



# TR-4983: iSCSI를 통한 NetApp ASA 에서의 간소화되고 자동화된 Oracle 배포 NetApp database solutions

NetApp  
August 18, 2025

# 목차

TR-4983: iSCSI를 통한 NetApp ASA 에서의 간소화되고 자동화된 Oracle 배포 .....	1
목적 .....	1
대상 .....	1
솔루션 테스트 및 검증 환경 .....	1
아키텍처 .....	2
하드웨어 및 소프트웨어 구성 요소 .....	2
랩 환경에서의 Oracle 데이터베이스 구성 .....	3
배포 고려사항의 핵심 요소 .....	3
솔루션 구축 .....	3
배포를 위한 전제 조건 .....	4
자동화 매개변수 파일 .....	4
매개변수 파일 구성 .....	5
플레이북 실행 .....	8
실행 후 검증 .....	10
SnapCenter 사용한 Oracle 백업, 복원 및 복제 .....	16
추가 정보를 찾을 수 있는 곳 .....	16

# TR-4983: iSCSI를 통한 NetApp ASA 에서의 간소화되고 자동화된 Oracle 배포

Allen Cao, Niyaz Mohamed, NetApp

이 솔루션은 iSCSI 프로토콜을 사용하는 기본 데이터베이스 스토리지로 NetApp ASA 어레이에 자동화된 Oracle을 배포하고 보호하기 위한 개요와 세부 정보를 제공하며, asm을 볼륨 관리자로 사용하여 독립형 ReStart에서 Oracle 데이터베이스를 구성합니다.

## 목적

NetApp ASA 시스템은 SAN 인프라에 현대적인 솔루션을 제공합니다. 이러한 솔루션은 규모에 맞게 간소화하고 데이터베이스와 같은 비즈니스에 중요한 애플리케이션을 가속화하며, 데이터를 항상 사용할 수 있도록 보장(가동 시간 99.9999%)하고 TCO와 탄소 발자국을 줄이는 데 도움이 됩니다. NetApp ASA 시스템에는 가장 높은 성능을 요구하는 애플리케이션을 위해 설계된 A-시리즈 모델과 비용 효율적인 대용량 배포를 위해 최적화된 C-시리즈 모델이 포함됩니다. ASA A-시리즈와 C-시리즈 시스템은 함께 뛰어난 성능을 제공하여 고객 경험을 개선하고 결과 도출 시간을 단축하며, 비즈니스에 중요한 데이터를 가용성, 보호 및 보안 상태로 유지하고 모든 작업 부하에 대해 보다 효과적인 용량을 제공하며, 업계에서 가장 효과적인 보증을 바탕으로 합니다.

이 문서는 Ansible 자동화를 사용하여 ASA 시스템으로 구축된 SAN 환경에서 Oracle 데이터베이스를 간소화된 방식으로 배포하는 방법을 보여줍니다. Oracle 데이터베이스는 iSCSI 프로토콜을 사용하여 데이터 액세스를 위한 독립형 ReStart 구성과 ASA 스토리지 어레이의 데이터베이스 디스크 관리를 위한 Oracle ASM으로 배포됩니다. 또한 NetApp ASA 시스템에서 스토리지 효율적인 데이터베이스 작업을 위해 NetApp SnapCenter UI 도구를 사용하여 Oracle 데이터베이스 백업, 복원 및 복제에 대한 정보도 제공합니다.

이 솔루션은 다음과 같은 사용 사례를 해결합니다.

- NetApp ASA 시스템에 자동화된 Oracle 데이터베이스를 기본 데이터베이스 저장소로 배포
- NetApp SnapCenter 도구를 사용하여 NetApp ASA 시스템에서 Oracle 데이터베이스 백업 및 복원
- NetApp SnapCenter 도구를 사용하여 NetApp ASA 시스템에서 개발/테스트 또는 기타 사용 사례를 위한 Oracle 데이터베이스 복제

## 대상

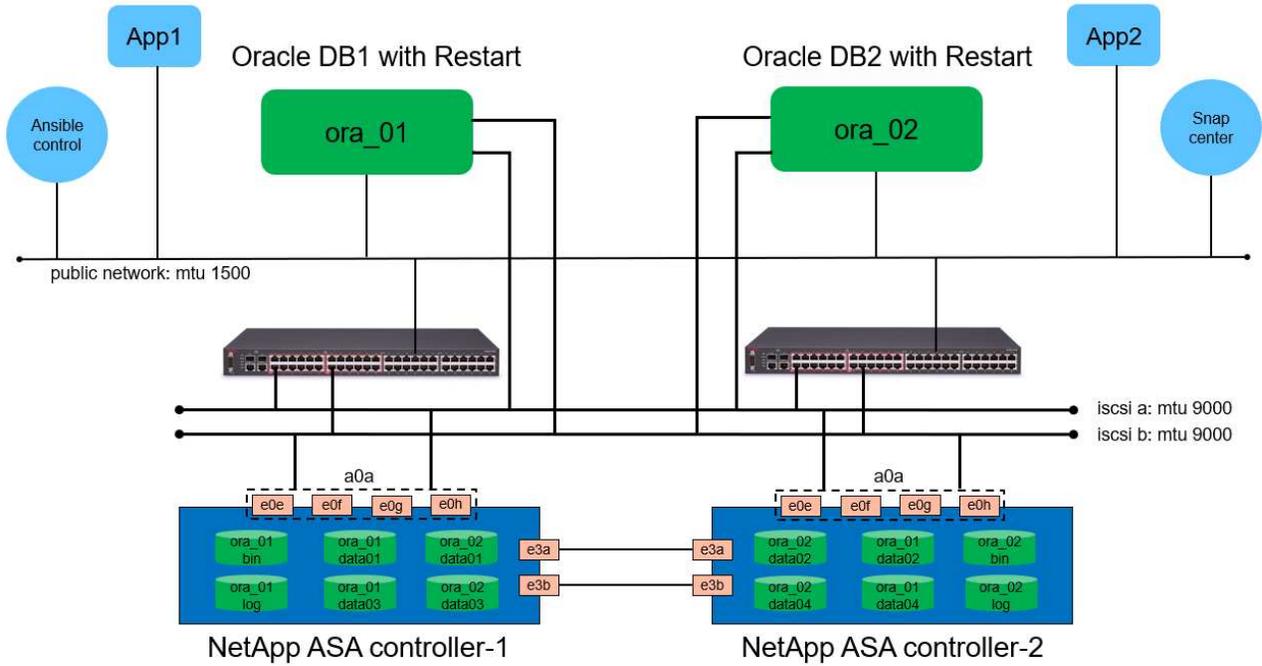
이 솔루션은 다음과 같은 사람들을 위해 만들어졌습니다.

- NetApp ASA 시스템에 Oracle을 배포하려는 DBA입니다.
- NetApp ASA 시스템에서 Oracle 워크로드를 테스트하려는 데이터베이스 솔루션 아키텍트입니다.
- NetApp ASA 시스템에 Oracle 데이터베이스를 배포하고 관리하려는 스토리지 관리자입니다.
- NetApp ASA 시스템에 Oracle 데이터베이스를 구축하려는 애플리케이션 소유자입니다.

## 솔루션 테스트 및 검증 환경

이 솔루션의 테스트와 검증은 최종 배포 환경과 일치하지 않을 수 있는 실험실 환경에서 수행되었습니다. 섹션을 참조하세요 [배포 고려사항의 핵심 요소](#) 자세한 내용은.

### Simplified, Automated Oracle Database Deployment on NetApp ASA with iSCSI



NetApp

### 하드웨어 및 소프트웨어 구성 요소

하드웨어		
NetApp ASA A400	버전 9.13.1P1	2개의 NS224 셀프, 총 69.3TiB 용량을 갖춘 48개의 NVMe AFF 드라이브
UCSB-B200-M4	인텔® 제온® CPU E5-2690 v4 @ 2.60GHz	4노드 VMware ESXi 클러스터
소프트웨어		
레드햇 리눅스	RHEL-8.6, 4.18.0-372.9.1.el8.x86_64 커널	테스트를 위해 RedHat 구독을 배포했습니다.
윈도우 서버	2022 표준, 10.0.20348 빌드 20348	SnapCenter 서버 호스팅
오라클 그리드 인프라	버전 19.18	RU 패치 p34762026_190000_Linux-x86-64.zip이 적용되었습니다.
오라클 데이터베이스	버전 19.18	RU 패치 p34765931_190000_Linux-x86-64.zip을 적용했습니다.
오라클 OPatch	버전 12.2.0.1.36	최신 패치 p6880880_190000_Linux-x86-64.zip
SnapCenter 서버	버전 4.9P1	작업 그룹 배포

VMware vSphere 하이퍼바이저	버전 6.5.0.20000	VMware Tools, 버전: 11365 - Linux, 12352 - Windows
JDK를 엽니다	버전 java-1.8.0-openjdk.x86_64	DB VM의 SnapCenter 플러그인 요구 사항

## 랩 환경에서의 Oracle 데이터베이스 구성

섬기는 사람	데이터 베이스	DB 스토리지
ora_01	NTAP1(NTAP1_PDB1, NTAP1_PDB2, NTAP1_PDB3)	ASA A400 의 iSCSI LUN
ora_02	NTAP2(NTAP2_PDB1, NTAP2_PDB2, NTAP2_PDB3)	ASA A400 의 iSCSI LUN

## 배포 고려사항의 핵심 요소

- 오라클 데이터베이스 저장 레이아웃. 이 자동화된 Oracle 배포에서는 기본적으로 Oracle 바이너리, 데이터 및 로그를 호스팅하기 위해 4개의 데이터베이스 볼륨을 프로비저닝합니다. 그런 다음 데이터와 로그 LUN에서 두 개의 ASM 디스크 그룹을 만듭니다. +DATA asm 디스크 그룹 내에서 각 ASA A400 클러스터 노드의 볼륨에 두 개의 데이터 LUN을 프로비저닝합니다. +LOGS asm 디스크 그룹 내에서 단일 ASA A400 노드의 로그 볼륨에 두 개의 LUN을 생성합니다. ONTAP 볼륨 내에 여러 개의 LUN을 배치하면 일반적으로 더 나은 성능을 제공합니다.
- 다중 DB 서버 배포. 자동화 솔루션은 단일 Ansible 플레이북 실행으로 Oracle 컨테이너 데이터베이스를 여러 DB 서버에 배포할 수 있습니다. DB 서버 수에 관계없이 플레이북 실행은 동일하게 유지됩니다. 여러 DB 서버를 배포하는 경우 플레이북은 ASA A400 의 듀얼 컨트롤러에 데이터베이스 LUN을 최적으로 배치하기 위한 알고리즘을 사용하여 빌드됩니다. 서버 호스트 인덱스에 있는 홀수 DB 서버의 바이너리와 로그 LUN은 컨트롤러 1에 위치합니다. 서버 호스트 인덱스의 짝수 DB 서버의 바이너리와 로그 LUN은 컨트롤러 2에 위치합니다. DB 데이터는 두 개의 컨트롤러에 균등하게 분산됩니다. Oracle ASM은 두 컨트롤러의 데이터 LUN을 단일 ASM 디스크 그룹으로 결합하여 두 컨트롤러의 처리 능력을 최대한 활용합니다.
- iSCSI 구성. 데이터베이스 VM은 스토리지 액세스를 위해 iSCSI 프로토콜을 통해 ASA 스토리지에 연결합니다. 중복성을 위해 각 컨트롤러 노드에 이중 경로를 구성하고 다중 경로 스토리지 액세스를 위해 DB 서버에 iSCSI 다중 경로를 설정해야 합니다. 성능과 처리량을 극대화하려면 스토리지 네트워크에서 점보 프레임 활성화하세요.
- 사용자가 생성하는 각 Oracle ASM 디스크 그룹에 사용할 Oracle ASM 중복 수준입니다. ASA A400 클러스터 디스크 수준에서 데이터 보호를 위해 RAID DP 에 스토리지를 구성하므로 다음을 사용해야 합니다. External Redundancy 즉, 해당 옵션은 Oracle ASM이 디스크 그룹의 내용을 미러링하는 것을 허용하지 않습니다.
- 데이터베이스 백업. NetApp 사용자 친화적인 UI 인터페이스를 통해 데이터베이스 백업, 복원 및 복제를 위한 SnapCenter software 제품군을 제공합니다. NetApp 빠른(1분 이내) 스냅샷 백업, 빠른(분) 데이터베이스 복원 및 데이터베이스 복제를 달성하기 위해 이러한 관리 도구를 구현할 것을 권장합니다.

## 솔루션 구축

다음 섹션에서는 iSCSI를 통해 단일 노드의 DB VM에 직접 마운트된 데이터베이스 LUN을 사용하여 NetApp ASA A400 에서 자동화된 Oracle 19c를 배포하고 보호하는 단계별 절차를 제공합니다. Oracle ASM을 데이터베이스 볼륨 관리자로 사용하여 구성을 다시 시작합니다.

## 배포를 위한 전제 조건

배포에는 다음과 같은 전제 조건이 필요합니다.

1. NetApp ASA 스토리지 어레이가 설치 및 구성되어 있다고 가정합니다. 여기에는 iSCSI 브로드캐스트 도메인, 두 컨트롤러 노드의 LACP 인터페이스 그룹 a0a, 두 컨트롤러 노드의 iSCSI VLAN 포트(a0a-`<iscsi-a-vlan-id>`, a0a-`<iscsi-b-vlan-id>`)가 포함됩니다. 도움이 필요할 경우 다음 링크에서 자세한 단계별 지침을 확인하세요. "[상세 가이드 - ASA A400](#)"
2. 최신 버전의 Ansible과 Git이 설치된 Ansible 컨트롤러 노드로 Linux VM을 프로비저닝합니다. 자세한 내용은 다음 링크를 참조하세요. "[NetApp 솔루션 자동화 시작하기](#)" 섹션에서 - Setup the Ansible Control Node for CLI deployments on RHEL / CentOS 또는 Setup the Ansible Control Node for CLI deployments on Ubuntu / Debian.
3. iSCSI를 위한 NetApp Oracle 배포 자동화 툴킷의 사본을 복제합니다.

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

4. 최신 버전의 NetApp SnapCenter UI 도구를 실행하기 위해 Windows 서버를 프로비저닝합니다. 자세한 내용은 다음 링크를 참조하세요. "[SnapCenter 서버 설치](#)"
5. 베어메탈 또는 가상화된 VM으로 RHEL Oracle DB 서버 두 개를 구축합니다. 비밀번호 권한이 없는 sudo로 DB 서버에 관리자 사용자를 만들고 Ansible 호스트와 Oracle DB 서버 호스트 간에 SSH 개인/공개 키 인증을 활성화합니다. DB 서버 /tmp/archive 디렉토리에 Oracle 19c 설치 파일을 다음 단계로 진행합니다.

```
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 준비할 충분한 공간을 확보하려면 Oracle VM 루트 볼륨에 최소 50G를 할당했는지 확인하세요.

6. 다음 영상을 시청해보세요:

[iSCSI를 사용한 NetApp ASA에서의 간소화되고 자동화된 Oracle 배포](#)

## 자동화 매개변수 파일

Ansible 플레이북은 미리 정의된 매개변수를 사용하여 데이터베이스 설치 및 구성 작업을 실행합니다. 이 Oracle 자동화 솔루션의 경우 플레이북을 실행하기 전에 사용자 입력이 필요한 사용자 정의 매개변수 파일이 3개 있습니다.

- 호스트 - 자동화 플레이북이 실행되는 대상을 정의합니다.
- vars/vars.yml - 모든 대상에 적용되는 변수를 정의하는 전역 변수 파일입니다.
- host\_vars/host\_name.yml - 로컬 대상에만 적용되는 변수를 정의하는 로컬 변수 파일입니다. 우리의 사용 사례에서는 이는 Oracle DB 서버입니다.

이러한 사용자 정의 변수 파일 외에도 필요하지 않은 한 변경할 필요가 없는 기본 매개변수가 포함된 여러 가지 기본 변수 파일이 있습니다. 다음 섹션에서는 사용자 정의 변수 파일이 구성되는 방식을 보여줍니다.

## 매개변수 파일 구성

## 1. Ansible 대상 hosts 파일 구성:

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

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

## 2. 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: on-prem

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

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

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

```

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

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

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

# Enter RHEL subscription to enable repo
redhat_sub_username: xxxxxxxx
redhat_sub_password: "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 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. 각각은 서로 다른 작업 블록을 수행하고 서로 다른 목적을 위해 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 실행하는 세 가지 옵션이 있습니다.

1. 모든 배포 플레이북을 한 번에 결합해서 실행합니다.

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

2. 1~4까지의 숫자 순서대로 플레이북을 하나씩 실행합니다.

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

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

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

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

3. 태그와 함께 0-all\_playbook.yml을 실행합니다.

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

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

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

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

#### 4. 환경 실행 취소

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

## 실행 후 검증

플레이백을 실행한 후 Oracle 사용자로 Oracle DB 서버에 로그인하여 Oracle 그리드 인프라와 데이터베이스가 성공적으로 생성되었는지 확인합니다. 다음은 호스트 ora\_01에서 Oracle 데이터베이스 검증의 예입니다.

1. 생성된 그리드 인프라와 리소스를 검증합니다.

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

```

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

racle/product/19.0.0

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

```



무시하다 Not All Endpoints Registered 국가 세부 정보. 이는 리스너와의 수동 및 동적 데이터베이스 등록 충돌로 인해 발생하며 무시해도 됩니다.

2. ASM 필터 드라이버가 예상대로 작동하는지 확인합니다.

```

[oracle@ora_01 ~]$ 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
327680    318644          0      318644          0
N  DATA/
MOUNTED   EXTERN  N      512     512    4096   4194304
81920     78880          0      78880          0
N  LOGS/
ASMCMDB> lsdsk
Path
AFD:ORA_01_DAT1_01
AFD:ORA_01_DAT1_03
AFD:ORA_01_DAT1_05
AFD:ORA_01_DAT1_07
AFD:ORA_01_DAT2_02
AFD:ORA_01_DAT2_04
AFD:ORA_01_DAT2_06
AFD:ORA_01_DAT2_08
AFD:ORA_01_LOGS_01
AFD:ORA_01_LOGS_02
ASMCMDB> afd_state
ASMCMDB-9526: The AFD state is 'LOADED' and filtering is 'ENABLED' on
host 'ora_01'
ASMCMDB>

```

3. Oracle Enterprise Manager Express에 로그인하여 데이터베이스를 검증하세요.



# ORACLE ENTERPRISE MANAGER DATABASE EXPRESS

Username

Password

Container Name

**Log in**



Copyright 2013, 2020, Oracle and/or its affiliates. All rights reserved.

## ORACLE Enterprise Manager Database Express

NTAP1 (19.18.0.0.0) Performance Storage

### Database Home

Time Zone

Browser (GMT-05:00)

1 min Auto-Refresh Refresh

#### Status

Up Time 1 hours, 7 minutes, 23 seconds

Type Single Instance (NTAP1)

CDB (3 PDB(s))

Version 19.18.0.0.0 Enterprise Edition

Platform Name Linux x86 64-bit

Thread 1

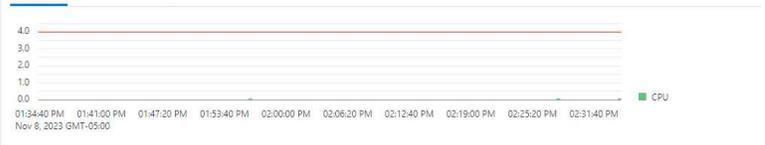
Archiver Stopped

Last Backup Time N/A

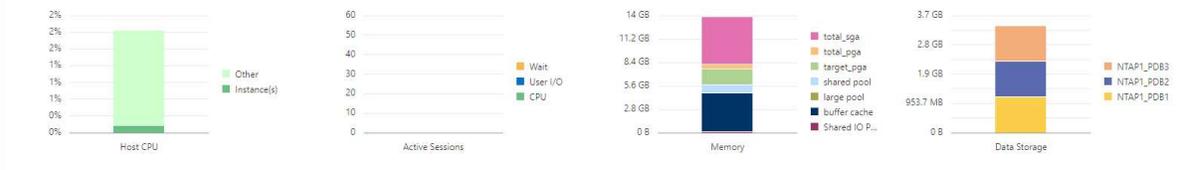
Incident(s) 4

#### Performance

Activity Services Containers



#### Resources



#### SQL Monitor - Last Hour (20 max)



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

```
SQL> show pdbs
```

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

```
SQL> alter session set container=NTAP1_PDB1;
```

Session altered.

```
SQL> select dbms_xdb_config.gethttpsport() from dual;
```

```
DBMS_XDB_CONFIG.GETHTTPSPO...
-----
                                0
```

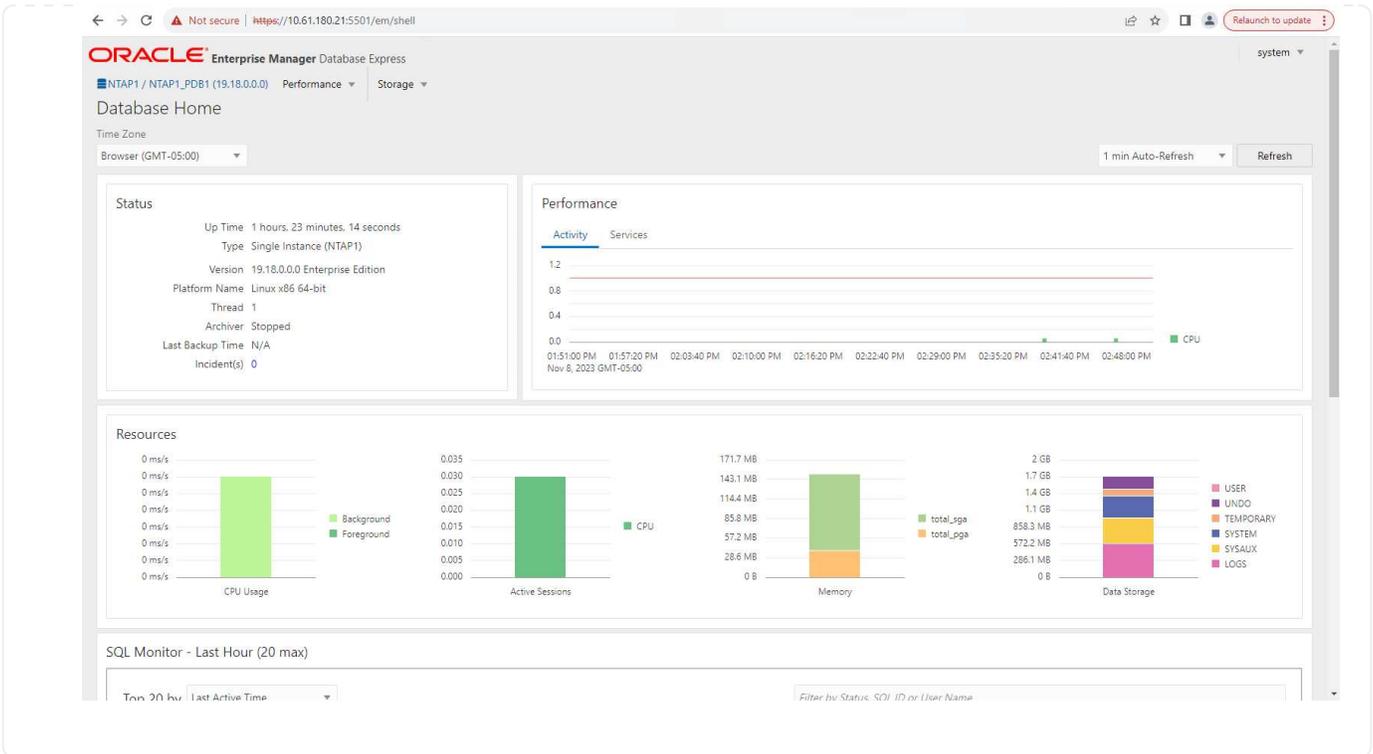
```
SQL> exec DBMS_XDB_CONFIG.SETHTTPSPO...;
```

PL/SQL procedure successfully completed.

```
SQL> select dbms_xdb_config.gethttpsport() from dual;
```

```
DBMS_XDB_CONFIG.GETHTTPSPO...
-----
                                5501
```

login to NTAP1\_PDB1 from port 5501.



## SnapCenter 사용한 Oracle 백업, 복원 및 복제

TR-4979를 참조하세요"게스트 마운트 FSx ONTAP 통해 AWS의 VMware Cloud에서 간소화되고 자체 관리되는 Oracle" 부분 Oracle backup, restore, and clone with SnapCenter SnapCenter 설정 및 데이터베이스 백업, 복원, 복제 워크플로우 실행에 대한 자세한 내용은 다음을 참조하세요.

## 추가 정보를 찾을 수 있는 곳

이 문서에 설명된 정보에 대해 자세히 알아보려면 다음 문서 및/또는 웹사이트를 검토하세요.

- NETAPP ASA: 울플래시 SAN 어레이

["https://www.netapp.com/data-storage/all-flash-san-storage-array/"](https://www.netapp.com/data-storage/all-flash-san-storage-array/)

- 새 데이터베이스 설치로 독립형 서버에 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)

- ONTAP 과 함께 Red Hat Enterprise Linux 8.2 사용

["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)

## 저작권 정보

Copyright © 2026 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.