



TR-4987: NFS를 사용한 Azure NetApp Files 에서의 간소화되고 자동화된 Oracle 배포 NetApp database solutions

NetApp
August 18, 2025

목차

TR-4987: NFS를 사용한 Azure NetApp Files 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	3
자동화 매개변수 파일	5
매개변수 파일 구성	5
플레이북 실행	8
실행 후 검증	10
Oracle 데이터베이스를 Azure로 마이그레이션	17
SnapCenter 사용한 Oracle 백업, 복원 및 복제	17
추가 정보를 찾을 수 있는 곳	18

TR-4987: NFS를 사용한 Azure NetApp Files 에서의 간소화되고 자동화된 Oracle 배포

Allen Cao, Niyaz Mohamed, NetApp

이 솔루션은 NFS 프로토콜을 사용하여 기본 데이터베이스 저장소로 Microsoft Azure NetApp Files 에 자동화된 Oracle을 배포하기 위한 개요와 세부 정보를 제공하고, Oracle 데이터베이스는 dNFS가 활성화된 컨테이너 데이터베이스로 배포됩니다.

목적

클라우드에서 성능 집약적이고 대기 시간에 민감한 Oracle 워크로드를 실행하는 것은 어려울 수 있습니다. Azure NetApp Files (ANF)를 사용하면 기업의 LOB(기간 업무) 및 스토리지 전문가가 코드를 변경하지 않고도 까다로운 Oracle 워크로드를 쉽게 마이그레이션하고 실행할 수 있습니다. Azure NetApp Files 온프레미스에서 Azure로 Oracle 데이터베이스를 새로 배포하거나 마이그레이션(리프트 앤 시프트)하는 등 다양한 시나리오에서 기본 공유 파일 스토리지 서비스로 널리 사용됩니다.

이 문서에서는 Ansible 자동화를 사용하여 NFS 마운트를 통해 Azure NetApp 파일에 Oracle 데이터베이스를 간소화하여 배포하는 방법을 보여줍니다. Oracle 데이터베이스는 Oracle dNFS 프로토콜을 활성화하여 컨테이너 데이터베이스(CDB) 및 플러그형 데이터베이스(PDB) 구성으로 배포되어 성능을 향상시킵니다. 또한, 온프레미스 Oracle 단일 인스턴스 데이터베이스 또는 PDB는 자동화된 PDB 재배치 방법을 사용하여 서비스 중단을 최소화하면서 Azure에 새로 배포된 컨테이너 데이터베이스로 마이그레이션할 수 있습니다. 또한 Azure Cloud의 NetApp SnapCenter UI 도구를 사용하여 Oracle 데이터베이스를 빠르게 백업, 복원 및 복제하는 방법에 대한 정보도 제공합니다.

이 솔루션은 다음과 같은 사용 사례를 해결합니다.

- Azure NetApp 파일에 자동화된 Oracle 컨테이너 데이터베이스 배포
- 온프레미스와 Azure 클라우드 간 자동화된 Oracle 데이터베이스 마이그레이션

대상

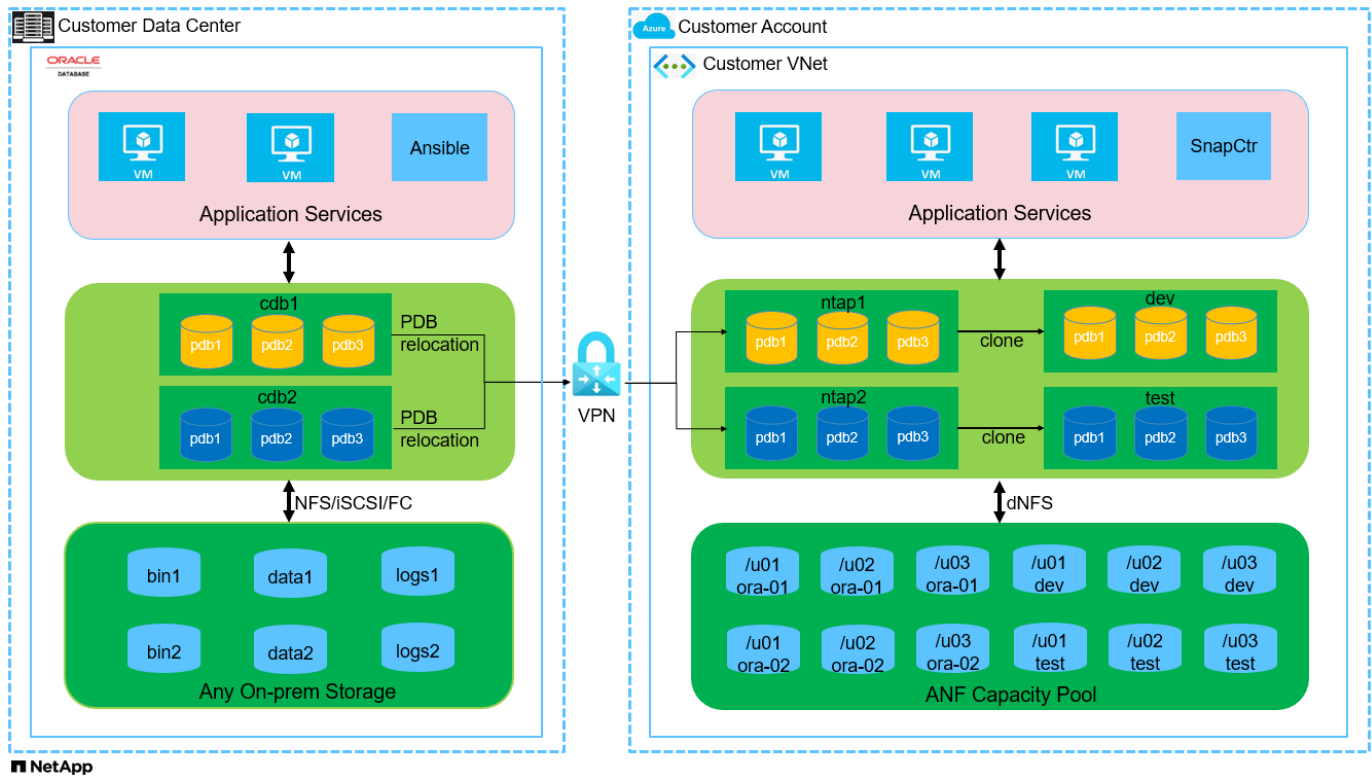
이 솔루션은 다음과 같은 사람들을 위해 만들어졌습니다.

- Azure NetApp Files 에 Oracle을 배포하려는 DBA입니다.
- Azure NetApp Files 에서 Oracle 워크로드를 테스트하려는 데이터베이스 솔루션 아키텍트입니다.
- Azure NetApp Files 에서 Oracle 데이터베이스를 배포하고 관리하려는 스토리지 관리자입니다.
- Azure NetApp Files 에 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.

아키텍처



하드웨어 및 소프트웨어 구성 요소

하드웨어		
Azure NetApp Files	Microsoft에서 Azure에 제공하는 현재 서비스	프리미엄 서비스 수준을 갖춘 1개의 용량 풀
DB 서버용 Azure VM	Standard_B4ms - 4개 vCPU, 16GiB	동시 배포를 위한 두 개의 Linux 가상 머신 인스턴스
SnapCenter 용 Azure VM	Standard_B4ms - 4개 vCPU, 16GiB	하나의 Windows 가상 머신 인스턴스
소프트웨어		
레드햇 리눅스	RHEL Linux 8.6(LVM) - x64 Gen2	테스트를 위해 RedHat 구독을 배포했습니다.
윈도우 서버	2022 DataCenter; Azure Edition 핫패치 - x64 Gen2	SnapCenter 서버 호스팅
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 5.0	작업 그룹 배포
JDK를 엽니다	버전 java-11-openjdk	DB VM의 SnapCenter 플러그인 요구 사항

NFS	버전 3.0	Oracle dNFS 활성화됨
앤서블	코어 2.16.2	파이썬 3.6.8

랩 환경에서의 Oracle 데이터베이스 구성

섬기는 사람	데이터 베이스	DB 스토리지
오라-01	NTAP1(NTAP1_PDB1, NTAP1_PDB2, NTAP1_PDB3)	/u01, /u02, /u03 ANF 용량 풀에 NFS 마운트
오라-02	NTAP2(NTAP2_PDB1, NTAP2_PDB2, NTAP2_PDB3)	/u01, /u02, /u03 ANF 용량 풀에 NFS 마운트

배포 고려사항의 핵심 요소

- 오라클 데이터베이스 저장 레이아웃. 이 자동화된 Oracle 배포에서는 기본적으로 Oracle 바이너리, 데이터 및 로그를 호스팅하기 위해 각 데이터베이스에 대해 3개의 데이터베이스 볼륨을 프로비저닝합니다. 볼륨은 NFS를 통해 Oracle DB 서버에 /u01 - 바이너리, /u02 - 데이터, /u03 - 로그로 마운트됩니다. 이중 제어 파일은 중복성을 위해 /u02 및 /u03 마운트 지점에 구성됩니다.
- 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. 다른 데이터베이스 인스턴스 ID(Oracle SID)로 배포를 반복하여 여러 컨테이너 데이터베이스를 단일 VM 인스턴스에 배포할 수 있습니다. 하지만 배포된 데이터베이스를 지원할 만큼 호스트에 충분한 메모리가 있는지 확인하세요.
- dNFS 구성. dNFS(Oracle 11g부터 사용 가능)를 사용하면 Azure Virtual Machine에서 실행되는 Oracle 데이터베이스가 기본 NFS 클라이언트보다 훨씬 더 많은 I/O를 처리할 수 있습니다. 자동화된 Oracle 배포는 기본적으로 NFSv3에서 dNFS를 구성합니다.
- 배포 속도를 높이려면 대용량 볼륨을 할당하세요. ANF 파일 시스템 IO 처리량은 볼륨 크기에 따라 조절됩니다. 초기 배포 시에는 대용량 볼륨을 할당하면 배포 속도를 높일 수 있습니다. 이후 볼륨은 애플리케이션에 영향을 주지 않고 동적으로 축소될 수 있습니다.
- 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 NFS를 통해 Azure VM에 직접 탑재된 데이터베이스 볼륨을 사용하여 Azure NetApp Files 에서 Oracle 19c를 자동으로 배포하고 데이터베이스를 마이그레이션하는 단계별 절차를 제공합니다.

배포를 위한 전제 조건

배포에는 다음과 같은 전제 조건이 필요합니다.

1. Azure 계정이 설정되었고, Azure 계정 내에 필요한 VNet 및 네트워크 세그먼트가 생성되었습니다.
2. Azure 클라우드 포털에서 Azure Linux VM을 Oracle DB 서버로 배포합니다. Oracle 데이터베이스에 대한 Azure NetApp Files 용량 풀과 데이터베이스 볼륨을 만듭니다. azureuser가 DB 서버에 대해 VM SSH 개인 /공개 키 인증을 활성화합니다. 환경 설정에 대한 자세한 내용은 이전 섹션의 아키텍처 다이어그램을 참조하세요. 또한 참조됨 ["Azure VM 및 Azure NetApp Files 대한 단계별 Oracle 배포 절차"](#) 자세한 내용은.



로컬 디스크 중복성을 사용하여 배포된 Azure VM의 경우 Oracle 설치 파일을 준비하고 OS 스왑 파일을 추가할 수 있는 충분한 공간을 확보하기 위해 VM 루트 디스크에 최소 128G를 할당했는지 확인하세요. /tmplv와 /rootlv OS 파티션을 그에 맞게 확장합니다. rootvg-homelv에 여유 공간이 1G 미만이면 1G를 추가합니다. 데이터베이스 볼륨 이름이 VMname-u01, VMname-u02, VMname-u03 규칙을 따르는지 확인하세요.

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

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

```
sudo lvresize -r -L +1G /dev/mapper/rootvg-homelv
```

3. Azure 클라우드 포털에서 Windows 서버를 프로비저닝하여 최신 버전의 NetApp SnapCenter UI 도구를 실행합니다. 자세한 내용은 다음 링크를 참조하세요. ["SnapCenter 서버 설치"](#)
4. 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.



Ansible 컨트롤러 노드는 ssh 포트를 통해 Azure DB VM에 연결할 수 있는 한 온프레미스 또는 Azure 클라우드에 위치할 수 있습니다.

5. NFS용 NetApp Oracle 배포 자동화 툴킷의 사본을 복제합니다.

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

6. Azure DB VM /tmp/archive 디렉터리에 777 권한이 있는 Oracle 19c 설치 파일을 다음 단계로 진행합니다.

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

7. 다음 영상을 시청해보세요:

[NFS를 사용한 Azure NetApp Files 에서의 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 - 지정된 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 파일을 구성하는 방법을 보여줍니다.

매개변수 파일 구성

1. Ansible 대상 hosts 파일 구성:

```
# 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 user configuration variables
#####
##### Consolidate all variables from ONTAP, linux and oracle
#####
#####
#####

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

# Prerequisite to create three volumes in NetApp ONTAP storage from
System Manager or cloud dashboard with following naming convention:
# db_hostname_u01 - Oracle binary
# db_hostname_u02 - Oracle data
# db_hostname_u03 - Oracle redo
# It is important to strictly follow the name convention or the
automation will fail.

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

redhat_sub_username: XXXXXXXXX
redhat_sub_password: XXXXXXXXX

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

# Database domain name
db_domain: solutions.netapp.com

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

3. 로컬 DB 서버 host_vars/host_name.yml ora_01.yml, ora_02.yml과 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

# Local NFS lif ip address to access database volumes
nfs_lif: 172.30.136.68
```

플레이북 실행

자동화 툴킷에는 총 5개의 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.
4-oracle_config.yml - install and configure Oracle on DB servers and
create a container database.
5-destroy.yml - optional to undo the environment to dismantle all.
```

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 azureuser -e
@vars/vars.yml
```

2. 1~4까지의 숫자 순서대로 플레이북을 하나씩 실행합니다.

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

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

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

3. 태그와 함께 0-all_playbook.yml을 실행합니다.

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

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

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

4. 환경 실행 취소

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

실행 후 검증

플레이북을 실행한 후 Oracle DB 서버 VM에 로그인하여 Oracle이 설치 및 구성되었고 컨테이너 데이터베이스가 성공적으로 생성되었는지 확인합니다. 다음은 호스트 ora-01에서 Oracle 데이터베이스 검증의 예입니다.

1. NFS 마운트 검증

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

#
# /etc/fstab
# Created by anaconda on Thu Sep 14 11:04:01 2023
#
# Accessible filesystems, by reference, are maintained under
# '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for
# more info.
#
# After editing this file, run 'systemctl daemon-reload' to update
# systemd
# units generated from this file.
#
/dev/mapper/rootvg-rootlv / xfs defaults
0 0
UUID=268633bd-f9bb-446d-9ald-8fca4609ale1 /boot
xfs defaults 0 0
UUID=89D8-B037 /boot/efi vfat
defaults,uid=0,gid=0,umask=077,shortname=winnt 0 2
/dev/mapper/rootvg-homelv /home xfs defaults
0 0
/dev/mapper/rootvg-tmplv /tmp xfs defaults
0 0
/dev/mapper/rootvg-usrlv /usr xfs defaults
0 0
/dev/mapper/rootvg-varlv /var xfs defaults
0 0
/mnt/swapfile swap swap defaults 0 0
172.30.136.68:/ora-01-u01 /u01 nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsiz=65536,wsiz=65536 0 0
172.30.136.68:/ora-01-u02 /u02 nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsiz=65536,wsiz=65536 0 0
172.30.136.68:/ora-01-u03 /u03 nfs
rw,bg,hard,vers=3,proto=tcp,timeo=600,rsiz=65536,wsiz=65536 0 0

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

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

2. Oracle 리스너 검증

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

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

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

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

```

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

```

3. Oracle 데이터베이스 및 dNFS 검증

```

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

```

```

instance.

# A colon, ':', is used as the field terminator.  A new line
terminates
# the entry.  Lines beginning with a pound sign, '#', are comments.
#
# Entries are of the form:
#   $ORACLE_SID:$ORACLE_HOME:<N|Y>:
#
# The first and second fields are the system identifier and home
# directory of the database respectively.  The third field indicates
# to the dbstart utility that the database should , "Y", or should
not,
# "N", be brought up at system boot time.
#
# Multiple entries with the same $ORACLE_SID are not allowed.
#
#
NTAP1:/u01/app/oracle/product/19.0.0/NTAP1:Y

```

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

```

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Feb 1 16:37:51 2024
Version 19.18.0.0.0

```

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

```
Connected to:
```

```

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

```

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

NAME	OPEN_MODE	LOG_MODE
NTAP1	READ WRITE	ARCHIVELOG

```
SQL> show pdbs
```

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

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

```
NAME
```

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

```
NAME
```

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

```
19 rows selected.
```

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

```
NAME
```

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

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

```
MEMBER
```

```
-----
-----
/u03/orareco/NTAP1/onlineelog/redo03.log
```

```
/u03/orareco/NTAP1/onlineelog/redo02.log
```

```
/u03/orareco/NTAP1/onlineelog/redo01.log
```

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

```
SVRNAME
```

```
-----  
-----
```

```
DIRNAME
```

```
-----  
-----
```

```
NFSVERSION
```

```
-----  
172.30.136.68
```

```
/ora-01-u02
```

```
NFSv3.0
```

```
172.30.136.68
```

```
/ora-01-u03
```

```
NFSv3.0
```

```
SVRNAME
```

```
-----  
-----
```

```
DIRNAME
```

```
-----  
-----
```

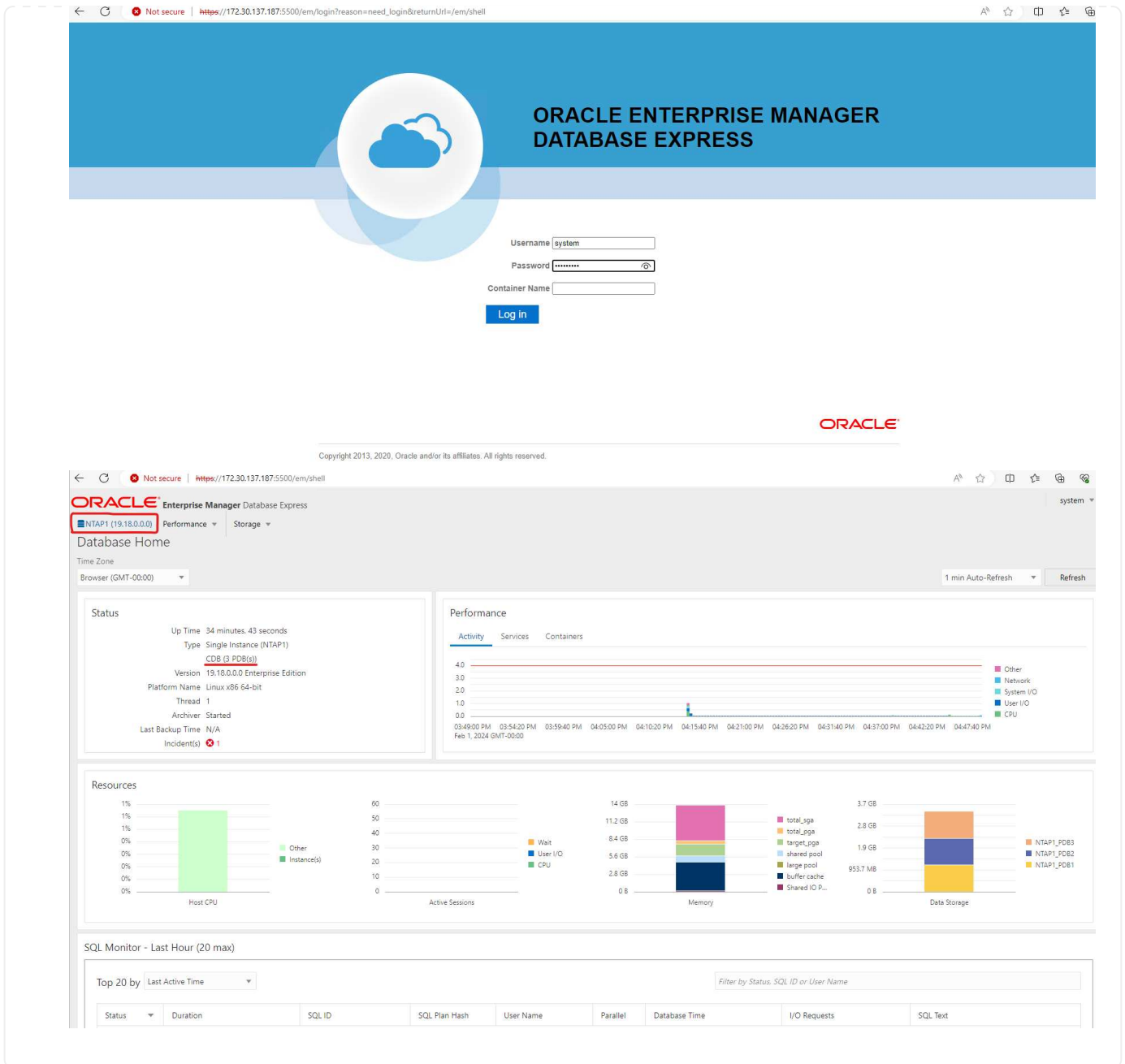
```
NFSVERSION
```

```
-----  
172.30.136.68
```

```
/ora-01-u01
```

```
NFSv3.0
```

4. Oracle Enterprise Manager Express에 로그인하여 데이터베이스를 검증하세요.



Oracle 데이터베이스를 Azure로 마이그레이션

온프레미스에서 클라우드로 Oracle 데이터베이스를 마이그레이션하는 것은 힘든 일입니다. 올바른 전략과 자동화를 활용하면 프로세스를 원활하게 진행하고 서비스 중단과 가동 중지 시간을 최소화할 수 있습니다. 이 자세한 지침을 따르세요 ["온프레미스에서 Azure 클라우드로 데이터베이스 마이그레이션"](#) 귀하의 데이터베이스 마이그레이션 여정을 안내합니다.

SnapCenter 사용한 Oracle 백업, 복원 및 복제

NetApp Azure 클라우드에 배포된 Oracle 데이터베이스를 관리하기 위해 SnapCenter UI 도구를 권장합니다. TR-4988을 참조하세요. "[SnapCenter 사용한 ANF에서의 Oracle 데이터베이스 백업, 복구 및 복제](#)" 자세한 내용은.

추가 정보를 찾을 수 있는 곳

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

- SnapCenter 사용한 ANF에서의 Oracle 데이터베이스 백업, 복구 및 복제

["SnapCenter 사용한 ANF에서의 Oracle 데이터베이스 백업, 복구 및 복제"](#)

- Azure NetApp Files

["https://azure.microsoft.com/en-us/products/netapp"](https://azure.microsoft.com/en-us/products/netapp)

- Oracle Direct NFS 배포

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

- 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)

저작권 정보

Copyright © 2025 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.