. 클론 EC2 인스턴스에서 클론 데이터베이스를 쿼리합니다. 운영 데이터베이스에서 발생한 테스트 트랜잭션이 복제 데이터베이스를 통과했음을 확인했습니다.

```

[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
-----

```

```
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>
```

이렇게 하면 개발, 테스트, 보고 또는 기타 사용 사례를 위해 FSx 스토리지의 Data Guard에 있는 대기 데이터베이스에서 새 Oracle 데이터베이스의 클론 및 검증이 완료됩니다. 여러 Oracle 데이터베이스를 Data Guard의 동일한 대기 데이터베이스에서 복제할 수 있습니다.

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 문서 및/또는 웹 사이트를 참조하십시오.

- Data Guard 개념 및 관리

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

- WP-7357: EC2에서 Oracle 데이터베이스 구축 및 FSx 모범 사례

["https://docs.netapp.com/us-en/netapp-solutions/databases/aws_ora_fsx_ec2_deploy_intro.html"](https://docs.netapp.com/us-en/netapp-solutions/databases/aws_ora_fsx_ec2_deploy_intro.html)

- NetApp ONTAP용 Amazon FSx

["https://aws.amazon.com/fsx/netapp-ontap/"](https://aws.amazon.com/fsx/netapp-ontap/)

- Amazon EC2

https://aws.amazon.com/pm/ec2/?trk=36c6da98-7b20-48fa-8225-4784bced9843&sc_channel=ps&s_kwcid=AL!4422!3!467723097970!e!!g!!aws%20ec2&ef_id=Cj0KCQiA54KfBhCKARIsAJzSrdqwQrghn6I71jiWzSeaT9Uh1-vY-VfhJixF-xnv5rWwn2S7RqZOTQ0aAh7eEALw_wbC:s&s_kwcid=AL!4422!3!467723097970!e!!g!!aws%20ec2

TR-4973: AWS FSx ONTAP에서 증분 병합을 사용하여 Oracle VLDB의 빠른 복구 및 복제

Allen Cao, Niyaz Mohamed, NetApp

목적

Oracle RMAN(Recovery Manager) 백업 툴을 사용하여 Oracle에서 매우 큰 데이터베이스(VLDB)를 복구하는 것은 매우 어려운 작업이 될 수 있습니다. 오류가 발생할 경우 백업 미디어에서 데이터베이스 복원 프로세스를 수행하는 데 시간이 오래 걸릴 수 있으므로 데이터베이스 복구가 지연되고 서비스 수준 계약(SLA)에 상당한 영향을 줄 수 있습니다. 그러나 Oracle은 버전 10g부터 사용자가 DB 서버 호스트에 있는 추가 디스크 스토리지에 Oracle 데이터베이스 데이터 파일의 스테이징된 이미지 복제본을 생성할 수 있는 RMAN 기능을 도입했습니다. 이러한 이미지 복제본은 RMAN을

사용하여 매일 점진적으로 업데이트할 수 있습니다. 장애가 발생할 경우 DBA(데이터베이스 관리자)가 Oracle 데이터베이스를 실패한 미디어에서 이미지 복사본으로 신속하게 전환하여 전체 데이터베이스 미디어 복원이 필요하지 않게 할 수 있습니다. 그 결과, 필요한 데이터베이스 스토리지가 2배로 증가했지만 SLA가 크게 개선되었습니다.

VLDB에 대한 SLA에 관심이 있고 Oracle 데이터베이스를 AWS와 같은 퍼블릭 클라우드로 이전하려는 경우 대기 데이터베이스 이미지 복사본을 스테이징하기 위해 AWS FSx ONTAP와 같은 리소스를 사용하여 유사한 데이터베이스 보호 구조를 설정할 수 있습니다. 이 문서에서는 운영 스토리지 장애 시 신속한 복구를 위해 대기 데이터베이스 복사본을 스테이징하기 위해 AWS FSx ONTAP에서 마운트된 NFS 파일 시스템을 Oracle 데이터베이스 서버에 프로비저닝하고 내보내는 방법을 설명합니다.

또한, NetApp FlexClone을 활용하여 추가 스토리지 투자 없이 동일한 대기 데이터베이스 이미지 복사본을 사용하여 개발/테스트 Oracle 환경을 구축하는 등의 다른 사용 사례에서 동일한 스테이징 NFS 파일 시스템의 복사본을 생성하는 방법도 살펴볼 수 있습니다.

이 솔루션은 다음과 같은 사용 사례를 해결합니다.

- AWS FSx ONTAP 스토리지의 NFS 마운트 지점에서 RMAN을 통해 Oracle VLDB 이미지 복제 증분 병합이 수행됩니다.
- 장애 발생 시 FSx ONTAP 스토리지의 데이터베이스 이미지 복사본으로 전환하여 Oracle VLDB를 빠르게 복구합니다.
- 클론 FSx ONTAP NFS 파일 시스템 볼륨에 Oracle VLDB 이미지 복제본을 저장하여 다른 활용 사례에서 다른 데이터베이스 인스턴스를 설치하는 데 사용됩니다.

대상

이 솔루션은 다음과 같은 사용자를 대상으로 합니다.

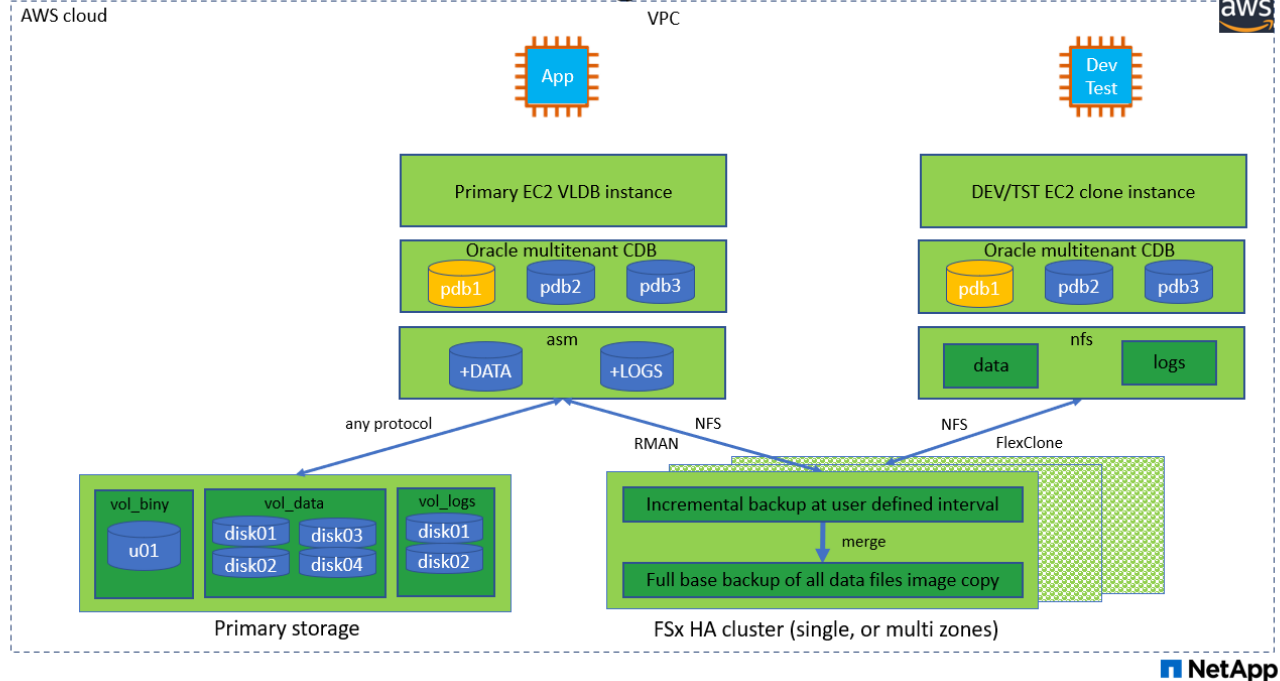
- 더 빠른 데이터베이스 복구를 위해 AWS에서 RMAN을 통해 Oracle VLDB 이미지 복사본 증분 병합을 설정하는 DBA.
- AWS 퍼블릭 클라우드에서 Oracle 워크로드를 테스트하는 데이터베이스 솔루션 설계자
- AWS FSx ONTAP 스토리지에 구축된 Oracle 데이터베이스를 관리하는 스토리지 관리자
- AWS FSx/EC2 환경에서 Oracle 데이터베이스를 가동하려는 애플리케이션 소유자입니다.

솔루션 테스트 및 검증 환경

이 솔루션의 테스트 및 검증은 최종 구축 환경과 일치하지 않을 수 있는 AWS FSx ONTAP 및 EC2 환경에서 수행되었습니다. 자세한 내용은 섹션을 참조하십시오 [\[Key Factors for Deployment Consideration\]](#).

있습니다

Oracle VLDB Incremental Merge via RMAN on AWS FSxN



하드웨어 및 소프트웨어 구성 요소

* 하드웨어 *		
FSX ONTAP 저장소	AWS에서 제공하는 현재 버전입니다	동일한 VPC 및 가용성 존에 FSx HA 클러스터 1개
컴퓨팅용 EC2 인스턴스	T2.xLarge/4vCPU/16G	EC2 T2 xLarge EC2 인스턴스 2개, 하나는 운영 DB 서버로, 다른 하나는 클론 DB 서버로
* 소프트웨어 *		
RedHat Linux	RHEL-8.6.0_HVM-20220503-x86_64-2-Hourly2-GP2	테스트를 위해 RedHat 서브스크립션을 배포했습니다
Oracle Grid Infrastructure	버전 19.18	RU 패치 p34762026_190000_Linux-x86-64.zip 를 적용했습니다
Oracle 데이터베이스	버전 19.18	RU 패치 p34765931_190000_Linux-x86-64.zip 를 적용했습니다
Oracle OPatch	버전 12.2.0.1.36	최신 패치 p6880880_190000_Linux-x86-64.zip

구축 시 고려해야 할 주요 요소

- * RMAN 증분 병합용 Oracle VLDB 스토리지 레이아웃 * 테스트 및 검증에서 Oracle 증분 백업 및 병합용 NFS 볼륨은 단일 FSx 파일 시스템에서 4Gbps 처리량, 160,000 원시 SSD IOPS 및 192TiB 용량 제한이 있는 할당됩니다. 임계값을 통한 구축의 경우 여러 FSx 파일 시스템을 여러 NFS 마운트 지점과 병렬로 연결하여 더 큰 용량을 제공할 수 있습니다.

- * RMAN 증분 병합을 사용하여 Oracle 복구 성능. * RMAN 증분 백업 및 병합은 일반적으로 RTO 및 RPO 목표에 따라 사용자 정의 빈도로 실행됩니다. 운영 데이터 스토리지 및/또는 아카이브 로그가 완전히 손실되면 데이터 손실이 발생할 수 있습니다. Oracle 데이터베이스는 FSx 데이터베이스 백업 이미지 복사본에서 사용할 수 있는 마지막 증분 백업까지 복구할 수 있습니다. 데이터 손실을 최소화하기 위해 FSx NFS 마운트 지점에 Oracle 플래시 복구 영역을 설정할 수 있으며 데이터베이스 이미지 복사본과 함께 아카이빙된 로그를 FSx NFS 마운트에 백업할 수 있습니다.
- * FSx NFS 파일 시스템에서 Oracle VLDB를 실행합니다. * 데이터베이스 백업을 위한 다른 대용량 스토리지와 달리 AWS FSx ONTAP는 높은 수준의 성능과 스토리지 효율성을 제공하는 클라우드 지원 운영 등급 스토리지입니다. Oracle VLDB가 운영 스토리지에서 FSx ONTAP NFS 파일 시스템의 이미지 복제본으로 전환되면 기본 스토리지 장애가 해결되는 동안 데이터베이스 성능을 높은 수준으로 유지할 수 있습니다. 기본 스토리지 장애로 인해 사용자 애플리케이션 환경이 저하되지 않는다는 것을 알면 편리할 수 있습니다.
- * FlexClone Oracle VLDB 기타 사용 사례에서 NFS 볼륨의 이미지 복사본 * AWS FSx ONTAP FlexClone은 쓰기 가능한 동일한 NFS 데이터 볼륨의 공유 복사본을 제공합니다. 따라서 Oracle 데이터베이스가 전환되더라도 Oracle VLDB 이미지 복제본을 스테이징하는 무결성을 유지하면서 다른 많은 활용 사례에 사용할 수 있습니다. 따라서 VLDB 스토리지 설치 공간이 크게 줄어 스토리지 비용이 크게 절감됩니다. NetApp은 Oracle 성능을 높은 수준으로 유지하기 위해 데이터베이스를 운영 스토리지에서 데이터베이스 이미지 복사본으로 전환하는 경우 FlexClone 활동을 최소화하는 것이 좋습니다.
- * EC2 컴퓨팅 인스턴스 * 이러한 테스트 및 검증에서는 AWS EC2 T2.xLarge 인스턴스를 Oracle 데이터베이스 컴퓨팅 인스턴스로 사용했습니다. NetApp에서는 데이터베이스 워크로드에 최적화되어 있으므로 운영 구축 시 Oracle용 컴퓨팅 인스턴스로 M5 유형 EC2 인스턴스를 사용할 것을 권장합니다. 실제 워크로드 요구 사항에 따라 vCPU 수와 RAM 양에 따라 EC2 인스턴스를 적절하게 사이징해야 합니다.
- * FSx 스토리지 HA 클러스터 단일 또는 다중 영역 배포. * 이러한 테스트 및 검증에서는 단일 AWS 가용성 영역에 FSx HA 클러스터를 구축했습니다. 프로덕션 배포를 위해 FSx HA 쌍을 두 가지 가용성 영역에 배포하는 것이 좋습니다. FSx HA 클러스터는 스토리지 레벨 이중화를 제공하기 위해 액티브-패시브 파일 시스템 쌍으로 미러링되는 HA 쌍으로 프로비저닝됩니다. 다중 영역 구축을 통해 단일 AWS 영역에서 장애가 발생할 경우 고가용성을 더욱 강화할 수 있습니다.
- * FSx 스토리지 클러스터 크기 조정 * ONTAP 스토리지 파일 시스템용 Amazon FSx는 최대 160,000개의 원시 SSD IOPS, 최대 4Gbps 처리량 및 최대 192TiB 용량을 제공합니다. 그러나 구현 시 실제 요구 사항에 따라 프로비저닝된 IOPS, 처리량 및 스토리지 제한(최소 1,024GiB)을 기준으로 클러스터 크기를 조정할 수 있습니다. 애플리케이션 가용성에 영향을 주지 않고 용량을 즉각적으로 동적으로 조정할 수 있습니다.
- * dNFS 구성. * dNFS는 Oracle 커널에 내장되어 있으며 Oracle을 NFS 스토리지에 구축할 때 Oracle 데이터베이스 성능을 크게 향상시키는 것으로 알려져 있습니다. dNFS는 Oracle 바이너리로 패키징되지만 기본적으로 활성화되어 있지 않습니다. NFS 기반 Oracle 데이터베이스 구축 시 이 기능을 설정해야 합니다. VLDB를 위한 여러 FSx 파일 시스템 배포의 경우 다른 FSx NFS 파일 시스템에 대한 dNFS 다중 경로를 올바르게 구성해야 합니다.

솔루션 구축

VPC 내에서 AWS EC2 환경에 Oracle VLDB를 이미 구축한 것으로 가정합니다. AWS에서의 Oracle 구축에 대한 도움이 필요한 경우 다음 기술 보고서에서 도움말을 참조하십시오.

- ["EC2 및 FSx Best Practices에 Oracle Database 구축"](#)
- ["iSCSI/ASM이 포함된 AWS FSx/EC2에서 Oracle 데이터베이스 구축 및 보호"](#)
- ["NFS/ASM을 사용하는 AWS FSx/EC2에서 독립 실행형 재시작 시 Oracle 19c"](#)

Oracle VLDB는 FSx ONTAP 또는 AWS EC2 에코시스템 내에서 선택 가능한 다른 스토리지에서 실행될 수 있습니다. 다음 섹션에서는 AWS FSx ONTAP 스토리지에서 NFS 마운트에 스테이징되는 Oracle VLDB의 이미지 복제본에 RMAN 증분 병합을 설정하는 단계별 구축 절차를 제공합니다.

배포에는 다음과 같은 사전 요구 사항이 필요합니다.

1. AWS 계정이 설정되었으며 AWS 계정 내에 필요한 VPC 및 네트워크 세그먼트가 생성되었습니다.
2. AWS EC2 콘솔에서 2개의 EC2 Linux 인스턴스를 구축해야 합니다. 하나는 운영 Oracle DB 서버로, 다른 하나는 선택적 대체 클론 타겟 DB 서버입니다. 환경 설정에 대한 자세한 내용은 이전 섹션의 아키텍처 다이어그램을 참조하십시오. 또한 를 검토합니다 "[Linux 인스턴스에 대한 사용자 가이드](#)" 를 참조하십시오.
3. AWS EC2 콘솔에서 ONTAP 스토리지 HA 클러스터용 Amazon FSx를 구축하여 Oracle 데이터베이스 대기 이미지 복사본을 저장하는 NFS 볼륨을 호스팅합니다. FSx 저장소 배포에 익숙하지 않은 경우 설명서를 참조하십시오 "[ONTAP 파일 시스템용 FSx 생성](#)" 을 참조하십시오.
4. 2단계와 3단계는 라는 EC2 인스턴스를 만드는 다음 Terraform 자동화 툴킷을 사용하여 수행할 수 있습니다 ora_01 이라는 FSx 파일 시스템이 있습니다 fsx_01. 실행 전에 지침을 주의 깊게 검토하고 환경에 맞게 변수를 변경하십시오. 사용자 고유의 배포 요구 사항에 맞게 템플릿을 쉽게 수정할 수 있습니다.

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



Oracle 설치 파일을 스테이징할 충분한 공간을 확보하기 위해 EC2 인스턴스 루트 볼륨에 50G 이상을 할당했는지 확인합니다.

EC2 DB 인스턴스 호스트에 마운트할 **NFS** 볼륨을 프로비저닝하고 내보냅니다

이 데모에서는 FSx 클러스터 관리 IP를 통해 ssh를 통해 fsxadmin 사용자로 FSx 클러스터에 로그인하여 명령줄에서 NFS 볼륨을 프로비저닝하는 방법을 보여 줍니다. 또는 AWS FSx 콘솔을 사용하여 볼륨을 할당할 수도 있습니다. 데이터베이스 크기를 수용하도록 둘 이상의 FSx 파일 시스템이 설정된 경우 다른 FSx 파일 시스템에서 이 절차를 반복합니다.

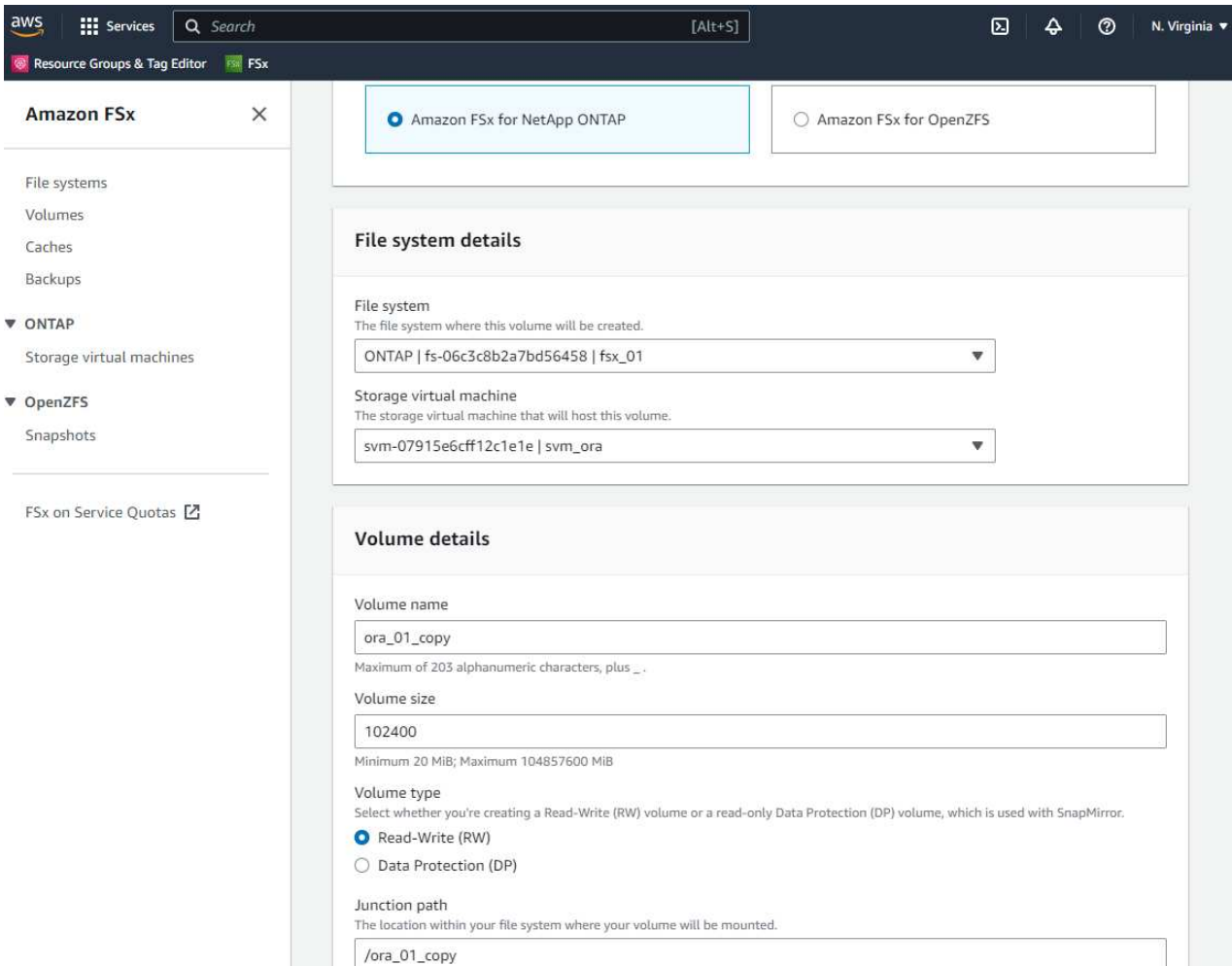
1. 먼저, SSH를 통해 FSx 클러스터에 fsxadmin 사용자로 로그인하여 CLI를 통해 NFS 볼륨을 프로비저닝합니다. FSx ONTAP UI 콘솔에서 검색할 수 있는 FSx 클러스터 관리 IP 주소로 변경합니다.

```
ssh fsxadmin@172.30.15.53
```

2. 운영 Oracle VLDB 데이터베이스 데이터 파일 이미지 복사본을 저장하기 위해 운영 스토리지와 동일한 크기의 NFS 볼륨을 생성합니다.

```
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. 또는 AWS FSx 콘솔 UI에서 스토리지 효율성 옵션을 사용하여 볼륨을 프로비저닝할 수 있습니다 Enabled, 보안 스타일 Unix, 스냅샷 정책 None `스토리지 계층화를 지원합니다 `Snapshot Only 아래 그림과 같이 표시됩니다.



4. 일별 스케줄과 30일 보존 기간을 사용하여 Oracle 데이터베이스에 대한 사용자 지정된 스냅샷 정책을 생성합니다. 스냅샷 빈도 및 보존 기간 측면에서 특정 요구에 맞게 정책을 조정해야 합니다.

```
snapshot policy create -policy oracle -enabled true -schedule1 daily
-count1 30
```

RMAN 증분 백업 및 병합을 위해 프로비저닝된 NFS 볼륨에 정책을 적용합니다.

```
vol modify -volume ora_01_copy -snapshot-policy oracle
```

5. EC2 인스턴스에 EC2-USER로 로그인하여 /nfsfsxn 디렉토리를 생성합니다. 추가 FSx 파일 시스템에 대한 추가 마운트 지점 디렉토리를 생성합니다.

```
sudo mkdir /nfsfsxn
```

6. FSx ONTAP NFS 볼륨을 EC2 DB 인스턴스 호스트에 마운트합니다. FSx 가상 서버 NFS lif 주소로 변경합니다. FSx ONTAP UI 콘솔에서 NFS lif 주소를 검색할 수 있습니다.

```
sudo mount 172.30.15.19:/ora_01_copy /nfsfsxn -o  
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsiz=262144,wsiz=262144,noi  
tr
```

7. 마운트 지점 소유권을 Oracle로 변경합니다. 즉, 필요에 따라 Oracle 사용자 이름 및 기본 그룹으로 변경하십시오.

```
sudo chown oracle:oinstall /nfsfsxn
```

FSx에서 이미지 복사본에 대한 **Oracle RMAN** 증분 병합을 설정합니다

RMAN 증분 병합은 증분 백업/병합 간격마다 스테이징 데이터베이스 데이터 파일 이미지 복제본을 지속적으로 업데이트합니다. 데이터베이스 백업의 이미지 복사본은 증분 백업/병합을 실행하는 빈도만큼 최신 상태가 됩니다. 따라서 RMAN 증분 백업 및 병합의 빈도를 결정할 때 데이터베이스 성능, RTO 및 RPO 목표를 고려해야 합니다.

1. Oracle 사용자로 운영 DB 서버 EC2 인스턴스에 로그인합니다
2. Oracle 플래시 복구 영역의 Oracle 데이터 파일 이미지 복제본 및 archlog 디렉토리를 저장하려면 마운트 지점 /nfsfsxn 아래에 oracopy 디렉토리를 생성합니다.

```
mkdir /nfsfsxn/oracopy
```

```
mkdir /nfsfsxn/archlog
```

3. sqlplus를 통해 Oracle 데이터베이스에 로그인하고, 블록 변경 추적을 설정하여 증분 백업을 가속화하고, Oracle 플래시 복구 영역을 현재 운영 스토리지에 있는 경우 FSxN 마운트로 변경합니다. 따라서 RMAN 기본 제어 파일/spfile 자동 백업 및 아카이빙된 로그를 복구를 위해 FSxN NFS 마운트에 백업할 수 있습니다.

```
sqlplus / as sysdba
```

sqlplus 프롬프트에서 다음 명령을 실행합니다.

```
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. RMAN 백업 및 증분 병합 스크립트를 생성합니다. 이 스크립트는 병렬 RMAN 백업 및 병합을 위해 여러 채널을 할당합니다. 첫 번째 실행에서는 초기 전체 기준선 이미지 복사본이 생성됩니다. 전체 실행에서 먼저 보존 기간 외부에 있는 폐기된 백업을 정리하여 스테이징 영역을 깨끗하게 유지합니다. 그런 다음 병합 및 백업 전에 현재 로그 파일을 전환합니다. 증분 백업은 병합 후에 데이터베이스 이미지 복사본이 현재 데이터베이스 상태를 하나의 백업/병합 주기로 후행 하도록 합니다. 병합 및 백업 순서를 반대로 하면 사용자가 원하는 대로 신속하게 복구할 수 있습니다. RMAN 스크립트는 기본 DB 서버의 crontab에서 실행할 간단한 셸 스크립트로 통합할 수 있습니다. RMAN 설정에서 제어 파일 자동 백업이 켜져 있는지 확인합니다.

```
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. EC2 DB 서버에서 RMAN 카탈로그를 사용하거나 사용하지 않고 Oracle 사용자로 RMAN에 로컬로 로그인합니다. 이 데모에서는 RMAN 카탈로그에 접속하지 않습니다.

```
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. RMAN 프롬프트에서 스크립트를 실행합니다. 첫 번째 실행에서는 기본 데이터베이스 이미지 복사본과 후속 실행이 기본 이미지 복사본을 점진적으로 병합하고 업데이트합니다. 다음은 스크립트 및 일반 출력을 실행하는 방법입니다. 호스트의 CPU 코어와 일치하는 채널 수를 설정합니다.

```
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. 백업 후 데이터베이스 이미지 복제본을 나열하여 FSx ONTAP NFS 마운트 지점에서 데이터베이스 이미지 복제본이 생성되었는지 확인합니다.

```
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. Oracle RMAN 명령 프롬프트에서 스키마를 보고하여 현재 활성 데이터베이스 데이터 파일이 운영 스토리지 ASM + 데이터 디스크 그룹에 있는지 확인합니다.

```

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.265.113666447
2	123	PDB\$SEED:TEMP	32767	+DATA/DB1/FB864A929AEB79B9E053630F1EAC7046/TEMPFILE/temp.269.1136667185
3	10240	DB1_PDB1:TEMP	32767	+DATA/DB1/FB867DA8C68C816EE053630F1EAC2BCF/TEMPFILE/temp.274.1136668051
4	123	DB1_PDB2:TEMP	32767	+DATA/DB1/FB867EA89ECF81C0E053630F1EACB901/TEMPFILE/temp.279.1136668067
5	123	DB1_PDB3:TEMP	32767	+DATA/DB1/FB867F8A4D4F821CE053630F1EAC69CC/TEMPFILE/temp.284.1136668081

RMAN>

9. OS NFS 마운트 지점에서 데이터베이스 이미지 복사본 검증

```
[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_1l1sd7dm
-rw-r----- 1 oracle asm 5251072 May 17 18:19 data_D-DB1_I-
1730530050_TS-USERS_FNO-16_1t1sd7dn
-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_101sd7d1
```

이것으로 Oracle 데이터베이스 대기 이미지 복사 백업 및 병합의 설치가 완료되었습니다.

빠른 복구를 위해 **Oracle DB**를 이미지 복제본으로 전환합니다

데이터 손실 또는 손상과 같은 운영 스토리지 문제로 인해 장애가 발생할 경우 데이터베이스를 FSx ONTAP NFS 마운트의 이미지 복제본으로 신속하게 전환하며 데이터베이스 복구 없이 현재 상태로 복구할 수 있습니다. 미디어 복원을 제거하면 VLDB의 데이터베이스 복구 속도가 엄청나게 빨라집니다. 이 활용 사례에서는 데이터베이스 호스트 인스턴스가 온전하고 데이터베이스 제어 파일, 아카이브된 로그 및 현재 로그를 모두 복구에 사용할 수 있다고 가정합니다.

1. Oracle 사용자로 EC2 DB 서버 호스트에 로그인하고 테스트 테이블을 생성한 후 전환합니다.

```
[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. 중단 데이터베이스를 종료하여 오류를 시뮬레이션한 다음 마운트 단계에서 Oracle을 시작합니다.

```
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. Oracle 사용자는 RMAN을 통해 Oracle 데이터베이스에 접속하여 데이터베이스를 복제본으로 전환합니다.

```
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_1o1sd7dl"
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_1l1sd7dm"
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_091sd711"
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. 복구 및 오픈 데이터베이스를 통해 마지막 증분 백업에서 최신 상태로 복원합니다.

```
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. 복구 후 sqlplus에서 데이터베이스 구조를 확인하여 제어, 임시 및 현재 로그 파일을 제외한 모든 데이터베이스 데이터 파일이 이제 FSx ONTAP NFS 파일 시스템에서 복제본으로 전환되는지 확인합니다.

```
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_091sd711
/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_1l1sd7dm  
/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_101sd7d1
```

43 rows selected.

SQL>

6. SQL plus에서 전환할 때 복사하도록 전환하기 전에 삽입한 테스트 테이블의 내용을 확인합니다

```
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. FSx ONTAP는 뛰어난 성능을 제공하는 이중화된 운영 등급 스토리지이므로 FSx NFS 마운트에서 Oracle 데이터베이스를 장기간 동안 성능 저하 없이 실행할 수 있습니다. 운영 스토리지 문제가 해결되면 가동 중지 시간을 최소화하면서 증분 백업 병합 프로세스를 반대로 수행하여 다시 되돌릴 수 있습니다.

이미지 복사본에서 다른 **EC2 DB** 인스턴스 호스트로 **Oracle DB** 복구

운영 스토리지와 EC2 DB 인스턴스 호스트가 모두 손실되는 경우 원래 서버에서 복구를 수행할 수 없습니다. 하지만 이중화 FSxN NFS 파일 시스템에 여전히 Oracle 데이터베이스 백업 이미지가 복제본이 있습니다. 동일한 EC2 DB 인스턴스를 다시 빠르게 프로비저닝하고 NFS를 통해 VLDB의 이미지 복사본을 새 EC2 DB 호스트에 쉽게 마운트하여 복구를 실행할 수 있습니다. 이 섹션에서는 이러한 작업을 위한 단계별 절차를 시연합니다.

1. 대체 호스트 검증으로 Oracle 데이터베이스를 복구하기 위해 이전에 만든 테스트 테이블에 행을 삽입합니다.

```
[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. Oracle 사용자로 RMAN 증분 백업을 실행하고 병합하여 FSxN NFS 마운트에 설정된 백업으로 트랜잭션을 플러시합니다.

```
[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. 운영 EC2 DB 인스턴스 호스트를 종료하여 스토리지 및 DB 서버 호스트의 총 장애를 시뮬레이션합니다.
4. AWS EC2 콘솔을 통해 OS와 버전이 동일한 새 EC2 DB 인스턴스 호스트 ora_02 를 사용합니다. 운영 EC2 DB 서버 호스트와 동일한 패치를 사용하여 운영 체제 커널 을 구성하고, Oracle은 RPM을 사전 설치하고, 호스트에 스왑 공간을 추가합니다. 소프트웨어만 제공하는 운영 EC2 DB 서버 호스트와 동일한 버전의 Oracle 패치를 설치합니다. 아래 링크에서 제공되는 NetApp 자동화 툴킷을 사용하여 이러한 작업을 자동화할 수 있습니다.

툴킷: ["NA_oracle19c_deploy 를 선택합니다"](#)
설명서: ["NFS에서 ONTAP용 Oracle19c의 자동 배포"](#)

5. oratab, oralnst.loc 및 Oracle user.bash_profile과 같은 Oracle 환경을 운영 EC2 DB 인스턴스 호스트

ora_01에 맞게 구성합니다. 이러한 파일을 FSxN NFS 마운트 지점에 백업하는 것이 좋습니다.

6. FSxN NFS 마운트의 Oracle 데이터베이스 백업 이미지 복사본은 이중화, 고가용성 및 고성능을 위해 AWS 가용성 영역을 확장하는 FSx 클러스터에 저장됩니다. 네트워킹에 연결할 수 있는 한 NFS 파일 시스템을 새 서버에 쉽게 마운트할 수 있습니다. 다음 절차에서는 복구를 위해 Oracle VLDB 백업의 이미지 복제본을 새로 프로장된 EC2 DB 인스턴스 호스트에 마운트합니다.

EC2-USER로 마운트 지점을 만듭니다.

```
sudo mkdir /nfsfsxn
```

EC2-USER로 Oracle VLDB 백업 이미지 복사본을 저장한 NFS 볼륨을 마운트합니다.

```
sudo mount 172.30.15.19:/ora_01_copy /nfsfsxn -o
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsz=262144,wsz=262144,noi
tr
```

7. FSxN NFS 마운트 지점에서 Oracle 데이터베이스 백업 이미지 복제본을 검증합니다.

```
[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_411t508t  
-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. 복구를 위해 FSxN NFS 마운트에서 사용 가능한 Oracle 아카이브 로그를 확인하고 마지막 로그 파일 로그 시퀀스 번호를 기록합니다. 이 경우 175입니다. 복구 지점은 최대 로그 시퀀스 번호 176입니다.

```

[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
ol_mf_1_143__02rotwyb_.arc
-r--r----- . 1 oracle 54331 165636608 May 30 15:49
ol_mf_1_144__02x563wh_.arc
-r--r----- . 1 oracle 54331 168408064 May 30 15:51
ol_mf_1_145__031kg2co_.arc
-r--r----- . 1 oracle 54331 169446400 May 30 15:54
ol_mf_1_146__035xpcdt_.arc
-r--r----- . 1 oracle 54331 167595520 May 30 15:56
ol_mf_1_147__03bds8qf_.arc
-r--r----- . 1 oracle 54331 169270272 May 30 15:59
ol_mf_1_148__03gyt7rx_.arc
-r--r----- . 1 oracle 54331 170712576 May 30 16:01
ol_mf_1_149__03mfx17v_.arc
-r--r----- . 1 oracle 54331 170744832 May 30 16:04
ol_mf_1_150__03qzz0ty_.arc
-r--r----- . 1 oracle 54331 169380864 May 30 16:06
ol_mf_1_151__03wgxdry_.arc
-r--r----- . 1 oracle 54331 169833984 May 30 16:09
ol_mf_1_152__040y85v3_.arc
-r--r----- . 1 oracle 54331 165134336 May 30 16:20
ol_mf_1_153__04ox946w_.arc
-r--r----- . 1 oracle 54331 169929216 May 30 16:22
ol_mf_1_154__04rbv7n8_.arc
-r--r----- . 1 oracle 54331 171903488 May 30 16:23
ol_mf_1_155__04tvlyvn_.arc
-r--r----- . 1 oracle 54331 179061248 May 30 16:25
ol_mf_1_156__04xgfjtl_.arc
-r--r----- . 1 oracle 54331 173593088 May 30 16:26
ol_mf_1_157__04zyg8hw_.arc
-r--r----- . 1 oracle 54331 175999488 May 30 16:27
ol_mf_1_158__052gp9mt_.arc
-r--r----- . 1 oracle 54331 179092992 May 30 16:29
ol_mf_1_159__0551wk7s_.arc
-r--r----- . 1 oracle 54331 175524352 May 30 16:30
ol_mf_1_160__057146my_.arc
-r--r----- . 1 oracle 54331 173949440 May 30 16:32
ol_mf_1_161__05b2dmwp_.arc
-r--r----- . 1 oracle 54331 184166912 May 30 16:33
ol_mf_1_162__05drbj8n_.arc
-r--r----- . 1 oracle 54331 173026816 May 30 16:35
ol_mf_1_163__05h8lm1h_.arc
-r--r----- . 1 oracle 54331 174286336 May 30 16:36
ol_mf_1_164__05krsqmh_.arc
-r--r----- . 1 oracle 54331 166092288 May 30 16:37
ol_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__063ltvgv_.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. Oracle 사용자로, 새 EC2 인스턴스 DB 호스트 ora_02, ORACLE_SID에 ORACLE_HOME 변수를 현재 Oracle 설치 상태로 설정합니다. 이 경우 db1입니다.
10. Oracle 사용자는 적절한 관리 디렉토리가 구성된 \$ORACLE_HOME/dbs 디렉토리에 일반 Oracle 초기화 파일을 생성합니다. 가장 중요한 것은 Oracle을 사용하는 것입니다 flash recovery area 운영 Oracle VLDB 인스턴스에 정의된 FSxN NFS 마운트 경로를 가리킵니다. flash recovery area 구성은 섹션에서 설명합니다 Setup Oracle RMAN incremental merge to image copy on FSx. Oracle 제어 파일을 FSx ONTAP NFS 파일 시스템으로 설정합니다.

```
vi $ORACLE_HOME/dbs/initdb1.ora
```

다음 예제 항목이 있는 경우:

```
*.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'
```

일치하지 않는 경우 위의 init 파일을 운영 Oracle DB 서버에서 복구된 백업 초기화 파일로 교체해야 합니다.

11. Oracle 사용자로 RMAN을 실행하여 새 EC2 DB 인스턴스 호스트에서 Oracle 복구를 실행합니다.


```

[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

```

- 데이터베이스 ID를 설정합니다. 데이터베이스 ID는 FSx NFS 마운트 지점에 있는 이미지 복제본의 Oracle 파일 이름에서 검색할 수 있습니다.

```

RMAN> set dbid = 1730530050;

executing command: SET DBID

```

- 자동 백업에서 제어 파일을 복원합니다. Oracle 제어 파일 및 spfile 자동 백업이 활성화된 경우 모든 증분 백업 및 병합 주기에 백업됩니다. 여러 복제본을 사용할 수 있는 경우 최신 백업이 복구됩니다.

```

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. 나중에 기본 DB 인스턴스와 일치하도록 매개 변수 파일을 업데이트하기 위해 spfile에서 /tmp 폴더로 초기화 파일을 복구합니다.

```

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. 제어 파일을 마운트하고 데이터베이스 백업 이미지 복제본을 확인합니다.

```
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_4nlt509m
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-
SYS_AUX_FNO-10_4ilt5083
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_4olt509m
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-
SYS_AUX_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-
SYS_AUX_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. 데이터베이스를 복원하지 않고 복구를 실행하도록 데이터베이스를 복제본으로 전환합니다.

```

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. 플래시 복구 영역에서 사용 가능한 마지막 아카이브 로그까지 Oracle 복구를 실행합니다.

```
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__04tvlyvn_.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__057146my_.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__05pmg741_.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__057146m
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__0631tvq
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
```



더 빠른 복구를 위해서는 `recovery_parallelism` 매개 변수를 사용하여 병렬 세션을 설정하거나 데이터베이스 복구를 위한 복구 명령에서 병렬 처리 수준을 지정합니다. `RECOVER DATABASE PARALLEL (DEGREE d INSTANCES DEFAULT);`. 일반적으로 병렬 처리 수는 호스트의 CPU 코어 수와 같아야 합니다.

18. RMAN을 종료하고 `sqlplus`를 통해 Oracle 사용자로 로그인하여 데이터베이스를 열고 불완전한 복구 후 로그를 재설정합니다.

```
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. 데이터베이스가 운영 데이터베이스 장애 전에 삽입한 행이 있는 새 호스트로 복구되었는지 확인합니다.


```
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
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. 기타 복구 후 작업

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,wsiz=262144,noin
tr 0          0
```

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

이로써 FSxN NFS 파일 시스템의 백업 이미지 복제본에서 새로운 EC2 DB 인스턴스 호스트로 Oracle VLDB 데이터베이스 복구가 완료됩니다.

다른 활용 사례를 위해 **Oracle** 대기 이미지 복제본을 복제합니다

Oracle VLDB 이미지 복제본을 스테이징하기 위해 AWS FSx ONTAP를 사용하는 또 다른 이점은 FlexCloned가 최소한의 추가 스토리지 투자로 다른 많은 용도로 사용할 수 있다는 것입니다. 다음 활용 사례에서는 FSx ONTAP에서 DEV, UAT 등과 같은 다른 Oracle 활용 사례에 대해 스테이징 NFS 볼륨을 스냅샷 및 복제하는 방법을 시연합니다

1. 먼저 이전에 만든 것과 동일한 테스트 테이블에 행을 삽입합니다.

```
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. RMAN 백업을 수행하고 FSx ONTAP 데이터베이스 이미지 복사본에 병합하여 FSx NFS 마운트의 백업 세트에 트랜잭션이 캡처되지만 클론 데이터베이스가 복구될 때까지 복제본으로 병합되지 않도록 합니다.

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

3. ssh를 통해 fsxadmin 사용자로 FSx 클러스터에 로그인하여 예약된 백업 정책으로 생성된 스냅샷을 관찰하고 Oracle은 1단계에서 커밋한 트랜잭션을 포함하도록 일회성 스냅샷을 생성합니다.

```
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. 대체 EC2 Oracle 호스트에서 새 DB1 클론 인스턴스를 설치하는 데 사용할 일회성 스냅샷에서 클론을 생성합니다. 볼륨 ora_01_copy에 대해 사용 가능한 모든 일별 스냅샷에서 클론을 생성할 수 있습니다.

```
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. 클론 복제된 볼륨을 보호하려는 경우를 제외하고 상위 볼륨 스냅샷 정책을 상속하므로 클론 복제된 볼륨에 대한 스냅샷 정책을 해제한 다음 그대로 둡니다.

```
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. 운영 Oracle EC2 인스턴스와 동일한 버전 및 패치 레벨로 사전 설치된 Oracle 소프트웨어를 사용하여 새로운 EC2 Linux 인스턴스에 로그인하고 복제된 볼륨을 마운트합니다.

```
[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,rsiz=262144,wsiz=262144,noi
tr
```

7. FSx NFS 마운트에서 데이터베이스 증분 백업 세트, 이미지 복사본 및 사용 가능한 아카이브 로그를 검증합니다.

```
[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_8l1tkrap
-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 9e1tv01d_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 9g1tv01t_304_1_1
-rw-r----- 1 oracle 54331    319488 Jun  5 15:31 9h1tv01t_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 9f1tv01t_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_17vwwvt9_.arc
-rw-r----- 1 oracle 54331 166651904 Jun  5 14:44
o1_mf_1_191_17vx6vmg_.arc
-rw-r----- 1 oracle 54331 167406080 Jun  5 14:47
o1_mf_1_192_17vxctms_.arc
-rw-r----- 1 oracle 54331 166868992 Jun  5 14:49
o1_mf_1_193_17vxjjps_.arc
-rw-r----- 1 oracle 54331 166087168 Jun  5 14:52
o1_mf_1_194_17vxnxrh_.arc
-rw-r----- 1 oracle 54331 175210496 Jun  5 14:54
o1_mf_1_195_17vxswv5_.arc
-rw-r----- 1 oracle 54331 167078400 Jun  5 14:57
o1_mf_1_196_17vxylwp_.arc
-rw-r----- 1 oracle 54331 169701888 Jun  5 14:59
o1_mf_1_197_17vy3cyw_.arc
-rw-r----- 1 oracle 54331 167845376 Jun  5 15:02
o1_mf_1_198_17vy8245_.arc
-rw-r----- 1 oracle 54331 170763776 Jun  5 15:05

```

```

o1_mf_1_199_17vydv4c_.arc
-rw-r----- 1 oracle 54331 193853440 Jun  5 15:07
o1_mf_1_200_17vykf23_.arc
-rw-r----- 1 oracle 54331 165523968 Jun  5 15:09
o1_mf_1_201_17vyp1dh_.arc
-rw-r----- 1 oracle 54331 161117184 Jun  5 15:12
o1_mf_1_202_17vyvrm5_.arc
-rw-r----- 1 oracle 54331  10098176 Jun  5 15:21
o1_mf_1_203_17vzdfwm_.arc

```

8. 이제 복구 프로세스는 기본 운영 인스턴스와 일치하도록 장애 설정 Oracle 환경(`oratab`, `$oracle_home`, `$oracle_sid`)이 설정된 후 새로운 EC2 DB 인스턴스에 대한 복구의 이전 사용 사례와 유사합니다. FSx NFS 마운트의 플래시 복구 디렉토리를 가리키는 `db_recovery_file_dest_size` 및 `db_recovery_file_dest`를 포함한 초기화 파일을 생성합니다. 그런 다음 복구를 실행하기 위해 RMAN을 사용합니다. 다음은 명령 단계와 출력입니다.

```

[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_17vzyb
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_17vzyb
vq_.bkp
channel ORA_DISK_1: control file restore from AUTOBACKUP complete
output file name=/nfsfsxn/oracopy/db1.ct1
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_8dltkri1
      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_8altkrhr
      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_7q1tkqsl
      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_7ultkr42
      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_17vwvvt9_.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_17vx6vmg_.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_17vxctms_.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_17vxjjps_.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_17vxnxrh_.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_17vxswv5_.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_17vxlwp_.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_17vy3cyw_.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_17vy8245_.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_17vydv4c_.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_17vykf23_.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_17vyp1dh_.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_17vyvrm5_.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_17vzdfwm_.arc
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_190_17vwvvt9
_.arc thread=1 sequence=190
archived log file

```
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_191_17vx6vmg
_.arc thread=1 sequence=191
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_192_17vxctms
_.arc thread=1 sequence=192
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_193_17vxjjps
_.arc thread=1 sequence=193
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_194_17vxnxrh
_.arc thread=1 sequence=194
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_195_17vxswv5
_.arc thread=1 sequence=195
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_196_17vxyllwp
_.arc thread=1 sequence=196
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_197_17vy3cyw
_.arc thread=1 sequence=197
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_198_17vy8245
_.arc thread=1 sequence=198
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_199_17vydv4c
_.arc thread=1 sequence=199
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_200_17vykf23
_.arc thread=1 sequence=200
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_201_17vyp1dh
_.arc thread=1 sequence=201
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_202_17vyvrm5
_.arc thread=1 sequence=202
archived log file
name=/nfsfsxn/archlog/DB1/archivelog/2023_06_05/o1_mf_1_203_17vzdfwm
_.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_l81bhgz6g_.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_7jltkqk6
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-22_7kltkqk6
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-23_7lltkqk6
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-24_7mltkqk6
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-25_7nltkqrh
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-26_7oltkqrj
```

NAME

```
-----
-----
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-27_7pltkqrq
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-28_7qltkqs1
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-29_7rltkr32
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-30_7sltkr3a
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-31_7tltkr3i
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-32_7ultkr42
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-33_7vltkra6
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-34_80ltkram
/nfsfsxn/oracopy/data_D-DB1_I-1730530050_TS-SOE_FNO-35_81ltkrap
/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. 복제된 데이터베이스 인스턴스의 이름을 바꾸고 데이터베이스 ID를 Oracle nid 유틸리티로 변경합니다. 데이터베이스 인스턴스 상태가 인 상태여야 합니다 mount 명령을 실행합니다.

```

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_7ultkr4 - 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_l81bhzh6g_.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_181bj13g_.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. oratab, init 파일에서 Oracle 데이터베이스 환경 구성을 새 데이터베이스 이름 또는 인스턴스 ID로 변경하고 새 인스턴스 ID와 일치하는 필요한 관리 디렉토리를 생성합니다. 그런 다음 resetlogs 옵션을 사용하여 인스턴스를 시작합니다.

```
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 pdba
```

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 pdba
```

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>
```

그러면 개발, UAT 또는 기타 활용 사례에 대해 FSx NFS 마운트의 스테이징 데이터베이스 복제본에서 새 Oracle 인스턴스의 클론이 완료됩니다. 동일한 스테이징 이미지 복사본에서 여러 Oracle 인스턴스를 복제할 수 있습니다.



오류가 발생한 경우 RMAN-06571: datafile 1 does not have recoverable copy 데이터베이스를 복제본으로 전환할 때 1차 운영 DB와 일치하는 데이터베이스 생성을 확인하십시오. 필요한 경우 RMAN 명령에 따라 감금치를 재설정합니다 reset database to incarnation n;.

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 문서 및/또는 웹 사이트를 참조하십시오.

- RMAN: 병합된 증분 백업 전략(Doc ID 745798.1)

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

- RMAN 백업 및 복구 사용자 가이드

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

- NetApp ONTAP용 Amazon FSx

["https://aws.amazon.com/fsx/netapp-ontap/"](https://aws.amazon.com/fsx/netapp-ontap/)

- Amazon EC2

https://aws.amazon.com/pm/ec2/?trk=36c6da98-7b20-48fa-8225-4784bced9843&sc_channel=ps&s_kwcid=AL14422!3!467723097970!e!!g!!aws%20ec2&ef_id=Cj0KCQiA54KfBhCKARIsAJzSrdqwQrghn6l71jiWzSeaT9Uh1-vY-VfhJixF-xnv5rWwn2S7RqZOTQ0aAh7eEALw_wcB:G:s&s_kwcid=AL14422!3!467723097970!e!!g!!aws%20ec2

TR-4974: NFS/ASM을 사용하는 AWS FSx/EC2에서 독립 실행형 재가동 시 Oracle 19c

Allen Cao, Niyaz Mohamed, NetApp

목적

ASM(Automatic Storage Management)은 많은 Oracle 설치에 사용되는 일반적인 Oracle 스토리지 볼륨 관리자입니다. 또한 Oracle에서 권장하는 스토리지 관리 솔루션이기도 합니다. 기존 볼륨 관리자 및 파일 시스템을 대체할 수 있습니다. Oracle 버전 11g부터 ASM은 데이터베이스가 아닌 그리드 인프라와 함께 패키지로 제공됩니다. 따라서 RAC를 사용하지 않는 스토리지 관리에 Oracle ASM을 활용하려면 Oracle Restart라고도 하는 독립 실행형 서버에 Oracle 그리드 인프라를 설치해야 합니다. 이렇게 하면 더욱 단순한 Oracle 데이터베이스 구축에 따르는 복잡성이 더욱 증가합니다. 그러나 이름에서 알 수 있듯이 Oracle이 재시작 모드로 구축되면 사용자 개입 없이 호스트를 재부팅한 후 오류가 발생한 Oracle 서비스가 다시 시작됩니다. 이를 통해 특정 수준의 고가용성 또는 HA 기능을 제공할 수 있습니다.

Oracle ASM은 일반적으로 FC, iSCSI 스토리지 프로토콜 및 LUN에 원시 스토리지 디바이스로 구축됩니다. 하지만, NFS 프로토콜의 ASM과 NFS 파일 시스템은 Oracle의 구성도 지원합니다. 이 문서에서는 EC2 컴퓨팅 인스턴스가 있는 ONTAP 스토리지 환경을 위한 Amazon FSx에서 NFS 프로토콜과 Oracle ASM을 사용하여 Oracle 19c 데이터베이스를 구축하는 방법을 보여 줍니다. 또한 NetApp BlueXP 콘솔을 통해 NetApp SnapCenter 서비스를 사용하여 개발/테스트 또는 기타 사용 사례에서 AWS 퍼블릭 클라우드에서 스토리지 효율적인 데이터베이스 작업을 위해 Oracle 데이터베이스를 백업, 복원, 복제하는 방법을 보여줍니다.

이 솔루션은 다음과 같은 사용 사례를 해결합니다.

- ONTAP 스토리지용 Amazon FSx 및 NFS/ASM을 사용하는 EC2 컴퓨팅 인스턴스에 Oracle 데이터베이스 구축
- NFS/ASM을 사용하여 퍼블릭 AWS 클라우드에서 Oracle 워크로드 테스트 및 검증
- AWS에 구축된 Oracle 데이터베이스 재시작 기능의 테스트 및 검증

대상

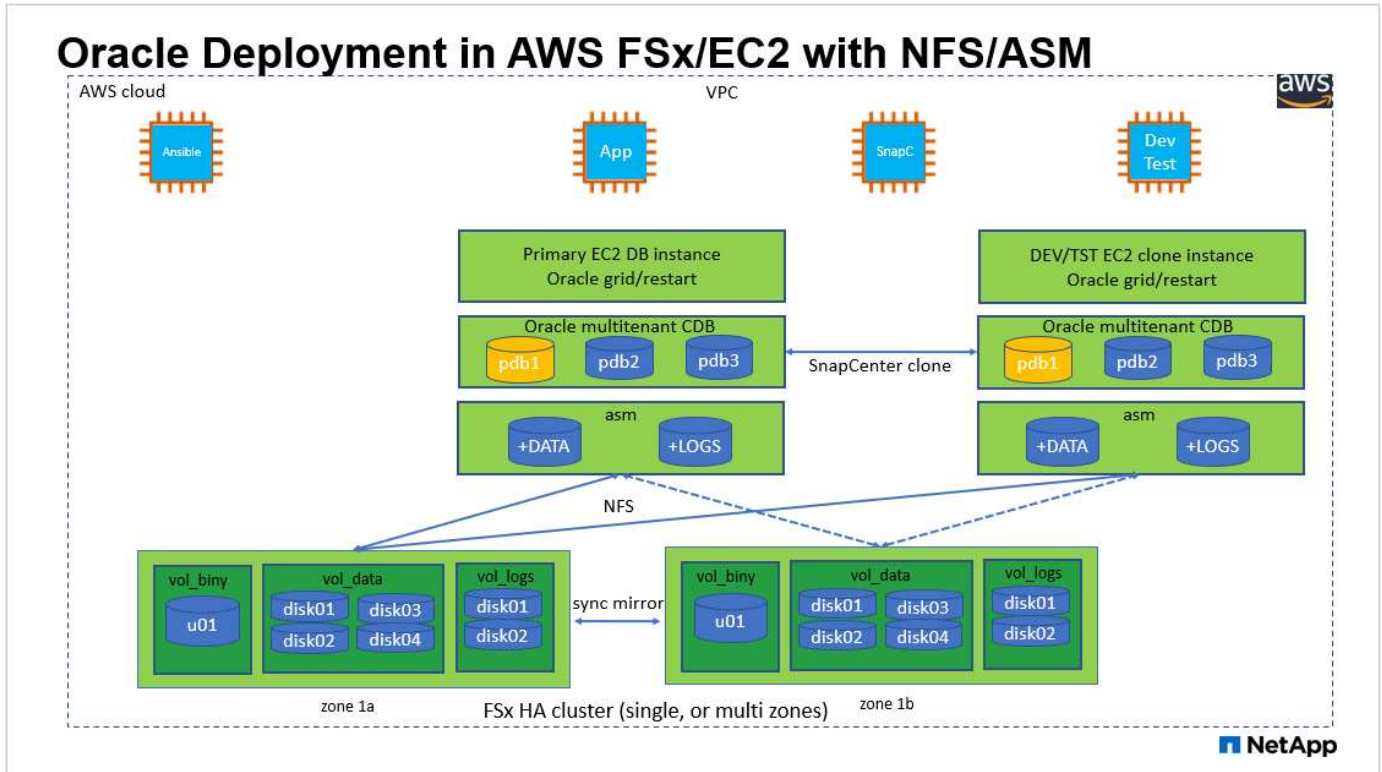
이 솔루션은 다음과 같은 사용자를 대상으로 합니다.

- NFS/ASM을 통해 AWS 퍼블릭 클라우드에 Oracle을 구축하고 싶은 DBA
- AWS 퍼블릭 클라우드에서 Oracle 워크로드를 테스트하려는 데이터베이스 솔루션 설계자
- AWS FSx 스토리지에 구축된 Oracle 데이터베이스를 구축하고 관리하려는 스토리지 관리자
- AWS FSx/EC2에서 Oracle 데이터베이스를 가동하려는 애플리케이션 소유자입니다.

솔루션 테스트 및 검증 환경

이 솔루션의 테스트 및 검증은 최종 구축 환경과 일치하지 않을 수 있는 AWS FSx 및 EC2 환경에서 수행되었습니다. 자세한 내용은 섹션을 참조하십시오 [\[Key Factors for Deployment Consideration\]](#).

있습니다



하드웨어 및 소프트웨어 구성 요소

* 하드웨어 *

FSX ONTAP 저장소	AWS에서 제공하는 현재 버전입니다	동일한 VPC 및 가용성 존에 FSx HA 클러스터 1개
컴퓨팅용 EC2 인스턴스	T2.xLarge/4vCPU/16G	EC2 T2 xLarge EC2 인스턴스 2개, 하나는 운영 DB 서버로, 다른 하나는 클론 DB 서버로
* 소프트웨어 *		
RedHat Linux	RHEL-8.6.0_HVM-20220503-x86_64-2-Hourly2-GP2	테스트를 위해 RedHat 서브스크립션을 배포했습니다
Oracle Grid Infrastructure	버전 19.18	RU 패치 p34762026_190000_Linux-x86-64.zip 를 적용했습니다
Oracle 데이터베이스	버전 19.18	RU 패치 p34765931_190000_Linux-x86-64.zip 를 적용했습니다
Oracle OPatch	버전 12.2.0.1.36	최신 패치 p6880880_190000_Linux-x86-64.zip
SnapCenter 서비스	버전	v2.3.1.2324

구축 시 고려해야 할 주요 요소

- * EC2 컴퓨팅 인스턴스 * 이러한 테스트 및 검증에서는 Oracle 데이터베이스 컴퓨팅 인스턴스에 AWS EC2 T2.xLarge 인스턴스 유형을 사용했습니다. 운영 구축 환경에서는 데이터베이스 워크로드에 최적화된 M5 유형 EC2 인스턴스를 Oracle의 컴퓨팅 인스턴스로 사용하는 것이 좋습니다. 실제 워크로드 요구 사항에 따라 vCPU 수와 RAM 양에 따라 EC2 인스턴스를 적절하게 사이징해야 합니다.
- * FSx 스토리지 HA 클러스터 단일 또는 다중 영역 배포. * 이러한 테스트 및 검증에서는 단일 AWS 가용성 영역에 FSx HA 클러스터를 구축했습니다. 프로덕션 배포를 위해 FSx HA 쌍을 두 가지 가용성 영역에 배포하는 것이 좋습니다. FSx HA 클러스터는 스토리지 레벨 이중화를 제공하기 위해 액티브-패시브 파일 시스템 쌍으로 미러링되는 HA 쌍으로 프로비저닝됩니다. 다중 영역 구축을 통해 단일 AWS 영역에서 장애가 발생할 경우 고가용성을 더욱 강화할 수 있습니다.
- * FSx 스토리지 클러스터 크기 조정 * ONTAP 스토리지 파일 시스템용 Amazon FSx는 최대 160,000개의 원시 SSD IOPS, 최대 4Gbps 처리량 및 최대 192TiB 용량을 제공합니다. 그러나 구현 시 실제 요구 사항에 따라 프로비저닝된 IOPS, 처리량 및 스토리지 제한(최소 1,024GiB)을 기준으로 클러스터 크기를 조정할 수 있습니다. 애플리케이션 가용성에 영향을 주지 않고 용량을 즉각적으로 동적으로 조정할 수 있습니다.
- * Oracle 데이터 및 로그 레이아웃 * 테스트 및 검증에서 각각 데이터 및 로그용 ASM 디스크 그룹 2개를 구축했습니다. DATA ASM 디스크 그룹 내에서 데이터 NFS 파일 시스템 마운트 지점에 4개의 디스크를 프로비저닝했습니다. logs ASM 디스크 그룹 내에서 로그 NFS 파일 시스템 마운트 지점에 디스크 2개를 프로비저닝했습니다. 대규모 데이터베이스 배포의 경우 ASM 디스크 그룹을 구축하여 여러 FSx 파일 시스템에 ASM NFS 디스크를 분산하고 FSx 파일 시스템에 고정된 여러 NFS 마운트 지점을 통해 분산할 수 있습니다. 이 설정은 4Gbps 이상의 처리량과 160,000개의 원시 SSD IOPS 요구 사항을 충족하도록 설계되었습니다.
- * dNFS 구성. * dNFS는 Oracle 커널에 내장되어 있으며 Oracle을 NFS 스토리지에 구축할 때 Oracle 데이터베이스 성능을 크게 향상시키는 것으로 알려져 있습니다. dNFS는 Oracle 바이너리로 패키징되지만 기본적으로 활성화되어 있지 않습니다. NFS 기반 Oracle 데이터베이스 구축 시 이 기능을 설정해야 합니다. 대형 데이터베이스에 여러 FSx 파일 시스템을 구축하는 경우 dNFS 다중 경로를 올바르게 구성해야 합니다.
- * 사용자가 생성한 각 Oracle ASM 디스크 그룹에 사용할 Oracle ASM 이중화 레벨 * FSx는 이미 FSx 클러스터 레벨에서 스토리지를 미러링하므로 반드시 필요합니다 ONLY External Redundancy(외부 이중화 사용): 이 옵션을 사용하면 Oracle ASM이 디스크 그룹의 내용을 미러링할 수 없습니다. 이는 Oracle 데이터베이스 데이터

스토리지용 NFS의 경우 Oracle 수준에서 ASM 콘텐츠를 미러링할 필요가 없는 하드 NFS 마운트 옵션이 필요하므로 특히 중요합니다.

- * 데이터베이스 백업 * NetApp은 클라우드에서 데이터베이스 백업, 복원, 복제를 위한 SnapCenter 소프트웨어 서비스의 SaaS 버전을 제공하며 NetApp BlueXP 콘솔 UI를 통해 이용할 수 있습니다. 신속한(1분 이내) 스냅샷 백업, 빠른(몇 분) 데이터베이스 복원, 데이터베이스 복제를 위해 이러한 서비스를 구현하는 것이 좋습니다.

솔루션 구축

다음 섹션에서는 단계별 배포 절차를 제공합니다.

배포를 위한 사전 요구 사항

배포에는 다음과 같은 사전 요구 사항이 필요합니다.

1. AWS 계정이 설정되었으며 AWS 계정 내에 필요한 VPC 및 네트워크 세그먼트가 생성되었습니다.
2. AWS EC2 콘솔에서 2개의 EC2 Linux 인스턴스를 구축해야 합니다. 하나는 운영 Oracle DB 서버로, 다른 하나는 선택적 대체 클론 타겟 DB 서버입니다. 환경 설정에 대한 자세한 내용은 이전 섹션의 아키텍처 다이어그램을 참조하십시오. 또한 를 검토합니다 "[Linux 인스턴스에 대한 사용자 가이드](#)" 를 참조하십시오.
3. AWS EC2 콘솔에서 ONTAP 스토리지 HA 클러스터용 Amazon FSx를 구축하여 Oracle 데이터베이스 볼륨을 호스팅합니다. FSx 저장소 배포에 익숙하지 않은 경우 설명서를 참조하십시오 "[ONTAP 파일 시스템용 FSx 생성](#)" 을 참조하십시오.
4. 2단계와 3단계는 라는 EC2 인스턴스를 만드는 다음 Terraform 자동화 툴킷을 사용하여 수행할 수 있습니다 ora_01 이라는 FSx 파일 시스템이 있습니다 fsx_01. 실행 전에 지침을 주의 깊게 검토하고 환경에 맞게 변수를 변경하십시오.

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



Oracle 설치 파일을 스테이징할 충분한 공간을 확보하기 위해 EC2 인스턴스 루트 볼륨에 50G 이상을 할당했는지 확인합니다.

EC2 인스턴스 커널 구성

사전 요구 사항이 용량 할당된 상태에서 EC2 인스턴스에 EC2-USER 및 sudo로 로그인하여 Oracle 설치를 위한 Linux 커널을 구성합니다.

1. 스테이징 디렉터리를 만듭니다 /tmp/archive 폴더를 지정하고 를 설정합니다 777 권한.

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

2. Oracle 바이너리 설치 파일 및 기타 필요한 rpm 파일을 에 다운로드하고 스테이징합니다 /tmp/archive 디렉토리.

에 명시된 설치 파일의 다음 목록을 참조하십시오 /tmp/archive EC2 인스턴스에서 선택합니다.

```
[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. 대부분의 커널 구성 요구 사항을 충족하는 Oracle 19c 사전 설치 RPM을 설치합니다.

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

4. 누락된 을 다운로드하고 설치합니다 compat-libcap1 Linux 8에서


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

5. NetApp에서 NetApp 호스트 유틸리티를 다운로드하고 설치합니다.

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

6. 설치합니다 `policycoreutils-python-utils` 이는 EC2 인스턴스에서 사용할 수 없습니다.

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

7. 열려 있는 JDK 버전 1.8을 설치합니다.

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

8. NFS-utils를 설치합니다.

```
yum install nfs-utils
```

9. 현재 시스템에서 투명 HugePages를 비활성화합니다.

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

에 다음 행을 추가합니다 /etc/rc.local 를 눌러 비활성화합니다 transparent_hugepage 재부팅 후:

```
# 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. SELinux를 변경하여 해제합니다 SELINUX=enforcing 를 선택합니다 SELINUX=disabled. 변경 사항을 적용하려면 호스트를 재부팅해야 합니다.

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

11. 에 다음 행을 추가합니다 `limit.conf` 따옴표 없이 파일 설명자 제한과 스택 크기를 설정하려면 다음을 수행합니다 " ".

```
vi /etc/security/limits.conf
**          hard    nofile      65536"
**          soft    stack       10240"
```

12. 다음 명령을 실행하여 EC2 인스턴스에 스왑 공간을 추가합니다. "[스왑 파일을 사용하여 Amazon EC2 인스턴스에서 스왑 공간으로 사용할 메모리를 어떻게 할당합니까?](#)" 정확한 추가 공간은 최대 16G RAM의 크기에 따라 달라집니다.
13. ASM `sysasm` 그룹에 사용할 ASM 그룹을 추가합니다

```
groupadd asm
```

14. Oracle 사용자를 수정하여 ASM을 보조 그룹으로 추가합니다(Oracle 사용자는 Oracle 사전 설치 RPM 설치 후 생성되어야 함).

```
usermod -a -G asm oracle
```

15. EC2 인스턴스를 재부팅합니다.

EC2 인스턴스 호스트에 마운트할 **NFS** 볼륨을 프로비저닝하고 내보냅니다

FSx 클러스터 관리 IP를 사용하여 ssh를 통해 FSx 클러스터에 로그인하여 Oracle 데이터베이스 바이너리, 데이터 및 로그 파일을 호스팅하여 명령줄에서 세 개의 볼륨을 프로비저닝합니다.

1. SSH를 통해 FSx 클러스터에 fsxadmin 사용자로 로그인합니다.

```
ssh fsxadmin@172.30.15.53
```

2. 다음 명령을 실행하여 Oracle 바이너리에 대한 볼륨을 생성합니다.

```
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. 다음 명령을 실행하여 Oracle 데이터용 볼륨을 생성합니다.

```
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. 다음 명령을 실행하여 Oracle 로그용 볼륨을 생성합니다.

```
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. 생성된 DB 볼륨을 확인합니다.

```
vol show
```

이 문제는 다음 항목을 반환해야 합니다.

```

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.

```

데이터베이스 스토리지 구성

이제 EC2 인스턴스 호스트에서 Oracle 그리드 인프라 및 데이터베이스 설치를 위한 FSx 스토리지를 가져와서 설정합니다.

1. SSH 키와 EC2 인스턴스 IP 주소를 사용하여 SSH를 통해 EC2 사용자로 EC2 인스턴스에 로그인합니다.

```
ssh -i ora_01.pem ec2-user@172.30.15.58
```

2. Oracle 바이너리 파일 시스템을 마운트할 /u01 디렉토리를 생성합니다

```
sudo mkdir /u01
```

3. 에 바이너리 볼륨을 마운트합니다 `/u01`에서 FSx NFS lif IP 주소로 변경되었습니다. NetApp 자동화 툴킷을 통해 FSx 클러스터를 구축한 경우, FSx 가상 스토리지 서버 NFS lif IP 주소가 리소스 프로비저닝 실행이 끝날 때 출력에 나열됩니다. 그렇지 않으면 AWS FSx 콘솔 UI에서 검색할 수 있습니다.

```
sudo mount -t nfs 172.30.15.19:/ora_01_biny /u01 -o rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsiz=65536
```

4. 변경 /u01 마운트 지점 소유권을 Oracle 사용자 및 연결된 기본 그룹에 지정합니다.

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

5. Oracle 데이터 파일 시스템을 마운트하기 위한 /oradata 디렉토리를 생성합니다

```
sudo mkdir /oradata
```

6. 에 데이터 볼륨을 마운트합니다 `/oradata`에서 FSx NFS lif IP 주소로 변경되었습니다

```
sudo mount -t nfs 172.30.15.19:/ora_01_data /oradata -o rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsiz=65536
```

7. 변경 /oradata 마운트 지점 소유권을 Oracle 사용자 및 연결된 기본 그룹에 지정합니다.

```
sudo chown oracle:oinstall /oradata
```

8. Oracle 로그 파일 시스템을 마운트하기 위한 디렉토리를 생성/또는 생성합니다

```
sudo mkdir /oralogs
```

9. 예 로그 볼륨을 마운트합니다 `/oralogs`에서 FSx NFS lif IP 주소로 변경되었습니다

```
sudo mount -t nfs 172.30.15.19:/ora_01_logs /oralogs -o  
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsiz=65536,wsiz=65536
```

10. 변경 /oralogs 마운트 지점 소유권을 Oracle 사용자 및 연결된 기본 그룹에 지정합니다.

```
sudo chown oracle:oinstall /oralogs
```

11. 예 마운트 지점을 추가합니다 /etc/fstab.

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

다음 줄을 추가합니다.

```
172.30.15.19:/ora_01_biny      /u01          nfs  
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsiz=65536,wsiz=65536  0  
0  
172.30.15.19:/ora_01_data    /oradata      nfs  
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsiz=65536,wsiz=65536  0  
0  
172.30.15.19:/ora_01_logs    /oralogs      nfs  
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsiz=65536,wsiz=65536  0  
0
```

12. sudo를 Oracle 사용자에게 생성하고 ASM 디스크 파일을 저장할 ASM 폴더를 생성합니다

```
sudo su  
su - oracle  
mkdir /oradata/asm  
mkdir /oralogs/asm
```

13. Oracle 사용자는 ASM 데이터 디스크 파일을 생성하고 블록 크기의 디스크 크기와 일치하도록 개수를 변경합니다.

```
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. 루트 사용자로 데이터 디스크 파일 권한을 640으로 변경합니다

```
chmod 640 /oradata/asm/*
```

15. Oracle 사용자는 ASM 로그 디스크 파일을 생성하고 블록 크기의 디스크 크기와 일치하도록 개수를 변경합니다.

```
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. 루트 사용자로 로그 디스크 파일 권한을 640으로 변경합니다

```
chmod 640 /oralogs/asm/*
```

17. EC2 인스턴스 호스트를 재부팅합니다.

Oracle 그리드 인프라 설치

1. SSH를 통해 EC2 인스턴스로 로그인하고 주석 처리를 해제하여 암호 인증을 활성화합니다
PasswordAuthentication yes 그런 다음 의견을 남기기 바랍니다 PasswordAuthentication no.

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

2. sshd 서비스를 다시 시작합니다.

```
sudo systemctl restart sshd
```

3. Oracle 사용자 암호를 재설정합니다.

```
sudo passwd oracle
```

4. Oracle Restart 소프트웨어 소유자 사용자(Oracle)로 로그인합니다. 다음과 같이 Oracle 디렉토리를 생성합니다.

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

5. 디렉터리 권한 설정을 변경합니다.

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

6. 그리드 홈 디렉토리를 만들고 변경합니다.

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

7. 그리드 설치 파일의 압축을 풉니다.

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

8. 그리드 홈에서 을 삭제합니다 OPatch 디렉토리.

```
rm -rf OPatch
```

9. 그리드 홈에서 복사합니다 p6880880_190000_Linux-x86-64.zip grid_home 으로 이동한 다음

압축을 풉니다.

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

10. 그리드 홈에서 수정합니다 `cv/admin/cvu_config`, 주석 취소 및 바꾸기 `CV_ASSUME_DISTID=OEL5` 와 함께 `CV_ASSUME_DISTID=OL7`.

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

11. 를 준비합니다 `gridsetup.rsp` 자동 설치용 파일 및 RSP 파일을 에 배치합니다 `/tmp/archive` 디렉토리. RSP 파일은 다음 정보를 사용하여 섹션 A, B 및 G를 포함해야 합니다.

```
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/*,/orlogs/asm/*
oracle.install.asm.diskGroup.diskDiscoveryString=/oradata/asm/nfs_data_
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. EC2 인스턴스에 루트 사용자로 로그인합니다.

13. 설치합니다 `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. EC2 인스턴스에 Oracle 사용자로 로그인하여 에서 패치를 추출합니다 `/tmp/archive` 폴더.

```
unzip p34762026_190000_Linux-x86-64.zip
```

15. 그리드 홈 `/u01/app/oracle/product/19.0.0/GRID`에서 Oracle 사용자로 를 실행합니다 `gridSetup.sh`

그리드 인프라 설치용.

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

그리드 인프라에 대해 잘못된 그룹에 대한 경고를 무시합니다. 단일 Oracle 사용자를 사용하여 Oracle Restart를 관리하고 있으므로 이 작업이 필요합니다.

16. 루트 사용자로 다음 스크립트를 실행합니다.

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

17. Oracle 사용자는 다음 명령을 실행하여 구성을 완료합니다.

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

18. Oracle 사용자로 로그 디스크 그룹을 생성합니다.

```
bin/asmca -silent -sysAsmPassword 'yourPWD' -asmsnmpPassword  
'yourPWD' -createDiskGroup -diskGroupName LOGS -disk  
'/orlogs/asm/nfs_logs_disk*' -redundancy EXTERNAL -au_size 4
```

19. Oracle 사용자는 설치 구성 후 그리드 서비스를 검증합니다.

```

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 데이터베이스 설치

1. Oracle 사용자로 로그인하고 설정을 해제합니다 \$ORACLE_HOME 및 \$ORACLE_SID 설정되어 있는지 확인합니다.

```
unset ORACLE_HOME
unset ORACLE_SID
```

2. Oracle DB 홈 디렉토리를 생성하고 변경합니다.

```
mkdir /u01/app/oracle/product/19.0.0/db1
cd /u01/app/oracle/product/19.0.0/db1
```

3. Oracle DB 설치 파일의 압축을 풉니다.

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

4. DB 홈에서 를 삭제합니다 OPatch 디렉토리.

```
rm -rf OPatch
```

5. DB 홈에서 복사합니다 p6880880_190000_Linux-x86-64.zip 를 선택합니다 `grid_home`그런 다음 압축을 풉니다.

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

6. DB 홈에서 수정한다 cv/admin/cvu_config`을 클릭합니다 `CV_ASSUME_DISTID=OEL5 와 함께 CV_ASSUME_DISTID=OL7.

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

7. 에서 /tmp/archive 디렉토리에서 DB 19.18 RU 패치의 포장을 풉니다.

```
unzip p34765931_190000_Linux-x86-64.zip
```

8. 에서 DB 자동 설치 RSP 파일을 준비합니다 /tmp/archive/dbinstall.rsp 다음 값이 있는 디렉터리:

```
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. db1 home/u01/app/oracle/product/19.0.0/db1에서 자동 소프트웨어 전용 DB 설치를 실행합니다.

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

10. 루트 사용자로 를 실행합니다 root.sh 소프트웨어 전용 설치 후 스크립트.

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

11. Oracle 사용자로 을 생성합니다 dbca.rsp 다음 항목이 있는 파일:

```
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
```



EC2 인스턴스 호스트에서 사용 가능한 메모리를 기준으로 총 메모리를 설정합니다. Oracle은 의 75%를 할당합니다 totalMemory DB 인스턴스 SGA 또는 버퍼 캐시로.

12. Oracle 사용자로 dbca를 사용하여 DB 생성을 시작합니다.

```
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. Oracle 사용자로서 DB 생성 후 Oracle Restart HA 서비스를 확인합니다.

```

[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.dbf.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. Oracle 사용자를 설정합니다 .bash_profile.


```
vi ~/.bash_profile
```

15. 다음 항목 추가:

```
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. 생성된 CDB/PDB를 검증합니다.

```
. ~/.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. Oracle 사용자로 Oracle 데이터베이스 홈 디렉토리 /u01/app/oracle/product/19.0.0/db1로 변경하고 dNFS를 활성화합니다

```
cd /u01/app/oracle/product/19.0.0/db1

mkdir rdbms/lib/odm

cp lib/libnfsodm19.so rdbms/lib/odm/
```

18. oranfstab 파일을 oracle_home에서 구성합니다

```
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. Oracle 사용자는 sqlplus에서 데이터베이스에 로그인하고 DB 복구 크기 및 위치를 +logs 디스크 그룹으로 설정합니다.

```
. ~/.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. 아카이브 로그 모드를 활성화하고 Oracle DB 인스턴스를 재부팅합니다

```
shutdown immediate;

startup mount;

alter database archivelog;

alter database open;

alter system switch logfile;
```

21. 인스턴스 재부팅 후 DB 로그 모드 및 dNFS를 확인합니다

```

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. Oracle ASM 확인

```

[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          /orlogs/asm/nfs_logs_disk01   MEMBER
  CACHED  NORMAL
LOGS_0001          /orlogs/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.

자동화된 구축 옵션

NetApp은 이 솔루션 구현을 지원하기 위해 Ansible과 함께 완전 자동화된 솔루션 배포 툴킷을 릴리즈할 것입니다. 도구 키트의 가용성을 다시 확인하십시오. 릴리스 후 링크가 여기에 게시됩니다.

SnapCenter 서비스를 사용한 Oracle 데이터베이스 백업, 복원 및 클론 복제

현재 NFS 및 ASM 스토리지 옵션이 포함된 Oracle 데이터베이스는 기존 SnapCenter 서버 UI 툴에서만 지원됩니다 (참조) "[SnapCenter을 사용한 하이브리드 클라우드 데이터베이스 솔루션](#)" NetApp SnapCenter UI 툴을 사용한

Oracle 데이터베이스 백업, 복원 및 클론에 대한 자세한 내용은

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 문서 및/또는 웹 사이트를 참조하십시오.

- 새 데이터베이스 설치를 통해 독립 실행형 서버용 Oracle Grid Infrastructure 설치

["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"](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)

- 응답 파일을 사용하여 Oracle 데이터베이스 설치 및 구성

["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"](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)

- NetApp ONTAP용 Amazon FSx

["https://aws.amazon.com/fsx/netapp-ontap/"](https://aws.amazon.com/fsx/netapp-ontap/)

- Amazon EC2

https://aws.amazon.com/pm/ec2/?trk=36c6da98-7b20-48fa-8225-4784bced9843&sc_channel=ps&s_kwcid=AL14422!3!467723097970!e!!g!!aws%20ec2&ef_id=Cj0KCQiA54KfBhCKARIsAJzSrdqwQrghn6l71jiWzSeaT9Uh1-vY-VfhJixF-xnv5rWwn2S7RqZOTQ0aAh7eEALw_wcB:G:s&s_kwcid=AL14422!3!467723097970!e!!g!!aws%20ec2

TR-4965: iSCSI/ASM을 사용하는 AWS FSx/EC2에서 Oracle 데이터베이스 구축 및 보호

Allen Cao, Niyaz Mohamed, NetApp

목적

ASM(Automatic Storage Management)은 많은 Oracle 설치에 사용되는 일반적인 Oracle 스토리지 볼륨 관리자입니다. 또한 Oracle에서 권장하는 스토리지 관리 솔루션이기도 합니다. 기존 볼륨 관리자 및 파일 시스템을 대체할 수 있습니다. Oracle 버전 11g 이후 ASM은 데이터베이스가 아닌 그리드 인프라와 함께 패키징되었습니다. 따라서 RAC를 사용하지 않는 스토리지 관리에 Oracle ASM을 활용하려면 Oracle Restart라고도 하는 독립 실행형 서버에 Oracle 그리드 인프라를 설치해야 합니다. 이렇게 하면 Oracle 데이터베이스 구현이 더 복잡해집니다. 그러나 이름에서 알 수 있듯이 Oracle이 재시작 모드로 구축되면 그리드 인프라스트럭처에 의해 또는 사용자 개입 없이 호스트를 재부팅한 후에 장애가 발생한 Oracle 서비스가 자동으로 다시 시작되어 특정 수준의 고가용성 또는 HA 기능을 제공합니다.

이 문서에서는 EC2 컴퓨팅 인스턴스가 있는 ONTAP 스토리지 환경을 위한 Amazon FSx에서 iSCSI 프로토콜과 Oracle ASM을 사용하여 Oracle 데이터베이스를 구축하는 방법을 보여 줍니다. 또한 NetApp BlueXP 콘솔을 통해 NetApp SnapCenter 서비스를 사용하여 개발/테스트 또는 기타 사용 사례에서 AWS 퍼블릭 클라우드에서 스토리지 효율적인 데이터베이스 작업을 위해 Oracle 데이터베이스를 백업, 복원, 복제하는 방법을 보여줍니다.

이 솔루션은 다음과 같은 사용 사례를 해결합니다.

- ONTAP 스토리지용 Amazon FSx 및 iSCSI/ASM을 사용하는 EC2 컴퓨팅 인스턴스에 Oracle 데이터베이스 구축

- iSCSI/ASM을 사용하여 퍼블릭 AWS 클라우드에서 Oracle 워크로드 테스트 및 검증
- AWS에 구축된 Oracle 데이터베이스 재시작 기능의 테스트 및 검증

대상

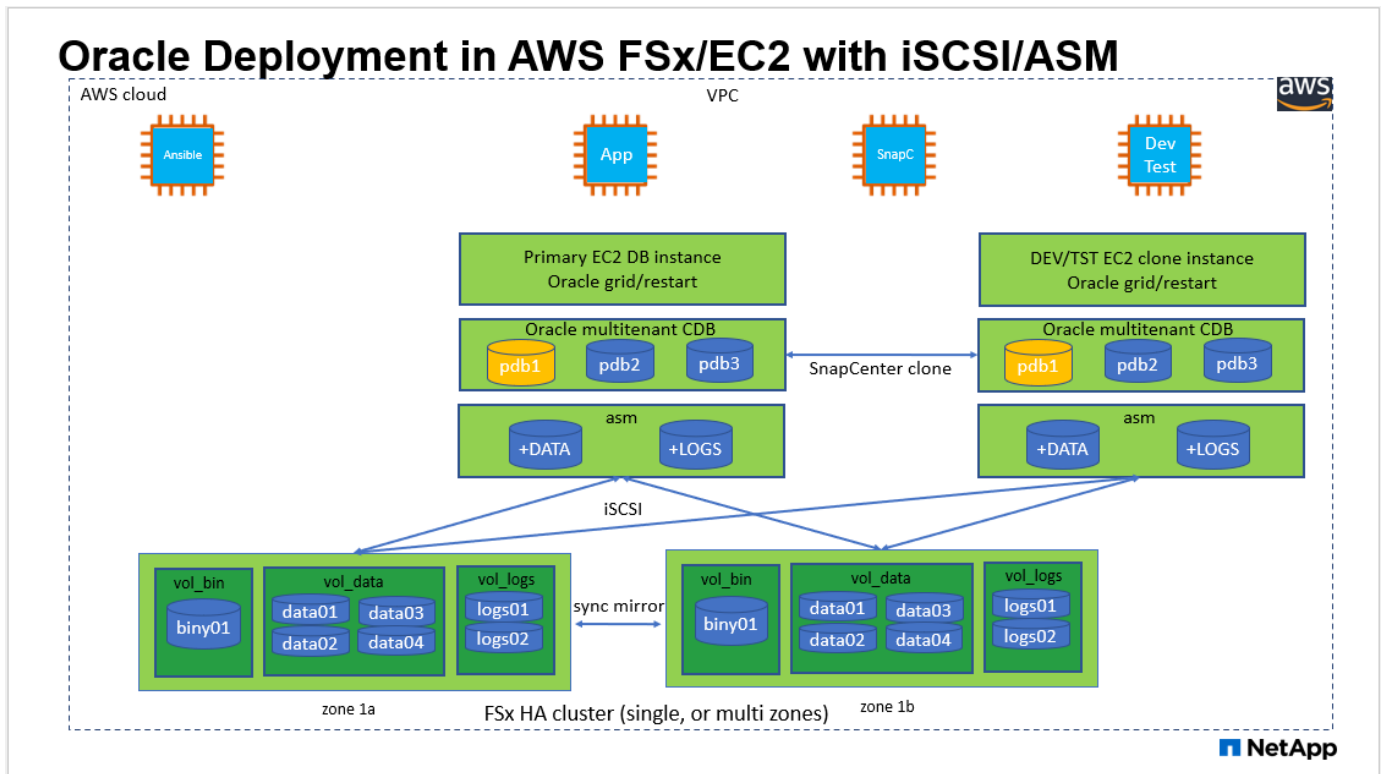
이 솔루션은 다음과 같은 사용자를 대상으로 합니다.

- iSCSI/ASM을 통해 AWS 퍼블릭 클라우드에 Oracle을 구축하려는 DBA
- AWS 퍼블릭 클라우드에서 Oracle 워크로드를 테스트하려는 데이터베이스 솔루션 설계자
- AWS FSx 스토리지에 구축된 Oracle 데이터베이스를 구축하고 관리하려는 스토리지 관리자
- AWS FSx/EC2에서 Oracle 데이터베이스를 가동하려는 애플리케이션 소유자입니다.

솔루션 테스트 및 검증 환경

이 솔루션의 테스트 및 검증은 최종 구축 환경과 일치하지 않을 수 있는 AWS FSx 및 EC2 환경에서 수행되었습니다. 자세한 내용은 섹션을 참조하십시오 [\[Key Factors for Deployment Consideration\]](#).

있습니다



하드웨어 및 소프트웨어 구성 요소

* 하드웨어 *		
FSX ONTAP 저장소	AWS에서 제공하는 현재 버전입니다	동일한 VPC 및 가용성 존에 FSx HA 클러스터 1개

컴퓨팅용 EC2 인스턴스	T2.xLarge/4vCPU/16G	EC2 T2 xLarge EC2 인스턴스 2개, 하나는 운영 DB 서버로, 다른 하나는 클론 DB 서버로
* 소프트웨어 *		
RedHat Linux	RHEL-8.6.0_HVM-20220503- x86_64-2-Hourly2-GP2	테스트를 위해 RedHat 서브스크립션을 배포했습니다
Oracle Grid Infrastructure	버전 19.18	RU 패치 p34762026_190000_Linux-x86- 64.zip 를 적용했습니다
Oracle 데이터베이스	버전 19.18	RU 패치 p34765931_190000_Linux-x86- 64.zip 를 적용했습니다
Oracle OPatch	버전 12.2.0.1.36	최신 패치 p6880880_190000_Linux- x86-64.zip
SnapCenter 서비스	버전	v2.3.1.2324

구축 시 고려해야 할 주요 요소

- * EC2 컴퓨팅 인스턴스 * 이러한 테스트 및 검증에서는 Oracle 데이터베이스 컴퓨팅 인스턴스에 AWS EC2 T2.xLarge 인스턴스 유형을 사용했습니다. 운영 구축 환경에서는 데이터베이스 워크로드에 최적화된 M5 유형 EC2 인스턴스를 Oracle의 컴퓨팅 인스턴스로 사용하는 것이 좋습니다. 실제 워크로드 요구 사항에 따라 vCPU 수와 RAM 양에 따라 EC2 인스턴스를 적절하게 사이징해야 합니다.
- * FSx 스토리지 HA 클러스터 단일 또는 다중 영역 배포. * 이러한 테스트 및 검증에서는 단일 AWS 가용성 영역에 FSx HA 클러스터를 구축했습니다. 프로덕션 배포를 위해 FSx HA 쌍을 두 가지 가용성 영역에 배포하는 것이 좋습니다. FSx HA 클러스터는 스토리지 레벨 이중화를 제공하기 위해 액티브-패시브 파일 시스템 쌍으로 미러링되는 HA 쌍으로 프로비저닝됩니다. 다중 영역 구축을 통해 단일 AWS 영역에서 장애가 발생할 경우 고가용성을 더욱 강화할 수 있습니다.
- * FSx 스토리지 클러스터 크기 조정 * ONTAP 스토리지 파일 시스템용 Amazon FSx는 최대 160,000개의 원시 SSD IOPS, 최대 4Gbps 처리량 및 최대 192TiB 용량을 제공합니다. 그러나 구현 시 실제 요구 사항에 따라 프로비저닝된 IOPS, 처리량 및 스토리지 제한(최소 1,024GiB)을 기준으로 클러스터 크기를 조정할 수 있습니다. 애플리케이션 가용성에 영향을 주지 않고 용량을 즉각적으로 동적으로 조정할 수 있습니다.
- * Oracle 데이터 및 로그 레이아웃 * 테스트 및 검증에서 각각 데이터 및 로그용 ASM 디스크 그룹 2개를 구축했습니다. DATA ASM 디스크 그룹 내에서 데이터 볼륨에 4개의 LUN을 프로비저닝했습니다. logs ASM 디스크 그룹 내에서 로그 볼륨에 2개의 LUN을 프로비저닝했습니다. 일반적으로 ONTAP 볼륨용 Amazon FSx 내에 여러 개의 LUN을 배치하면 성능이 향상됩니다.
- * iSCSI 구성. * EC2 인스턴스 데이터베이스 서버는 iSCSI 프로토콜을 사용하여 FSx 스토리지에 연결합니다. EC2 인스턴스는 일반적으로 단일 네트워크 인터페이스 또는 ENI로 구축합니다. 단일 NIC 인터페이스는 iSCSI 및 애플리케이션 트래픽을 모두 전달합니다. 애플리케이션 및 iSCSI 트래픽-처리량 요구 사항을 모두 충족하는 올바른 EC2 컴퓨팅 인스턴스를 선택하려면 Oracle AWR 보고서를 신중하게 분석하여 Oracle 데이터베이스의 최대 I/O 처리량 요구 사항을 측정하는 것이 중요합니다. 또한 다중 경로가 올바르게 구성된 두 FSx iSCSI 엔드포인트 모두에 4개의 iSCSI 연결을 할당하는 것이 좋습니다.
- * 사용자가 생성한 각 Oracle ASM 디스크 그룹에 사용할 Oracle ASM 이중화 레벨 * FSx는 이미 FSx 클러스터 레벨에서 스토리지를 미러링하므로 외부 이중화를 사용해야 합니다. 이는 옵션이 Oracle ASM이 디스크 그룹의 내용을 미러링하지 못하도록 허용하지 않는다는 것을 의미합니다.
- * 데이터베이스 백업 * NetApp은 클라우드에서 데이터베이스 백업, 복원, 복제를 위한 SnapCenter 소프트웨어 서비스의 SaaS 버전을 제공하며 NetApp BlueXP 콘솔 UI를 통해 이용할 수 있습니다. 신속한(1분 이내) 스냅샷

백업, 빠른(몇 분) 데이터베이스 복원, 데이터베이스 복제를 위해 이러한 서비스를 구현하는 것이 좋습니다.

솔루션 구축

다음 섹션에서는 단계별 배포 절차를 제공합니다.

배포를 위한 사전 요구 사항

배포에는 다음과 같은 사전 요구 사항이 필요합니다.

1. AWS 계정이 설정되었으며 AWS 계정 내에 필요한 VPC 및 네트워크 세그먼트가 생성되었습니다.
2. AWS EC2 콘솔에서 2개의 EC2 Linux 인스턴스를 구축해야 합니다. 하나는 운영 Oracle DB 서버로, 다른 하나는 선택적 대체 클론 타겟 DB 서버입니다. 환경 설정에 대한 자세한 내용은 이전 섹션의 아키텍처 다이어그램을 참조하십시오. 또한 를 검토합니다 "[Linux 인스턴스에 대한 사용자 가이드](#)" 를 참조하십시오.
3. AWS EC2 콘솔에서 ONTAP 스토리지 HA 클러스터용 Amazon FSx를 구축하여 Oracle 데이터베이스 볼륨을 호스팅합니다. FSx 저장소 배포에 익숙하지 않은 경우 설명서를 참조하십시오 "[ONTAP 파일 시스템용 FSx 생성](#)" 을 참조하십시오.
4. 2단계와 3단계는 라는 EC2 인스턴스를 만드는 다음 Terraform 자동화 툴킷을 사용하여 수행할 수 있습니다 ora_01 이라는 FSx 파일 시스템이 있습니다 fsx_01. 실행 전에 지침을 주의 깊게 검토하고 환경에 맞게 변수를 변경하십시오.

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



Oracle 설치 파일을 스테이징할 충분한 공간을 확보하기 위해 EC2 인스턴스 루트 볼륨에 50G 이상을 할당했는지 확인합니다.

EC2 인스턴스 커널 구성

사전 요구 사항이 용량 할당된 상태에서 EC2 인스턴스에 EC2-USER 및 sudo로 로그인하여 Oracle 설치를 위한 Linux 커널을 구성합니다.

1. 스테이징 디렉터리를 만듭니다 /tmp/archive 폴더를 지정하고 를 설정합니다 777 권한.

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

2. Oracle 바이너리 설치 파일 및 기타 필요한 rpm 파일을 에 다운로드하고 스테이징합니다 /tmp/archive 디렉토리.

에 명시된 설치 파일의 다음 목록을 참조하십시오 /tmp/archive EC2 인스턴스에서 선택합니다.

```
[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. 대부분의 커널 구성 요구 사항을 충족하는 Oracle 19c 사전 설치 RPM을 설치합니다.

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

4. 누락된 을 다운로드하고 설치합니다 compat-libcap1 Linux 8에서

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

5. NetApp에서 NetApp 호스트 유틸리티를 다운로드하고 설치합니다.

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

6. 설치합니다 `policycoreutils-python-utils` 이는 EC2 인스턴스에서 사용할 수 없습니다.

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

7. 열려 있는 JDK 버전 1.8을 설치합니다.

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

8. iSCSI 초기자 유틸리티를 설치합니다.

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

9. 설치합니다 sg3_utils.

```
yum install sg3_utils
```

10. 설치합니다 device-mapper-multipath.

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

11. 현재 시스템에서 투명 HugePages를 비활성화합니다.

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

에 다음 행을 추가합니다 /etc/rc.local 를 눌러 비활성화합니다 transparent_hugepage 재부팅 후:

```
# 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
```

- SELinux를 변경하여 해제합니다 SELINUX=enforcing 를 선택합니다 SELINUX=disabled. 변경 사항을 적용하려면 호스트를 재부팅해야 합니다.

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

- 에 다음 행을 추가합니다 limit.conf 따옴표 없이 파일 설명자 제한과 스택 크기를 설정하려면 다음을 수행합니다 " " .

```
vi /etc/security/limits.conf
**                hard    nofile           65536"
**                soft    stack           10240"
```

- 다음 명령을 실행하여 EC2 인스턴스에 스왑 공간을 추가합니다. "스왑 파일을 사용하여 Amazon EC2 인스턴스에서 스왑 공간으로 사용할 메모리를 어떻게 할당합니까?" 정확한 추가 공간은 최대 16G RAM의 크기에 따라 달라집니다.
- 변경 node.session.timeo.replacement_timeout 에 있습니다 iscsi.conf 120 ~ 5초 사이의 구성 파일.

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

- EC2 인스턴스에서 iSCSI 서비스를 설정 및 시작합니다.

```
systemctl enable iscsid
systemctl start iscsid
```

- 데이터베이스 LUN 매핑에 사용할 iSCSI 이니시에이터 주소를 검색합니다.

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

- ASM sysasm 그룹에 사용할 ASM 그룹을 추가합니다.

```
groupadd asm
```

19. Oracle 사용자를 수정하여 ASM을 보조 그룹으로 추가합니다(Oracle 사용자는 Oracle 사전 설치 RPM 설치 후 생성되어야 함).

```
usermod -a -G asm oracle
```

20. Linux 방화벽이 활성화된 경우 중지하고 비활성화합니다.

```
systemctl stop firewalld  
systemctl disable firewalld
```

21. EC2 인스턴스를 재부팅합니다.

데이터베이스 볼륨 및 **LUN**을 **EC2** 인스턴스 호스트에 프로비저닝 및 매핑합니다

FSx 클러스터 관리 IP를 사용하여 ssh를 통해 FSx 클러스터에 로그인하여 Oracle 데이터베이스 바이너리, 데이터 및 로그 파일을 호스팅하여 명령줄에서 세 개의 볼륨을 프로비저닝합니다.

1. SSH를 통해 FSx 클러스터에 fsxadmin 사용자로 로그인합니다.

```
ssh fsxadmin@172.30.15.53
```

2. 다음 명령을 실행하여 Oracle 바이너리에 대한 볼륨을 생성합니다.

```
vol create -volume ora_01_biny -aggregate aggr1 -size 50G -state  
online -type RW -snapshot-policy none -tiering-policy snapshot-only
```

3. 다음 명령을 실행하여 Oracle 데이터용 볼륨을 생성합니다.

```
vol create -volume ora_01_data -aggregate aggr1 -size 100G -state  
online -type RW -snapshot-policy none -tiering-policy snapshot-only
```

4. 다음 명령을 실행하여 Oracle 로그용 볼륨을 생성합니다.

```
vol create -volume ora_01_logs -aggregate aggr1 -size 100G -state  
online -type RW -snapshot-policy none -tiering-policy snapshot-only
```

5. 데이터베이스 바이너리 볼륨 내에 바이너리 LUN을 생성합니다.

```
lun create -path /vol/ora_01_biny/ora_01_biny_01 -size 40G -ostype  
linux
```

6. 데이터베이스 데이터 볼륨 내에 데이터 LUN을 생성합니다.

```
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. 데이터베이스 로그 볼륨 내에 로그 LUN을 생성합니다.

```
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. 위의 EC2 커널 구성의 14단계에서 검색된 이니시에이터를 사용하여 EC2 인스턴스에 대한 igroup을 생성합니다.

```
igroup create -igroup ora_01 -protocol iscsi -ostype linux  
-initiator iqn.1994-05.com.redhat:f65fed7641c2
```

9. LUN을 위에서 생성한 igroup에 매핑합니다. 볼륨 내의 각 추가 LUN에 대해 LUN ID를 순차적으로 증분합니다.

```
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. LUN 매핑을 확인합니다.

```
mapping show
```

이 문제는 다음 항목을 반환해야 합니다.

```
FsxId02ad7bf3476b741df::> mapping show
```

```
(lun mapping show)
```

Vserver Protocol	Path	Igroup	LUN ID
svm_ora iscsi	/vol/ora_01_biny/ora_01_biny_01	ora_01	0
svm_ora iscsi	/vol/ora_01_data/ora_01_data_01	ora_01	1
svm_ora iscsi	/vol/ora_01_data/ora_01_data_02	ora_01	2
svm_ora iscsi	/vol/ora_01_data/ora_01_data_03	ora_01	3
svm_ora iscsi	/vol/ora_01_data/ora_01_data_04	ora_01	4
svm_ora iscsi	/vol/ora_01_logs/ora_01_logs_01	ora_01	5
svm_ora iscsi	/vol/ora_01_logs/ora_01_logs_02	ora_01	6

데이터베이스 스토리지 구성

이제 EC2 인스턴스 호스트에서 Oracle 그리드 인프라 및 데이터베이스 설치를 위한 FSx 스토리지를 가져와서 설정합니다.

1. SSH 키와 EC2 인스턴스 IP 주소를 사용하여 SSH를 통해 EC2 사용자로 EC2 인스턴스에 로그인합니다.

```
ssh -i ora_01.pem ec2-user@172.30.15.58
```

2. SVM iSCSI IP 주소를 사용하여 FSx iSCSI 엔드포인트를 검색합니다. 그런 다음 환경별 포털 주소로 변경합니다.

```
sudo iscsiadm iscsiadm --mode discovery --op update --type  
sendtargets --portal 172.30.15.51
```

3. 각 타겟에 로그인하여 iSCSI 세션을 설정합니다.

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

명령의 예상 출력은 다음과 같습니다.

```
[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. 활성 iSCSI 세션 목록을 보고 확인합니다.

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

iSCSI 세션을 반환합니다.

```
[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. LUN을 호스트로 가져왔는지 확인합니다.

```
sudo sanlun lun show
```

그러면 FSx의 Oracle LUN 목록이 반환됩니다.

```

[ec2-user@ip-172-30-15-58 ~]$ sudo sanlun lun show
controller(7mode/E-Series)/                               device
host                lun
vservers(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. 를 구성합니다 multipath.conf 다음 기본 항목과 블랙리스트 항목이 있는 파일입니다.

```

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. 다중 경로 서비스를 시작합니다.

```
sudo systemctl start multipathd
```

이제 다중 경로 장치가 에 나타납니다 /dev/mapper 디렉토리.

```

[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. SSH를 통해 FSx 클러스터에 fsxadmin 사용자로 로그인하여 각 LUN의 일련 번호 16진수 번호를 검색합니다. 6c574xxx...로 시작합니다. 16진수 번호는 AWS 공급업체 ID인 3600a0980으로 시작합니다.

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

그리고 다음과 같이 돌아옵니다.

```
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. 를 업데이트합니다 /dev/multipath.conf 파일 - 다중 경로 장치에 대한 사용자 친화적인 이름을 추가합니다.

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

다음 항목이 있는 경우:

```

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. 다중 경로 서비스를 재부팅하여 에서 장치가 에 있는지 확인합니다 /dev/mapper LUN 이름 대 직렬 16진수 ID로 변경되었습니다.

```
sudo systemctl restart multipathd
```

확인합니다 /dev/mapper 다음과 같이 돌아가려면:

```
[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. 단일 주 파티션으로 바이너리 LUN을 파티셔닝합니다.

```
sudo fdisk /dev/mapper/ora_01_biny_01
```

12. XFS 파일 시스템을 사용하여 분할된 바이너리 LUN을 포맷합니다.

```
sudo mkfs.xfs /dev/mapper/ora_01_biny_01p1
```

13. 에 바이너리 LUN을 마운트합니다 /u01.

```
sudo mount -t xfs /dev/mapper/ora_01_biny_01p1 /u01
```

14. 변경 /u01 마운트 지점 소유권을 Oracle 사용자에게 부여하며, 기본 그룹이 사용됩니다.

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

15. 바이너리 LUN의 UUID를 찾습니다.

```
sudo blkid /dev/mapper/ora_01_biny_01p1
```

16. 에 마운트 지점을 추가합니다 /etc/fstab.

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

다음 줄을 추가합니다.

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



EC2 인스턴스 재부팅 중에 발생할 수 있는 루트 잠금 문제를 방지하려면 UUID와 NOFAIL 옵션을 사용하여 바이너리를 마운트하는 것이 중요합니다.

17. 루트 사용자로 Oracle 디바이스에 대한 udev 규칙을 추가합니다.

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

다음 항목 포함:

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

18. 루트 사용자로 udev 규칙을 다시 로드합니다.

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

19. 루트 사용자로 udev 규칙을 트리거합니다.

```
udevadm trigger
```

20. 루트 사용자로 multipathd를 다시 로드합니다.

```
systemctl restart multipathd
```

21. EC2 인스턴스 호스트를 재부팅합니다.

Oracle 그리드 인프라 설치

1. SSH를 통해 EC2 인스턴스로 로그인하고 주석 처리를 해제하여 암호 인증을 활성화합니다
PasswordAuthentication yes 그런 다음 의견을 남기기 바랍니다 PasswordAuthentication no.

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

2. sshd 서비스를 다시 시작합니다.

```
sudo systemctl restart sshd
```

3. Oracle 사용자 암호를 재설정합니다.

```
sudo passwd oracle
```

4. Oracle Restart 소프트웨어 소유자 사용자(Oracle)로 로그인합니다. 다음과 같이 Oracle 디렉토리를 생성합니다.

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

5. 디렉터리 권한 설정을 변경합니다.

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

6. 그리드 홈 디렉토리를 만들고 변경합니다.

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

7. 그리드 설치 파일의 압축을 풉니다.

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

8. 그리드 홈에서 을 삭제합니다 OPatch 디렉토리.

```
rm -rf OPatch
```

9. 그리드 홈에서 압축을 풉니다 p6880880_190000_Linux-x86-64.zip.

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

10. 그리드 홈에서 수정합니다 cv/admin/cvu_config, 주석 취소 및 바꾸기 CV_ASSUME_DISTID=OEL5 와 함께 CV_ASSUME_DISTID=OL7.

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

11. 를 준비합니다 gridsetup.rsp 자동 설치용 파일 및 RSP 파일을 에 배치합니다 /tmp/archive 디렉토리. RSP 파일은 다음 정보를 사용하여 섹션 A, B 및 G를 포함해야 합니다.

```
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/mapper/ora_01_data_02,/dev/mapper/ora_01_data_03,/dev/mapper/ora_01_data_04
oracle.install.asm.diskGroup.diskDiscoveryString=/dev/mapper/*
oracle.install.asm.monitorPassword="SetPWD"
oracle.install.asm.configureAFD=true
```

12. EC2 인스턴스에 루트 사용자로 로그인하여 설정합니다 ORACLE_HOME 및 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. Oracle ASM 필터 드라이버와 함께 사용할 디스크 디바이스를 프로비저닝합니다.

```
./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. 설치합니다 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. 설정 해제 \$ORACLE_BASE.

```
unset ORACLE_BASE
```

16. EC2 인스턴스에 Oracle 사용자로 로그인하여 에서 패치를 추출합니다 /tmp/archive 폴더.

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

17. 그리드 홈 /u01/app/oracle/product/19.0.0/GRID에서 Oracle 사용자로 를 실행합니다 gridSetup.sh 그리드 인프라 설치용.

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

그리드 인프라에 대해 잘못된 그룹에 대한 경고를 무시합니다. 단일 Oracle 사용자를 사용하여 Oracle Restart를 관리하고 있으므로 이 작업이 필요합니다.

18. 루트 사용자로 다음 스크립트를 실행합니다.

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

19. 루트 사용자로 multipathd를 다시 로드합니다.

```
systemctl restart multipathd
```

20. Oracle 사용자는 다음 명령을 실행하여 구성을 완료합니다.

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

21. Oracle 사용자로 로그 디스크 그룹을 생성합니다.

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

22. Oracle 사용자는 설치 구성 후 그리드 서비스를 검증합니다.

```
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 필터 드라이버 상태입니다.

```
[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
ASMCMDS> 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/
ASMCMDS> afd_state
ASMCMDS-9526: The AFD state is 'LOADED' and filtering is 'ENABLED' on
host 'ip-172-30-15-58.ec2.internal'
```

Oracle 데이터베이스 설치

1. Oracle 사용자로 로그인하고 설정을 해제합니다 \$ORACLE_HOME 및 \$ORACLE_SID 설정되어 있는지 확인합니다.

```
unset ORACLE_HOME
unset ORACLE_SID
```

2. Oracle DB 홈 디렉토리를 생성하고 변경합니다.

```
mkdir /u01/app/oracle/product/19.0.0/db1
cd /u01/app/oracle/product/19.0.0/db1
```

3. Oracle DB 설치 파일의 압축을 풉니다.

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

4. DB 홈에서 OPatch 디렉토리를 삭제합니다.

```
rm -rf OPatch
```

5. DB HOME에서 압축을 풉니다 p6880880_190000_Linux-x86-64.zip.

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

6. DB 홈에서 수정한다 cv/admin/cvu_config`을 클릭합니다 `CV_ASSUME_DISTID=OEL5 와 함께 CV_ASSUME_DISTID=OL7.

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

7. 에서 /tmp/archive 디렉토리에서 DB 19.18 RU 패치의 포장을 풉니다.

```
unzip p34765931_190000_Linux-x86-64.zip
```

8. 에서 DB 자동 설치 RSP 파일을 준비합니다 /tmp/archive/dbinstall.rsp 다음 값이 있는 디렉터리:

```
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. db1 home/u01/app/oracle/product/19.0.0/db1에서 자동 소프트웨어 전용 DB 설치를 실행합니다.

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

10. 루트 사용자로 를 실행합니다 root.sh 소프트웨어 전용 설치 후 스크립트.

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

11. Oracle 사용자로 을 생성합니다 dbca.rsp 다음 항목이 있는 파일:

```
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. Oracle 사용자로 dbca를 사용하여 DB 생성을 시작합니다.


```
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. Oracle 사용자로서 DB 생성 후 Oracle Restart HA 서비스를 확인합니다.

```
[oracle@ip-172-30-15-58 db1]$ ../grid/bin/crsctl stat res -t
```

Name	Target	State	Server	State
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.dbf.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. Oracle 사용자를 설정합니다 .bash_profile.

```
vi ~/.bash_profile
```

15. 다음 항목 추가:

```
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. 생성된 CDB/PDB를 검증합니다.

```
/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.265.1132177009
```

```
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/sysaux.266.1132177009
```

```
+DATA/DB1/DATAFILE/users.259.1132176247
```

```
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/undotbs1.267.1132177009
```

```
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/system.271.1132177853
```

```
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/sysaux.272.1132177853
```

```
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/undotbs1.270.1132177853
```

```
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/users.274.1132177871
```

```
NAME
```

```
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/system.276.1132177871
```

```
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/sysaux.277.1132177871
```

```
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/undotbs1.275.1132177871
```

```
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/users.279.1132177889
```

```
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/system.281.1132177889
```

```
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/sysaux.282.1132177889
```

```
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/undotbs1.280.1132177889
```

```
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/users.284.1132177907
```

```
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. DB 복구 대상 크기를 +logs 디스크 그룹 크기로 설정합니다.

```
alter system set db_recovery_file_dest_size = 80G scope=both;
```

18. sqlplus를 사용하여 데이터베이스에 로그인하고 아카이브 로그 모드를 설정합니다.

```
sqlplus /as sysdba.  
  
shutdown immediate;  
  
startup mount;  
  
alter database archivelog;  
  
alter database open;
```

이것으로 ONTAP 및 EC2 컴퓨팅 인스턴스의 Amazon FSx에서 Oracle 19c 버전 19.18 재시작 구축이 완료되었습니다. 필요한 경우 Oracle 제어 파일 및 온라인 로그 파일을 + 로그 디스크 그룹으로 재배치하는 것이 좋습니다.

자동화된 구축 옵션

을 참조하십시오 ["TR-4986: iSCSI를 지원하는 Amazon FSx ONTAP에서 간편하고 자동화된 Oracle 배포"](#) 를 참조하십시오.

SnapCenter 서비스를 사용한 Oracle 데이터베이스 백업, 복원 및 클론 복제

을 참조하십시오 ["Oracle용 SnapCenter 서비스"](#) Oracle 데이터베이스 백업, 복원 및 NetApp BlueXP 콘솔을 사용한 클론에 대한 자세한 내용

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 문서 및/또는 웹 사이트를 참조하십시오.

- 새 데이터베이스 설치를 통해 독립 실행형 서버용 Oracle Grid Infrastructure 설치

["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"](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)

- 응답 파일을 사용하여 Oracle 데이터베이스 설치 및 구성

["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"](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)

- NetApp ONTAP용 Amazon FSx

["https://aws.amazon.com/fsx/netapp-ontap/"](https://aws.amazon.com/fsx/netapp-ontap/)

- Amazon EC2

https://aws.amazon.com/pm/ec2/?trk=36c6da98-7b20-48fa-8225-4784bced9843&sc_channel=ps&s_kwcid=AL14422!3!467723097970!e!!g!!aws%20ec2&ef_id=Cj0KCQiA54KfBhCKARIsAJzSrdqwQrghn6l71ji!WzSeaT9Uh1-vY-VfhJixF-xnv5rWwn2S7RqZOTQ0aAh7eEALw_wcB:G:s&s_kwcid=AL14422!3!467723097970!e!!g!!aws%20ec2

AWS EC2 및 FSx Best Practices에 Oracle Database 구축

WP-7357: EC2 및 FSx Best Practices Introduction에서 Oracle Database Deployment를 참조하십시오

Allen Cao, Niyaz Mohamed, Jeffrey Steiner, NetApp

많은 미션 크리티컬 엔터프라이즈 Oracle 데이터베이스는 여전히 사내에서 호스팅되며, 많은 기업은 이러한 Oracle 데이터베이스를 퍼블릭 클라우드로 마이그레이션하려고 합니다. 종종 이러한 Oracle 데이터베이스는 애플리케이션 중심이므로 많은 서비스형 데이터베이스 퍼블릭 클라우드 오퍼링에서 누락되는 사용자별 구성이 필요합니다. 따라서 현재의 데이터베이스 환경에서는 고유한 요구 사항을 수용할 수 있는 고성능의 확장 가능한 컴퓨팅 및 스토리지 서비스를 통해 구축된 퍼블릭 클라우드 기반 Oracle 데이터베이스 솔루션이 필요합니다. AWS EC2 컴퓨팅 인스턴스 및 AWS FSx 스토리지 서비스는 미션 크리티컬 Oracle 데이터베이스 워크로드를 퍼블릭 클라우드로 구축 및 마이그레이션하는 데 활용할 수 있는 이 퍼즐의 누락된 조각일 수 있습니다.

Amazon EC2(Amazon Elastic Compute Cloud)는 클라우드에서 안전하고 크기 조절이 가능한 컴퓨팅 용량을 제공하는 웹 서비스입니다. 이 솔루션은 기업이 웹 기반 클라우드 컴퓨팅을 보다 쉽게 사용할 수 있도록 설계되었습니다. 간단한 Amazon EC2 웹 서비스 인터페이스를 사용하면 최소한의 마찰로 용량을 확보하고 구성할 수 있습니다. 컴퓨팅 리소스를 완벽하게 제어하고 Amazon의 검증된 컴퓨팅 환경에서 실행할 수 있습니다.

ONTAP용 Amazon FSx는 업계 최고의 NetApp ONTAP 블록 및 파일 스토리지를 사용하는 AWS 스토리지 서비스로, NFS, SMB 및 iSCSI를 제공합니다. 이처럼 강력한 스토리지 엔진을 사용하여 1밀리초 미만의 응답 시간, 수 Gbps의

처리량, 데이터베이스 인스턴스당 100,000 이상의 IOPS를 제공하는 미션 크리티컬 Oracle 데이터베이스 애플리케이션을 AWS로 재배치하는 것이 그 어느 때보다 쉬워졌습니다. 또한 FSx 스토리지 서비스에는 기본 복제 기능이 포함되어 있어 온프레미스 Oracle 데이터베이스를 AWS로 쉽게 마이그레이션하거나 미션 크리티컬 Oracle 데이터베이스를 HA 또는 DR용 보조 AWS 가용성 영역으로 복제할 수 있습니다.

이 문서의 목표는 FSx 스토리지와 내부 시스템과 유사한 성능을 제공하는 EC2 인스턴스를 사용하여 Oracle 데이터베이스를 구축하고 구성하는 방법에 대한 단계별 프로세스, 절차 및 모범 사례 지침을 제공하는 것입니다. 또한 NetApp은 AWS 퍼블릭 클라우드에서 Oracle 데이터베이스 워크로드를 구현, 구성, 관리하는 데 필요한 대부분의 작업을 자동화하는 자동화 툴킷을 제공합니다.

솔루션 및 사용 사례에 대해 자세히 알아보려면 다음 개요 비디오 를 참조하십시오.

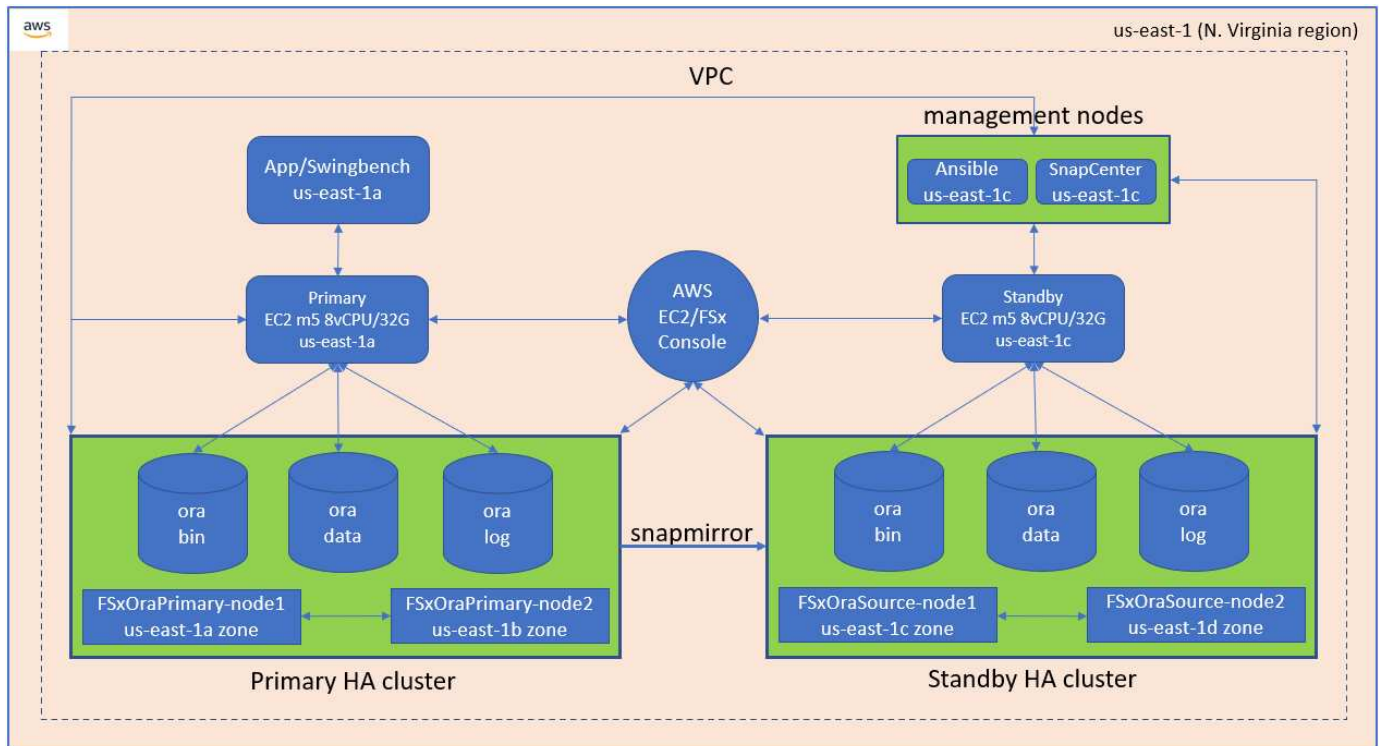
["AWS 및 FSx ONTAP, Part1 - 활용 사례 및 솔루션 아키텍처의 하이브리드 클라우드로 Oracle 데이터베이스를 현대화하십시오"](#)

솔루션 아키텍처

다음 아키텍처 다이어그램은 FSx 스토리지 서비스가 있는 AWS EC2 인스턴스에 고가용성 Oracle 데이터베이스 구축을 보여 줍니다. 유사한 배포 체계이지만 다른 지역의 대기 구성을 재해 복구에 설정할 수 있습니다.

환경 내에서 Oracle 컴퓨팅 인스턴스는 AWS EC2 인스턴스 콘솔을 통해 구축됩니다. 콘솔에서 여러 EC2 인스턴스 유형을 사용할 수 있습니다. RedHat Enterprise Linux 8과 함께 M5 Ami 이미지와 같은 데이터베이스 기반 EC2 인스턴스 유형 및 최대 10Gbps의 네트워크 대역폭을 구축하는 것이 좋습니다.

반면에 FSx 볼륨의 Oracle 데이터베이스 스토리지는 AWS FSx 콘솔 또는 CLI와 함께 구축됩니다. 이후에 Oracle 바이너리, 데이터 또는 로그 볼륨을 제공하고 EC2 인스턴스 Linux 호스트에 마운트합니다. 사용된 기본 스토리지 프로토콜에 따라 각 데이터 또는 로그 볼륨에 여러 개의 LUN이 할당될 수 있습니다.



FSx 스토리지 클러스터는 기본 스토리지 클러스터와 대기 스토리지 클러스터가 모두 두 개의 서로 다른 가용성 영역에

구축되도록 이중 중복성으로 설계되었습니다. 데이터베이스 볼륨은 모든 Oracle 바이너리, 데이터 및 로그 볼륨에 대해 사용자가 구성할 수 있는 간격으로 기본 FSx 클러스터에서 대기 FSx 클러스터로 복제됩니다.

이 고가용성 Oracle 환경은 Ansible 컨트롤러 노드와 SnapCenter 백업 서버 및 UI 툴을 사용하여 관리됩니다. Ansible 플레이북 기반 툴 키트를 사용하여 Oracle 설치, 구성, 복제를 자동화할 수 있습니다. Oracle EC2 인스턴스 커널 운영 체제 또는 Oracle 패칭에 대한 모든 업데이트를 병렬로 실행하여 운영 및 대기 상태를 동기화할 수 있습니다. 실제로 초기 자동화 설정을 쉽게 확장하여 필요한 경우 일상적인 Oracle 작업을 몇 가지 반복 수행할 수 있습니다.

SnapCenter는 Oracle 데이터베이스 시점 복구 또는 필요한 경우 운영 또는 대기 영역에서 데이터베이스 클론 복제를 위한 워크플로우를 제공합니다. SnapCenter UI를 통해 Oracle 데이터베이스 백업 및 복제를 스탠바이 FSx 스토리지에 구성하여 RTO 또는 RPO 목표에 따라 고가용성 또는 재해 복구를 수행할 수 있습니다.

이 솔루션은 Oracle RAC 및 Data Guard 구축에서 사용할 수 있는 기능과 유사한 기능을 제공하는 대체 프로세스를 제공합니다.

Oracle 데이터베이스 구축에 고려해야 할 요인

퍼블릭 클라우드에는 다양한 컴퓨팅 및 스토리지 옵션을 제공하므로 정확한 유형의 컴퓨팅 인스턴스 및 스토리지 엔진을 사용하여 데이터베이스를 구축할 수 있습니다. 또한 Oracle 데이터베이스에 최적화된 컴퓨팅 및 스토리지 구성을 선택해야 합니다.

다음 섹션에서는 FSx 스토리지가 있는 EC2 인스턴스의 AWS 퍼블릭 클라우드에서 Oracle 데이터베이스를 구축할 때의 주요 고려 사항에 대해 설명합니다.

VM 성능

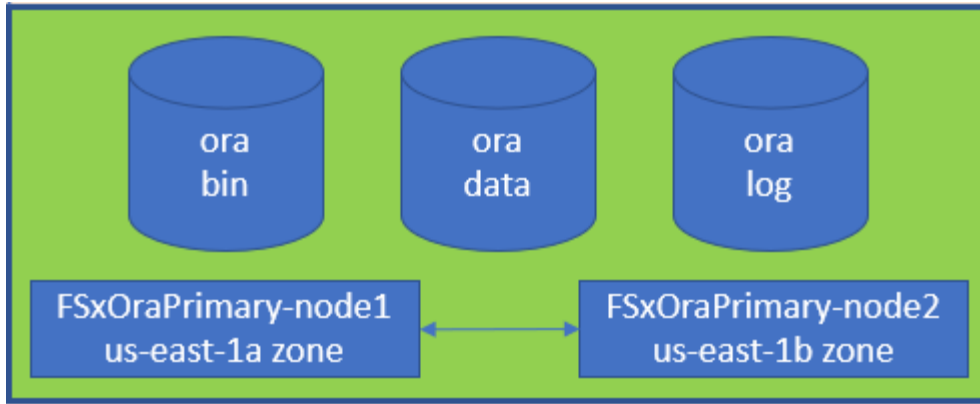
공용 클라우드에서 관계형 데이터베이스의 성능을 최적화하려면 올바른 VM 크기를 선택하는 것이 중요합니다. 더 나은 성능을 위해 데이터베이스 워크로드에 최적화된 EC2 M5 Series 인스턴스를 Oracle 구축에 사용하는 것이 좋습니다. AWS에서 Oracle용 RDS 인스턴스를 실행하는 데에도 같은 인스턴스 유형이 사용됩니다.

- 워크로드 특성에 따라 올바른 vCPU 및 RAM 조합을 선택합니다.
- VM에 스왑 공간을 추가합니다. 기본 EC2 인스턴스 구축은 데이터베이스에 적합하지 않은 스왑 공간을 생성하지 않습니다.

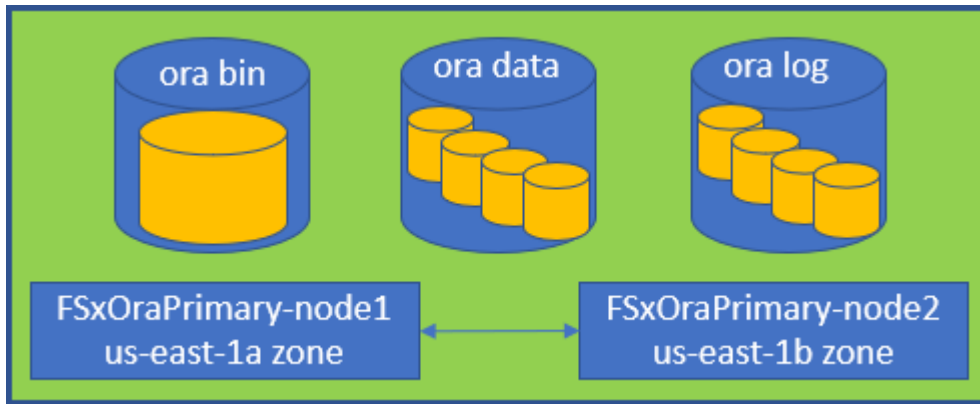
스토리지 레이아웃 및 설정

권장되는 스토리지 레이아웃은 다음과 같습니다.

- NFS 스토리지의 경우 권장되는 볼륨 레이아웃은 세 개의 볼륨입니다. 하나는 Oracle 바이너리용이고, 다른 하나는 Oracle 데이터 및 중복 제어 파일용이며, 다른 하나는 Oracle 액티브 로그, 아카이브 로그 및 제어 파일용입니다.



- iSCSI 스토리지의 경우 권장되는 볼륨 레이아웃은 Oracle 바이너리용 볼륨 1개, Oracle 데이터 및 중복 제어 파일용 볼륨 1개, Oracle 액티브 로그, 아카이브 로그 및 제어 파일용 볼륨 3개입니다. 그러나 각 데이터 및 로그 볼륨에는 4개의 LUN이 포함되어 있어야 합니다. LUN은 HA 클러스터 노드에서 균형을 유지하는 것이 이상적입니다.



- 스토리지 IOPS 및 처리량의 경우 FSx 스토리지 클러스터의 프로비저닝된 IOPS 및 처리량에 대한 임계값을 선택할 수 있으며, 워크로드가 변경될 때마다 이러한 매개 변수를 즉석에서 조정할 수 있습니다.
 - 자동 IOPS 설정은 할당된 스토리지 용량 또는 최대 80,000개의 사용자 정의 스토리지 용량(GiB)당 3IOPS입니다.
 - 처리량 수준은 128, 256, 512, 1024, 2045Mbps와 같이 증가합니다.

를 검토합니다 "[NetApp ONTAP 성능을 위한 Amazon FSx](#)" 처리량 및 IOPS 사이징 관련 문서

NFS 구성

가장 일반적인 운영 체제인 Linux에는 네이티브 NFS 기능이 포함되어 있습니다. Oracle은 Oracle에 기본적으로 통합된 직접 NFS(dNFS) 클라이언트를 제공합니다. Oracle은 20년 이상 NFSv3을 지원해 왔습니다. dNFS는 모든 버전의 Oracle에서 NFSv3에서 지원됩니다. NFSv4 표준을 따르는 모든 OS에서 NFSv4가 지원됩니다. NFSv4를 지원하는 dNFS를 지원하려면 Oracle 12.1.0.2 이상이 필요합니다. NFSv4.1은 특정 OS 지원이 필요합니다. 지원되는 OS에 대해서는 NetApp 상호 운용성 매트릭스 툴(IMT)을 참조하십시오. NFSv4.1에 대한 dNFS 지원을 위해서는 Oracle 버전 19.3.0.0 이상이 필요합니다.

NetApp 자동화 툴킷을 사용하여 자동화된 Oracle 구축은 NFSv3에서 dNFS를 자동으로 구성합니다.

기타 고려 사항:

- TCP 슬롯 테이블은 호스트 버스 어댑터(HBA) 큐 길이(queue depth)와 동등한 NFS의 기능입니다. 이들 테이블은

한 번에 수행될 수 있는 최대 NFS 작업의 수를 제어합니다. 기본값은 보통 16이며 최적의 성능을 발휘하기에 너무 낮습니다. 최신 Linux 커널에서는 반대의 문제가 발생하는데, 요청을 통해 NFS 서버를 포화시키는 수준까지 TCP 슬롯 테이블의 한계를 자동으로 높일 수 있습니다.

최적의 성능을 제공하고 성능 문제를 방지하려면 TCP 슬롯 테이블을 제어하는 커널 매개 변수를 128로 조정합니다.

```
sysctl -a | grep tcp.*.slot_table
```

- 다음 표에는 Linux NFSv3 - 단일 인스턴스에 대한 권장 NFS 마운트 옵션이 나와 있습니다.

File Type	Mount Options
<ul style="list-style-type: none"> • Control files • Data files • Redo logs 	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsiz=65536
<ul style="list-style-type: none"> • ORACLE_HOME • ORACLE_BASE 	rw,bg,hard,vers=3,proto=tcp,timeo=600,rsize=65536,wsiz=65536



dNFS를 사용하기 전에 Oracle Doc 1495104.1에 설명된 패치가 설치되어 있는지 확인하십시오. NFSv3 및 NFSv4에 대한 NetApp Support Matrix에는 특정 운영 체제가 포함되어 있지 않습니다. RFC를 따르는 모든 OS가 지원됩니다. 온라인 IMT에서 NFSv3 또는 NFSv4 지원을 검색할 때 일치하는 항목이 표시되지 않으므로 특정 OS를 선택하지 마십시오. 모든 OS는 일반 정책에 의해 암시적으로 지원됩니다.

고가용성

솔루션 아키텍처에 설명된 대로 HA는 스토리지 레벨 복제를 기반으로 합니다. 따라서 Oracle의 시작 및 가용성은 컴퓨팅 및 스토리지를 얼마나 빠르게 가동 및 복구할 수 있는지에 따라 다릅니다. 다음 주요 요소를 참조하십시오.

- 대기 컴퓨팅 인스턴스를 준비하고 두 호스트에 대한 Ansible 병렬 업데이트를 통해 기본 컴퓨팅 인스턴스와 동기화하십시오.
- 맨 마지막에 Oracle을 설치할 필요가 없도록 기본 볼륨에서 바이너리 볼륨을 복제하고 설치 및 패치 작업을 수행해야 하는 항목을 파악할 수 있습니다.
- 복제 빈도는 서비스를 사용할 수 있도록 Oracle 데이터베이스를 복구하는 속도를 나타냅니다. 복제 빈도와 스토리지 소비는 서로 상충됩니다.
- 자동화를 활용하여 복구를 수행하고 인적 오류 없이 신속하게 대기 상태로 전환합니다. NetApp은 이러한 목적을 위한 자동화 툴킷을 제공합니다.

AWS EC2 및 FSx에서 Oracle Deployment Procedures를 단계별로 수행합니다

이 섹션에서는 FSx 스토리지와 함께 Oracle RDS 사용자 지정 데이터베이스를 구축하는 구축 절차를 설명합니다.

EC2 콘솔을 통해 Oracle용 EC2 Linux 인스턴스를 구축합니다

AWS를 처음 사용하는 경우, 먼저 AWS 환경을 설정해야 합니다. AWS 웹 사이트 랜딩 페이지의 문서 탭에는 AWS

EC2 콘솔을 통해 Oracle 데이터베이스를 호스팅하는 데 사용할 수 있는 Linux EC2 인스턴스를 구축하는 방법에 대한 EC2 지침 링크가 제공됩니다. 다음 섹션은 이러한 단계를 요약한 것입니다. 자세한 내용은 연결된 AWS EC2 관련 문서를 참조하십시오.

AWS EC2 환경 설정

EC2 및 FSx 서비스에서 Oracle 환경을 실행하는 데 필요한 리소스를 프로비저닝하려면 AWS 계정을 만들어야 합니다. 필요한 세부 정보는 다음 AWS 문서를 참조하십시오.

- ["Amazon EC2를 사용하도록 설정합니다"](#)

주요 주제:

- AWS에 등록하십시오.
- 키 쌍을 생성합니다.
- 보안 그룹을 만듭니다.

AWS 계정 특성에 여러 가용성 영역 설정

아키텍처 다이어그램에 표시된 Oracle 고가용성 구성의 경우 한 지역에서 최소 4개의 가용성 영역을 활성화해야 합니다. 여러 가용성 영역을 여러 지역에 배치하여 재해 복구에 필요한 거리를 충족할 수도 있습니다.

The screenshot displays the AWS Management Console interface. The main content area shows 'Resources' for the US East (N. Virginia) Region, listing various EC2-related items such as instances, key pairs, security groups, and volumes. A 'Launch instance' button is visible. On the right, the 'Account attributes' panel is open, showing supported platforms, default VPC, and settings. Below that, the 'Explore AWS' section provides links to various AWS services and cost optimization tips. The 'Zones' table is also visible, listing available availability zones in the region.

Zone name	Zone ID
us-east-1a	use1-az6
us-east-1b	use1-az3
us-east-1c	use1-az2
us-east-1d	use1-az4

Oracle 데이터베이스를 호스팅하기 위한 EC2 인스턴스 생성 및 연결

자습서를 참조하십시오 ["Amazon EC2 Linux 인스턴스를 시작합니다"](#) 을 참조하십시오.

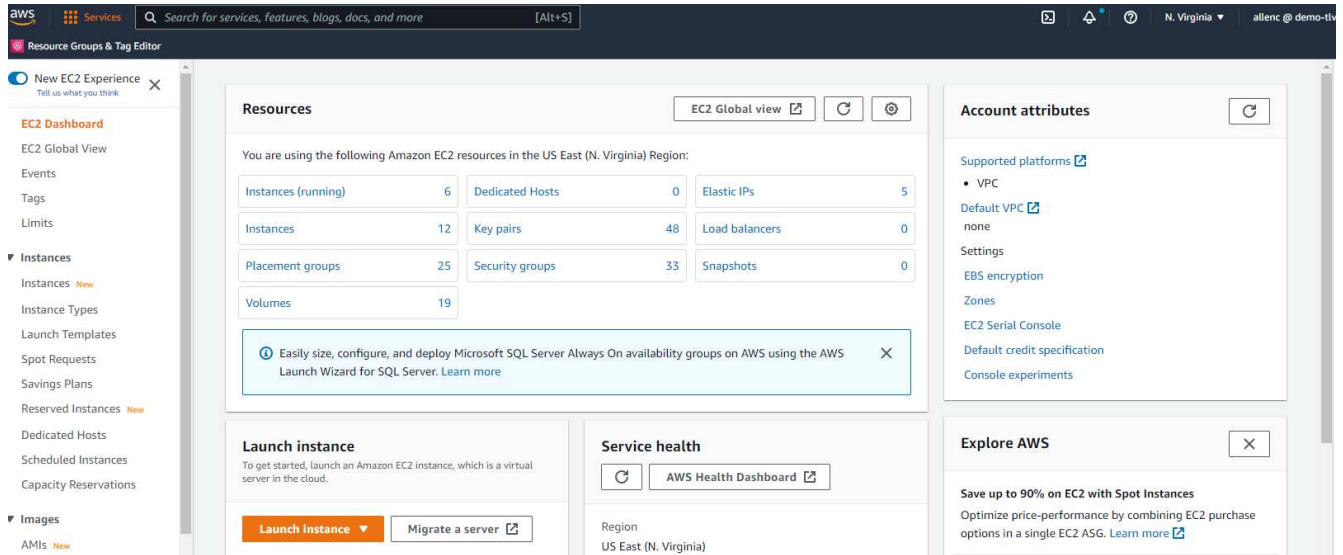
주요 주제:

- 개요.

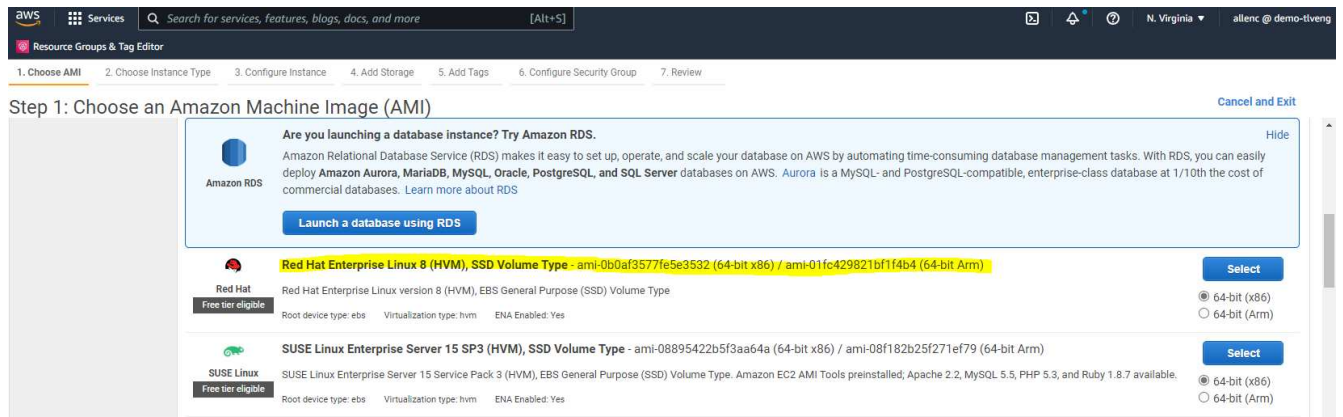
- 필수 구성 요소.
- 1단계: 인스턴스를 시작합니다.
- 2단계: 인스턴스에 연결합니다.
- 3단계: 인스턴스를 정리하십시오.

다음 스크린샷에서는 Oracle 실행을 위해 EC2 콘솔을 사용하여 M5 형식의 Linux 인스턴스를 구축하는 방법을 보여줍니다.

1. EC2 대시보드에서 노란색 Launch Instance 버튼을 클릭하여 EC2 인스턴스 구축 워크플로우를 시작합니다.



2. 1단계에서 "Red Hat Enterprise Linux 8(HVM), SSD 볼륨 유형 - AMI-0b0af3577fe5e3532(64비트 x86)/AMI-01fc429821bf1f4b4(64비트 ARM)"를 선택합니다.



3. 2단계에서 Oracle 데이터베이스 작업 부하에 따라 적절한 CPU 및 메모리 할당이 있는 M5 인스턴스 유형을 선택합니다. "다음: 인스턴스 세부 정보 구성"을 클릭합니다.

	Instance type	Instance size class	Number of vCPUs	Memory (GiB)	Storage	Network bandwidth (Gbps)	Private IP addresses	Private IPv6 addresses
<input type="checkbox"/>	m4	m4.16xlarge	64	256	EBS only	Yes	25 Gigabit	Yes
<input type="checkbox"/>	m5	m5.large	2	8	EBS only	Yes	Up to 10 Gigabit	Yes
<input type="checkbox"/>	m5	m5.xlarge	4	16	EBS only	Yes	Up to 10 Gigabit	Yes
<input checked="" type="checkbox"/>	m5	m5.2xlarge	8	32	EBS only	Yes	Up to 10 Gigabit	Yes
<input type="checkbox"/>	m5	m5.4xlarge	16	64	EBS only	Yes	Up to 10 Gigabit	Yes
<input type="checkbox"/>	m5	m5.8xlarge	32	128	EBS only	Yes	10 Gigabit	Yes
<input type="checkbox"/>	m5	m5.12xlarge	48	192	EBS only	Yes	10 Gigabit	Yes
<input type="checkbox"/>	m5	m5.16xlarge	64	256	EBS only	Yes	20 Gigabit	Yes
<input type="checkbox"/>	m5	m5.24xlarge	96	384	EBS only	Yes	25 Gigabit	Yes
<input type="checkbox"/>	m5	m5.metal	96	384	EBS only	Yes	25 Gigabit	Yes

4. 3단계에서 인스턴스를 배치할 VPC 및 서브넷을 선택하고 공용 IP 할당을 활성화합니다. "다음: 스토리지 추가"를 클릭합니다.

Step 3: Configure Instance Details

No default VPC found. Select another VPC, or create a new default VPC.

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1 Launch into Auto Scaling Group

Purchasing option: Request Spot instances

Network: vpc-0474064fc537e5182 Create new VPC
No default VPC found. Create a new default VPC.

Subnet: subnet-08c952541f4ab282d | us-east-1a Create new subnet
250 IP Addresses available

Auto-assign Public IP: Enable

Hostname type: Use subnet setting (IP name)

DNS Hostname: Enable IP name IPv4 (A record) DNS requests
 Enable resource-based IPv4 (A record) DNS requests
 Enable resource-based IPv6 (AAAA record) DNS requests

Placement group: Add instance to placement group

Capacity Reservation: Open

Domain join directory: No directory Create new directory

IAM role: None Create new IAM role

Cancel Previous **Review and Launch** Next: Add Storage

5. 4단계에서 루트 디스크에 충분한 공간을 할당합니다. 스왑을 추가하려면 공간이 필요할 수 있습니다. 기본적으로 EC2 인스턴스는 0 스왑 공간을 할당하며, 이 공간은 Oracle 실행에 최적화되어 있지 않습니다.

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/sda1	snap-03a3ad00558b4d17c	50	General Purpose SSD (gp2)	150 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

▼ **Shared file systems**

You currently don't have any file systems on this instance. Select "Add file system" button below to add a file system.

[Add file system](#)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)

6. 5단계에서 필요한 경우 인스턴스 식별용 태그를 추가합니다.

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. A copy of a tag can be applied to volumes, instances or both. Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key (128 characters maximum)	Value (256 characters maximum)	Instances	Volumes	Network Interfaces
<p><i>This resource currently has no tags</i></p> <p>Choose the Add tag button or click to add a Name tag. Make sure your IAM policy includes permissions to create tags.</p>				

[Add Tag](#) (Up to 50 tags maximum)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Security Group](#)

7. 6단계에서 기존 보안 그룹을 선택하거나 인스턴스에 대해 원하는 인바운드 및 아웃바운드 정책을 사용하여 새 보안 그룹을 생성합니다.

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group Select an existing security group

Security Group ID	Name	Description	Actions
<input type="checkbox"/> sg-0d746a0908b897c48	AviOcm03112021OCCM1635951256631-OCCMSecurityGroup-B9QFHUJRUUV	NetApp OCCM Instance External Security Group	Copy to new
<input type="checkbox"/> sg-07b0625cd544aee16	AviOcm0311OCCM1635943382952-OCCMSecurityGroup-1L8D4QX2SC945	NetApp OCCM Instance External Security Group	Copy to new
<input type="checkbox"/> sg-0618122caefc50e9	AviOcm1103OCCM1635944222133-OCCMSecurityGroup-DX5PHX6CKVKC	NetApp OCCM Instance External Security Group	Copy to new
<input type="checkbox"/> sg-0d53ea8c78987e660	AviOcm1209OCCM1631452667252-OCCMSecurityGroup-TSKVZ1Q4SH48	NetApp OCCM Instance External Security Group	Copy to new
<input type="checkbox"/> sg-0aed9f8836b48c52d	AviOcmFSXOCCM1638110371156-OCCMSecurityGroup-N0ENZJW3TVYB	NetApp OCCM Instance External Security Group	Copy to new
<input type="checkbox"/> sg-083a6ea5cba912375	connector1OCCM1631455604110-OCCMSecurityGroup-1790QV45PH3ZW	NetApp OCCM Instance External Security Group	Copy to new
<input checked="" type="checkbox"/> sg-08148ca915189ac87	default	default VPC security group	Copy to new
<input type="checkbox"/> sg-07f6c527620e3bb22	fsx02OCCM1633339531669-OCCMSecurityGroup-1XZYC5WM15NP7	NetApp OCCM Instance External Security Group	Copy to new
<input type="checkbox"/> sg-0f359d2ba38db749f	SG-Version10-0CE6MEs-NetAppExternalSecurityGroup-N8B50KGTk5U	ONTAP Cloud firewall rules for management and data interface	Copy to new

Inbound rules for sg-08148ca915189ac87 (Selected security groups: sg-08148ca915189ac87)

Type	Protocol	Port Range	Source	Description
All traffic	All	All	192.168.1.0/24	
All traffic	All	All	sg-08148ca915189ac87 (default)	

[Cancel](#) [Previous](#) [Review and Launch](#)

8. 7단계에서 인스턴스 구성 요약을 검토하고 시작을 클릭하여 인스턴스 배포를 시작합니다. 인스턴스에 액세스하기 위해 키 쌍을 생성하거나 키 쌍을 선택하라는 메시지가 표시됩니다.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details [Edit AMI](#)

Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-0b0af3577fe5e3532
 Free tier eligible Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type
 Root Device Type: ebs Virtualization type: hvm

Instance Type [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
m5.2xlarge	-	8	32	EBS only	Yes	Up to 10 Gigabit

Security Groups [Edit security groups](#)

Security Group ID	Name	Description
sg-08148ca915189ac87	default	default VPC security group

All selected security groups inbound rules

Type	Protocol	Port Range	Source	Description
All traffic	All	All	192.168.1.0/24	
All traffic	All	All	sg-08148ca915189ac87 (default)	

Instance Details [Edit instance details](#)

Storage [Edit storage](#)

[Cancel](#) [Previous](#) [Launch](#)

Select an existing key pair or create a new key pair ✕

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance. Amazon EC2 supports ED25519 and RSA key pair types.

Note: The selected key pair will be added to the set of keys authorized for this instance. [Learn more about removing existing key pairs from a public AMI.](#)

Select a key pair

I acknowledge that I have access to the corresponding private key file, and that without this file, I won't be able to log into my instance.

Cancel
Launch Instances

9. SSH 키 쌍을 사용하여 EC2 인스턴스에 로그인합니다. 필요에 따라 키 이름과 인스턴스 IP 주소를 변경합니다.

```
ssh -i ora-dblv2.pem ec2-user@54.80.114.77
```

아키텍처 다이어그램에 표시된 대로 지정된 가용성 영역에 2개의 EC2 인스턴스를 운영 및 대기 Oracle 서버로 만들어야 합니다.

Oracle 데이터베이스 스토리지용 **ONTAP** 파일 시스템용 **FSx**를 프로비저닝합니다

EC2 인스턴스 구축은 운영 체제에 EBS 루트 볼륨을 할당합니다. ONTAP 파일 시스템용 FSX는 Oracle 바이너리, 데이터 및 로그 볼륨을 비롯한 Oracle 데이터베이스 스토리지 볼륨을 제공합니다. FSx 스토리지 NFS 볼륨은 AWS FSx 콘솔 또는 Oracle 설치에서 프로비저닝될 수 있으며 사용자가 자동화 매개 변수 파일에서 구성할 때 볼륨을 할당하는 구성 자동화도 가능합니다.

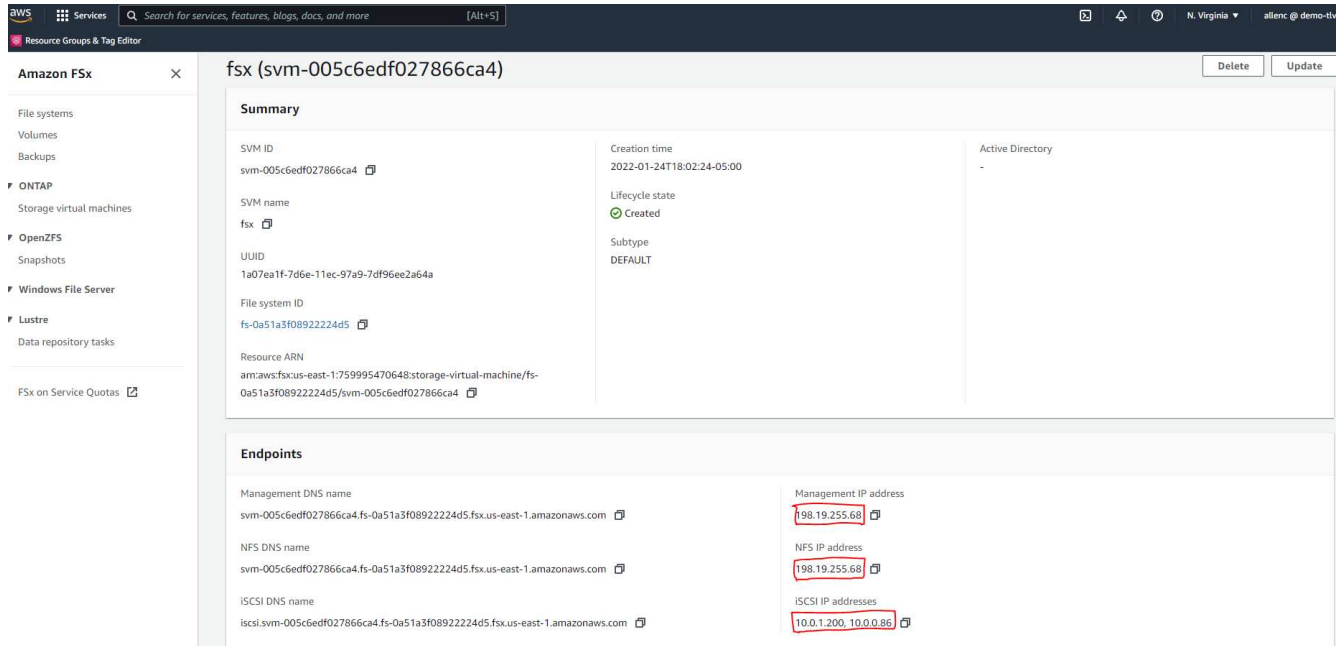
ONTAP 파일 시스템용 FSx 생성

이 문서를 참조했습니다 "[ONTAP 파일 시스템용 FSx 관리](#)" ONTAP 파일 시스템용 FSx를 생성하는 데 사용됩니다.

주요 고려 사항:

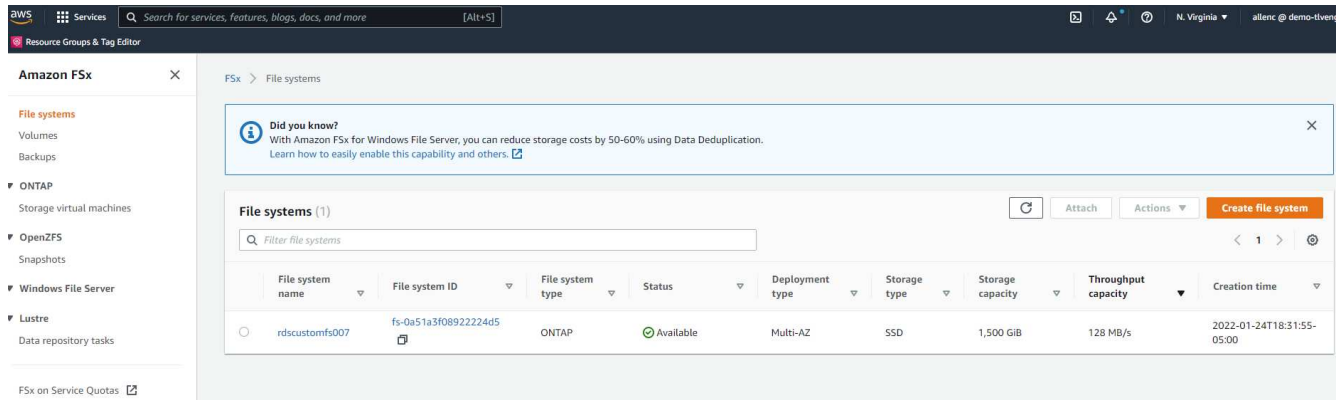
- SSD 스토리지 용량: 최소 1024GiB, 최대 192TiB
- 프로비저닝된 SSD IOPS입니다. 워크로드 요구사항에 따라 파일 시스템당 최대 80,000 SSD IOPS
- 처리량 용량:
- 관리자 fsxadmin/vsadmin 암호를 설정합니다. FSx 구성 자동화에 필요합니다.

- 백업 및 유지 관리. 자동 일일 백업을 사용하지 않도록 설정합니다. 데이터베이스 스토리지 백업은 SnapCenter 일정을 통해 실행됩니다.
- SVM 세부 정보 페이지에서 SVM 관리 IP 주소와 프로토콜별 액세스 주소를 검색합니다. FSx 구성 자동화에 필요합니다.

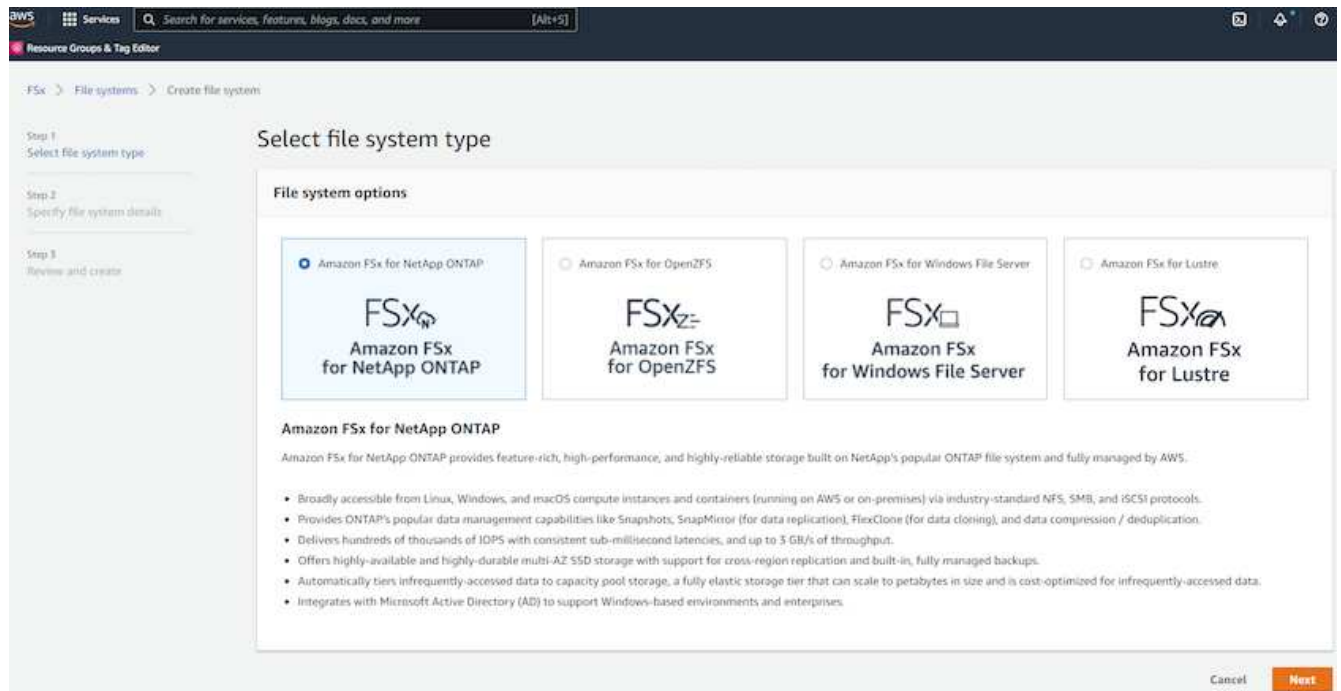


기본 또는 대기 HA FSx 클러스터를 설정하려면 다음 단계별 절차를 참조하십시오.

1. FSx 콘솔에서 파일 시스템 생성 을 클릭하여 FSx 프로비저닝 워크플로우를 시작합니다.



2. NetApp ONTAP용 Amazon FSx 를 선택합니다. 다음 을 클릭합니다.



3. Standard Create를 선택하고 File System Details에서 파일 시스템의 이름을 Multi-AZ HA로 지정합니다. 데이터베이스 워크로드에 따라 자동 또는 사용자 프로비저닝 IOPS 최대 80,000 SSD IOPS를 선택합니다. FSX 스토리지는 백엔드에서 최대 2TiB NVMe 캐싱과 함께 제공되므로 더욱 높은 측정 IOPS를 제공할 수 있습니다.

File system details

File system name - optional [Info](#)

Maximum of 256 Unicode letters, whitespace, and numbers, plus + - = . _ : /

Deployment type [Info](#)

Multi-AZ

Single-AZ

SSD storage capacity [Info](#)

Minimum 1024 GiB; Maximum 192 TiB.

Provisioned SSD IOPS

Amazon FSx provides 3 IOPS per GiB of storage capacity. You can also provision additional SSD IOPS as needed.

Automatic (3 IOPS per GiB of SSD storage)

User-provisioned

Maximum 80,000 IOPS

Throughput capacity [Info](#)

The sustained speed at which the file server hosting your file system can serve data. The file server can also burst to higher speeds for periods of time.

Recommended throughput capacity

128 MB/s

Specify throughput capacity

Throughput capacity

4. 네트워크 및 보안 섹션에서 VPC, 보안 그룹 및 서브넷을 선택합니다. FSx 배포 전에 만들어야 합니다. FSx 클러스터(기본 또는 대기)의 역할에 따라 FSx 스토리지 노드를 적절한 영역에 배치합니다.

Network & security

Virtual Private Cloud (VPC) [Info](#)

Specify the VPC from which your file system is accessible.

vpc-0474064fc537e5182 ▼

VPC Security Groups [Info](#)

Specify VPC Security Groups to associate with your file system's network interfaces.

Choose VPC security group(s) ▼

sg-08148ca915189ac87 (default) ✕

Preferred subnet [Info](#)

Specify the preferred subnet for your file system.

subnet-08c952541f4ab282d (us-east-1a) ▼

Standby subnet

subnet-0a84d6eeeb0f4e5c0 (us-east-1b) ▼

VPC route tables

Specify the VPC route tables associated with your file system.

VPC's default route table

Select one or more VPC route tables

Endpoint IP address range

Specify the IP address range in which the endpoints to access your file system will be created

No preference

Select an IP address range

5. 보안 및 암호화 섹션에서 기본값을 적용하고 fsxadmin 암호를 입력합니다.

Security & encryption

Encryption key [Info](#)
 AWS Key Management Service (KMS) encryption key that protects your file system data at rest.

aws/fsx (default) ▼

Description	Account	KMS key ID
Default master key that protects my FSx resources when no other key is defined	759995470648	5b31feff-6759-4306-a852-9c99a743982a

File system administrative password
 Password for this file system's "fsxadmin" user, which you can use to access the ONTAP CLI or REST API.

Don't specify a password
 Specify a password

Password

Confirm password

6. SVM 이름과 vsadmin 암호를 입력합니다.

Default storage virtual machine configuration

Storage virtual machine name

SVM administrative password
 Password for this SVM's "vsadmin" user, which you can use to access the ONTAP CLI or REST API.

Don't specify a password
 Specify a password

Password

Confirm password

Active Directory
 Joining an Active Directory enables access from Windows and MacOS clients over the SMB protocol.

Do not join an Active Directory
 Join an Active Directory

7. 볼륨 구성은 비워 둡니다. 이 시점에서는 볼륨을 생성할 필요가 없습니다.

Default volume configuration

Volume name

Maximum of 203 alphanumeric characters, plus _.

Junction path

The location within your file system where your volume will be mounted.

Volume size

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.

▶ Backup and maintenance - optional

▶ Tags - optional

Cancel Back Next

8. Summary 페이지를 검토하고 Create File System을 클릭하여 FSx 파일 시스템 프로비저닝을 완료합니다.

aws Services Search for services, features, blogs, docs, and more [Alt+S]

Resource Groups & Tag Editor

Step 1 Select file system type

Step 2 Specify file system details

Step 3 Review and create

Create file system

Summary
Verify the following attributes before proceeding

Attribute	Value	Editable after creation
File system type	Amazon FSx for NetApp ONTAP	
File system name	aws_ora_prod	✔
Deployment type	Multi-AZ	
Storage type	SSD	
SSD storage capacity	1,024 GiB	✔
Minimum SSD IOPS	40000 IOPS	✔
Throughput capacity	512 MB/s	✔
Virtual Private Cloud (VPC)	vpc-0474064fc537e5182	
VPC Security Groups	sg-08148ca915189ac87	✔
Preferred subnet	subnet-08c952541f4ab282d	
Standby subnet	subnet-0a84d6eeeb0f4e5c0	
VPC route tables	VPC's default route table	
Endpoint IP address range	No preference	
KMS key ID	arn:aws:kms:us-east-1:759995470648:key/5b31feff-6759-4306-a852-9c99a743982a	
Daily automatic backup window	No preference	✔
Automatic backup	7 day(s)	✔

Oracle 데이터베이스용 데이터베이스 볼륨 프로비저닝

을 참조하십시오 ["ONTAP 볼륨용 FSx 관리 - 볼륨 생성"](#) 를 참조하십시오.

주요 고려 사항:

- 데이터베이스 볼륨의 크기를 적절하게 조정합니다.
- 성능 구성을 위해 용량 풀 계층화 정책을 사용하지 않도록 설정합니다.
- NFS 스토리지 볼륨에 대해 Oracle dNFS를 사용하도록 설정합니다.
- iSCSI 스토리지 볼륨에 대한 다중 경로 설정

FSx 콘솔에서 데이터베이스 볼륨을 생성합니다

AWS FSx 콘솔에서 Oracle 데이터베이스 파일 스토리지용 볼륨 3개를 생성할 수 있습니다. 하나는 Oracle 바이너리용이고, 다른 하나는 Oracle 데이터용이고, 다른 하나는 Oracle 로그용입니다. 볼륨 이름이 올바른 식별을 위해 Oracle 호스트 이름(자동화 툴킷의 hosts 파일에 정의되어 있음)과 일치하는지 확인하십시오. 이 예에서는 EC2 인스턴스의 일반적인 IP 주소 기반 호스트 이름 대신 db1을 EC2 Oracle 호스트 이름으로 사용합니다.

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

Create volume
✕

File system

ONTAP | fs-0a51a3f08922224d5 | rdscustomfs007 ▼

Storage virtual machine

svm-005c6edf027866ca4 | fsx ▼

Volume name

db1_log

Maximum of 203 alphanumeric characters, plus _.

Junction path

/db1_log

The location within your file system where your volume will be mounted.

Volume size

256000

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



iSCSI LUN 생성은 현재 FSx 콘솔에서 지원되지 않습니다. Oracle용 iSCSI LUN 구축의 경우 NetApp 자동화 툴킷을 통해 ONTAP용 자동화 를 사용하여 볼륨 및 LUN을 생성할 수 있습니다.

FSx 데이터베이스 볼륨이 있는 **EC2** 인스턴스에 **Oracle**을 설치 및 구성합니다

NetApp 자동화 팀은 모범 사례에 따라 EC2 인스턴스에서 Oracle 설치 및 구성을 실행할 수 있는 자동화 키트를 제공합니다. 현재 버전의 자동화 키트는 기본 RU 패치 19.8을 사용하여 NFS에서 Oracle 19c를 지원합니다. 필요한 경우 자동화 키트를 다른 RU 패치에 쉽게 적용할 수 있습니다.

자동화를 실행하도록 **Ansible** 컨트롤러를 준비합니다

다음 섹션의 지침을 따르십시오. "[Oracle 데이터베이스를 호스팅하기 위한 EC2 인스턴스 생성 및 연결](#)" Ansible 컨트롤러를 실행할 작은 EC2 Linux 인스턴스를 프로비저닝합니다. RedHat을 사용하는 대신 2vCPU 및 8G RAM을 사용하는 Amazon Linux T2.Large로 충분합니다.

NetApp Oracle 구현 자동화 툴킷을 살펴보십시오

1단계에서 EC2-USER로 프로비저닝한 EC2 Ansible 컨트롤러 인스턴스에 로그인하고 EC2-user 홈 디렉토리에서 "git clone" 명령을 실행하여 자동화 코드 복사본을 복제합니다.

```
git clone https://github.com/NetApp-Automation/na_oracle19c_deploy.git
```

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

자동화 툴킷을 사용하여 자동화된 **Oracle 19c** 구축을 실행합니다

자세한 지침을 참조하십시오 "[CLI 구축 Oracle 19c Database](#)" CLI 자동화를 통해 Oracle 19c를 구축합니다. 호스트 액세스 인증에 암호 대신 SSH 키 쌍을 사용하고 있기 때문에 플레이북 실행을 위한 명령 구문이 약간 변경됩니다. 다음 목록은 요약 정보입니다.

1. 기본적으로 EC2 인스턴스는 액세스 인증을 위해 SSH 키 쌍을 사용합니다. Ansible 컨트롤러 자동화 루트 디렉토리 '/home/EC2-user/na_oracle19c_deploy' 및 '/home/EC2-user/na_RDS_FSX_oranfs_config'에서 단계에 구축된 Oracle 호스트에 대한 SSH 키 'accessstkey.pem'의 복사본을 만듭니다. "[Oracle 데이터베이스를 호스팅하기 위한 EC2 인스턴스 생성 및 연결](#) 있습니다."
2. EC2 인스턴스 DB 호스트에 EC2-USER로 로그인하여 python3 라이브러리를 설치합니다.

```
sudo yum install python3
```

3. 루트 디스크 드라이브에서 16G 스왑 공간을 만듭니다. 기본적으로 EC2 인스턴스는 0 스왑 공간을 만듭니다. 다음 AWS 설명서를 참조하십시오. "[스왑 파일을 사용하여 Amazon EC2 인스턴스에서 스왑 공간으로 사용할 메모리를 어떻게 할당합니까?](#)".
4. Ansible 컨트롤러('cd/home/EC2-user/na_RDS_FSX_oranfs_config')로 돌아가서 적절한 요구 사항과 'linux_config' 태그를 사용하여 사전 클론 플레이북을 실행합니다.

```
ansible-playbook -i hosts rds_preclone_config.yml -u ec2-user --private-key accessstkey.pem -e @vars/fsx_vars.yml -t requirements_config
```

```
ansible-playbook -i hosts rds_preclone_config.yml -u ec2-user --private-key accessstkey.pem -e @vars/fsx_vars.yml -t linux_config
```

5. '/home/EC2-user/na_oracle19c_deploy-master' 디렉토리로 전환하고 README 파일을 읽은 다음 글로벌

'vars.yml' 파일에 관련 글로벌 매개 변수를 채웁니다.

6. host_name.yml 파일을 host_vars 디렉토리에 관련 파라미터로 채웁니다.
7. Linux용 플레이북을 실행하고 vsadmin 암호를 묻는 메시지가 표시되면 Enter 키를 누릅니다.

```
ansible-playbook -i hosts all_playbook.yml -u ec2-user --private-key accesststkey.pem -t linux_config -e @vars/vars.yml
```

8. Oracle용 플레이북을 실행하고 vsadmin 암호를 묻는 메시지가 표시되면 Enter 키를 누릅니다.

```
ansible-playbook -i hosts all_playbook.yml -u ec2-user --private-key accesststkey.pem -t oracle_config -e @vars/vars.yml
```

필요한 경우 SSH 키 파일의 사용 권한 비트를 400으로 변경합니다. host_vars' 파일의 Oracle 호스트 ('abilities_host')를 EC2 인스턴스 공용 주소로 변경합니다.

기본 및 대기 **FSx HA** 클러스터 간에 **SnapMirror**를 설정합니다

고가용성 및 재해 복구를 위해 기본 및 대기 FSx 스토리지 클러스터 간에 SnapMirror 복제를 설정할 수 있습니다. 다른 클라우드 스토리지 서비스와 달리 FSx를 사용하면 원하는 빈도와 복제 처리량으로 스토리지 복제를 제어 및 관리할 수 있습니다. 또한 사용자는 가용성에 영향을 주지 않고 HA/DR을 테스트할 수 있습니다.

다음 단계에서는 운영 FSx 스토리지 클러스터와 대기 FSx 스토리지 클러스터 간에 복제를 설정하는 방법을 보여 줍니다.

1. 기본 및 대기 클러스터 피어링을 설정합니다. fsxadmin 사용자로 운영 클러스터에 로그인하고 다음 명령을 실행합니다. 이 상호 생성 프로세스는 운영 클러스터와 대기 클러스터 모두에서 create 명령을 실행합니다. 'standby_cluster_name'을 환경에 적합한 이름으로 바꿉니다.

```
cluster peer create -peer-addr  
standby_cluster_name,inter_cluster_ip_address -username fsxadmin  
-initial-allowed-vserver-peers *
```

2. 기본 클러스터와 대기 클러스터 간에 SVM 피어링을 설정합니다. vsadmin 사용자로 운영 클러스터에 로그인하고 다음 명령을 실행합니다. 기본_vserver_name, 'standby_vserver_name', 'tandby_cluster_name'을 환경에 적합한 이름으로 바꾸십시오.

```
vserver peer create -vserver primary_vserver_name -peer-vserver  
standby_vserver_name -peer-cluster standby_cluster_name -applications  
snapmirror
```

3. 클러스터 및 SVM 발길이 올바르게 설정되었는지 확인합니다.

```

FsxId00164454fac5591e6::> cluster peer show
Peer Cluster Name          Cluster Serial Number Availability Authentication
-----
FsxId0b6a95149d07aa82e    1-80-000011             Available         ok

FsxId00164454fac5591e6::> vserver peer show
Vserver Peer Peer Peer Peering Remote
Vserver Vserver State Peer Cluster Applications Vserver
-----
svm_FSxOraSource
svm_FSxOraTarget
peered FsxId0b6a95149d07aa82e
snapmirror svm_FSxOraTarget

FsxId00164454fac5591e6::>

```

4. 기본 FSx 클러스터의 각 소스 볼륨에 대해 스탠바이 FSx 클러스터에서 타겟 NFS 볼륨을 생성합니다. 환경에 맞게 볼륨 이름을 바꿉니다.

```

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. iSCSI 프로토콜을 데이터 액세스에 사용하는 경우 Oracle 바이너리, Oracle 데이터 및 Oracle 로그에 대한 iSCSI 볼륨 및 LUN을 생성할 수도 있습니다. 볼륨에 약 10%의 여유 공간을 남겨 둡니다.

```

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-XR-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. iSCSI LUN의 경우 바이너리 LUN을 예로 사용하여 각 LUN에 대한 Oracle 호스트 이니시에이터에 대한 매핑을 생성합니다. 사용자 환경에 적합한 이름으로 igroup을 교체하고 각 추가 LUN에 대해 LUN-ID를 늘립니다.

```
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. 기본 데이터베이스 볼륨과 대기 데이터베이스 볼륨 사이에 SnapMirror 관계를 생성합니다. 해당 환경에 적합한 SVM 이름을 교체합니다

```
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
```

이 SnapMirror 설정은 NFS 데이터베이스 볼륨용 NetApp 자동화 툴킷을 사용하여 자동화할 수 있습니다. 이 툴킷은 NetApp 공개 GitHub 사이트에서 다운로드할 수 있습니다.

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

설정 및 페일오버 테스트를 시도하기 전에 README 지침을 주의 깊게 읽으십시오.



Oracle 바이너리를 기본 클러스터에서 대기 클러스터로 복제하면 Oracle 라이선스가 영향을 받을 수 있습니다. 자세한 내용은 Oracle 라이선스 담당자에게 문의하십시오. 또는 복구 및 페일오버 시 Oracle을 설치 및 구성해야 합니다.

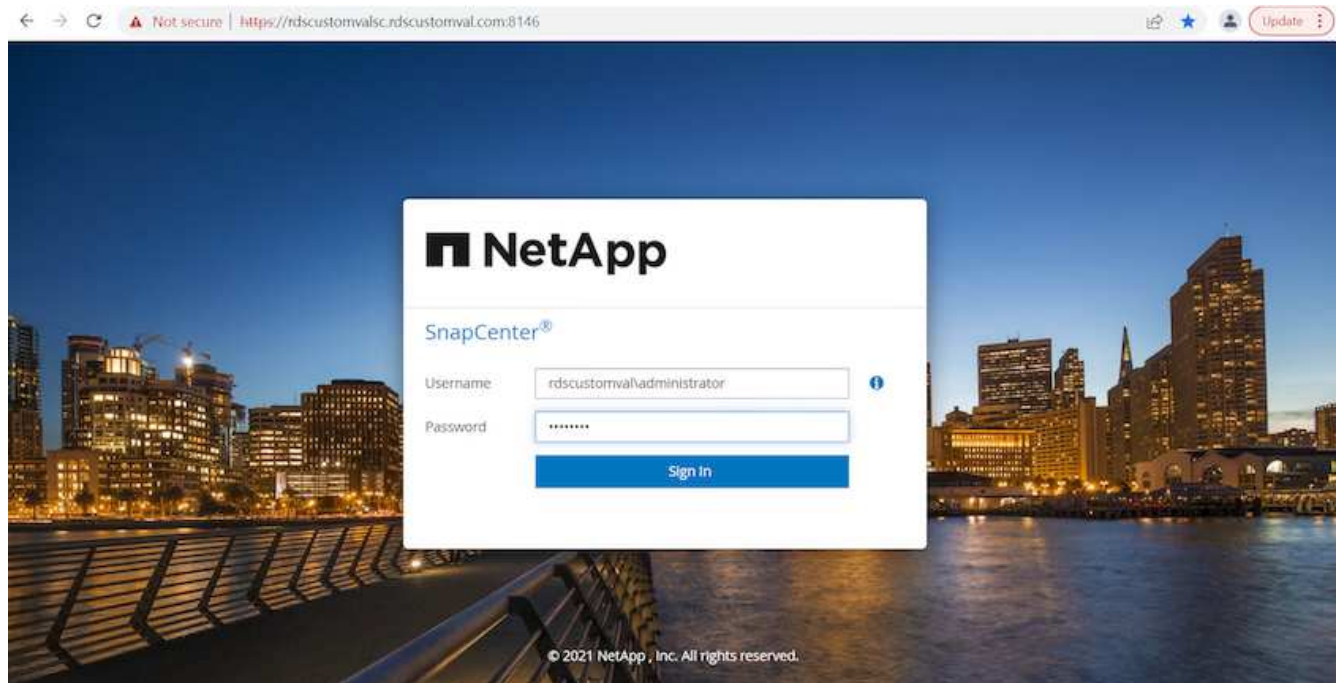
SnapCenter 배포

SnapCenter 설치

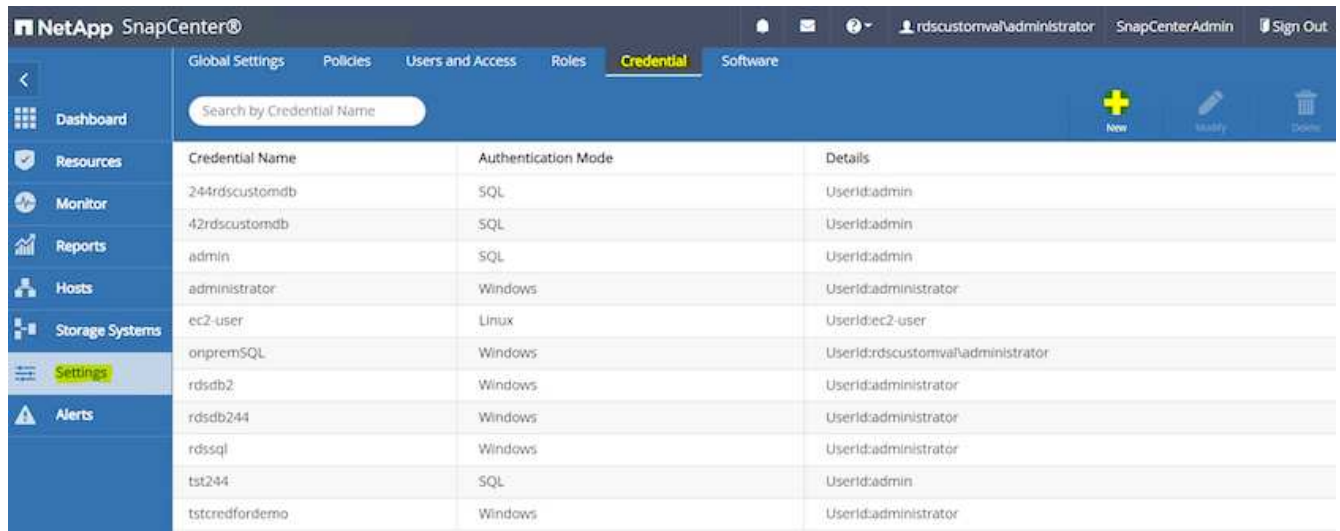
를 따릅니다 "[SnapCenter 서버 설치](#)" SnapCenter 서버를 설치합니다. 이 문서에서는 독립 실행형 SnapCenter 서버를 설치하는 방법에 대해 설명합니다. SnapCenter SaaS 버전은 베타 검토 중이며 곧 제공될 예정입니다. 필요한 경우 NetApp 담당자에게 문의하십시오.

EC2 Oracle 호스트용 SnapCenter 플러그인을 구성합니다

1. 자동화된 SnapCenter 설치 후 SnapCenter 서버가 설치된 Windows 호스트의 관리 사용자로 SnapCenter에 로그인합니다.



2. 왼쪽 메뉴에서 설정, 자격 증명 및 새로 만들기 를 차례로 클릭하여 SnapCenter 플러그인 설치를 위한 EC2 사용자 자격 증명을 추가합니다.



3. EC2 인스턴스 호스트에서 '/etc/ssh/sshd_config' 파일을 편집하여 EC2-user 암호를 재설정하고 암호 SSH 인증을 활성화합니다.
4. "Sudo 권한 사용" 확인란이 선택되어 있는지 확인합니다. 이전 단계에서 EC2-user 암호를 재설정했습니다.

Credential x

Credential Name

Authentication Mode ▼

Username i

Password

Use sudo privileges i

5. 이름 확인을 위해 SnapCenter 서버 이름과 IP 주소를 EC2 인스턴스 호스트 파일에 추가합니다.

```

[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. SnapCenter 서버 Windows 호스트에서 Windows 호스트 파일 'C:\Windows\System32\drivers\etc\hosts'에 EC2 인스턴스 호스트 IP 주소를 추가합니다.

```

10.0.0.151    ip-10-0-0-151.ec2.internal
```

7. 왼쪽 메뉴에서 호스트 > 관리 호스트 를 선택한 다음 추가 를 클릭하여 EC2 인스턴스 호스트를 SnapCenter에 추가합니다.

Name	Type	System	Plug-in	Version	Overall Status
RDSAMAZ-VJ0DQK0	Windows	Stand-alone	Microsoft Windows Server, Microsoft SQL Server	4.5	Host down
rdscustommssql1.rdscustomval.com	Windows	Stand-alone	Microsoft Windows Server, Microsoft SQL Server	4.5	Running

Oracle Database를 선택하고 제출하기 전에 기타 옵션을 클릭합니다.

Add Host

Host Type: Linux

Host Name: 10.0.0.151

Credentials: ec2-user

Select Plug-ins to Install SnapCenter Plug-ins Package 4.5 P2 for Linux

- Oracle Database
- SAP HANA

More Options: Port, Install Path, Custom Plug-Ins...

Submit Cancel

사전 설치 검사 건너뛰기 를 선택합니다. Preinstall Checks(사전 설치 검사)를 건너뛰는지 확인한 다음 Save(저장) 후 Submit(제출)을 클릭합니다.

More Options

Port: ⓘ

Installation Path: ⓘ

Skip preinstall checks

Custom Plug-ins

Choose a File

지문 확인 메시지가 표시되면 확인 및 제출 을 클릭합니다.

Confirm Fingerprint

Authenticity of the host cannot be determined ⓘ

Host name	Fingerprint	Valid
ip-10-0-0-151.ec2.internal	ssh-rsa 2048 97:6F:3C:7D:38:42:F6:54:B7:AF:E3:61:61:BA:2E:6F	

플러그인 구성이 성공적으로 완료되면 관리 호스트의 전체 상태가 실행 중 으로 표시됩니다.

Managed Hosts							
Search by Name		+		-		↺	
Name	Type	System	Plug-in	Version	Overall Status		
ip-10-0-0-151.ec2.internal	Linux	Stand-alone	UNIX, Oracle Database	4.5	Running		

Oracle 데이터베이스에 대한 백업 정책을 구성합니다

이 섹션을 참조하십시오 **"SnapCenter에서 데이터베이스 백업 정책을 설정합니다"** Oracle 데이터베이스 백업 정책 구성에 대한 자세한 내용은 를 참조하십시오.

일반적으로 전체 스냅샷 Oracle 데이터베이스 백업에 대한 정책과 Oracle 아카이브 로그 전용 스냅샷 백업에 대한 정책을 생성해야 합니다.



백업 정책에서 Oracle 아카이브 로그 잘라내기 기능을 활성화하여 로그 아카이브 공간을 제어할 수 있습니다. HA 또는 DR을 위해 대기 위치에 복제해야 하는 경우 "2차 복제 옵션 선택"에서 "로컬 스냅샷 복사본을 생성한 후 SnapMirror 업데이트"를 선택합니다.

Oracle 데이터베이스 백업 및 예약을 구성합니다

SnapCenter의 데이터베이스 백업은 사용자가 구성할 수 있으며 리소스 그룹에서 개별적으로 또는 그룹으로 설정할 수 있습니다. 백업 간격은 RTO 및 RPO 목표에 따라 달라집니다. 전체 데이터베이스 백업을 몇 시간마다 실행하고 빠른 복구를 위해 10-15분 등의 높은 빈도로 로그 백업을 아카이브하는 것이 좋습니다.

의 Oracle 섹션을 참조하십시오 ["백업 정책을 구현하여 데이터베이스를 보호합니다"](#) 섹션에 생성된 백업 정책을 구현하기 위한 자세한 단계별 프로세스를 참조하십시오 [Oracle 데이터베이스에 대한 백업 정책을 구성합니다](#) 백업 작업 스케줄링에 대한 것입니다.

다음 이미지는 Oracle 데이터베이스를 백업하도록 설정된 리소스 그룹의 예입니다.



EC2 및 FSx Oracle 데이터베이스 관리

AWS EC2 및 FSx 관리 콘솔 외에도 Ansible 제어 노드 및 SnapCenter UI 툴이 이 Oracle 환경에서 데이터베이스 관리를 위해 구축되었습니다.

Ansible 제어 노드를 사용하면 커널 또는 패치 업데이트를 위해 운영 인스턴스와 대기 인스턴스를 동기화된 상태로 유지하는 병렬 업데이트를 통해 Oracle 환경 구성을 관리할 수 있습니다. NetApp 자동화 툴킷을 통해 파일오버, 재동기화 및 파일백을 자동화하여 Ansible을 통해 빠른 애플리케이션 복구 및 가용성을 아카이브할 수 있습니다. 플레이북을 사용하여 반복적인 데이터베이스 관리 작업을 실행하여 사람의 실수를 줄일 수 있습니다.

SnapCenter UI 툴은 Oracle 데이터베이스용 SnapCenter 플러그인을 사용하여 데이터베이스 스냅샷 백업, 시점 복구, 데이터베이스 클론 복제 등을 수행할 수 있습니다. Oracle 플러그인 기능에 대한 자세한 내용은 ["Oracle 데이터베이스용 SnapCenter 플러그인 개요"](#)를 참조하십시오.

다음 섹션에서는 SnapCenter UI를 통해 Oracle 데이터베이스 관리의 주요 기능을 수행하는 방법에 대해 자세히 설명합니다.

- 데이터베이스 스냅샷 백업
- 데이터베이스 시점 복원
- 데이터베이스 클론 생성

데이터베이스 클론 복제는 논리적 데이터 오류 또는 손상 시 데이터 복구를 위해 별도의 EC2 호스트에 운영 데이터베이스의 복제본을 생성하고 애플리케이션 테스트, 디버깅, 패치 검증 등에 클론을 사용할 수도 있습니다.

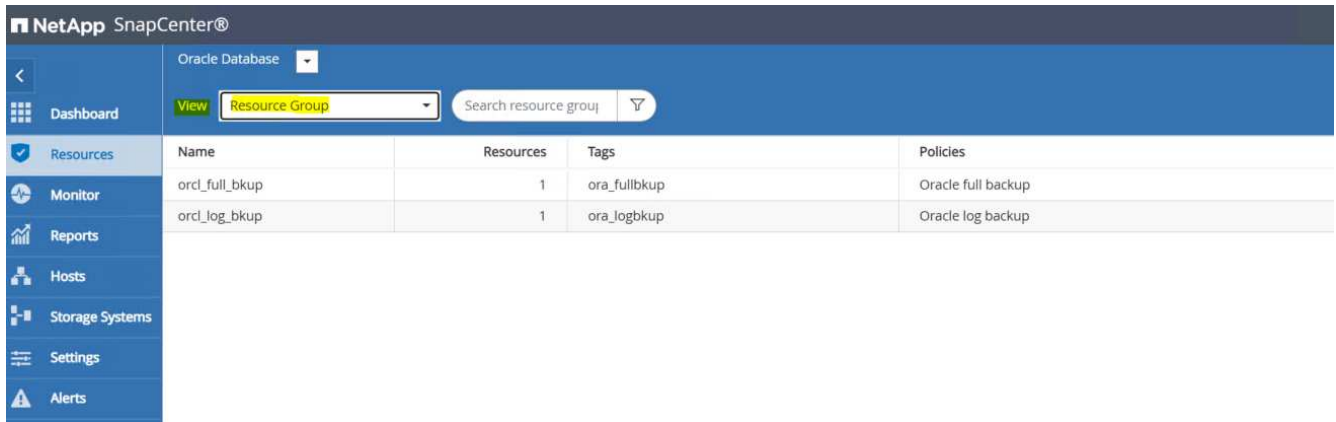
스냅샷 찍기

사용자가 구성한 간격으로 EC2/FSx Oracle 데이터베이스가 정기적으로 백업됩니다. 언제든지 한 번의 스냅샷 백업을 수행할 수도 있습니다. 이는 전체 데이터베이스 스냅샷 백업과 아카이브 로그 전용 스냅샷 백업에 모두 적용됩니다.

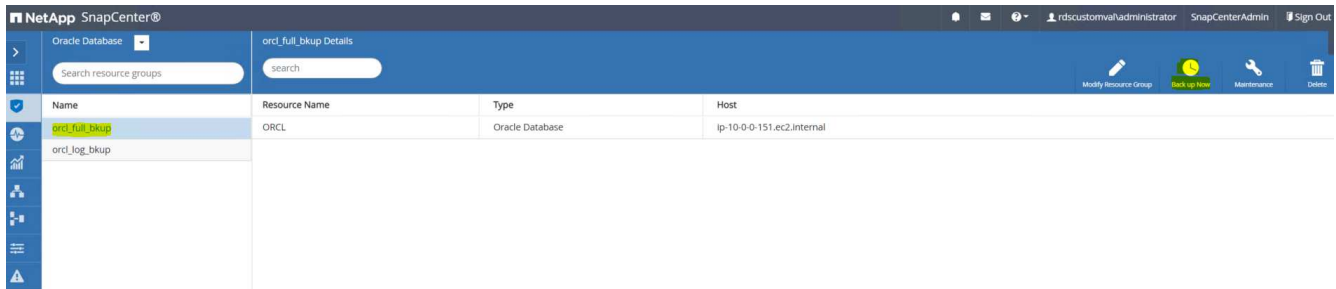
전체 데이터베이스 스냅샷 생성

전체 데이터베이스 스냅샷에는 데이터 파일, 제어 파일 및 아카이브 로그 파일을 포함한 모든 Oracle 파일이 포함됩니다.

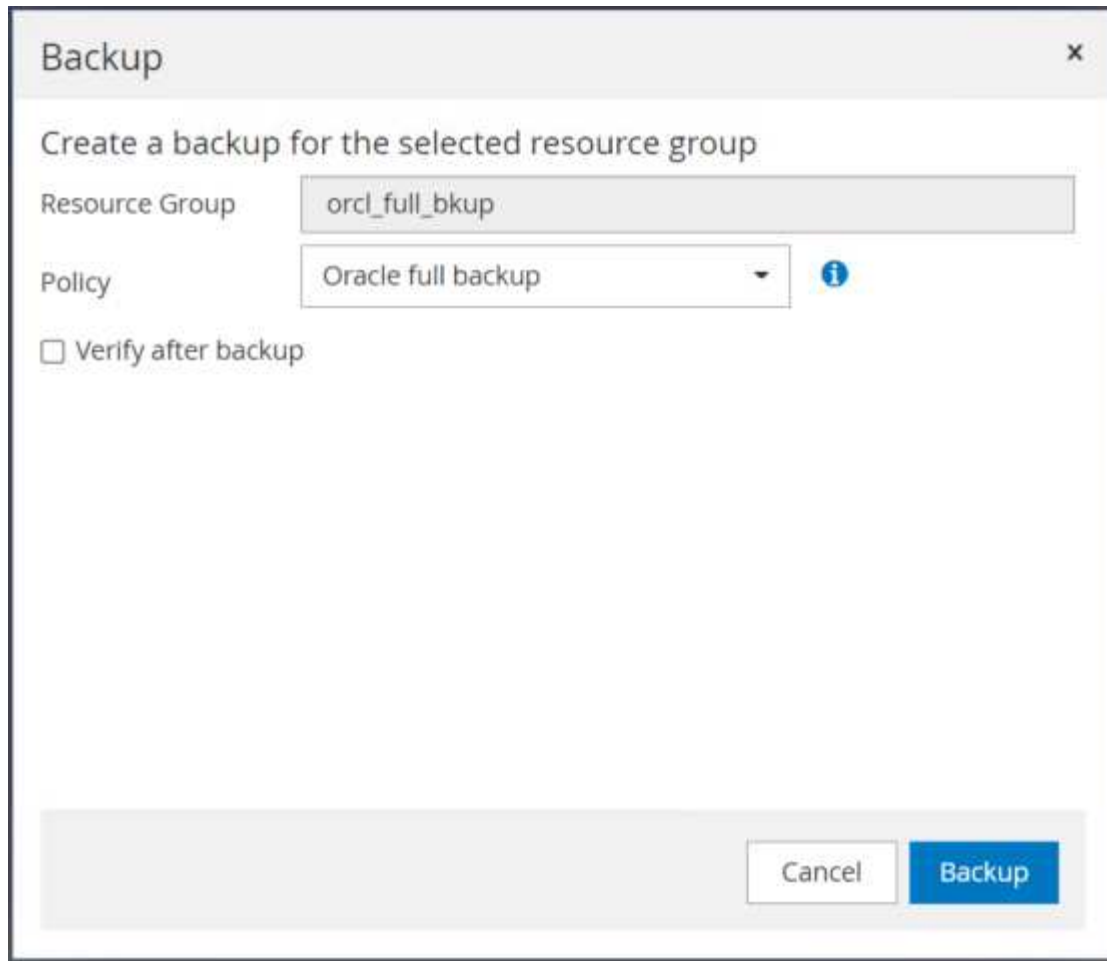
1. SnapCenter UI에 로그인하고 왼쪽 메뉴에서 리소스 를 클릭합니다. 보기 드롭다운에서 자원 그룹 보기로 변경합니다.



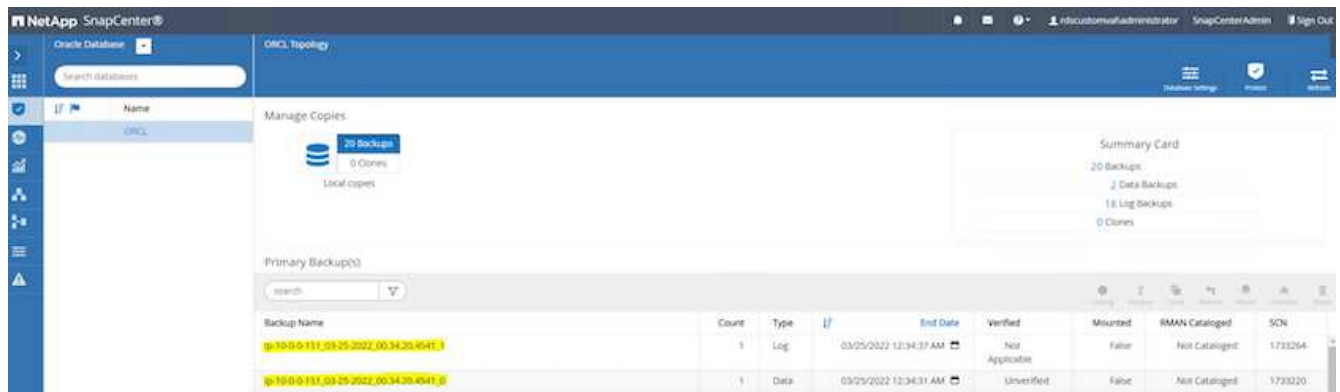
2. 전체 백업 리소스 이름을 클릭한 다음 지금 백업 아이콘을 클릭하여 임시 백업을 시작합니다.



3. 백업을 클릭한 다음 백업을 확인하여 전체 데이터베이스 백업을 시작합니다.



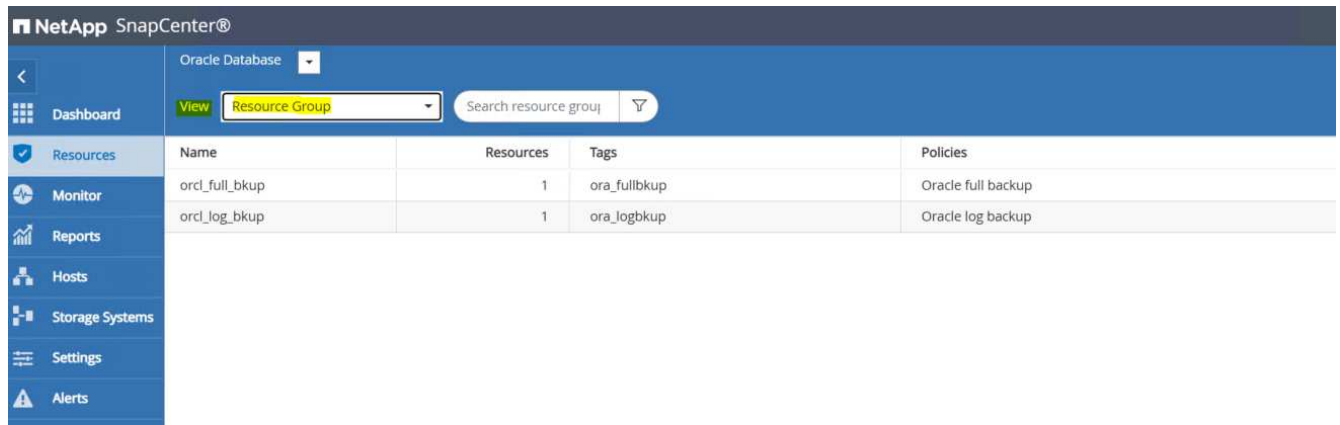
데이터베이스의 리소스 보기에서 데이터베이스 관리 백업 복사본 페이지를 열어 일회성 백업이 성공적으로 완료되었는지 확인합니다. 전체 데이터베이스 백업에서는 데이터 볼륨용 스냅샷 하나와 로그 볼륨용 스냅샷 두 개를 생성합니다.



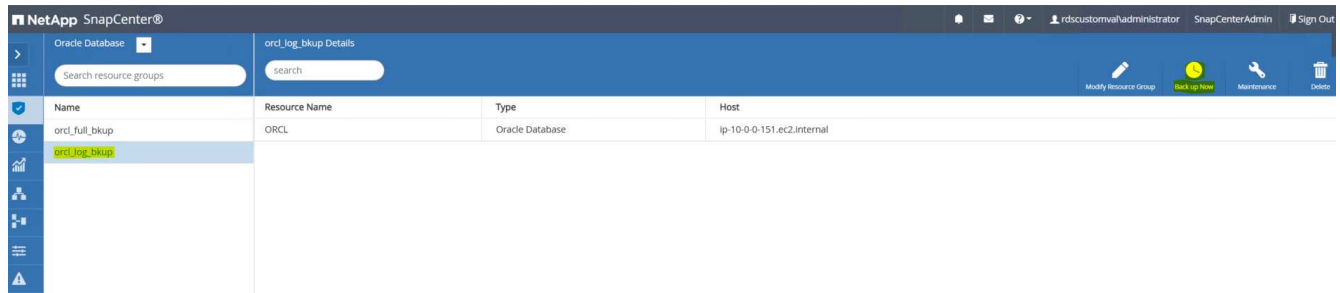
아카이브 로그 스냅샷 생성

아카이브 로그 스냅샷은 Oracle 아카이브 로그 볼륨에 대해서만 생성됩니다.

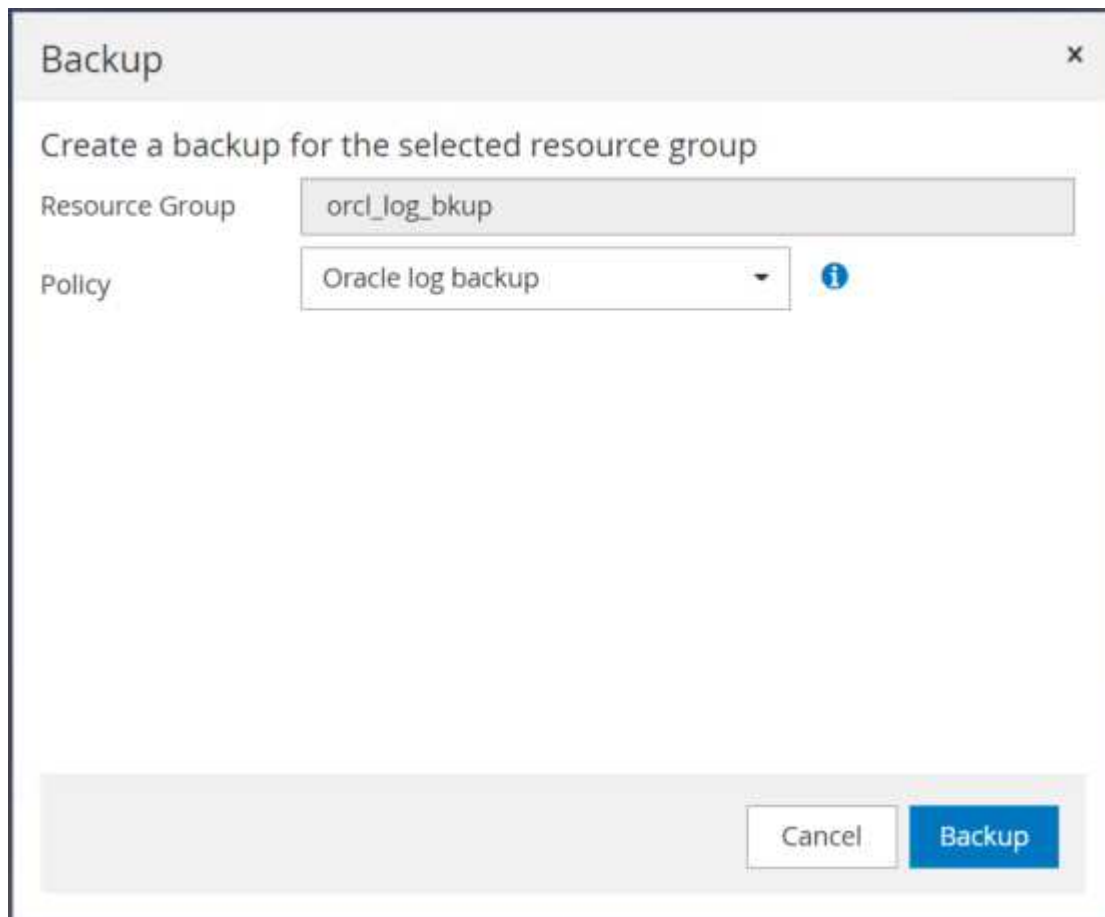
1. SnapCenter UI에 로그인하고 왼쪽 메뉴 모음에서 리소스 탭을 클릭합니다. 보기 드롭다운에서 자원 그룹 보기로 변경합니다.



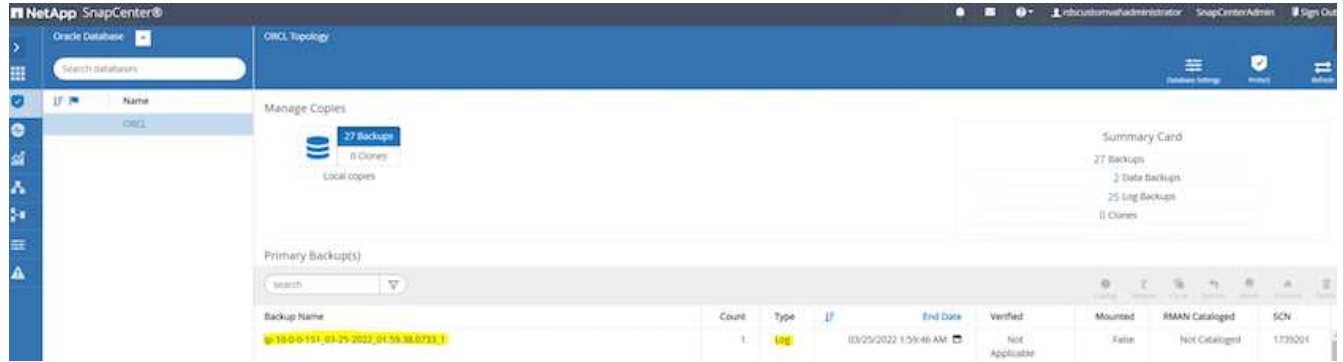
2. 로그 백업 리소스 이름을 클릭한 다음 지금 백업 아이콘을 클릭하여 아카이브 로그에 대한 임시 백업을 시작합니다.



3. 백업을 클릭한 다음 백업 확인 을 클릭하여 아카이브 로그 백업을 시작합니다.



데이터베이스의 리소스 보기에서 데이터베이스 관리 백업 복사본 페이지를 열어 일회성 아카이브 로그 백업이 성공적으로 완료되었는지 확인합니다. 아카이브 로그 백업에서는 로그 볼륨에 대해 하나의 스냅샷을 생성합니다.



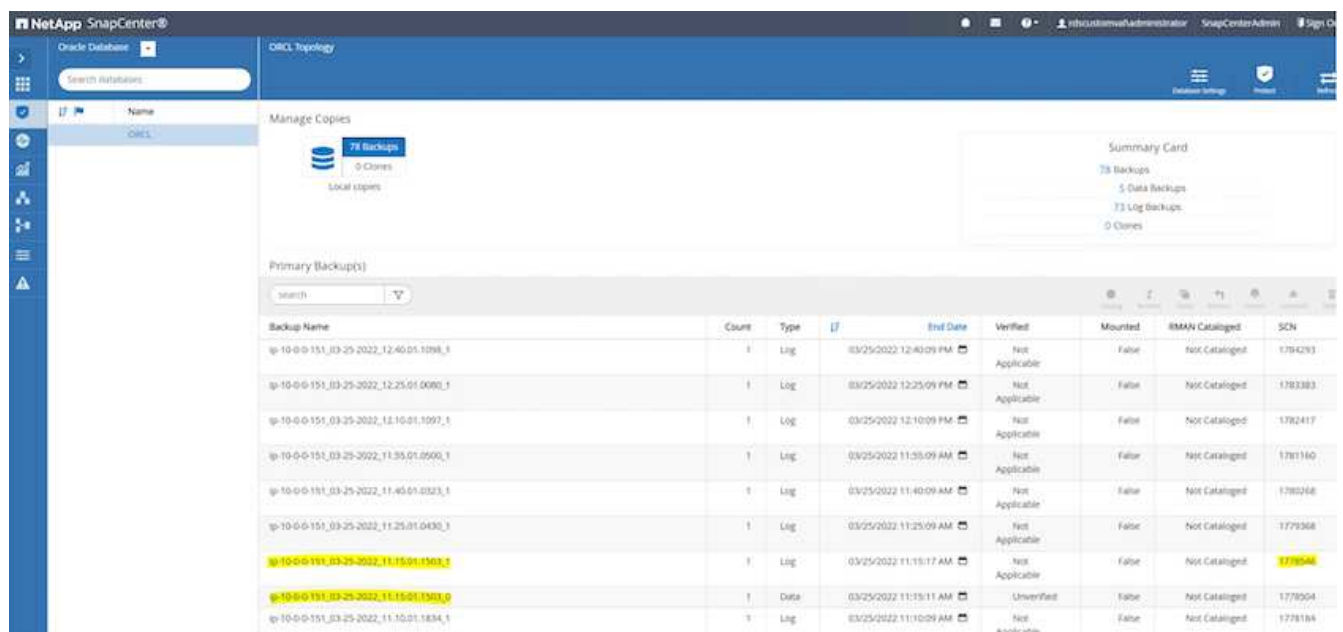
특정 시점으로 복원 중

동일한 EC2 인스턴스 호스트에서 특정 시점으로 SnapCenter 기반 복원이 실행됩니다. 복구를 수행하려면 다음 단계를 수행하십시오.

1. SnapCenter 리소스 탭 > 데이터베이스 보기에서 데이터베이스 이름을 클릭하여 데이터베이스 백업을 엽니다.



2. 데이터베이스 백업 사본과 복원할 원하는 시점을 선택합니다. 또한 해당 시점의 해당 SCN 번호를 표시합니다. 시점 복원은 시간 또는 SCN을 사용하여 수행할 수 있습니다.



3. 로그 볼륨 스냅샷을 강조 표시하고 마운트 버튼을 클릭하여 볼륨을 마운트합니다.

Manage Copies

78 Backups
0 Clones
Local copies

Summary Card

78 Backups
5 Data Backups
73 Log Backups
0 Clones

Primary Backup(s)

Backup Name	Count	Type	End Date	Verified	Mounted	RMAN Cataloged	SCN
ip-10-0-0-151_03-25-2022_12-40-01.1098_1	1	Log	03/25/2022 12:40:09 PM	Not Applicable	False	Not Cataloged	1784293
ip-10-0-0-151_03-25-2022_12-25-01.0080_1	1	Log	03/25/2022 12:25:09 PM	Not Applicable	False	Not Cataloged	1783383
ip-10-0-0-151_03-25-2022_12-10-01.1097_1	1	Log	03/25/2022 12:10:09 PM	Not Applicable	False	Not Cataloged	1782417
ip-10-0-0-151_03-25-2022_11-55-01.0500_1	1	Log	03/25/2022 11:55:09 AM	Not Applicable	False	Not Cataloged	1781160
ip-10-0-0-151_03-25-2022_11-40-01.0323_1	1	Log	03/25/2022 11:40:09 AM	Not Applicable	False	Not Cataloged	1780268
ip-10-0-0-151_03-25-2022_11-25-01.0430_1	1	Log	03/25/2022 11:25:09 AM	Not Applicable	False	Not Cataloged	1779368
ip-10-0-0-151_03-25-2022_11-15-01.1503_1	1	Log	03/25/2022 11:15:17 AM	Not Applicable	False	Not Cataloged	1778546
ip-10-0-0-151_03-25-2022_11-15-01.1503_0	1	Data	03/25/2022 11:15:11 AM	Unverified	False	Not Cataloged	1778504
ip-10-0-0-151_03-25-2022_11-10-01.1834_1	1	Log	03/25/2022 11:10:09 AM	Not Applicable	False	Not Cataloged	1778184

4. 로그 볼륨을 마운트할 운영 EC2 인스턴스를 선택합니다.

Mount backups

Choose the host to mount the backup: ip-10-0-0-151.ec2.internal

Mount path: /var/opt/snapcenter/sco/backup_mount/ip-10-0-0-151_03-25-2022_11-15-01.1503_1/ORCL

Mount Cancel

5. 마운트 작업이 성공적으로 완료되었는지 확인합니다. 또한 EC2 인스턴스 호스트에서 해당 로그 볼륨과 마운트 지정 경로를 확인합니다.

NetApp SnapCenter

Jobs Schedules Events Logs

Dashboard search by name

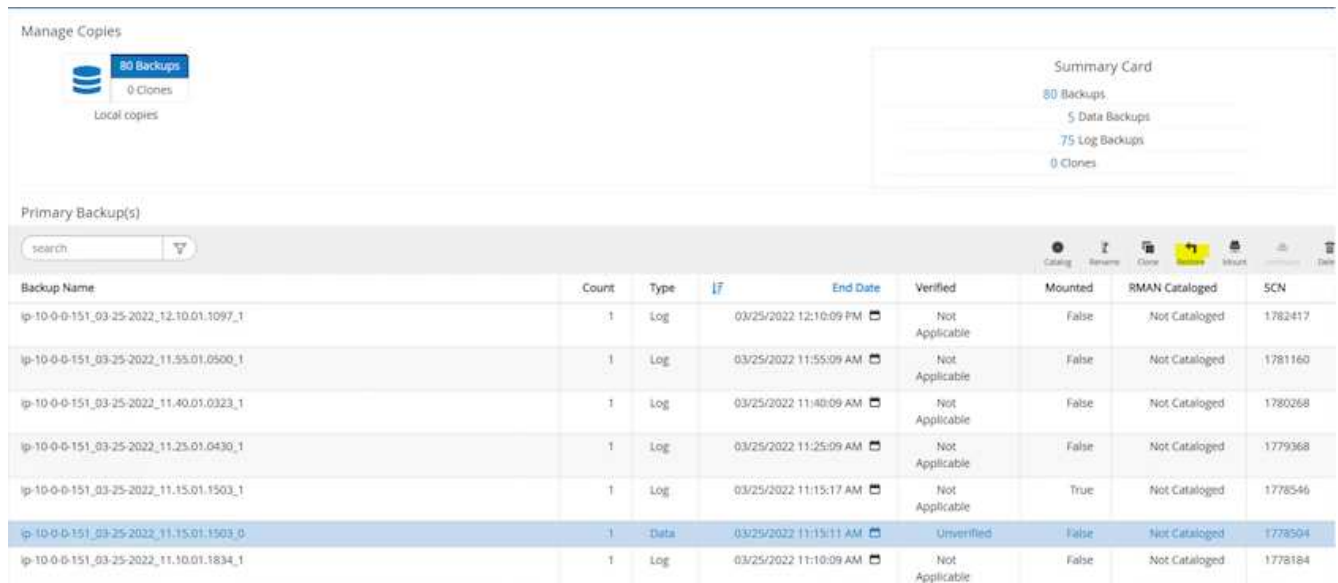
ID	Status	Name	Start date	End date	Owner
4391	Success	Backup of Resource Group 'orcl_bkup' with policy 'Oracle log backup'	3/25/2022 1:40:01 PM	3/25/2022 1:40:13 PM	itbcustomer@administrator
4391	Success	Mount backup ip-10-0-0-151_03-25-2022_11-15-01.1503_1	03-25-2022 1:38:30 PM	03-25-2022 1:38:53 PM	itbcustomer@administrator


```
[root@ip-10-0-0-151 ec2-user]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        7.6G  0    7.6G   0% /dev
tmpfs           16G   7.0G  8.3G  46% /dev/shm
tmpfs           7.7G  604K  7.6G   1% /run
tmpfs           7.7G   0    7.7G   0% /sys/fs/cgroup
/dev/nvme0n1p1  9.8G  5.4G  4.3G  56% /
198.19.255.68:/ora_nfs_log 48G  95M  48G   1% /ora_nfs_log
198.19.255.68:/ora_nfs_data 48G  3.4G  45G   8% /ora_nfs_data
/dev/mapper/dbdata01-lydbdata01 40G  471M  39G   2% /rdsdbdata
/dev/nvme5n1    25G  12G  13G  49% /rdsdbbin
tmpfs           1.6G   0    1.6G   0% /run/user/61001
tmpfs           1.6G   0    1.6G   0% /run/user/61005
198.19.255.68:/Scf91c793-5583-480d-9a34-6275dab17f5b 48G  91M  48G   1% /var/opt/snapcenter/sco/backup_mount/ip-10-0-0-151_03-25-2022_11.15.01.1503_1/ORCL/1
[root@ip-10-0-0-151 ec2-user]#
```

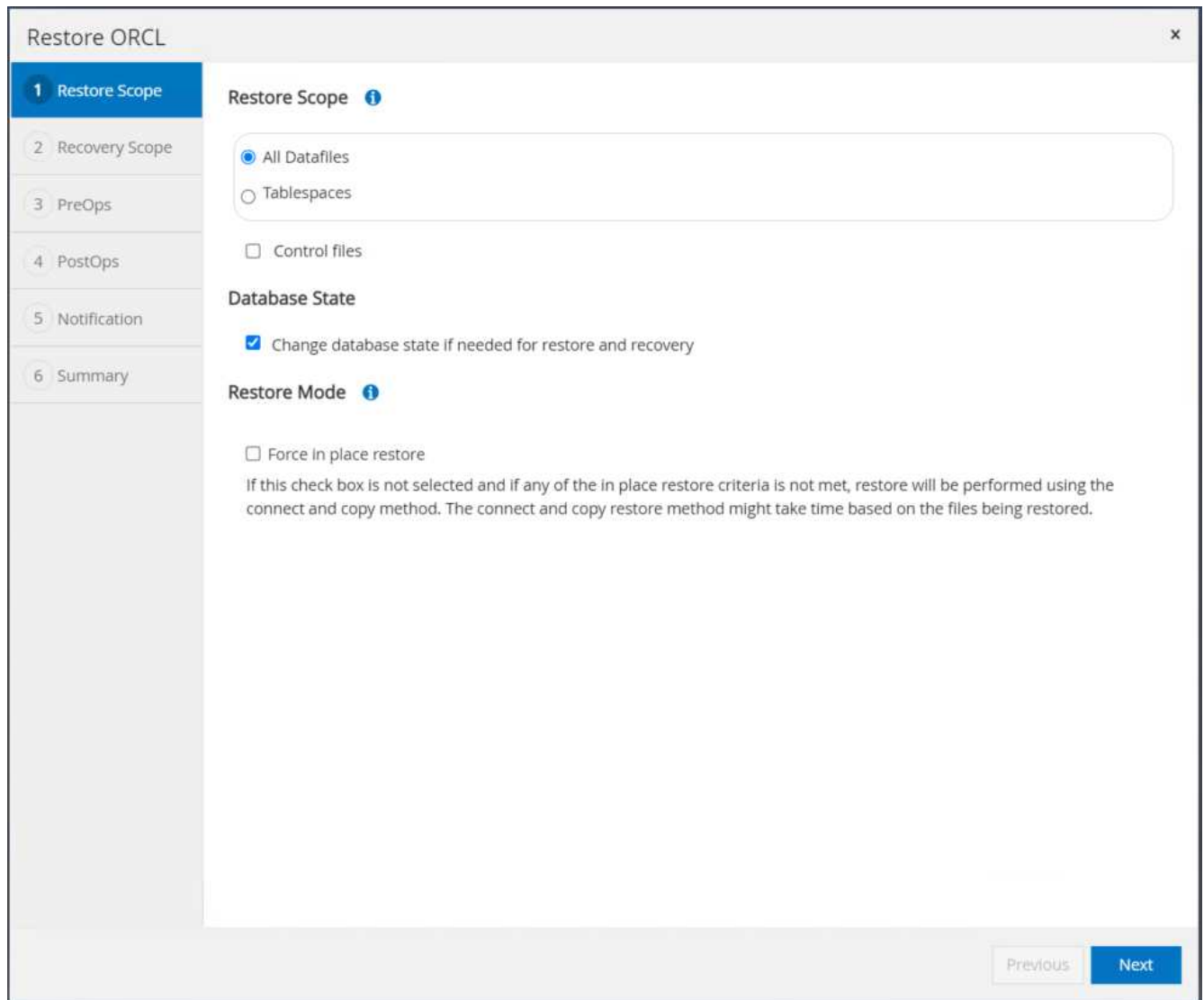
6. 마운트된 로그 볼륨에서 현재 아카이브 로그 디렉토리로 아카이브 로그를 복사합니다.

```
[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. SnapCenter 리소스 탭 > 데이터베이스 백업 페이지로 돌아가서 데이터 스냅샷 복사본을 강조 표시하고 복원 버튼을 클릭하여 데이터베이스 복원 워크플로우를 시작합니다.



8. "모든 데이터 파일" 및 "복원 및 복구에 필요한 경우 데이터베이스 상태 변경"을 선택하고 "다음"을 클릭합니다.



9. SCN 또는 시간을 사용하여 원하는 복구 범위를 선택합니다. 6단계에서 설명된 대로 마운트된 아카이브 로그를 현재 로그 디렉토리에 복사하는 대신, 마운트된 아카이브 로그 경로는 복구를 위해 "외부 아카이브 로그 파일 위치 지정"에 나열될 수 있습니다.

Restore ORCL

1 Restore Scope

2 Recovery Scope

3 PreOps

4 PostOps

5 Notification

6 Summary

Choose Recovery Scope

All Logs ?

Until SCN (System Change Number)

SCN ?

Date and Time

No recovery

Specify external archive log files locations + - ?

Previous **Next**

10. 필요한 경우 실행할 옵션 처방을 지정합니다.

Restore ORCL x

1 Restore Scope
2 Recovery Scope
3 PreOps
4 PostOps
5 Notification
6 Summary

Specify optional scripts to run before performing a restore job ⓘ

Prescript full path

Arguments

Script timeout

11. 필요한 경우 실행할 선택적 애프터스크립트를 지정합니다. 복구 후 열려 있는 데이터베이스를 확인합니다.

Restore ORCL x

- 1 Restore Scope
- 2 Recovery Scope
- 3 PreOps
- 4 PostOps**
- 5 Notification
- 6 Summary

Specify optional scripts to run after performing a restore job ⓘ

Postscript full path

Arguments

Open the database or container database in READ-WRITE mode after recovery

12. 작업 알림이 필요한 경우 SMTP 서버 및 이메일 주소를 제공합니다.

Restore ORCL x

- 1 Restore Scope
- 2 Recovery Scope
- 3 PreOps
- 4 PostOps
- 5 Notification**
- 6 Summary

Provide email settings ⓘ

Email preference:

From:

To:

Subject:

Attach job report

13. 작업 요약을 복원합니다. 마침 을 클릭하여 복원 작업을 시작합니다.

Restore ORCL
✕

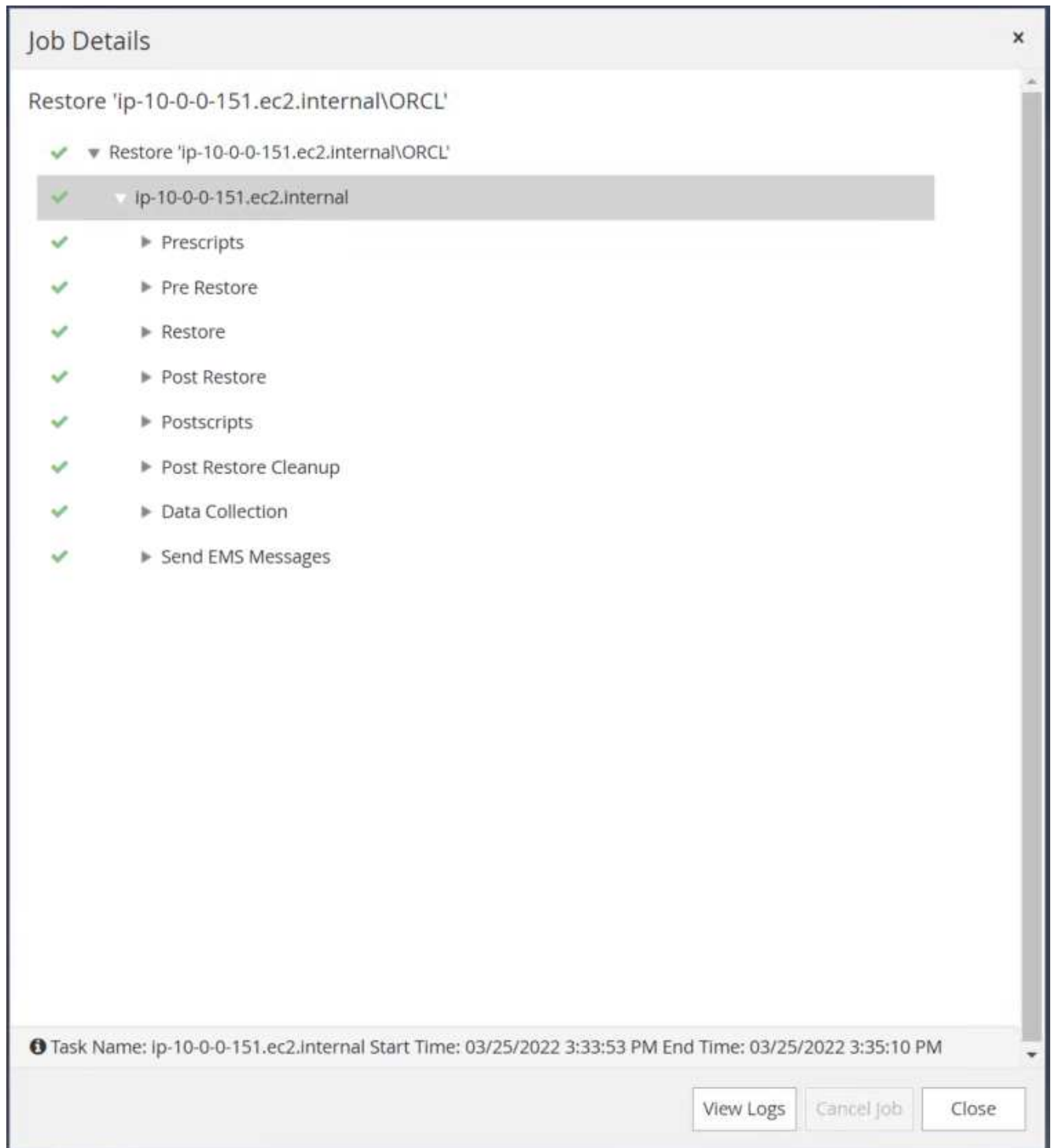
- 1 Restore Scope
- 2 Recovery Scope
- 3 PreOps
- 4 PostOps
- 5 Notification
- 6 Summary

Summary

Backup name	ip-10-0-0-151_03-25-2022_11.15.01.1503_0
Backup date	03/25/2022 11:15:11 AM
Restore scope	All DataFiles
Recovery scope	Until SCN 1778546
Auxiliary destination	
Options	Change database state if necessary , Open the database or container database in READ-WRITE mode after recovery
Prescript full path	None
Prescript arguments	
Postscript full path	None
Postscript arguments	
Send email	No

Previous
Finish

14. SnapCenter에서 복원을 검증합니다.



15. EC2 인스턴스 호스트에서 복원을 확인합니다.


```

-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_TIME    OPEN_MODE
-----
ORCL          1778547            25-MAR-22         READ WRITE

SQL>

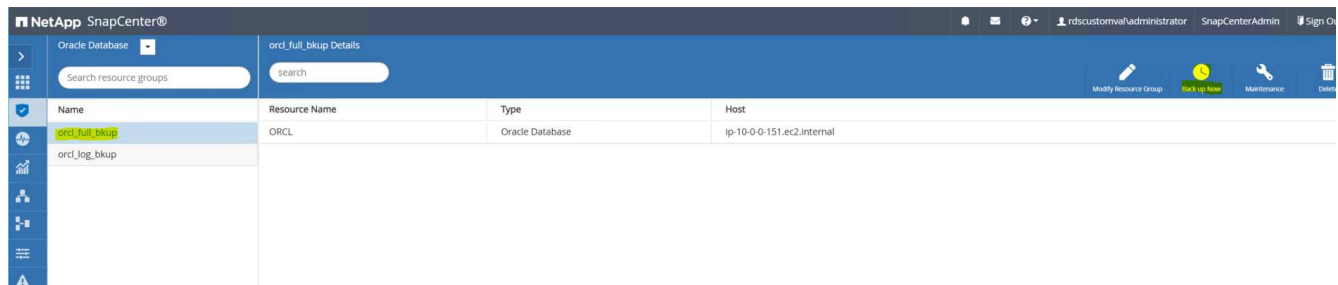
```

16. 복구 로그 볼륨을 마운트 해제하려면 4단계의 단계를 역순으로 수행합니다.

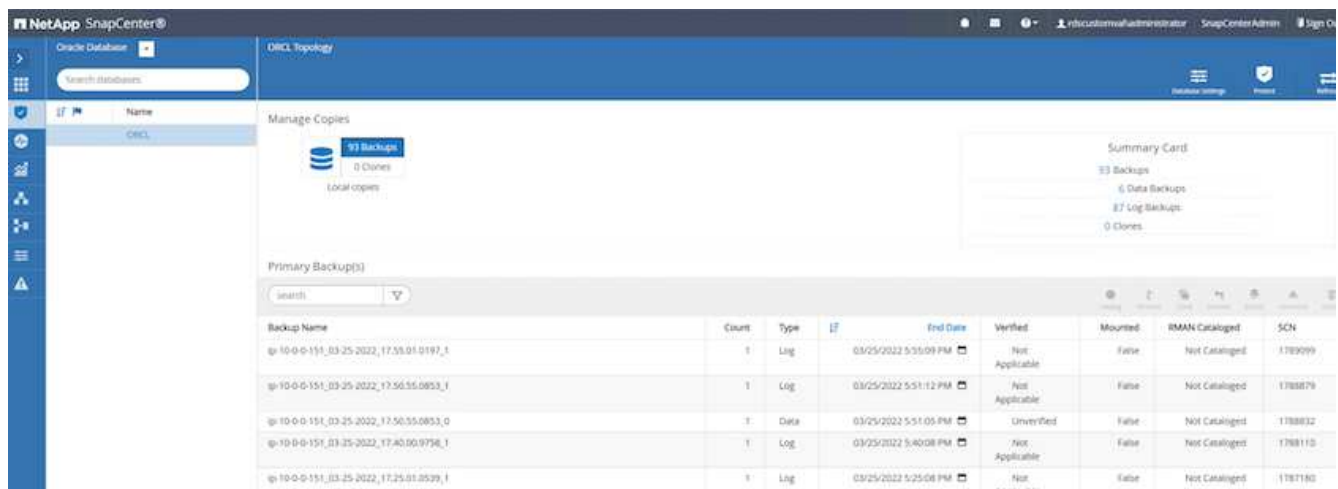
데이터베이스 클론 생성

다음 섹션에서는 SnapCenter 클론 워크플로우를 사용하여 운영 데이터베이스에서 대기 EC2 인스턴스로 데이터베이스 클론을 생성하는 방법을 보여 줍니다.

1. SnapCenter에서 전체 백업 리소스 그룹을 사용하여 기본 데이터베이스의 전체 스냅샷 백업을 수행합니다.



2. SnapCenter 리소스 탭 > 데이터베이스 보기에서 복제본을 생성할 기본 데이터베이스에 대한 데이터베이스 백업 관리 페이지를 엽니다.



3. 4단계에서 생성한 로그 볼륨 스냅샷을 스탠바이 EC2 인스턴스 호스트에 마운트합니다.

ORCL Topology

Manage Copies

95 Backups
0 Clones
Local copies

Summary Card

95 Backups
6 Data Backups
89 Log Backups
0 Clones

Primary Backup(s)

Backup Name	Count	Type	End Date	Verified	Mounted	RMAN Cataloged	SCN
ip-10-0-0-151_03-25-2022_18:55:01.0309_1	1	Log	03/25/2022 6:55:09 PM	Not Applicable	False	Not Cataloged	1892563
ip-10-0-0-151_03-25-2022_18:40:00.9602_1	1	Log	03/25/2022 6:40:23 PM	Not Applicable	False	Not Cataloged	1891375
ip-10-0-0-151_03-25-2022_17:55:01.0197_1	1	Log	03/25/2022 5:55:09 PM	Not Applicable	False	Not Cataloged	1789099
ip-10-0-0-151_03-25-2022_17:50:55.0853_1	1	Log	03/25/2022 5:51:12 PM	Not Applicable	False	Not Cataloged	1788879
ip-10-0-0-151_03-25-2022_17:50:55.0853_0	1	Data	03/25/2022 5:51:05 PM	Unverified	False	Not Cataloged	1788832
ip-10-0-0-151_03-25-2022_17:40:00.9758_1	1	Log	03/25/2022 5:40:08 PM	Not Applicable	False	Not Cataloged	1788110

Mount backups

Choose the host to mount the backup: ip-10-0-0-47.ec2.internal

Mount path: /var/opt/snapcenter/sco/backup_mount/ip-10-0-0-151_03-25-2022_17:50:55.0853_1/ORCL

Mount Cancel

4. 복제본에 대해 클론 복제할 스냅샷 복제본을 강조 표시하고 클론 버튼을 클릭하여 클론 절차를 시작합니다.

ORCL Topology

Manage Copies

93 Backups
0 Clones
Local copies

Summary Card

93 Backups
6 Data Backups
87 Log Backups
0 Clones

Primary Backup(s)

Backup Name	Count	Type	End Date	Verified	Mounted	RMAN Cataloged	SCN
ip-10-0-0-151_03-25-2022_17:55:01.0197_1	1	Log	03/25/2022 5:55:09 PM	Not Applicable	False	Not Cataloged	1789099
ip-10-0-0-151_03-25-2022_17:50:55.0853_1	1	Log	03/25/2022 5:51:12 PM	Not Applicable	False	Not Cataloged	1788879
ip-10-0-0-151_03-25-2022_17:50:55.0853_0	1	Data	03/25/2022 5:51:05 PM	Unverified	False	Not Cataloged	1788832
ip-10-0-0-151_03-25-2022_17:40:00.9758_1	1	Log	03/25/2022 5:40:08 PM	Not Applicable	False	Not Cataloged	1788110
ip-10-0-0-151_03-25-2022_17:25:01.0539_1	1	Log	03/25/2022 5:25:08 PM	Not Applicable	False	Not Cataloged	1787180

5. 기본 데이터베이스 이름과 다르게 복제본 이름을 변경합니다. 다음 을 클릭합니다.

Clone from ORCL

1 Name Provide clone database SID

2 Locations Clone SID ORCLREAD

3 Credentials

4 PreOps

5 PostOps

6 Notification

7 Summary

Previous Next

6. 클론 호스트를 스탠바이 EC2 호스트로 변경하고 기본 이름을 그대로 사용하고 Next를 클릭합니다.

Clone from ORCL
✕

- 1 Name
- 2 Locations
- 3 Credentials
- 4 PreOps
- 5 PostOps
- 6 Notification
- 7 Summary

Select the host to create a clone

Clone host

Datafile locations ⓘ

Control files ⓘ

Redo logs ⓘ

Group	Size	Unit	Number of files
RedoGroup 1	128	MB	1
<input type="text" value="/ora_nfs_data_ORCLREAD/ORCLREAD/redolog/redo04.log"/>			
RedoGroup 2	128	MB	1

7. Oracle 홈 설정을 타겟 Oracle 서버 호스트에 대해 구성된 설정과 일치하도록 변경하고 Next를 클릭합니다.

Clone from ORCL

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

7 Summary

Database Credentials for the clone

Credential name for sys user: None + i

Database port: 1521

Oracle Home Settings i

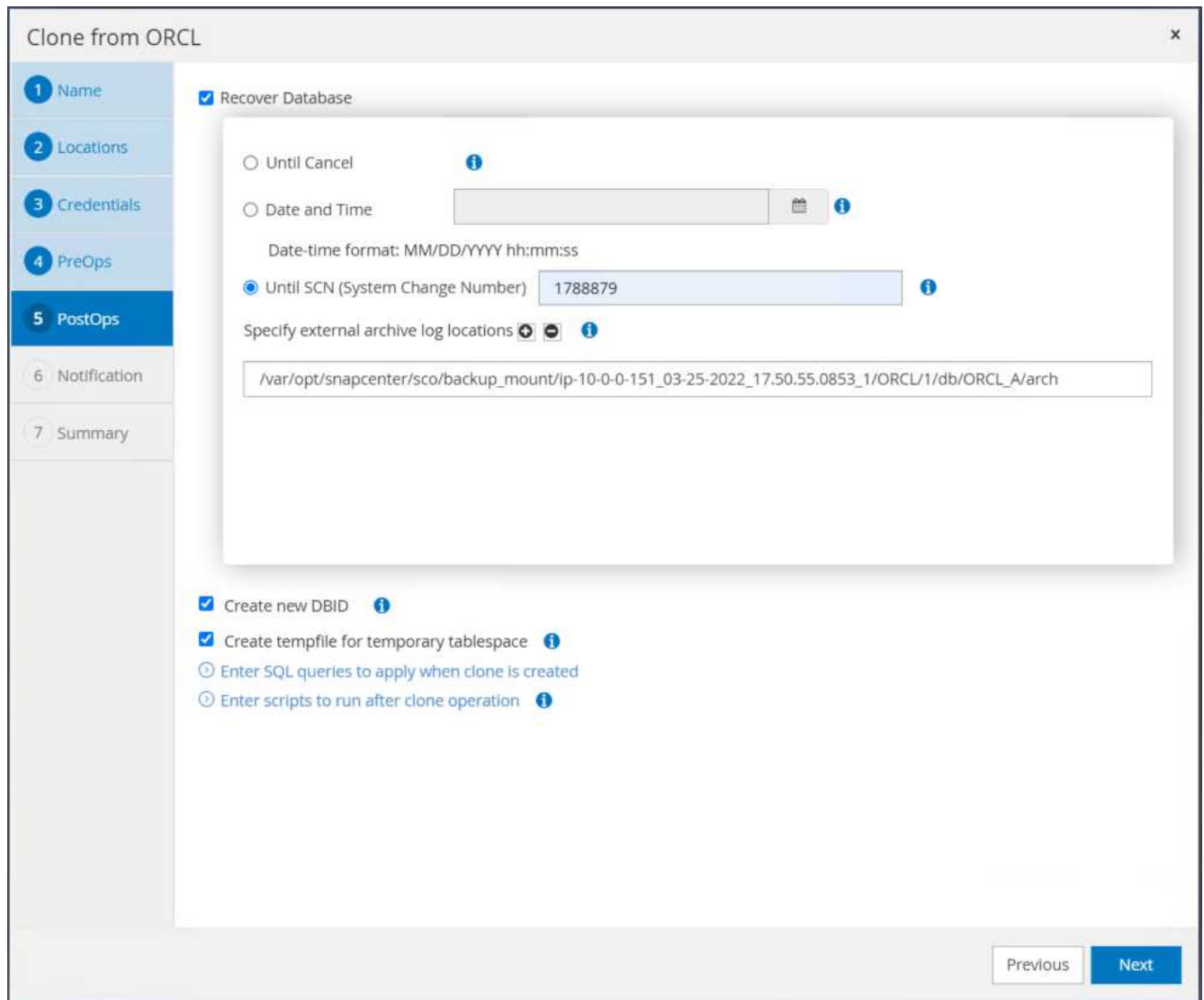
Oracle Home: /rdsdbbin/oracle

Oracle OS User: rdsdb

Oracle OS Group: database

Previous Next

8. 시간 또는 SCN 및 마운트된 아카이브 로그 경로를 사용하여 복구 지점을 지정합니다.



9. 필요한 경우 SMTP 이메일 설정을 전송합니다.

Clone from ORCL x

- 1 Name
- 2 Locations
- 3 Credentials
- 4 PreOps
- 5 PostOps
- 6 Notification
- 7 Summary

Provide email settings i

Email preference ▼
Never

From From email

To Email to

Subject Notification

Attach job report

Previous Next

10. 작업 요약을 클론하고 마침 을 클릭하여 클론 작업을 시작합니다.

Clone from ORCL

1 Name

2 Locations

3 Credentials

4 PreOps

5 PostOps

6 Notification

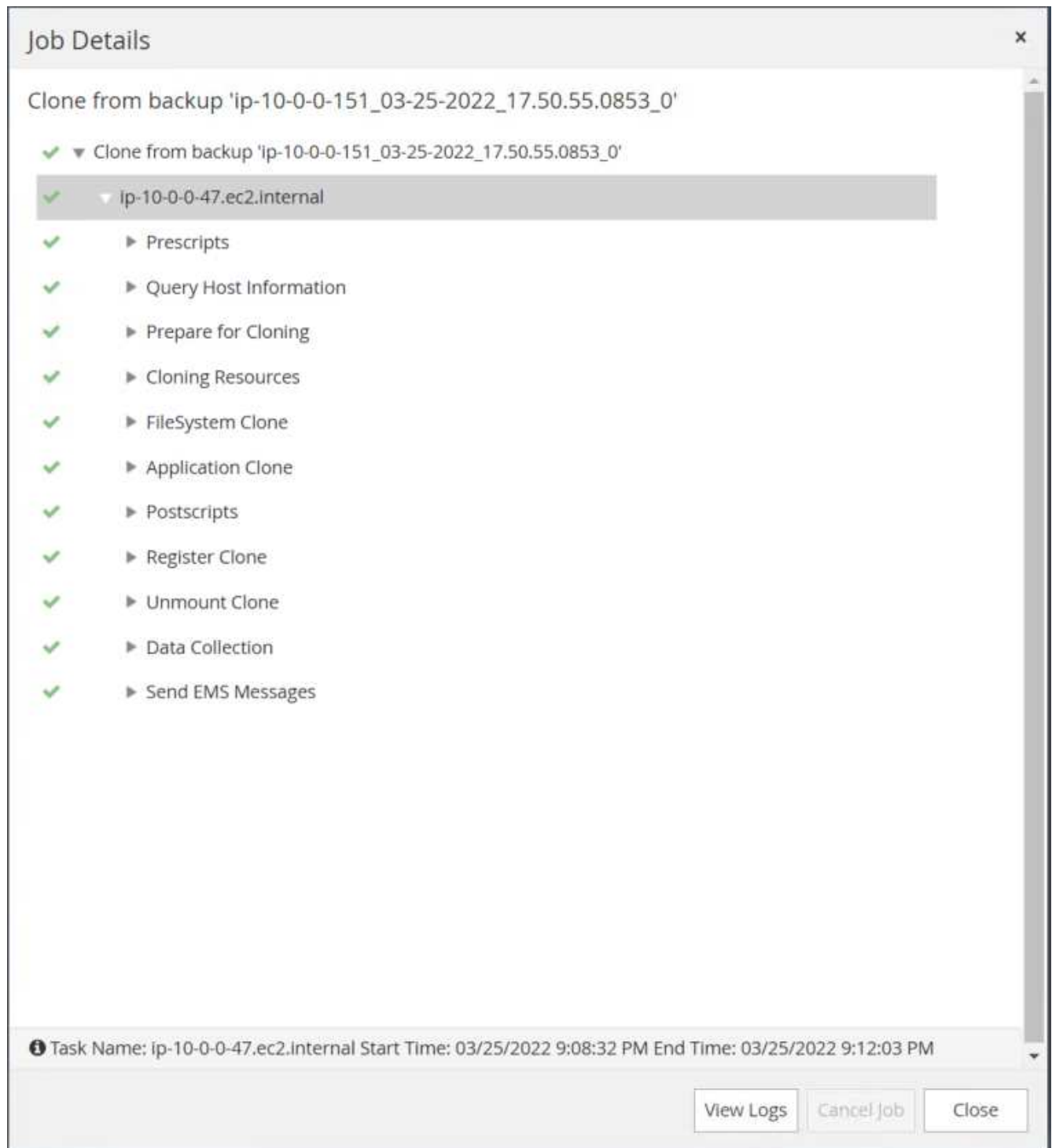
7 Summary

Summary

Clone from backup	ip-10-0-0-151_03-25-2022_17.50.55.0853_0
Clone SID	ORCLREAD
Clone server	ip-10-0-0-47.ec2.internal
Oracle home	/rdsdbbin/oracle
Oracle OS user	rdsdb
Oracle OS group	database
Datafile mountpaths	/ora_nfs_data_ORCLREAD
Control files	/ora_nfs_data_ORCLREAD/ORCLREAD/control/control01.ctl
Redo groups	RedoGroup =1 TotalSize =128 Path =/ora_nfs_data_ORCLREAD/ORCLREAD/redolog/redo04.log RedoGroup =2 TotalSize =128 Path =/ora_nfs_data_ORCLREAD/ORCLREAD/redolog/redo03.log RedoGroup =3 TotalSize =128 Path =/ora_nfs_data_ORCLREAD/ORCLREAD/redolog/redo02.log RedoGroup =4 TotalSize =128 Path =/ora_nfs_data_ORCLREAD/ORCLREAD/redolog/redo01.log
Recovery scope	Until SCN 1788879
Prescript full path	none
Prescript arguments	
Postscript full path	none
Postscript arguments	
Send email	No

Previous Finish

11. 클론 작업 로그를 검토하여 복제본 클론을 확인합니다.



복제된 데이터베이스는 즉시 SnapCenter에 등록됩니다.



12. Oracle 아카이브 로그 모드를 해제합니다. EC2 인스턴스에 Oracle 사용자로 로그인하여 다음 명령을 실행합니다.

```
sqlplus / as sysdba
```

```
shutdown immediate;
```

```
startup mount;
```

```
alter database noarchivelog;
```

```
alter database open;
```



대신 기본 Oracle 백업 복제본을 사용하여 동일한 절차를 통해 타겟 FSx 클러스터의 복제된 보조 백업 복제본에서 클론을 생성할 수도 있습니다.

HA가 대기 및 재동기화로 페일오버됩니다

대기 Oracle HA 클러스터는 컴퓨팅 계층 또는 스토리지 계층에서 운영 사이트에 장애가 발생할 경우고가용성을 제공합니다. 이 솔루션의 중요한 이점 중 하나는 사용자가 언제든지 빈도로 인프라를 테스트하고 검증할 수 있다는 것입니다. 페일오버는 실제 장애로 인해 사용자 시뮬레이션하거나 트리거될 수 있습니다. 페일오버 프로세스는 동일하며 빠른 애플리케이션 복구를 위해 자동화될 수 있습니다.

다음 페일오버 절차 목록을 참조하십시오.

1. 시뮬레이트된 페일오버의 경우 로그 스냅샷 백업을 실행하여 섹션에 설명된 대로 최신 트랜잭션을 대기 사이트로 플러시합니다 [아카이브 로그 스냅샷 생성](#). 실제 장애로 인해 트리거된 페일오버의 경우 마지막으로 성공한 예약 로그 볼륨 백업을 사용하여 복구 가능한 마지막 데이터가 대기 사이트에 복제됩니다.
2. 기본 FSx 클러스터와 대기 FSx 클러스터 간의 SnapMirror를 중단하십시오.
3. 스탠바이 EC2 인스턴스 호스트에 복제된 대기 데이터베이스 볼륨을 마운트합니다.
4. 복제된 Oracle 바이너리가 Oracle 복구에 사용되는 경우 Oracle 바이너리를 다시 링크합니다.
5. 대기 Oracle 데이터베이스를 사용 가능한 마지막 아카이브 로그로 복구합니다.
6. 애플리케이션 및 사용자 액세스를 위해 대기 Oracle 데이터베이스를 엽니다.
7. 실제 운영 사이트 장애의 경우 대기 Oracle 데이터베이스는 이제 새로운 운영 사이트의 역할을 수행하며, 데이터베이스 볼륨을 사용하여 역방향 SnapMirror 방법을 사용하여 장애가 발생한 운영 사이트를 새로운 대기 사이트로 재구축할 수 있습니다.
8. 테스트 또는 검증을 위해 시뮬레이션된 운영 사이트 오류의 경우 테스트 연습을 완료한 후 대기 Oracle 데이터베이스를 종료합니다. 그런 다음 대기 EC2 인스턴스 호스트에서 대기 데이터베이스 볼륨을 마운트 해제하고 운영 사이트에서 대기 사이트로 복제를 다시 동기화합니다.

이러한 절차는 NetApp Automation Toolkit을 사용하여 퍼블릭 NetApp GitHub 사이트에서 다운로드할 수 있습니다.

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

설정 및 페일오버 테스트를 시도하기 전에 README 지침을 주의 깊게 읽으십시오.

사내에서 퍼블릭 클라우드로 데이터베이스 마이그레이션

데이터베이스 마이그레이션은 어떤 방법으로든 어려운 작업입니다. Oracle 데이터베이스를 사내에서 클라우드로 마이그레이션하는 경우도 예외는 아닙니다.

다음 섹션에서는 Oracle 데이터베이스를 AWS EC2 컴퓨팅 및 FSx 스토리지 플랫폼을 통해 AWS 퍼블릭 클라우드로 마이그레이션할 때 고려해야 할 주요 요소를 제공합니다.

ONTAP 스토리지는 온프레미스에서 사용할 수 있습니다

온프레미스 Oracle 데이터베이스가 ONTAP 스토리지 어레이에 있는 경우 AWS FSx ONTAP 스토리지에 내장된 NetApp SnapMirror 기술을 사용하여 데이터베이스 마이그레이션을 위한 복제를 쉽게 설정할 수 있습니다. NetApp BlueXP 콘솔을 사용하여 마이그레이션 프로세스를 조정할 수 있습니다.

1. 온프레미스 인스턴스와 일치하는 타겟 컴퓨팅 EC2 인스턴스를 구축합니다.
2. FSx 콘솔에서 동일한 크기의 일치하는 데이터베이스 볼륨을 프로비저닝합니다.
3. FSx 데이터베이스 볼륨을 EC2 인스턴스에 마운트합니다.
4. 온프레미스 데이터베이스 볼륨에서 타겟 FSx 데이터베이스 볼륨으로의 SnapMirror 복제를 설정합니다. 초기 동기화는 운영 소스 데이터를 이동하는 데 시간이 다소 걸릴 수 있지만 다음과 같은 증분 업데이트는 훨씬 더 빠릅니다.
5. 전환 시 운영 애플리케이션을 종료하여 모든 트랜잭션을 중지합니다. Oracle sqlplus CLI 인터페이스에서 Oracle 온라인 로그 스위치를 실행하고 SnapMirror 동기화가 타겟 볼륨에 마지막으로 아카이빙된 로그를 푸시하도록 허용합니다.
6. 미러링된 볼륨을 분할하고 타겟에서 Oracle 복구를 실행하고 서비스를 위해 데이터베이스를 불러옵니다.
7. 애플리케이션을 클라우드의 Oracle 데이터베이스에 지정

다음 비디오에서는 NetApp BlueXP 콘솔 및 SnapMirror 복제를 사용하여 Oracle 데이터베이스를 사내에서 AWS FSx/EC2로 마이그레이션하는 방법을 보여줍니다.

온프레미스 Oracle DB를 AWS로 마이그레이션

ONTAP 스토리지는 온프레미스에서 사용할 수 없습니다

사내 Oracle 데이터베이스가 ONTAP 이외의 타사 스토리지에서 호스팅되는 경우 데이터베이스 마이그레이션은 Oracle 데이터베이스 백업 복사본의 복원을 기반으로 합니다. 전환하기 전에 아카이브 로그를 재생하여 최신 상태로 만들어야 합니다.

AWS S3를 데이터베이스 이동 및 마이그레이션을 위한 스테이징 스토리지 영역으로 사용할 수 있습니다. 이 방법에 대한 자세한 내용은 다음 단계를 참조하십시오.

1. 사내 인스턴스와 유사한 새로운 EC2 인스턴스를 프로비저닝합니다.

2. FSx 스토리지에서 동일한 데이터베이스 볼륨을 프로비저닝하고 EC2 인스턴스에 볼륨을 마운트합니다.
3. 디스크 레벨의 Oracle 백업 복사본을 생성합니다.
4. 백업 복사본을 AWS S3 스토리지로 이동합니다.
5. S3 스토리지에서 데이터와 아카이브 로그를 가져와 Oracle 제어 파일을 다시 생성하고 데이터베이스를 복원 및 복구합니다.
6. 타겟 Oracle 데이터베이스를 사내 소스 데이터베이스와 동기화합니다.
7. 전환 시 애플리케이션과 소스 Oracle 데이터베이스를 종료합니다. 최근 몇 개의 아카이브 로그를 복사하여 대상 Oracle 데이터베이스에 적용하여 최신 상태로 만듭니다.
8. 사용자 액세스를 위해 대상 데이터베이스를 시작합니다.
9. 대상 데이터베이스로 애플리케이션을 리디렉션하여 전환을 완료합니다.

최대한의 가용성과 함께 **PDB** 재배치를 사용하여 온프레미스 **Oracle** 데이터베이스를 **AWS FSx/EC2**로 마이그레이션합니다

이 마이그레이션 방식은 PDB/CDB 멀티 테넌트 모델에 이미 배포된 Oracle 데이터베이스에 가장 적합하며 ONTAP 스토리지는 온-프레미스에서 사용할 수 없습니다. PDB 재배치 방법은 Oracle PDB 핫 클론 기술을 활용하여 소스 CDB와 대상 CDB 간에 PDB를 이동하는 동시에 서비스 중단을 최소화합니다.

먼저, 사내에서 마이그레이션할 PDB를 호스팅할 충분한 스토리지를 가진 AWS FSx/EC2에서 CDB를 생성합니다. 여러 온프레미스 PDB를 한 번에 하나씩 재배치할 수 있습니다.

1. 멀티 테넌트 PDB/CDB 모델이 아닌 단일 인스턴스에 온-프레미스 데이터베이스가 배포된 경우의 지침을 따릅니다. "[멀티 테넌트 CDB에서 단일 인스턴스 비 CDB를 PDB로 변환합니다](#)" 단일 인스턴스를 멀티 테넌트 PDB/CDB로 변환합니다. 그런 다음 다음 다음 단계에 따라 변환된 PDB를 AWS FSx/EC2에서 CDB로 마이그레이션합니다.
2. 멀티 테넌트 PDB/CDB 모델에 온-프레미스 데이터베이스가 이미 배포된 경우의 지침을 따릅니다. "[PDB 재배치를 통해 온프레미스 Oracle 데이터베이스를 클라우드로 마이그레이션합니다](#)" 마이그레이션을 수행합니다.

다음 비디오에서는 최대 가용성과 함께 PDB 재배치를 사용하여 Oracle 데이터베이스(PDB)를 FSx/EC2로 마이그레이션하는 방법을 보여 줍니다.

["온프레미스 Oracle PDB를 최대 가용성으로 AWS CDB로 마이그레이션합니다"](#)



1단계와 2단계의 지침이 Azure 퍼블릭 클라우드의 맥락에서 설명되지만 이 절차는 변경 없이 AWS 클라우드에 적용할 수 있습니다.

NetApp 솔루션 자동화 팀에서는 사내에서 AWS 클라우드로 Oracle 데이터베이스를 쉽게 마이그레이션할 수 있는 마이그레이션 툴킷을 제공합니다. 다음 명령을 사용하여 PDB 재배치용 Oracle 데이터베이스 마이그레이션 툴킷을 다운로드합니다.

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

저작권 정보

Copyright © 2024 NetApp, Inc. All Rights Reserved. 미국에서 인쇄된 본 문서의 어떠한 부분도 저작권 소유자의 사전 서면 승인 없이는 어떠한 형식이나 수단(복사, 녹음, 녹화 또는 전자 검색 시스템에 저장하는 것을 비롯한 그래픽, 전자적 또는 기계적 방법)으로도 복제될 수 없습니다.

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이선스와 고지사항이 적용됩니다.

본 소프트웨어는 NetApp에 의해 '있는 그대로' 제공되며 상품성 및 특정 목적에의 적합성에 대한 명시적 또는 묵시적 보증을 포함하여(이에 제한되지 않음) 어떠한 보증도 하지 않습니다. NetApp은 대체품 또는 대체 서비스의 조달, 사용 불능, 데이터 손실, 이익 손실, 영업 중단을 포함하여(이에 국한되지 않음), 이 소프트웨어의 사용으로 인해 발생하는 모든 직접 및 간접 손해, 우발적 손해, 특별 손해, 징벌적 손해, 결과적 손해의 발생에 대하여 그 발생 이유, 책임론, 계약 여부, 엄격한 책임, 불법 행위(과실 또는 그렇지 않은 경우)와 관계없이 어떠한 책임도 지지 않으며, 이와 같은 손실의 발생 가능성이 통지되었다 하더라도 마찬가지입니다.

NetApp은 본 문서에 설명된 제품을 언제든지 예고 없이 변경할 권리를 보유합니다. NetApp은 NetApp의 명시적인 서면 동의를 받은 경우를 제외하고 본 문서에 설명된 제품을 사용하여 발생하는 어떠한 문제에도 책임을 지지 않습니다. 본 제품의 사용 또는 구매의 경우 NetApp에서는 어떠한 특허권, 상표권 또는 기타 지적 재산권이 적용되는 라이선스도 제공하지 않습니다.

본 설명서에 설명된 제품은 하나 이상의 미국 특허, 해외 특허 또는 출원 중인 특허로 보호됩니다.

제한적 권리 표시: 정부에 의한 사용, 복제 또는 공개에는 DFARS 252.227-7013(2014년 2월) 및 FAR 52.227-19(2007년 12월)의 기술 데이터-비상업적 품목에 대한 권리(Rights in Technical Data -Noncommercial Items) 조항의 하위 조항 (b)(3)에 설명된 제한사항이 적용됩니다.

여기에 포함된 데이터는 상업용 제품 및/또는 상업용 서비스(FAR 2.101에 정의)에 해당하며 NetApp, Inc.의 독점 자산입니다. 본 계약에 따라 제공되는 모든 NetApp 기술 데이터 및 컴퓨터 소프트웨어는 본질적으로 상업용이며 개인 비용만으로 개발되었습니다. 미국 정부는 데이터가 제공된 미국 계약과 관련하여 해당 계약을 지원하는 데에만 데이터에 대한 전 세계적으로 비독점적이고 양도할 수 없으며 재사용이 불가능하며 취소 불가능한 라이선스를 제한적으로 가집니다. 여기에 제공된 경우를 제외하고 NetApp, Inc.의 사전 서면 승인 없이는 이 데이터를 사용, 공개, 재생산, 수정, 수행 또는 표시할 수 없습니다. 미국 국방부에 대한 정부 라이선스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

상표 정보

NETAPP, NETAPP 로고 및 <http://www.netapp.com/TM>에 나열된 마크는 NetApp, Inc.의 상표입니다. 기타 회사 및 제품 이름은 해당 소유자의 상표일 수 있습니다.