



TR-5003: ANF 上的高吞吐量 Oracle VLDB 实现

NetApp database solutions

NetApp
August 18, 2025

目录

- TR-5003: ANF 上的高吞吐量 Oracle VLDB 实现 1
 - 目的 1
 - 受众 1
 - 解决方案测试和验证环境 1
 - 架构 1
 - 硬件和软件组件 2
 - Oracle VLDB Data Guard 配置，模拟纽约到洛杉矶的灾难恢复设置 3
 - 部署考虑的关键因素 3
 - 解决方案部署 3
 - 部署先决条件 3
 - Data Guard 的主要 Oracle VLDB 配置 6
 - 针对 Data Guard 的备用 Oracle VLDB 配置 15
 - 设置 Data Guard Broker 24
 - 通过自动化克隆其他用例的备用数据库 27
 - 在哪里可以找到更多信息 28

TR-5003：ANF 上的高吞吐量 Oracle VLDB 实现

Allen Cao、Niyaz Mohamed，NetApp

该解决方案提供了在 Azure 云中使用 Oracle Data Guard 在 Microsoft Azure NetApp Files (ANF) 上配置高吞吐量 Oracle 超大型数据库 (VLDB) 的概述和详细信息。

目的

高吞吐量和任务关键型 Oracle VLDB 对后端数据库存储提出了很高的要求。为了满足服务水平协议 (SLA)，数据库存储必须提供所需的容量和每秒高输入/输出操作数 (IOPS)，同时保持亚毫秒级延迟性能。在具有共享存储资源环境的公共云中部署此类数据库工作负载时，这一点尤其具有挑战性。并非所有存储平台都是平等的。高级 Azure NetApp Files 存储与 Azure 基础架构相结合，可以满足如此高要求的 Oracle 工作负载的需求。在经过验证的性能基准测试中 (["Azure NetApp Files 多个卷上的 Oracle 数据库性能"](#))，ANF 通过 SLOB 工具在合成的 100% 随机选择工作负载中提供了 250 万次读取 IOPS，延迟为 700 微秒。对于标准 8k 块大小，这相当于大约 20 GiB/s 的吞吐量。

在本文档中，我们演示了如何在具有多个 NFS 卷和 Oracle ASM 的 ANF 存储上设置具有 Data Guard 配置的 Oracle VLDB 以实现存储负载平衡。备用数据库可以通过快照快速（几分钟）备份，并根据需要克隆以供用例进行读/写访问。NetApp 解决方案工程团队提供了自动化工具包，可按照用户定义的计划轻松创建和刷新克隆。

此解决方案适用于以下用例：

- 在跨 Azure 区域的 Microsoft Azure NetApp Files 存储上的 Data Guard 设置中实现 Oracle VLDB。
- 快照备份并克隆物理备用数据库，以通过自动化服务于报告、开发、测试等用例。

受众

此解决方案适用于以下人群：

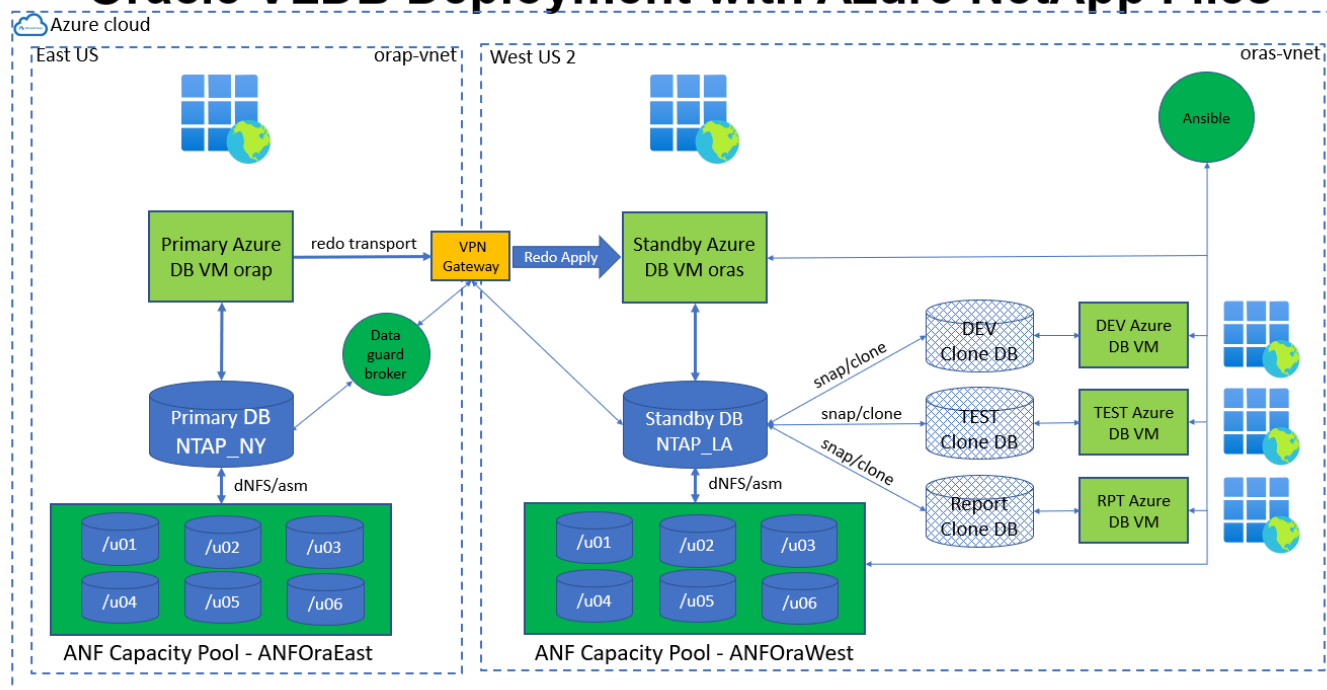
- 在 Azure 云中使用 Data Guard 设置 Oracle VLDB 以实现高可用性、数据保护和灾难恢复的 DBA。
- 对 Azure 云中具有 Data Guard 配置的 Oracle VLDB 感兴趣的数据库解决方案架构师。
- 管理支持 Oracle 数据库的 Azure NetApp Files 存储的存储管理员。
- 喜欢在 Azure 云环境中使用 Data Guard 建立 Oracle VLDB 的应用程序所有者。

解决方案测试和验证环境

该解决方案的测试和验证是在 Azure 云实验室设置中进行的，该设置可能与实际用户部署环境不匹配。有关更多信息，请参阅[部署考虑的关键因素](#)。

架构

Oracle VLDB Deployment with Azure NetApp Files



硬件和软件组件

硬件		
Azure NetApp Files	Microsoft 提供的当前版本	两个 4 TiB 容量池、高级服务级别、自动 QoS
用于数据库服务器的 Azure VM	标准 B4ms (4 个 vcpus, 16 GiB 内存)	三个数据库虚拟机，一个作为主数据库服务器，一个作为备用数据库服务器，第三个作为克隆数据库服务器
软件		
红帽Linux	Red Hat Enterprise Linux 8.6 (LVM) -x64 Gen2	部署 RedHat 订阅进行测试
Oracle 网络基础设施	19.18 版	已应用RU补丁p34762026_190000_Linux-x86-64.zip
Oracle 数据库	19.18 版	已应用RU补丁p34765931_190000_Linux-x86-64.zip
dNFS一次性补丁	p32931941_190000_Linux-x86-64.zip	适用于网络和数据库
Oracle OPatch	版本 12.2.0.1.36	最新补丁p6880880_190000_Linux-x86-64.zip
Ansible	版本核心 2.16.2	python版本-3.10.13

NFS	3.0 版	为 Oracle 启用 dNFS
-----	-------	------------------

Oracle VLDB Data Guard 配置，模拟纽约到洛杉矶的灾难恢复设置

数据库	数据库唯一名称	Oracle 网络服务名称
主云	NTAP_NY	NTAP_NY.internal.cloudapp.net
备用	NTAP_LA	NTAP_LA.internal.cloudapp.net

部署考虑的关键因素

- **Azure NetApp Files配置。** Azure NetApp Files在 Azure NetApp存储帐户中分配为 Capacity Pools。在这些测试和验证中，我们部署了一个 2 TiB 容量池来托管东部区域的 Oracle 主数据库，并部署了一个 4 TiB 容量池来托管西部 2 区域的备用数据库和数据库克隆。ANF 容量池有三个服务级别：Standard、Premium 和 Ultra。ANF 容量池的 IO 容量取决于容量池的大小及其服务等级。在创建容量池时，可以将 QoS 设置为“自动”或“手动”，并将静态数据加密设置为“单次”或“两次”。
- ***调整数据库卷的大小。***对于生产部署，NetApp 建议根据 Oracle AWR 报告对 Oracle 数据库吞吐量要求进行全面评估。在确定数据库的 ANF 卷大小时，要同时考虑数据库大小以及吞吐量要求。通过为 ANF 自动配置 QoS，可保证在分配超级服务级别的情况下，每 TiB 卷容量的带宽为 128 MiB/s。更高的吞吐量可能需要更大的容量才能满足要求。
- ***单卷或多卷。***由于 QoS 是根据卷大小和容量池服务级别严格执行的，因此单个大卷可以提供与具有相同聚合大小的多个卷类似的性能级别。建议为 Oracle VLDB 实现多个卷（多个 NFS 挂载点），以更好地利用共享后端 ANF 存储资源池。实现 Oracle ASM 以实现多个 NFS 卷上的 IO 负载平衡。
- **应用程序卷组。**为 Oracle 部署应用程序卷组 (AVG) 以优化性能。按应用程序卷组部署的卷放置在区域或区域基础架构中，以实现应用程序虚拟机的优化延迟和吞吐量。
- **Azure VM 考虑事项**在这些测试和验证中，我们使用了具有 4 个 vCPU 和 16 GiB 内存的 Azure VM - Standard_B4ms。您需要为具有高吞吐量要求的 Oracle VLDB 选择适当的 Azure DB VM。除了 vCPU 数量和 RAM 数量之外，在达到数据库存储容量之前，VM 网络带宽（入口和出口或 NIC 吞吐量限制）也可能成为瓶颈。
- **dNFS 配置。**通过使用 dNFS，在具有 ANF 存储的 Azure 虚拟机上运行的 Oracle 数据库可以比本机 NFS 客户端驱动更多的 I/O。确保应用 Oracle dNFS 补丁 p32931941 来解决潜在的错误。

解决方案部署

假设您已将主 Oracle 数据库部署在 VNet 内的 Azure 云环境中，作为设置 Oracle Data Guard 的起点。理想情况下，主数据库部署在具有 NFS 挂载的 ANF 存储上。您的主要 Oracle 数据库也可以在 NetApp ONTAP 存储或 Azure 生态系统或私有数据中心内的任何其他存储上运行。以下部分演示了在 Azure 中具有 ANF 存储的主 Oracle DB 与 Azure 中具有 ANF 存储的物理备用 Oracle DB 之间的 Oracle Data Guard 设置中 ANF 上的 Oracle VLDB 的配置。

部署先决条件

部署需要以下先决条件。

1. 已设置 Azure 云帐户，并在您的 Azure 帐户内创建了必要的 VNet 和网络子网。
2. 从 Azure 云门户控制台，您需要部署至少三个 Azure Linux VM，一个作为主 Oracle DB 服务器，一个作为备用 Oracle DB 服务器，以及一个用于报告、开发和测试等的克隆目标 DB 服务器。有关环境设置的更多详细信息，请参阅上一节中的体系结构图。还审查了微软["Azure 虚拟机"](#)了解更多信息。
3. 主 Oracle 数据库应该已经在主 Oracle DB 服务器中安装并配置。另一方面，在备用 Oracle DB 服务器或克隆 Oracle DB 服务器中，仅安装了 Oracle 软件，而没有创建 Oracle 数据库。理想情况下，Oracle 文件目录布局应该与所有 Oracle DB 服务器完全匹配。有关NetApp对 Azure 云和 ANF 中自动化 Oracle 部署的的建议的详细信息，请参阅以下技术报告以获取帮助。
 - ["TR-4987：使用 NFS 在Azure NetApp Files上简化、自动化 Oracle 部署"](#)



确保在 Azure VM 根卷中分配了至少 128G，以便有足够的空间来暂存 Oracle 安装文件。

4. 从 Azure 云门户控制台部署两个 ANF 存储容量池来托管 Oracle 数据库卷。ANF 存储容量池应位于不同的区域，以模拟真正的 DataGuard 配置。如果您不熟悉 ANF 存储的部署，请参阅文档["快速入门：设置Azure NetApp Files并创建 NFS 卷"](#)以获得分步说明。

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. 当主 Oracle 数据库和备用 Oracle 数据库位于两个不同的区域时，应配置 VPN 网关以允许两个单独的 VNet 之间的数据流量流动。Azure 中的详细网络配置超出了本文档的范围。以下屏幕截图提供了有关如何在实验室中配置、连接 VPN 网关以及如何确认数据流量流动的一些参考。

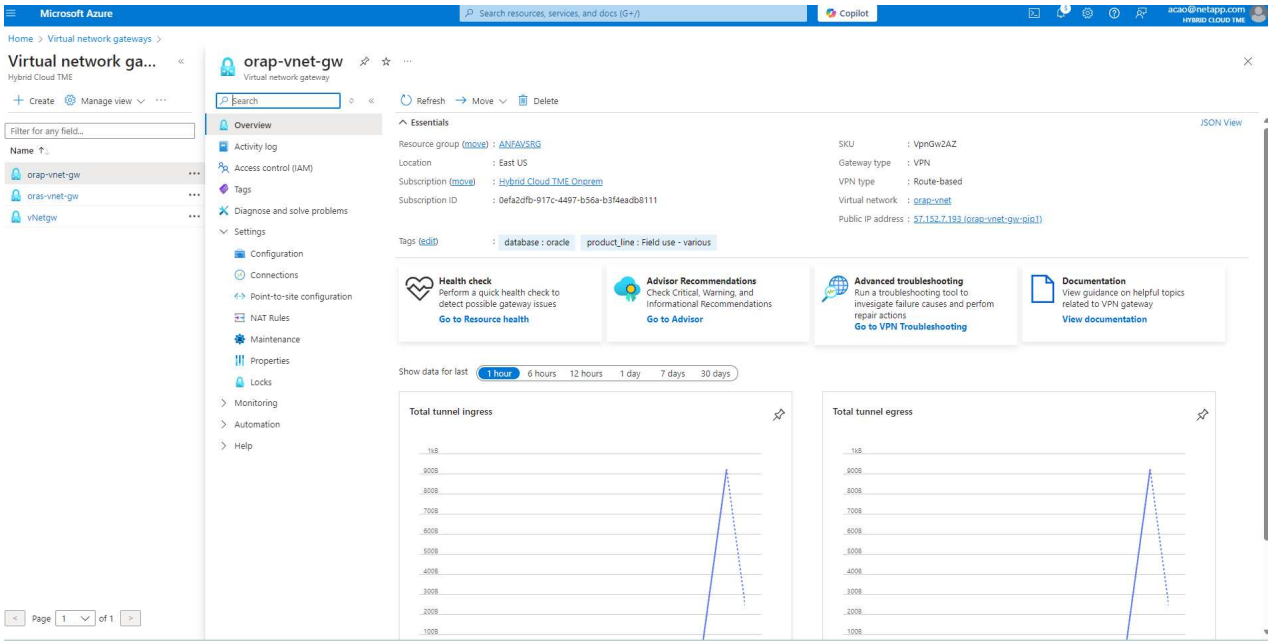
实验室 VPN 网关

:

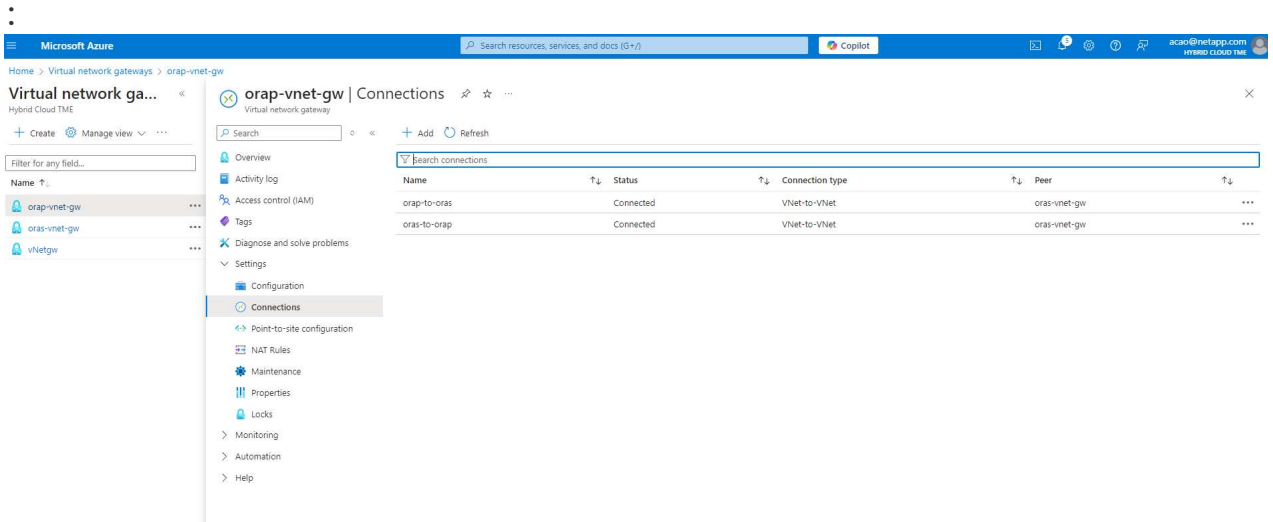
Name	Virtual network	Gateway type	Resource group	Location	Subscription
orap-vnet-gw	orap-vnet	Vpn	ANFAVSRG	East US	Hybrid Cloud TME Onprem
oras-vnet-gw	oras-vnet	Vpn	ANFAVSRG	West US 2	Hybrid Cloud TME Onprem
vNetgw	EHCVnet	Vpn	NSOL	Central US	Hybrid Cloud TME Onprem

主 VNet 网关

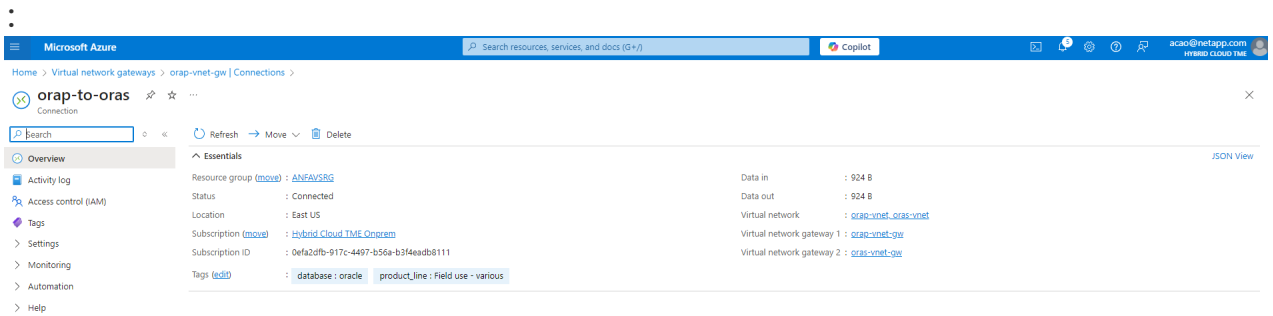
:



Vnet网关连接状态



验证流量是否已建立（单击三个点打开页面）



6. 参考此文档"为 Oracle 部署应用程序卷组"为 Oracle 部署应用程序卷组。

Data Guard 的主要 Oracle VLDB 配置

在此演示中，我们在主 Azure DB 服务器上设置了一个名为 NTAP 的主 Oracle 数据库，该数据库有六个 NFS 挂载点：/u01 用于 Oracle 二进制文件，/u02、/u04、/u05、/u06 用于 Oracle 数据文件和 Oracle 控制文件，/u03 用于 Oracle 活动日志、存档日志文件和冗余 Oracle 控制文件。此设置可作为参考配置。您的实际部署应该考虑到容量池大小、服务级别、数据库卷数量和每个卷的大小方面的特定需求和要求。

有关使用 ASM 在 NFS 上设置 Oracle Data Guard 的详细分步过程，请参阅["TR-5002- 使用 Azure NetApp Files 降低 Oracle Active Data Guard 成本"](#)和["TR-4974- 使用 NFS/ASM 在 AWS FSx/EC2 上独立重启 Oracle 19c"](#)相关章节。尽管 TR-4974 中的程序已在 Amazon FSx ONTAP 上得到验证，但它们同样适用于 ANF。下面说明了 Data Guard 配置中主 Oracle VLDB 的详细信息。

1. 主 Azure DB 服务器 orap.internal.cloudapp.net 上的主数据库 NTAP 最初部署为独立数据库，其中 NFS 上的 ANF 和 ASM 作为数据库存储。

```
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. 以 oracle 用户身份登录主数据库服务器。验证电网配置。

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

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

3. ASM 磁盘组配置。

```
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. 主数据库上 Data Guard 的参数设置。

```

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. 主数据库配置。

```
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. 主数据库上的 dNFS 配置。


```
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.
```

这完成了在具有 NFS/ASM 的 ANF 主站点上为 VLDB NTAP 设置 Data Guard 的演示。

针对 **Data Guard** 的备用 **Oracle VLDB** 配置

Oracle Data Guard 需要操作系统内核配置和 Oracle 软件堆栈（包括备用数据库服务器上的补丁集）以与主数据库服务器匹配。为了便于管理和简单起见，备用数据库服务器的数据库存储配置理想情况下也应该与主数据库服务器匹配，例如数据库目录布局和 NFS 挂载点的大小。

再次强调，有关使用 ASM 在 NFS 上设置 Oracle Data Guard 备用服务器的详细分步过程，请参阅["TR-5002 - 使用 Azure NetApp Files 降低 Oracle Active Data Guard 成本"](#)和["TR-4974 - 使用 NFS/ASM 在 AWS FSx/EC2 上独立重启 Oracle 19c"](#)相关章节。下面说明了 Data Guard 设置中备用数据库服务器上备用 Oracle VLDB 配置的详细信息。

1. 演示实验室中备用站点的备用 Oracle DB 服务器配置。

```
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. 备用数据库服务器上的网络基础设施配置。

```

[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        STABLE
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. 备用数据库服务器上的 ASM 磁盘组配置。

```

[oracle@oras ~]$ 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
1146880    1136912      0      1136912      0
N  DATA/
MOUNTED    EXTERN    N      512      512      4096      4194304
286720     284228      0      284228      0
N  LOGS/
ASMCMDB> lsdk
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. 备用数据库上的 Data Guard 参数设置。

```
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. 备用数据库配置。

```
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. 验证备用数据库恢复状态。注意 `recovery logmerger` 在 `APPLYING\_LOG` 行动。


```
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. 备用数据库上的 dNFS 配置。

```
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
```

这样就完成了 VLDB NTAP 的 Data Guard 设置演示，并在备用站点启用了托管备用恢复。

设置 Data Guard Broker

Oracle Data Guard 代理是一个分布式管理框架，可自动并集中执行 Oracle Data Guard 配置的创建、维护和监控。以下部分演示如何设置 Data Guard Broker 来管理 Data Guard 环境。

1. 通过 sqlplus 使用以下命令在主数据库和备用数据库上启动数据保护代理。

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

2. 从主数据库，以 SYSDBA 身份连接到 Data Guard Broker。

```
[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. 创建并启用 Data Guard Broker 配置。

```
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. 在 Data Guard Broker 管理框架内验证数据库状态。

```

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>

```

如果发生故障，可以使用 Data Guard Broker 立即将主数据库故障转移到备用数据库。如果 `Fast-Start Failover` 启用后，当检测到故障时，Data Guard Broker 可以将主数据库故障转移到备用数据库，而无需用户干预。

通过自动化克隆其他用例的备用数据库

以下自动化工具包专门用于创建或刷新部署到 ANF 的 Oracle Data Guard 备用数据库的克隆，并使用 NFS/ASM 配置实现完整的克隆生命周期管理。

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



目前，只有具有 bitbucket 访问权限的NetApp内部用户才能访问该工具包。对于感兴趣的外部用户，请向您的客户团队请求访问权限或联系NetApp解决方案工程团队。

在哪里可以找到更多信息

要了解有关本文档中描述的信息的更多信息，请查看以下文档和/或网站：

- TR-5002：使用Azure NetApp Files降低 Oracle Active Data Guard 成本

["TR-5002：使用Azure NetApp Files降低 Oracle Active Data Guard 成本"](#)

- TR-4974：在 AWS FSx/EC2 上使用 NFS/ASM 进行独立重启时 Oracle 19c

["TR-4974：在 AWS FSx/EC2 上使用 NFS/ASM 进行独立重启时 Oracle 19c"](#)

- Azure NetApp Files

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

- Oracle Data Guard 概念和管理

["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)

版权信息

版权所有 © 2025 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。