



TR-5003: ANF에 대한 고처리량 Oracle VLDB 구현

NetApp database solutions

NetApp
August 18, 2025

목차

TR-5003: ANF에 대한 고처리량 Oracle VLDB 구현	1
목적	1
대상	1
솔루션 테스트 및 검증 환경	1
아키텍처	2
하드웨어 및 소프트웨어 구성 요소	2
시뮬레이션된 NY-LA DR 설정을 사용한 Oracle VLDB Data Guard 구성	3
배포 고려사항의 핵심 요소	3
솔루션 구축	3
배포를 위한 전제 조건	4
Data Guard를 위한 기본 Oracle VLDB 구성	7
Data Guard를 위한 대기 Oracle VLDB 구성	16
Data Guard Broker 설정	25
자동화를 통해 다른 사용 사례에 대한 대기 데이터베이스 복제	28
추가 정보를 찾을 수 있는 곳	29

TR-5003: ANF에 대한 고처리량 Oracle VLDB 구현

Allen Cao, Niyaz Mohamed, NetApp

이 솔루션은 Azure 클라우드에서 Oracle Data Guard를 사용하여 Microsoft Azure NetApp Files (ANF)에서 고처리량 Oracle Very Large Database(VLDB)를 구성하는 데 대한 개요와 세부 정보를 제공합니다.

목적

높은 처리량과 미션 크리티컬 Oracle VLDB는 백엔드 데이터베이스 스토리지에 큰 부담을 줍니다. 서비스 수준 계약(SLA)을 충족하려면 데이터베이스 저장소는 밀리초 미만의 지연 시간 성능을 유지하면서 필요한 용량과 초당 높은 입출력 작업 수(IOPS)를 제공해야 합니다. 특히 공유 스토리지 리소스 환경이 있는 퍼블릭 클라우드에 이러한 데이터베이스 워크로드를 배포하는 경우 이는 특히 어렵습니다. 모든 저장 플랫폼이 동일하게 만들어진 것은 아닙니다. Azure 인프라와 결합된 프리미엄 Azure NetApp Files 스토리지는 이처럼 매우 까다로운 Oracle 워크로드의 요구 사항을 충족할 수 있습니다. 검증된 성능 벤치마크에서 ("[Azure NetApp Files 여러 볼륨에서 Oracle 데이터베이스 성능](#)"), ANF는 SLOB 도구를 통해 합성 100% 임의 선택 워크로드에서 700마이크로초 지연 시간으로 250만 읽기 IOPS를 제공했습니다. 표준 8k 블록 크기를 사용하면 처리량은 약 20GiB/s로 변환됩니다.

이 문서에서는 여러 NFS 볼륨과 스토리지 부하 분산을 위한 Oracle ASM을 사용하여 ANF 스토리지에 Data Guard 구성을 사용하여 Oracle VLDB를 설정하는 방법을 설명합니다. 대기 데이터베이스는 스냅샷을 통해 빠르게(분) 백업되고 원하는 사용 사례에 맞게 읽기/쓰기 액세스를 위해 복제될 수 있습니다. NetApp 솔루션 엔지니어링 팀은 사용자가 정의한 일정에 따라 쉽게 복제본을 만들고 새로 고칠 수 있는 자동화 툴킷을 제공합니다.

이 솔루션은 다음과 같은 사용 사례를 해결합니다.

- Azure 지역 전체의 Microsoft Azure NetApp Files 저장소에서 Data Guard 설정으로 Oracle VLDB를 구현합니다.
- 자동화를 통해 보고, 개발, 테스트 등의 사용 사례에 맞게 물리적 스탠바이 데이터베이스를 스냅샷 백업하고 복제합니다.

대상

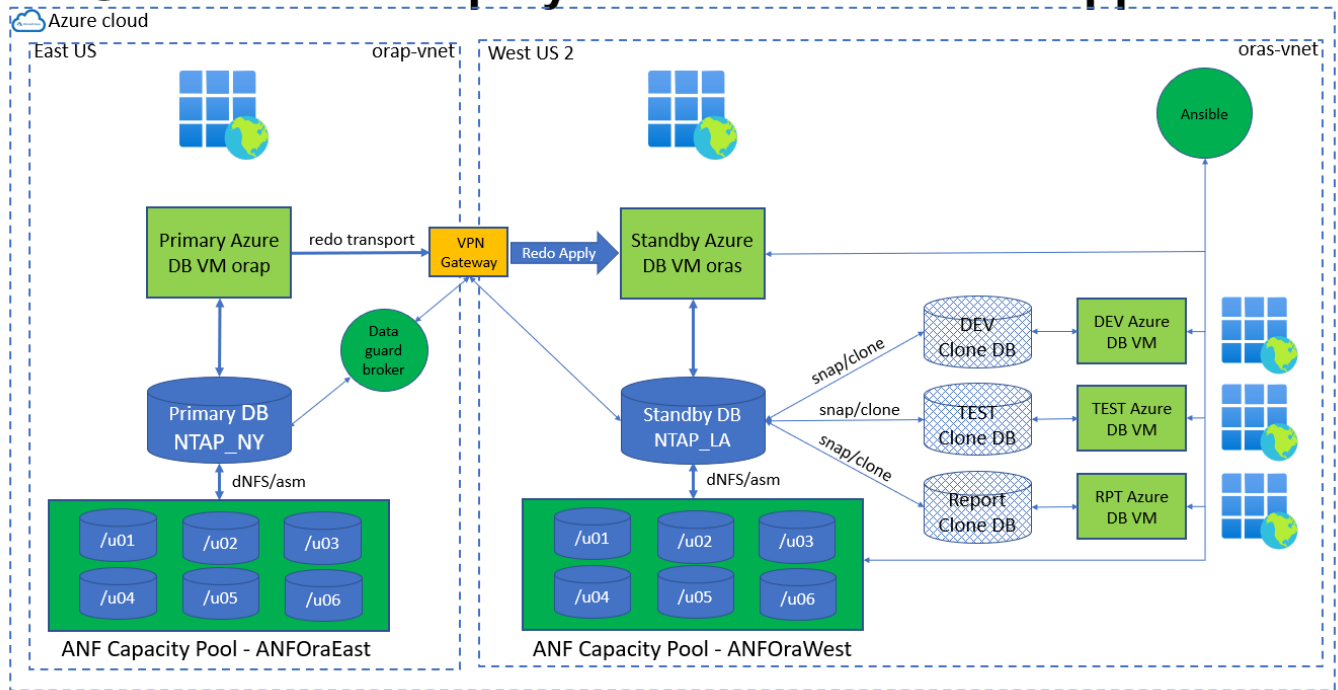
이 솔루션은 다음과 같은 사람들을 위해 만들어졌습니다.

- 고가용성, 데이터 보호 및 재해 복구를 위해 Azure 클라우드에서 Data Guard와 함께 Oracle VLDB를 설정하는 DBA입니다.
- Azure 클라우드에서 Data Guard 구성을 갖춘 Oracle VLDB에 관심이 있는 데이터베이스 솔루션 아키텍트입니다.
- Oracle 데이터베이스를 지원하는 Azure NetApp Files 스토리지를 관리하는 스토리지 관리자입니다.
- Azure 클라우드 환경에서 Data Guard와 함께 Oracle VLDB를 구축하고 싶어하는 애플리케이션 소유자입니다.

솔루션 테스트 및 검증 환경

이 솔루션의 테스트와 검증은 실제 사용자 배포 환경과 일치하지 않을 수 있는 Azure 클라우드 랩 환경에서 수행되었습니다. 자세한 내용은 다음 섹션을 참조하세요. [배포 고려사항의 핵심 요소](#).

Oracle VLDB Deployment with Azure NetApp Files



하드웨어 및 소프트웨어 구성 요소

하드웨어		
Azure NetApp Files	Microsoft에서 제공하는 현재 버전	2개의 4TiB 용량 풀, 프리미엄 서비스 수준, 자동 QoS
DB 서버용 Azure VM	표준 B4ms(4개 vcpus, 16GiB 메모리)	3개의 DB VM(하나는 기본 DB 서버, 하나는 대기 DB 서버, 세 번째는 복제 DB 서버)
소프트웨어		
레드햇 리눅스	Red Hat Enterprise Linux 8.6(LVM) - x64 Gen2	테스트를 위해 RedHat 구독을 배포했습니다.
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.
dNFS OneOff 패치	p32931941_190000_Linux-x86-64.zip	그리드와 데이터베이스 모두에 적용됨
오라클 OPatch	버전 12.2.0.1.36	최신 패치 p6880880_190000_Linux-x86-64.zip
앤서블	버전 코어 2.16.2	파이썬 버전 - 3.10.13

NFS	버전 3.0	Oracle에 대해 dNFS 활성화됨
-----	--------	----------------------

시뮬레이션된 NY-LA DR 설정을 사용한 Oracle VLDB Data Guard 구성

데이터 베이스	DB_고유_이름	Oracle Net 서비스 이름
주요한	NTAP_NY	NTAP_NY.internal.cloudapp.net
대기	NTAP_LA	NTAP_LA.internal.cloudapp.net

배포 고려사항의 핵심 요소

- *** Azure NetApp Files 구성.** * Azure NetApp Files Azure NetApp 저장소 계정에 다음과 같이 할당됩니다.
Capacity Pools. 이러한 테스트와 검증에서 우리는 동부 지역에 Oracle 기본 데이터베이스를 호스팅하기 위해 2TiB 용량 풀을 배포했고, 서부 2 지역에 대기 데이터베이스와 DB 복제본을 호스팅하기 위해 4TiB 용량 풀을 배포했습니다. ANF 용량 풀에는 표준, 프리미엄, 울트라 세 가지 서비스 수준이 있습니다. ANF 용량 풀의 IO 용량은 용량 풀의 크기와 서비스 수준에 따라 결정됩니다. 용량 풀을 생성할 때 QoS를 자동 또는 수동으로 설정하고 저장 시 데이터 암호화를 단일 또는 이중으로 설정할 수 있습니다.
- 데이터베이스 볼륨 크기 조정. 실제 운영에 배포하는 경우 NetApp Oracle AWR 보고서에서 Oracle 데이터베이스 처리량 요구 사항을 전체적으로 평가할 것을 권장합니다. 데이터베이스의 ANF 볼륨 크기를 조정할 때 데이터베이스 크기와 처리량 요구 사항을 모두 고려하세요. ANF에 대한 자동 QoS 구성을 사용하면 Ultra Service Level로 할당된 TiB 볼륨 용량당 128MiB/s의 대역폭이 보장됩니다. 처리량이 높아지면 요구 사항을 충족하기 위해 더 큰 볼륨 크기가 필요할 수 있습니다.
- 단일 권 또는 여러 권. QoS는 볼륨 크기 및 용량 풀 서비스 수준에 따라 엄격하게 적용되므로 단일 대용량 볼륨은 동일한 총 크기를 가진 여러 볼륨과 유사한 성능 수준을 제공할 수 있습니다. Oracle VLDB에 대해 공유 백엔드 ANF 스토리지 리소스 풀을 보다 효율적으로 활용하기 위해 여러 볼륨(여러 NFS 마운트 포인트)을 구현하는 것이 좋습니다. 여러 NFS 볼륨에서 IO 부하 분산을 위해 Oracle ASM을 구현합니다.
- 애플리케이션 볼륨 그룹. 성능 최적화를 위해 Oracle용 AVG(Application Volume Group)를 배포합니다. 애플리케이션 볼륨 그룹별로 배포된 볼륨은 애플리케이션 VM의 대기 시간과 처리량을 최적화하기 위해 지역 또는 영역 인프라에 배치됩니다.
- **Azure VM** 고려 사항. 이러한 테스트와 검증에서는 4개의 vCPU와 16GiB 메모리를 갖춘 Azure VM-Standard_B4ms를 사용했습니다. 높은 처리량 요구 사항을 갖춘 Oracle VLDB에 적합한 Azure DB VM을 선택해야 합니다. vCPU 수와 RAM 용량 외에도 VM 네트워크 대역폭(수신 및 송신 또는 NIC 처리량 제한)이 데이터베이스 저장 용량에 도달하기 전에 병목 현상이 발생할 수 있습니다.
- **dNFS** 구성. dNFS를 사용하면 ANF 스토리지가 있는 Azure Virtual Machine에서 실행되는 Oracle 데이터베이스는 기본 NFS 클라이언트보다 훨씬 더 많은 I/O를 처리할 수 있습니다. 잠재적인 버그를 해결하기 위해 Oracle dNFS 패치 p32931941이 적용되었는지 확인하세요.

솔루션 구축

Oracle Data Guard를 설정하기 위한 시작점으로 VNet 내의 Azure 클라우드 환경에 기본 Oracle 데이터베이스가 이미 배포되어 있다고 가정합니다. 이상적으로 기본 데이터베이스는 NFS 마운트가 있는 ANF 스토리지에 배포됩니다. 기본 Oracle 데이터베이스는 NetApp ONTAP 스토리지가 Azure 에코시스템 또는 프라이빗 데이터 센터 내의 다른 스토리지에서 실행할 수도 있습니다. 다음 섹션에서는 ANF 스토리지가 있는 Azure의 기본 Oracle DB와 ANF 스토리지가 있는 Azure의 물리적 대기 Oracle DB 사이에서 Oracle Data Guard 설정의 ANF 기반 Oracle VLDB에 대한 구성을 보여줍니다.

배포를 위한 전제 조건

배포에는 다음과 같은 전제 조건이 필요합니다.

1. Azure 클라우드 계정이 설정되었고, Azure 계정 내에 필요한 VNet 및 네트워크 서브넷이 생성되었습니다.
2. Azure 클라우드 포털 콘솔에서 최소 3개의 Azure Linux VM을 배포해야 합니다. 하나는 기본 Oracle DB 서버, 다른 하나는 대기 Oracle DB 서버로, 나머지 하나는 보고, 개발, 테스트 등을 위한 복제 대상 DB 서버로 사용해야 합니다. 환경 설정에 대한 자세한 내용은 이전 섹션의 아키텍처 다이어그램을 참조하세요. 또한 Microsoft를 검토하세요 ["Azure 가상 머신"](#) 자세한 내용은.
3. 기본 Oracle 데이터베이스는 기본 Oracle DB 서버에 설치 및 구성되어야 합니다. 반면, 스탠바이 Oracle DB 서버나 복제 Oracle DB 서버에서는 Oracle 소프트웨어만 설치되고 Oracle 데이터베이스는 생성되지 않습니다. 이상적으로는 Oracle 파일 디렉토리 레이아웃이 모든 Oracle DB 서버에서 정확히 일치해야 합니다. Azure 클라우드와 ANF에서 자동화된 Oracle 배포에 대한 NetApp 권장 사항에 대한 자세한 내용은 다음 기술 보고서를 참조하세요.

◦ ["TR-4987: NFS를 사용한 Azure NetApp Files 에서의 간소화되고 자동화된 Oracle 배포"](#)



Oracle 설치 파일을 준비할 충분한 공간을 확보하려면 Azure VM 루트 볼륨에 최소 128G를 할당했는지 확인하세요.

4. Azure 클라우드 포털 콘솔에서 두 개의 ANF 스토리지 용량 풀을 배포하여 Oracle 데이터베이스 볼륨을 호스팅합니다. ANF 스토리지 용량 풀은 실제 DataGuard 구성을 모방하기 위해 서로 다른 지역에 위치해야 합니다. ANF 스토리지 배포에 익숙하지 않은 경우 설명서를 참조하세요. ["빠른 시작: Azure NetApp Files 설정 및 NFS 볼륨 생성"](#) 단계별 지침을 확인하세요.

The screenshot shows the Azure portal interface for 'Azure NetApp Files'. It displays a table with two records for NetApp accounts. The table has columns for Name, Type, Resource group, Location, and Subscription. The records are for 'ANFOraEast' and 'ANFOraWest'.

Name	Type	Resource group	Location	Subscription
ANFOraEast	NetApp account	ANFAVSRG	East US	Hybrid Cloud TME Onprem
ANFOraWest	NetApp account	ANFAVSRG	West US 2	Hybrid Cloud TME Onprem

5. 기본 Oracle 데이터베이스와 대기 Oracle 데이터베이스가 서로 다른 두 지역에 있는 경우, 두 개의 별도 VNet 간에 데이터 트래픽 흐름을 허용하도록 VPN 게이트웨이를 구성해야 합니다. Azure의 자세한 네트워킹 구성은 이 문서의 범위를 벗어납니다. 다음 스크린샷은 VPN 게이트웨이가 어떻게 구성되고, 연결되고, 실험실에서 데이터 트래픽 흐름이 확인되는지에 대한 몇 가지 참고 자료를 제공합니다.

랩 VPN 게이트웨이

:

The screenshot shows the Azure portal interface for 'Virtual network gateways'. It displays a table with three records for VPN gateways. The table has columns for Name, Virtual network, Gateway type, Resource group, Location, and Subscription. The records are for 'orap-vnet-gw', 'oras-vnet-gw', and 'vNetgw'.

Name	Virtual network	Gateway type	Resource group	Location	Subscription
orap-vnet-gw	orap-vnet	Vpn	ANFAVSRG	East US	Hybrid Cloud TME Onprem
oras-vnet-gw	oras-vnet	Vpn	ANFAVSRG	West US 2	Hybrid Cloud TME Onprem
vNetgw	EHCVnet	Vpn	NSOL	Central US	Hybrid Cloud TME Onprem

기본 vnet 게이트웨이

:

The screenshot shows the Azure portal interface for a Virtual network gateway named 'orap-vnet-gw'. The left sidebar contains navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Configuration, Connections, Point-to-site configuration, NAT Rules, Maintenance, Properties, Locks, Monitoring, Automation, and Help. The main content area displays the gateway's details, including its location (East US) and subscription ID. Below this, there are sections for Health check, Advisor Recommendations, and Advanced troubleshooting. At the bottom, two line graphs show 'Total tunnel ingress' and 'Total tunnel egress' traffic over a 30-day period.

Vnet 게이트웨이 연결 상태

:

The screenshot shows the 'Connections' page for the 'orap-vnet-gw' Virtual network gateway. The left sidebar is the same as the previous screenshot. The main content area displays a table of connections. The table has columns for Name, Status, Connection type, and Peer. There are two connections listed: 'orap-to-oras' and 'oras-to-orap', both with a status of 'Connected' and a connection type of 'VNet-to-VNet'.

트래픽 흐름이 설정되었는지 확인합니다(세 개의 점을 클릭하여 페이지를 엽니다).

The screenshot shows the 'Overview' page for the 'orap-to-oras' Connection. The left sidebar contains navigation options like Overview, Activity log, Access control (IAM), Tags, Settings, Monitoring, Automation, and Help. The main content area displays the connection's details, including its location (East US) and subscription ID. Below this, there are sections for Activity log, Access control (IAM), Tags, Settings, Monitoring, Automation, and Help.

- 이 문서를 참조하세요"Oracle용 애플리케이션 볼륨 그룹 배포" Oracle용 Application Volume Group을 배포합니다.

Data Guard를 위한 기본 **Oracle VLDB** 구성

이 데모에서는 6개의 NFS 탑재 지점을 사용하여 기본 Azure DB 서버에 NTAP라는 기본 Oracle 데이터베이스를 설정했습니다. /u01은 Oracle 바이너리용, /u02, /u04, /u05, /u06은 Oracle 데이터 파일 및 Oracle 제어 파일용, /u03은 Oracle 활성 로그, 보관 로그 파일 및 중복 Oracle 제어 파일용입니다. 이 설정은 참조 구성으로 사용됩니다. 실제 배포 시에는 용량 풀 크기, 서비스 수준, 데이터베이스 볼륨 수, 각 볼륨 크기 등 특정 요구 사항과 필요 사항을 고려해야 합니다.

ASM을 사용하여 NFS에 Oracle Data Guard를 설정하는 방법에 대한 자세한 단계별 절차는 다음을 참조하세요. ["TR-5002- Azure NetApp Files 통한 Oracle Active Data Guard 비용 절감"](#) 그리고 ["TR-4974- NFS/ASM을 사용하는 AWS FSx/EC2에서 독립 실행형 재시작 시 Oracle 19c"](#) 관련 섹션. TR-4974의 절차는 Amazon FSx ONTAP 에서 검증되었지만 ANF에도 동일하게 적용할 수 있습니다. 다음은 Data Guard 구성에서 기본 Oracle VLDB의 세부 정보를 보여줍니다.

1. 기본 Azure DB 서버 orap.internal.cloudapp.net의 기본 데이터베이스 NTAP는 처음에는 NFS의 ANF와 데이터베이스 저장소인 ASM과 함께 독립 실행형 데이터베이스로 배포됩니다.

```
orap.internal.cloudapp.net:
resource group: ANFAVSRG
Location: East US
size: Standard B4ms (4 vcpus, 16 GiB memory)
OS: Linux (redhat 8.6)
pub_ip: 172.190.207.231
pri_ip: 10.0.0.4

[oracle@orap ~]$ df -h
Filesystem                                Size  Used Avail Use% Mounted on
devtmpfs                                  7.7G   0    7.7G   0% /dev
tmpfs                                     7.8G  1.1G   6.7G  15% /dev/shm
tmpfs                                     7.8G   17M   7.7G   1% /run
tmpfs                                     7.8G   0    7.8G   0% /sys/fs/cgroup
/dev/mapper/rootvg-rootlv                 22G   20G   2.1G  91% /
/dev/mapper/rootvg-usrlv                  10G   2.3G   7.8G  23% /usr
/dev/sda1                                 496M  181M  315M  37% /boot
/dev/mapper/rootvg-varlv                   8.0G   1.1G   7.0G  13% /var
/dev/sda15                                495M   5.8M  489M   2% /boot/efi
/dev/mapper/rootvg-homelv                  2.0G    47M   2.0G   3% /home
/dev/mapper/rootvg-tmplv                   12G    11G   1.9G  85% /tmp
/dev/sdb1                                 32G    49M   30G   1% /mnt
10.0.2.38:/orap-u06                       300G  282G   19G  94% /u06
10.0.2.38:/orap-u04                       300G  282G   19G  94% /u04
10.0.2.36:/orap-u01                       400G   21G  380G   6% /u01
10.0.2.37:/orap-u02                       300G  282G   19G  94% /u02
10.0.2.36:/orap-u03                       400G  282G  119G  71% /u03
10.0.2.39:/orap-u05                       300G  282G   19G  94% /u05

[oracle@orap ~]$ 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
NTAP:/u01/app/oracle/product/19.0.0/NTAP:N

```

2. Oracle 사용자로 기본 DB 서버에 로그인합니다. 그리드 구성을 검증합니다.

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

```

[oracle@orap ~]$ $GRID_HOME/bin/crsctl stat res -t
-----
-----
Name          Target  State          Server          State
details
-----
-----
Local Resources
-----
-----
ora.DATA.dg
          ONLINE  ONLINE        orap            STABLE
ora.LISTENER.lsnr
          ONLINE  ONLINE        orap            STABLE
ora.LOGS.dg
          ONLINE  ONLINE        orap            STABLE
ora.asm
          ONLINE  ONLINE        orap
Started, STABLE
ora.ons
          OFFLINE OFFLINE        orap            STABLE
-----
-----
Cluster Resources
-----
-----
ora.cssd
      1      ONLINE  ONLINE        orap            STABLE
ora.diskmon
      1      OFFLINE OFFLINE
ora.evmd
      1      ONLINE  ONLINE        orap            STABLE
ora.ntap.db
      1      OFFLINE OFFLINE
Instance Shutdown, ST
          ABLE
-----
-----
[oracle@orap ~]$

```

3. ASM 디스크 그룹 구성.

```
asmcmd
```

```

[oracle@orap ~]$ 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
1146880    1136944      0       1136944      0
N  DATA/
MOUNTED    EXTERN    N       512      512      4096      4194304
286720     283312      0       283312      0
N  LOGS/
ASMCMD> lsdk
Path
/u02/oradata/asm/orap_data_disk_01
/u02/oradata/asm/orap_data_disk_02
/u02/oradata/asm/orap_data_disk_03
/u02/oradata/asm/orap_data_disk_04
/u03/oralogs/asm/orap_logs_disk_01
/u03/oralogs/asm/orap_logs_disk_02
/u03/oralogs/asm/orap_logs_disk_03
/u03/oralogs/asm/orap_logs_disk_04
/u04/oradata/asm/orap_data_disk_05
/u04/oradata/asm/orap_data_disk_06
/u04/oradata/asm/orap_data_disk_07
/u04/oradata/asm/orap_data_disk_08
/u05/oradata/asm/orap_data_disk_09
/u05/oradata/asm/orap_data_disk_10
/u05/oradata/asm/orap_data_disk_11
/u05/oradata/asm/orap_data_disk_12
/u06/oradata/asm/orap_data_disk_13
/u06/oradata/asm/orap_data_disk_14
/u06/oradata/asm/orap_data_disk_15
/u06/oradata/asm/orap_data_disk_16
ASMCMD>

```

4. 기본 DB의 Data Guard에 대한 매개변수 설정.

```

SQL> show parameter name

```

NAME	TYPE	VALUE
-----	-----	

cdb_cluster_name	string	
cell_offloadgroup_name	string	

db_file_name_convert	string	
db_name	string	NTAP
db_unique_name	string	NTAP_NY
global_names	boolean	FALSE
instance_name	string	NTAP
lock_name_space	string	
log_file_name_convert	string	
pdb_file_name_convert	string	
processor_group_name	string	

NAME	TYPE	VALUE
------	------	-------

service_names	string	
NTAP_NY.internal.cloudapp.net		

SQL> sho parameter log_archive_dest

NAME	TYPE	VALUE
------	------	-------

log_archive_dest	string	
log_archive_dest_1	string	
LOCATION=USE_DB_RECOVERY_FILE_		DEST
VALID_FOR=(ALL_LOGFILES,A		LL_ROLES)
DB_UNIQUE_NAME=NTAP_		NY

log_archive_dest_10	string	
log_archive_dest_11	string	
log_archive_dest_12	string	
log_archive_dest_13	string	
log_archive_dest_14	string	
log_archive_dest_15	string	

NAME	TYPE	VALUE
------	------	-------

log_archive_dest_16	string	
log_archive_dest_17	string	
log_archive_dest_18	string	
log_archive_dest_19	string	
log_archive_dest_2	string	SERVICE=NTAP_LA
ASYNCR_VALID_FO		

```
R=(ONLINE_LOGFILES,PRIMARY_ROL
E)

DB_UNIQUE_NAME=NTAP_LA
log_archive_dest_20          string
log_archive_dest_21          string
log_archive_dest_22          string
```

5. 기본 DB 구성.

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

NAME	OPEN_MODE	LOG_MODE
NTAP	READ WRITE	ARCHIVELOG

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	NTAP_PDB1	READ WRITE	NO
4	NTAP_PDB2	READ WRITE	NO
5	NTAP_PDB3	READ WRITE	NO

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

NAME
+DATA/NTAP/DATAFILE/system.257.1189724205
+DATA/NTAP/DATAFILE/sysaux.258.1189724249
+DATA/NTAP/DATAFILE/undotbs1.259.1189724275
+DATA/NTAP/86B637B62FE07A65E053F706E80A27CA/DATAFILE/system.266.1189725235
+DATA/NTAP/86B637B62FE07A65E053F706E80A27CA/DATAFILE/sysaux.267.1189725235
+DATA/NTAP/DATAFILE/users.260.1189724275
+DATA/NTAP/86B637B62FE07A65E053F706E80A27CA/DATAFILE/undotbs1.268.1189725235
+DATA/NTAP/2B1302C26E089A59E0630400000A4D5C/DATAFILE/system.272.1189726217
+DATA/NTAP/2B1302C26E089A59E0630400000A4D5C/DATAFILE/sysaux.273.1189726217

```

+DATA/NTAP/2B1302C26E089A59E0630400000A4D5C/DATAFILE/undotbs1.271.11
89726217
+DATA/NTAP/2B1302C26E089A59E0630400000A4D5C/DATAFILE/users.275.11897
26243

NAME
-----
-----
+DATA/NTAP/2B13047FB98B9AAFE0630400000AFA5F/DATAFILE/system.277.1189
726245
+DATA/NTAP/2B13047FB98B9AAFE0630400000AFA5F/DATAFILE/sysaux.278.1189
726245
+DATA/NTAP/2B13047FB98B9AAFE0630400000AFA5F/DATAFILE/undotbs1.276.11
89726245
+DATA/NTAP/2B13047FB98B9AAFE0630400000AFA5F/DATAFILE/users.280.11897
26269
+DATA/NTAP/2B13061057039B10E0630400000AA001/DATAFILE/system.282.1189
726271
+DATA/NTAP/2B13061057039B10E0630400000AA001/DATAFILE/sysaux.283.1189
726271
+DATA/NTAP/2B13061057039B10E0630400000AA001/DATAFILE/undotbs1.281.11
89726271
+DATA/NTAP/2B13061057039B10E0630400000AA001/DATAFILE/users.285.11897
26293

```

19 rows selected.

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

```

MEMBER
-----
-----
+DATA/NTAP/ONLINELOG/group_3.264.1189724351
+LOGS/NTAP/ONLINELOG/group_3.259.1189724361
+DATA/NTAP/ONLINELOG/group_2.263.1189724351
+LOGS/NTAP/ONLINELOG/group_2.257.1189724359
+DATA/NTAP/ONLINELOG/group_1.262.1189724351
+LOGS/NTAP/ONLINELOG/group_1.258.1189724359
+DATA/NTAP/ONLINELOG/group_4.286.1190297279
+LOGS/NTAP/ONLINELOG/group_4.262.1190297283
+DATA/NTAP/ONLINELOG/group_5.287.1190297293
+LOGS/NTAP/ONLINELOG/group_5.263.1190297295
+DATA/NTAP/ONLINELOG/group_6.288.1190297307

```

```
MEMBER
-----
```

```
-----  
+LOGS/NTAP/ONLINELOG/group_6.264.1190297309  
+DATA/NTAP/ONLINELOG/group_7.289.1190297325  
+LOGS/NTAP/ONLINELOG/group_7.265.1190297327
```

14 rows selected.

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

NAME

```
-----  
-----  
+DATA/NTAP/CONTROLFILE/current.261.1189724347  
+LOGS/NTAP/CONTROLFILE/current.256.1189724347
```

6. 기본 DB에 대한 dNFS 구성.

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

```
SVRNAME
```

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

```
DIRNAME
```

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

```
10.0.2.39
```

```
/orap-u05
```

```
10.0.2.38
```

```
/orap-u04
```

```
10.0.2.38
```

```
/orap-u06
```

```
SVRNAME
```

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

```
DIRNAME
```

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

```
10.0.2.37
```

```
/orap-u02
```

```
10.0.2.36
```

```
/orap-u03
```

```
10.0.2.36
```

```
/orap-u01
```

```
6 rows selected.
```

이로써 NFS/ASM이 있는 ANF의 기본 사이트에서 VLDB NTAP에 대한 Data Guard 설정 데모가 완료되었습니다.

Data Guard를 위한 대기 Oracle VLDB 구성

Oracle Data Guard를 사용하려면 기본 DB 서버와 일치하도록 대기 DB 서버에 패치 세트를 포함한 Oracle 소프트웨어 스택과 OS 커널 구성이 필요합니다. 쉬운 관리와 단순성을 위해, 대기 DB 서버의 데이터베이스 스토리지 구성은 기본 DB 서버와 일치해야 합니다. 예를 들어, 데이터베이스 디렉터리 레이아웃과 NFS 마운트 지점의 크기가 그렇습니다.

또한 ASM을 사용하여 NFS에서 Oracle Data Guard 대기 모드를 설정하는 방법에 대한 자세한 단계별 절차는 다음을 참조하십시오. ["TR-5002 - Azure NetApp Files 통한 Oracle Active Data Guard 비용 절감"](#) 그리고 ["TR-4974 - NFS/ASM을 사용하는 AWS FSx/EC2에서 독립형 재시작 시 Oracle 19c"](#) 관련 섹션. 다음은 Data Guard 설정에서 대기 DB 서버의 대기 Oracle VLDB 구성에 대한 세부 정보를 보여줍니다.

1. 데모 랩의 대기 사이트에 있는 대기 Oracle DB 서버 구성입니다.

```
oras.internal.cloudapp.net:
resource group: ANFAVSRG
Location: West US 2
size: Standard B4ms (4 vcpus, 16 GiB memory)
OS: Linux (redhat 8.6)
pub_ip: 172.179.119.75
pri_ip: 10.0.1.4

[oracle@oras ~]$ df -h
Filesystem                                Size  Used Avail Use% Mounted on
devtmpfs                                  7.7G   0    7.7G   0% /dev
tmpfs                                     7.8G  1.1G   6.7G  15% /dev/shm
tmpfs                                     7.8G   25M   7.7G   1% /run
tmpfs                                     7.8G   0    7.8G   0% /sys/fs/cgroup
/dev/mapper/rootvg-rootlv                 22G   17G   5.6G  75% /
/dev/mapper/rootvg-usrlv                  10G   2.3G   7.8G  23% /usr
/dev/mapper/rootvg-varlv                   8.0G   1.1G   7.0G  13% /var
/dev/mapper/rootvg-homelv                  2.0G   52M   2.0G   3% /home
/dev/sda1                                 496M  181M  315M  37% /boot
/dev/sda15                                495M   5.8M  489M   2% /boot/efi
/dev/mapper/rootvg-tmplv                   12G   11G   1.8G  86% /tmp
/dev/sdb1                                 32G   49M   30G   1% /mnt
10.0.3.36:/oras-u03                       400G  282G  119G  71% /u03
10.0.3.36:/oras-u04                       300G  282G   19G  94% /u04
10.0.3.36:/oras-u05                       300G  282G   19G  94% /u05
10.0.3.36:/oras-u02                       300G  282G   19G  94% /u02
10.0.3.36:/oras-u01                       100G   21G   80G  21% /u01
10.0.3.36:/oras-u06                       300G  282G   19G  94% /u06

[oracle@oras ~]$ cat /etc/oratab
#Backup file is
/u01/app/oracle/crsdata/oras/output/oratab.bak.oras.oracle line
added by Agent
#
```

```

# 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
NTAP:/u01/app/oracle/product/19.0.0/NTAP:N                # line added
by Agent

```

2. 대기 DB 서버에서 그리드 인프라 구성.

```
[oracle@oras ~]$ $GRID_HOME/bin/crsctl stat res -t
```

```
-----
Name          Target  State          Server          State
details
-----
Local Resources
-----
ora.DATA.dg
          ONLINE  ONLINE         oras            STABLE
ora.LISTENER.lsnr
          ONLINE  ONLINE         oras            STABLE
ora.LOGS.dg
          ONLINE  ONLINE         oras            STABLE
ora.asm
          ONLINE  ONLINE         oras
Started, STABLE
ora.ons
          OFFLINE OFFLINE        oras            STABLE
-----
Cluster Resources
-----
ora.cssd
      1      ONLINE  ONLINE         oras            STABLE
ora.diskmon
      1      OFFLINE OFFLINE
ora.evmd
      1      ONLINE  ONLINE         oras            STABLE
ora.ntap_la.db
      1      ONLINE  INTERMEDIATE oras
Dismounted, Mount Ini
tiated, HOME=/u01/app
/oracle/product/19.0
.0/NTAP, STABLE
-----
-----
```

3. 대기 DB 서버에서 ASM 디스크 그룹 구성.

```

[oracle@oras ~]$ 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      4194304
1146880    1136912      0      1136912      0
N  DATA/
MOUNTED    EXTERN    N      512      512      4096      4194304
286720     284228      0      284228      0
N  LOGS/
ASMCMDS> lsdsd
Path
/u02/oradata/asm/oras_data_disk_01
/u02/oradata/asm/oras_data_disk_02
/u02/oradata/asm/oras_data_disk_03
/u02/oradata/asm/oras_data_disk_04
/u03/oralogs/asm/oras_logs_disk_01
/u03/oralogs/asm/oras_logs_disk_02
/u03/oralogs/asm/oras_logs_disk_03
/u03/oralogs/asm/oras_logs_disk_04
/u04/oradata/asm/oras_data_disk_05
/u04/oradata/asm/oras_data_disk_06
/u04/oradata/asm/oras_data_disk_07
/u04/oradata/asm/oras_data_disk_08
/u05/oradata/asm/oras_data_disk_09
/u05/oradata/asm/oras_data_disk_10
/u05/oradata/asm/oras_data_disk_11
/u05/oradata/asm/oras_data_disk_12
/u06/oradata/asm/oras_data_disk_13
/u06/oradata/asm/oras_data_disk_14
/u06/oradata/asm/oras_data_disk_15
/u06/oradata/asm/oras_data_disk_16

```

4. 대기 DB의 Data Guard에 대한 매개변수 설정.

```
SQL> show parameter name
```

NAME	TYPE	VALUE
-----	-----	

cdb_cluster_name	string	
cell_offloadgroup_name	string	
db_file_name_convert	string	
db_name	string	NTAP
db_unique_name	string	NTAP_LA
global_names	boolean	FALSE
instance_name	string	NTAP
lock_name_space	string	
log_file_name_convert	string	
pdb_file_name_convert	string	
processor_group_name	string	

NAME	TYPE	VALUE
-----	-----	

service_names	string	
NTAP_LA.internal.cloudapp.net		

```
SQL> show parameter log_archive_config
```

NAME	TYPE	VALUE
-----	-----	

log_archive_config	string	
DG_CONFIG=(NTAP_NY,NTAP_LA)		

```
SQL> show parameter fal_server
```

NAME	TYPE	VALUE
-----	-----	

fal_server	string	NTAP_NY

5. 대기 DB 구성.

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

NAME	OPEN_MODE	LOG_MODE
-----	-----	-----
NTAP	MOUNTED	ARCHIVELOG

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	MOUNTED	
3	NTAP_PDB1	MOUNTED	
4	NTAP_PDB2	MOUNTED	
5	NTAP_PDB3	MOUNTED	

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

NAME

```

+DATA/NTAP_LA/DATAFILE/system.261.1190301867
+DATA/NTAP_LA/DATAFILE/sysaux.262.1190301923
+DATA/NTAP_LA/DATAFILE/undotbs1.263.1190301969
+DATA/NTAP_LA/2B12C97618069248E0630400000AC50B/DATAFILE/system.264.1
190301987
+DATA/NTAP_LA/2B12C97618069248E0630400000AC50B/DATAFILE/sysaux.265.1
190302013
+DATA/NTAP_LA/DATAFILE/users.266.1190302039
+DATA/NTAP_LA/2B12C97618069248E0630400000AC50B/DATAFILE/undotbs1.267
.1190302045
+DATA/NTAP_LA/2B1302C26E089A59E0630400000A4D5C/DATAFILE/system.268.1
190302071
+DATA/NTAP_LA/2B1302C26E089A59E0630400000A4D5C/DATAFILE/sysaux.269.1
190302099
+DATA/NTAP_LA/2B1302C26E089A59E0630400000A4D5C/DATAFILE/undotbs1.270
.1190302125
+DATA/NTAP_LA/2B1302C26E089A59E0630400000A4D5C/DATAFILE/users.271.11
90302133

```

NAME

```

+DATA/NTAP_LA/2B13047FB98B9AAFE0630400000AFA5F/DATAFILE/system.272.1
190302137
+DATA/NTAP_LA/2B13047FB98B9AAFE0630400000AFA5F/DATAFILE/sysaux.273.1
190302163
+DATA/NTAP_LA/2B13047FB98B9AAFE0630400000AFA5F/DATAFILE/undotbs1.274
.1190302189
+DATA/NTAP_LA/2B13047FB98B9AAFE0630400000AFA5F/DATAFILE/users.275.11
90302197
+DATA/NTAP_LA/2B13061057039B10E0630400000AA001/DATAFILE/system.276.1
190302201
+DATA/NTAP_LA/2B13061057039B10E0630400000AA001/DATAFILE/sysaux.277.1

```

```

190302229
+DATA/NTAP_LA/2B13061057039B10E0630400000AA001/DATAFILE/undotbs1.278
.1190302255
+DATA/NTAP_LA/2B13061057039B10E0630400000AA001/DATAFILE/users.279.11
90302263

```

19 rows selected.

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

NAME

```

-----
-----
+DATA/NTAP_LA/CONTROLFILE/current.260.1190301831
+LOGS/NTAP_LA/CONTROLFILE/current.257.1190301833

```

```
SQL> select group#, type, member from v$logfile order by 2, 1;
```

GROUP# TYPE MEMBER

```

-----
-----
1 ONLINE +DATA/NTAP_LA/ONLINELOG/group_1.280.1190302305
1 ONLINE +LOGS/NTAP_LA/ONLINELOG/group_1.259.1190302309
2 ONLINE +DATA/NTAP_LA/ONLINELOG/group_2.281.1190302315
2 ONLINE +LOGS/NTAP_LA/ONLINELOG/group_2.258.1190302319
3 ONLINE +DATA/NTAP_LA/ONLINELOG/group_3.282.1190302325
3 ONLINE +LOGS/NTAP_LA/ONLINELOG/group_3.260.1190302329
4 STANDBY +DATA/NTAP_LA/ONLINELOG/group_4.283.1190302337
4 STANDBY +LOGS/NTAP_LA/ONLINELOG/group_4.261.1190302339
5 STANDBY +DATA/NTAP_LA/ONLINELOG/group_5.284.1190302347
5 STANDBY +LOGS/NTAP_LA/ONLINELOG/group_5.262.1190302349
6 STANDBY +DATA/NTAP_LA/ONLINELOG/group_6.285.1190302357

```

GROUP# TYPE MEMBER

```

-----
-----
6 STANDBY +LOGS/NTAP_LA/ONLINELOG/group_6.263.1190302359
7 STANDBY +DATA/NTAP_LA/ONLINELOG/group_7.286.1190302367
7 STANDBY +LOGS/NTAP_LA/ONLINELOG/group_7.264.1190302369

```

14 rows selected.

6. 대기 데이터베이스 복구 상태를 검증합니다. 주목하세요 recovery logmerger ~에 APPLYING_LOG 행동.

```
SQL> SELECT ROLE, THREAD#, SEQUENCE#, ACTION FROM
V$DATAGUARD_PROCESS;
```

ROLE	THREAD#	SEQUENCE#	ACTION
recovery logmerger	1	32	APPLYING_LOG
recovery apply slave	0	0	IDLE
RFS async	1	32	IDLE
recovery apply slave	0	0	IDLE
recovery apply slave	0	0	IDLE
RFS ping	1	32	IDLE
archive redo	0	0	IDLE
managed recovery	0	0	IDLE
archive redo	0	0	IDLE
archive redo	0	0	IDLE
recovery apply slave	0	0	IDLE

ROLE	THREAD#	SEQUENCE#	ACTION
redo transport monitor	0	0	IDLE
log writer	0	0	IDLE
archive local	0	0	IDLE
redo transport timer	0	0	IDLE
gap manager	0	0	IDLE
RFS archive	0	0	IDLE

17 rows selected.

7. 대기 DB에 대한 dNFS 구성.

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

```
SVRNAME
```

```
-----
```

```
-----
```

```
DIRNAME
```

```
-----
```

```
-----
```

```
10.0.3.36
```

```
/oras-u05
```

```
10.0.3.36
```

```
/oras-u04
```

```
10.0.3.36
```

```
/oras-u02
```

```
10.0.3.36
```

```
/oras-u06
```

```
10.0.3.36
```

```
/oras-u03
```

이로써 대기 사이트에서 관리형 대기 복구가 활성화된 VLDB NTAP에 대한 Data Guard 설정 데모가 완료되었습니다.

Data Guard Broker 설정

Oracle Data Guard 브로커는 Oracle Data Guard 구성의 생성, 유지 관리 및 모니터링을 자동화하고 중앙화하는 분산 관리 프레임워크입니다. 다음 섹션에서는 Data Guard 환경을 관리하기 위해 Data Guard Broker를 설정하는 방법을 보여줍니다.

1. sqlplus를 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 SYSDBA로 Data Guard Broker에 연결합니다.

```
[oracle@orap ~]$ dgmgrl sys@NTAP_NY
DGMGRL for Linux: Release 19.0.0.0.0 - Production on Wed Dec 11
20:53:20 2024
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 "NTAP_NY"
Connected as SYSDBA.
DGMGRL>
```

3. Data Guard Broker 구성을 만들고 활성화합니다.

```
DGMGRL> create configuration dg_config as primary database is
NTAP_NY connect identifier is NTAP_NY;
Configuration "dg_config" created with primary database "ntap_ny"
DGMGRL> add database NTAP_LA as connect identifier is NTAP_LA;
Database "ntap_la" added
DGMGRL> enable configuration;
Enabled.
DGMGRL> show configuration;

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
  ntap_ny - Primary database
  ntap_la - Physical standby database

Fast-Start Failover: Disabled

Configuration Status:
SUCCESS (status updated 3 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>
```

장애가 발생하는 경우, Data Guard Broker를 사용하면 기본 데이터베이스를 즉시 대기 데이터베이스로 장애 조치할 수 있습니다. 만약에 Fast-Start Failover 이 기능이 활성화되면 Data Guard Broker는 사용자 개입 없이 장애가 감지되면 기본 데이터베이스를 대기 데이터베이스로 장애 조치할 수 있습니다.

자동화를 통해 다른 사용 사례에 대한 대기 데이터베이스 복제

다음 자동화 툴킷은 NFS/ASM 구성을 사용하여 ANF에 배포된 Oracle Data Guard 대기 DB의 복제본을 만들거나 새로 고치도록 특별히 설계되어 복제본 수명 주기를 완벽하게 관리합니다.

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



현재 이 툴킷은 bitbucket 액세스 권한이 있는 NetApp 내부 사용자만 액세스할 수 있습니다. 관심 있는 외부 사용자는 계정 팀에 액세스를 요청하거나 NetApp 솔루션 엔지니어링 팀에 문의하세요.

추가 정보를 찾을 수 있는 곳

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

- TR-5002: Azure NetApp Files 통한 Oracle Active Data Guard 비용 절감

["TR-5002: Azure NetApp Files 통한 Oracle Active Data Guard 비용 절감"](#)

- TR-4974: NFS/ASM을 사용하는 AWS FSx/EC2에서 독립형 재시작 시 Oracle 19c

["TR-4974: NFS/ASM을 사용하는 AWS FSx/EC2에서 독립형 재시작 시 Oracle 19c"](#)

- Azure NetApp Files

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

- Oracle 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)

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.