TR-4965：使用 iSCSI/ASM 在 AWS FSx/EC2 中部署和保护 Oracle 数据库

NetApp database solutions

NetApp
August 18, 2025

This PDF was generated from https://docs.netapp.com/zh-cn/netapp-solutions-databases/oracle/aws-ora-fsx-ec2-iscsi-asm.html on August 18, 2025. Always check docs.netapp.com for the latest.

目录

TR-4965：使用 iSCSI/ASM 在 AWS FSx/EC2 中部署和保护 Oracle 数据库 . . . 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
部署考虑的关键因素 . . . 3
解决方案部署 . . . 3
部署先决条件 . . . 3
EC2 实例内核配置 . . . 4
配置数据库卷和 LUN 并将其映射到 EC2 实例主机 . . . 8
数据库存储配置 . . . 11
Oracle 网格基础设施安装 . . . 19
Oracle数据库安装 . . . 24
自动部署选项 . . . 31
使用SnapCenter服务备份、恢复和克隆 Oracle 数据库 . . . 31
在哪里可以找到更多信息 . . . 32

TR-4965：使用 iSCSI/ASM 在 AWS FSx/EC2 中部署和保护 Oracle 数据库

Allen Cao、Niyaz Mohamed， NetApp

该解决方案提供了在 AWS FSx ONTAP存储和 EC2 计算实例中使用 iSCSI 协议进行 Oracle 数据库部署和保护的概述和详细信息，以及在独立 ReStart 中使用 asm 作为卷管理器配置的 Oracle 数据库。

目的

ASM（自动存储管理）是一种流行的 Oracle 存储卷管理器，在许多 Oracle 安装中都有使用。它也是Oracle推荐的存储管理解决方案。它为传统的卷管理器和文件系统提供了替代方案。自 Oracle 11g 版本以来，ASM 与网格基础架构打包在一起，而不是与数据库打包在一起。因此，为了在没有 RAC 的情况下利用 Oracle ASM 进行存储管理，必须在独立服务器中安装 Oracle 网格基础架构（也称为 Oracle Restart）。这样做无疑会增加 Oracle 数据库部署的复杂性。但是，顾名思义，当 Oracle 以重启模式部署时，失败的 Oracle 服务会由网格基础架构自动重新启动或在主机重启后无需用户干预，从而提供一定程度的高可用性或 HA 功能。

在本文档中，我们演示了如何在具有 EC2 计算实例的Amazon FSx ONTAP存储环境中使用 iSCSI 协议和 Oracle ASM 部署 Oracle 数据库。我们还演示了如何通过NetApp BlueXP控制台使用NetApp SnapCenter服务来备份、恢复和克隆 Oracle 数据库，以用于开发/测试或 AWS 公共云中存储高效的数据库操作的其他用例。

此解决方案适用于以下用例：

- 使用 iSCSI/ASM 在Amazon FSx ONTAP存储和 EC2 计算实例中部署 Oracle 数据库
- 使用 iSCSI/ASM 在公共 AWS 云中测试和验证 Oracle 工作负载
- 测试和验证在 AWS 中部署的 Oracle 数据库重启功能

受众

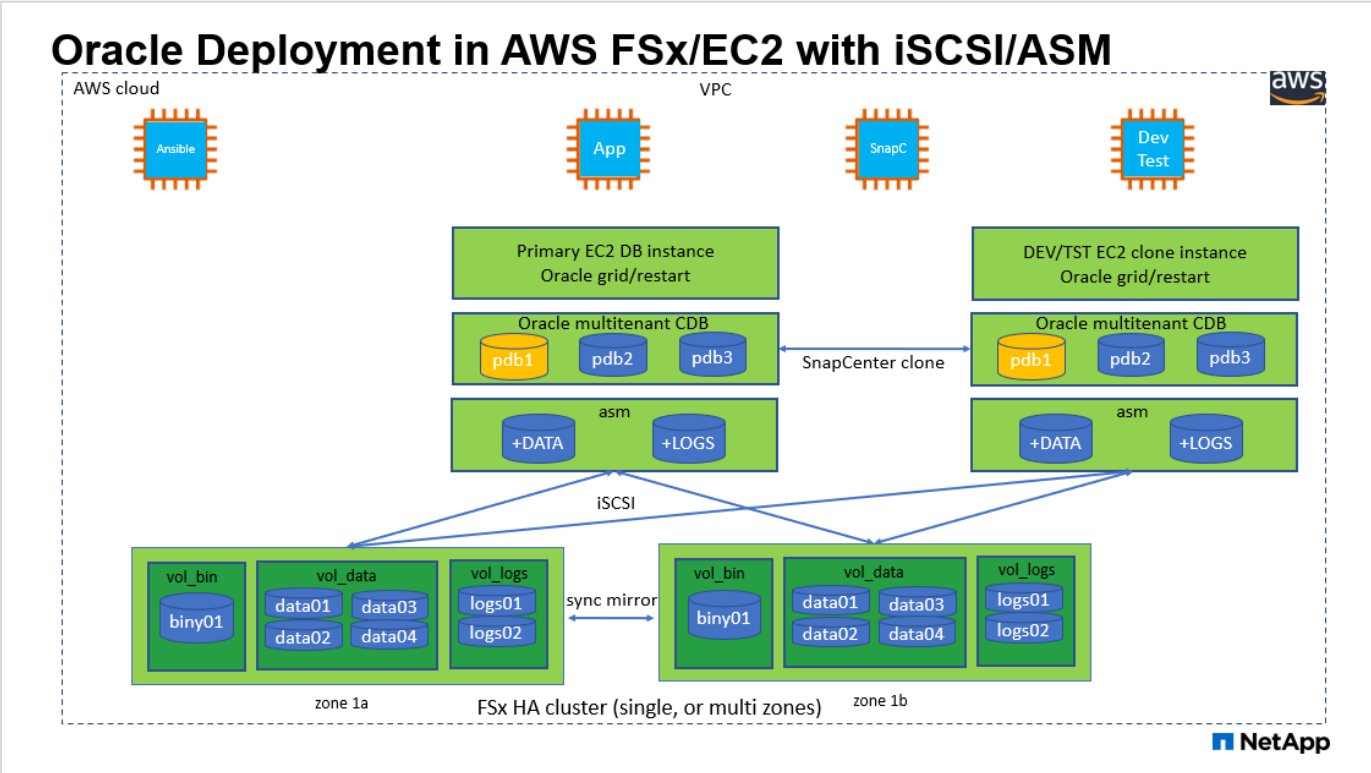
此解决方案适用于以下人群：

- 一位 DBA 想要使用 iSCSI/ASM 在 AWS 公共云中部署 Oracle。
- 一位数据库解决方案架构师想要在 AWS 公共云中测试 Oracle 工作负载。
- 想要部署和管理部署到 AWS FSx 存储的 Oracle 数据库的存储管理员。
- 希望在 AWS FSx/EC2 中建立 Oracle 数据库的应用程序所有者。

解决方案测试和验证环境

该解决方案的测试和验证是在可能与最终部署环境不匹配的 AWS FSx 和 EC2 环境中进行的。有关更多信息，请参阅[部署考虑的关键因素] 。

架构



硬件和软件组件

硬件		
FSx ONTAP存储	AWS 提供的当前版本	同一 VPC 和可用区中的一个 FSx HA 集群
用于计算的 EC2 实例	t2.xlarge/4vCPU/16G	两个 EC2 T2 xlarge EC2 实例，一个作为主数据库服务器，另一个作为克隆数据库服务器
软件		
红帽Linux	RHEL-8.6.0_HVM-20220503-x86_64-2-Hourly2-GP2	部署 RedHat 订阅进行测试
Oracle 网格基础设施	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
Oracle OPatch	版本 12.2.0.1.36	最新补丁p6880880_190000_Linux-x86-64.zip
SnapCenter服务	版本	v2.3.1.2324

部署考虑的关键因素

- EC2 计算实例。在这些测试和验证中，我们使用 AWS EC2 t2.xlarge 实例类型作为 Oracle 数据库计算实例。 NetApp建议在生产部署中使用 M5 类型的 EC2 实例作为 Oracle 的计算实例，因为它针对数据库工作负载进行了优化。您需要根据实际工作负载需求，适当调整 EC2 实例的 vCPU 数量和 RAM 容量。
- *FSx 存储 HA 集群单区域或多区域部署。*在这些测试和验证中，我们在单个 AWS 可用区中部署了一个 FSx HA 集群。对于生产部署， NetApp建议在两个不同的可用区部署 FSx HA 对。 FSx HA 集群始终在 HA 对中配置，该 HA 对在主动-被动文件系统中同步镜像，以提供存储级冗余。多区域部署进一步增强了单个 AWS 区域发生故障时的高可用性。
- FSx 存储集群大小。 Amazon FSx ONTAP存储文件系统提供高达 160,000 个原始 SSD IOPS、高达 4GBps 的吞吐量以及最大 192TiB 的容量。但是，您可以根据部署时的实际需求，根据预配置的 IOPS、吞吐量和存储限制（最小 1,024 GiB）来确定集群大小。容量可以动态调整，而不会影响应用程序的可用性。
- Oracle 数据和日志布局。在我们的测试和验证中，我们分别部署了两个 ASM 磁盘组用于数据和日志。在 +DATA asm 磁盘组中，我们在数据卷中配置了四个 LUN。在 +LOGS asm 磁盘组中，我们在日志卷中配置了两个 LUN。通常， Amazon FSx ONTAP卷内布局的多个 LUN 可提供更好的性能。
- iSCSI 配置。 EC2 实例数据库服务器通过 iSCSI 协议连接到 FSx 存储。 EC2 实例通常部署单个网络接口或 ENI。单个 NIC 接口同时承载 iSCSI 和应用程序流量。通过仔细分析 Oracle AWR 报告来衡量 Oracle 数据库峰值 I/O 吞吐量要求非常重要，这样才能选择出能够同时满足应用程序和 iSCSI 流量吞吐量要求的正确 EC2 计算实例。 NetApp还建议为两个 FSx iSCSI 端点分配四个 iSCSI 连接，并正确配置多路径。
- *您创建的每个 Oracle ASM 磁盘组使用的 Oracle ASM 冗余级别。*由于 FSx 已经在 FSx 集群级别镜像了存储，因此您应该使用外部冗余，这意味着该选项不允许 Oracle ASM 镜像磁盘组的内容。
- 数据库备份。 NetApp提供了 SaaS 版本的SnapCenter software服务，用于在云中进行数据库备份、恢复和克隆，可通过NetApp BlueXP控制台 UI 使用。 NetApp建议实施这样的服务，以实现快速（一分钟内）的 SnapShot 备份、快速（几分钟）的数据库恢复和数据库克隆。

解决方案部署

以下部分提供了分步部署过程。

部署先决条件

部署需要以下先决条件。

1. 已设置 AWS 账户，并在您的 AWS 账户内创建了必要的 VPC 和网络段。
2. 从 AWS EC2 控制台，您必须部署两个 EC2 Linux 实例，一个作为主 Oracle DB 服务器，另一个作为可选的备用克隆目标 DB 服务器。有关环境设置的更多详细信息，请参阅上一节中的架构图。还请查看"Linux 实例用户指南"了解更多信息。
3. 从 AWS EC2 控制台部署Amazon FSx ONTAP存储 HA 集群来托管 Oracle 数据库卷。如果您不熟悉 FSx 存储的部署，请参阅文档"创建 FSx ONTAP文件系统"以获得分步说明。
4. 步骤 2 和 3 可以使用以下 Terraform 自动化工具包执行，该工具包创建一个名为 `ora_01`以及一个名为 `fsx_01`。在执行之前，请仔细检查说明并更改变量以适合您的环境。

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

确保您已在 EC2 实例根卷中分配至少 50G，以便有足够的空间来暂存 Oracle 安装文件。

EC2 实例内核配置

满足先决条件后，以 ec2-user 身份登录 EC2 实例，并使用 sudo 以 root 用户身份配置 Linux 内核以进行 Oracle 安装。

1. 创建暂存目录 `/tmp/archive`文件夹并设置 `777`允许。

```
mkdir /tmp/archive

chmod 777 /tmp/archive
```

2. 下载 Oracle 二进制安装文件和其他所需的 rpm 文件并将其暂存到 `/tmp/archive`目录。

   请参阅以下安装文件列表以说明 `/tmp/archive`在 EC2 实例上。

```
[ec2-user@ip-172-30-15-58 ~]$ ls -l /tmp/archive
total 10537316
-rw-rw-r--. 1 ec2-user ec2-user      19112 Mar 21 15:57 compat-
libcap1-1.10-7.el7.x86_64.rpm
-rw-rw-r--  1 ec2-user ec2-user 3059705302 Mar 21 22:01
LINUX.X64_193000_db_home.zip
-rw-rw-r--  1 ec2-user ec2-user 2889184573 Mar 21 21:09
LINUX.X64_193000_grid_home.zip
-rw-rw-r--. 1 ec2-user ec2-user     589145 Mar 21 15:56
netapp_linux_unified_host_utilities-7-1.x86_64.rpm
-rw-rw-r--. 1 ec2-user ec2-user      31828 Mar 21 15:55 oracle-
database-preinstall-19c-1.0-2.el8.x86_64.rpm
-rw-rw-r--  1 ec2-user ec2-user 2872741741 Mar 21 22:31
p34762026_190000_Linux-x86-64.zip
-rw-rw-r--  1 ec2-user ec2-user 1843577895 Mar 21 22:32
p34765931_190000_Linux-x86-64.zip
-rw-rw-r--  1 ec2-user ec2-user  124347218 Mar 21 22:33
p6880880_190000_Linux-x86-64.zip
-rw-r--r--  1 ec2-user ec2-user     257136 Mar 22 16:25
policycoreutils-python-utils-2.9-9.el8.noarch.rpm
```

3. 安装 Oracle 19c 预安装 RPM，满足大多数内核配置要求。

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

4. 下载并安装缺失的 `compat-libcap1`在 Linux 8 中。

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

5. 从NetApp下载并安装NetApp主机实用程序。

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

6. 安装 `policycoreutils-python-utils`，这在 EC2 实例中不可用。

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

7. 安装开放的JDK版本1.8。

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

8. 安装 iSCSI 启动器实用程序。

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

9. 安装 `sg3_utils`。

```
yum install sg3_utils
```

10. 安装 `device-mapper-multipath`。

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

11. 在当前系统中禁用透明大页面。

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

添加以下行 `/etc/rc.local`禁用 `transparent_hugepage`重启后：

```
  # Disable transparent hugepages
          if test -f /sys/kernel/mm/transparent_hugepage/enabled;
then
            echo never > /sys/kernel/mm/transparent_hugepage/enabled
          fi
          if test -f /sys/kernel/mm/transparent_hugepage/defrag;
then
            echo never > /sys/kernel/mm/transparent_hugepage/defrag
          fi
```

12. 通过更改禁用 selinux `SELINUX=enforcing`到 `SELINUX=disabled`。您必须重新启动主机才能使更改生效。

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

13. 添加以下行到 `limit.conf`设置文件描述符限制和堆栈大小（不带引号） `" "`。

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

14. 按照以下说明向 EC2 实例添加交换空间："如何使用交换文件分配内存作为 Amazon EC2 实例中的交换空间？"要添加的具体空间量取决于 RAM 的大小，最高可达 16G。
15. 改变 `node.session.timeo.replacement_timeout`在 `iscsi.conf`配置文件从120秒减少到5秒。

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

16. 在 EC2 实例上启用并启动 iSCSI 服务。

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

17. 检索用于数据库 LUN 映射的 iSCSI 启动器地址。

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

18. 添加要用于 asm sysasm 组的 ASM 组。

```
groupadd asm
```

19. 修改 oracle 用户以添加 ASM 作为辅助组（oracle 用户应该在 Oracle 预安装 RPM 安装后创建）。

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

20. 如果 Linux 防火墙处于活动状态，请停止并禁用它。

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

21. 重启 EC2 实例。

配置数据库卷和 **LUN** 并将其映射到 **EC2** 实例主机

通过 ssh 以 fsxadmin 用户身份使用 FSx 集群管理 IP 登录 FSx 集群，从命令行配置三个卷来托管 Oracle 数据库二进制文件、数据和日志文件。

1. 以 fsxadmin 用户身份通过 SSH 登录 FSx 集群。

```
ssh fsxadmin@172.30.15.53
```

2. 执行以下命令为 Oracle 二进制文件创建卷。

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

3. 执行以下命令为 Oracle 数据创建卷。

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

4. 执行以下命令为 Oracle 日志创建卷。

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

5. 在数据库二进制卷内创建二进制 LUN。

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

6. 在数据库数据卷内创建数据 LUN。

```
lun create -path /vol/ora_01_data/ora_01_data_01 -size 20G -ostype
linux

lun create -path /vol/ora_01_data/ora_01_data_02 -size 20G -ostype
linux

lun create -path /vol/ora_01_data/ora_01_data_03 -size 20G -ostype
linux

lun create -path /vol/ora_01_data/ora_01_data_04 -size 20G -ostype
linux
```

7. 在数据库日志卷内创建日志 LUN。

```
lun create -path /vol/ora_01_logs/ora_01_logs_01 -size 40G -ostype
linux

lun create -path /vol/ora_01_logs/ora_01_logs_02 -size 40G -ostype
linux
```

8. 使用从上面的 EC2 内核配置的第 14 步检索到的启动器为 EC2 实例创建一个 igroup。

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

9. 将 LUN 映射到上面创建的 igroup。为卷内的每个附加 LUN 连续增加 LUN ID。

```
lun map -path /vol/ora_01_biny/ora_01_biny_01 -igroup ora_01
-vserver svm_ora -lun-id 0
lun map -path /vol/ora_01_data/ora_01_data_01 -igroup ora_01
-vserver svm_ora -lun-id 1
lun map -path /vol/ora_01_data/ora_01_data_02 -igroup ora_01
-vserver svm_ora -lun-id 2
lun map -path /vol/ora_01_data/ora_01_data_03 -igroup ora_01
-vserver svm_ora -lun-id 3
lun map -path /vol/ora_01_data/ora_01_data_04 -igroup ora_01
-vserver svm_ora -lun-id 4
lun map -path /vol/ora_01_logs/ora_01_logs_01 -igroup ora_01
-vserver svm_ora -lun-id 5
lun map -path /vol/ora_01_logs/ora_01_logs_02 -igroup ora_01
-vserver svm_ora -lun-id 6
```

10. 验证 LUN 映射。

```
mapping show
```

预计返回结果如下：

```
FsxId02ad7bf3476b741df::> mapping show
  (lun mapping show)
Vserver    Path                                        Igroup   LUN ID
Protocol
---------- ------------------------------------------  -------  ------
--------
svm_ora    /vol/ora_01_biny/ora_01_biny_01             ora_01        0
iscsi
svm_ora    /vol/ora_01_data/ora_01_data_01             ora_01        1
iscsi
svm_ora    /vol/ora_01_data/ora_01_data_02             ora_01        2
iscsi
svm_ora    /vol/ora_01_data/ora_01_data_03             ora_01        3
iscsi
svm_ora    /vol/ora_01_data/ora_01_data_04             ora_01        4
iscsi
svm_ora    /vol/ora_01_logs/ora_01_logs_01             ora_01        5
iscsi
svm_ora    /vol/ora_01_logs/ora_01_logs_02             ora_01        6
iscsi
```

数据库存储配置

现在，在 EC2 实例主机上导入并设置 Oracle 网格基础设施和数据库安装的 FSx 存储。

1. 使用您的 SSH 密钥和 EC2 实例 IP 地址，以 ec2-user 身份通过 SSH 登录 EC2 实例。

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

2. 使用 SVM iSCSI IP 地址发现 FSx iSCSI 端点。然后更改为特定于您的环境的门户地址。

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

3. 通过登录每个目标建立 iSCSI 会话。

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

该命令的预期输出是：

```
[ec2-user@ip-172-30-15-58 ~]$ sudo iscsiadm --mode node -l all
Logging in to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
172.30.15.51,3260]
Logging in to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
172.30.15.13,3260]
Login to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
172.30.15.51,3260] successful.
Login to [iface: default, target: iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3, portal:
172.30.15.13,3260] successful.
```

4. 查看并验证活动 iSCSI 会话列表。

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

返回 iSCSI 会话。

```
[ec2-user@ip-172-30-15-58 ~]$ sudo iscsiadm --mode session
tcp: [1] 172.30.15.51:3260,1028 iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3 (non-flash)
tcp: [2] 172.30.15.13:3260,1029 iqn.1992-
08.com.netapp:sn.1f795e65c74911edb785affbf0a2b26e:vs.3 (non-flash)
```

5. 验证 LUN 是否已导入主机。

```
sudo sanlun lun show
```

这将返回来自 FSx 的 Oracle LUN 列表。

```
[ec2-user@ip-172-30-15-58 ~]$ sudo sanlun lun show
controller(7mode/E-Series)/                                    device
host                  lun
vserver(cDOT/FlashRay)        lun-pathname
filename        adapter    protocol   size    product

svm_ora                       /vol/ora_01_logs/ora_01_logs_02
/dev/sdn        host3      iSCSI      40g     cDOT
svm_ora                       /vol/ora_01_logs/ora_01_logs_01
/dev/sdm        host3      iSCSI      40g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_03
/dev/sdk        host3      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_04
/dev/sdl        host3      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_01
/dev/sdi        host3      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_02
/dev/sdj        host3      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_biny/ora_01_biny_01
/dev/sdh        host3      iSCSI      40g     cDOT
svm_ora                       /vol/ora_01_logs/ora_01_logs_02
/dev/sdg        host2      iSCSI      40g     cDOT
svm_ora                       /vol/ora_01_logs/ora_01_logs_01
/dev/sdf        host2      iSCSI      40g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_04
/dev/sde        host2      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_02
/dev/sdc        host2      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_03
/dev/sdd        host2      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_data/ora_01_data_01
/dev/sdb        host2      iSCSI      20g     cDOT
svm_ora                       /vol/ora_01_biny/ora_01_biny_01
/dev/sda        host2      iSCSI      40g     cDOT
```

6. 配置 `multipath.conf`包含以下默认和黑名单条目的文件。

```
sudo vi /etc/multipath.conf

defaults {
    find_multipaths yes
    user_friendly_names yes
}

blacklist {
    devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
    devnode "^hd[a-z]"
    devnode "^cciss.*"
}
```

7. 启动多路径服务。

```
sudo systemctl start multipathd
```

现在多路径设备出现在 `/dev/mapper`目录。

```
[ec2-user@ip-172-30-15-58 ~]$ ls -l /dev/mapper
total 0
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e68512d -> ../dm-0
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685141 -> ../dm-1
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685142 -> ../dm-2
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685143 -> ../dm-3
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685144 -> ../dm-4
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685145 -> ../dm-5
lrwxrwxrwx 1 root root       7 Mar 21 20:13
3600a09806c574235472455534e685146 -> ../dm-6
crw------- 1 root root 10, 236 Mar 21 18:19 control
```

8. 通过 SSH 以 fsxadmin 用户身份登录 FSx 集群，检索以 6c574xxx… 开头的每个 LUN 的序列十六进制数，十六进制数以 3600a0980 开头，这是 AWS 供应商 ID。

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

并返回如下结果：

```
FsxId02ad7bf3476b741df::> lun show -fields serial-hex
vserver path                            serial-hex
------- ------------------------------- ------------------------
svm_ora /vol/ora_01_biny/ora_01_biny_01 6c574235472455534e68512d
svm_ora /vol/ora_01_data/ora_01_data_01 6c574235472455534e685141
svm_ora /vol/ora_01_data/ora_01_data_02 6c574235472455534e685142
svm_ora /vol/ora_01_data/ora_01_data_03 6c574235472455534e685143
svm_ora /vol/ora_01_data/ora_01_data_04 6c574235472455534e685144
svm_ora /vol/ora_01_logs/ora_01_logs_01 6c574235472455534e685145
svm_ora /vol/ora_01_logs/ora_01_logs_02 6c574235472455534e685146
7 entries were displayed.
```

9. 更新 `/dev/multipath.conf`文件来为多路径设备添加一个用户友好的名称。

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

包含以下条目：

```
multipaths {
        multipath {
                wwid            3600a09806c574235472455534e68512d
                alias           ora_01_biny_01
        }
        multipath {
                wwid            3600a09806c574235472455534e685141
                alias           ora_01_data_01
        }
        multipath {
                wwid            3600a09806c574235472455534e685142
                alias           ora_01_data_02
        }
        multipath {
                wwid            3600a09806c574235472455534e685143
                alias           ora_01_data_03
        }
        multipath {
                wwid            3600a09806c574235472455534e685144
                alias           ora_01_data_04
        }
        multipath {
                wwid            3600a09806c574235472455534e685145
                alias           ora_01_logs_01
        }
        multipath {
                wwid            3600a09806c574235472455534e685146
                alias           ora_01_logs_02
        }
}
```

10. 重新启动多路径服务以验证 `/dev/mapper`已更改为 LUN 名称而非序列十六进制 ID。

```
sudo systemctl restart multipathd
```

查看 `/dev/mapper`返回如下：

```
[ec2-user@ip-172-30-15-58 ~]$ ls -l /dev/mapper
total 0
crw------- 1 root root 10, 236 Mar 21 18:19 control
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_biny_01 -> ../dm-
0
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_data_01 -> ../dm-
1
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_data_02 -> ../dm-
2
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_data_03 -> ../dm-
3
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_data_04 -> ../dm-
4
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_logs_01 -> ../dm-
5
lrwxrwxrwx 1 root root       7 Mar 21 20:41 ora_01_logs_02 -> ../dm-
6
```

11. 使用单个主分区对二进制 LUN 进行分区。

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

12. 使用 XFS 文件系统格式化分区的二进制 LUN。

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

13. 挂载二进制 LUN 到 `/u01`。

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

14. 改变 `/u01`挂载点所有权归 Oracle 用户及其关联的主要组。

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

15. 查找二进制 LUN 的 UUI。

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

16. 添加挂载点 `/etc/fstab`。

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

添加以下行。

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

> 重要的是仅使用 UUID 和 nofail 选项挂载二进制文件，以避免 EC2 实例重启期间可能出现的根锁问题。

17. 以 root 用户身份添加 Oracle 设备的 udev 规则。

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

包括以下条目：

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

18. 以 root 用户身份重新加载 udev 规则。

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

19. 以root用户身份触发udev规则。

```
udevadm trigger
```

20. 以 root 用户身份重新加载 multipathd。

```
systemctl restart multipathd
```

21. 重新启动 EC2 实例主机。

Oracle 网格基础设施安装

1. 通过 SSH 以 ec2-user 身份登录 EC2 实例，并通过取消注释来启用密码验证 `PasswordAuthentication yes`然后注释掉 `PasswordAuthentication no`。

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

2. 重新启动 sshd 服务。

```
sudo systemctl restart sshd
```

3. 重置 Oracle 用户密码。

```
sudo passwd oracle
```

4. 以 Oracle Restart 软件所有者用户 (oracle) 身份登录。创建 Oracle 目录如下：

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

5. 更改目录权限设置。

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

6. 创建网格主目录并更改至该目录。

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

7. 解压网格安装文件。

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

8. 从网格主页中删除 `OPatch`目录。

```
rm -rf OPatch
```

9. 从网格主页，解压缩 `p6880880_190000_Linux-x86-64.zip`。

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

10. 从网格主页，修改 `cv/admin/cvu_config`，取消注释并替换 `CV_ASSUME_DISTID=OEL5`和 `CV_ASSUME_DISTID=OL7`。

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

11. 更新 `gridsetup.rsp`文件进行静默安装，并将 rsp 文件放在 `/tmp/archive`目录。 rsp 文件应涵盖 A、B 和 G 部分，并包含以下信息：

```
INVENTORY_LOCATION=/u01/app/oraInventory
oracle.install.option=HA_CONFIG
ORACLE_BASE=/u01/app/oracle
oracle.install.asm.OSDBA=dba
oracle.install.asm.OSOPER=oper
oracle.install.asm.OSASM=asm
oracle.install.asm.SYSASMPassword="SetPWD"
oracle.install.asm.diskGroup.name=DATA
oracle.install.asm.diskGroup.redundancy=EXTERNAL
oracle.install.asm.diskGroup.AUSize=4
oracle.install.asm.diskGroup.disks=/dev/mapper/ora_01_data_01,/dev/m
apper/ora_01_data_02,/dev/mapper/ora_01_data_03,/dev/mapper/ora_01_d
ata_04
oracle.install.asm.diskGroup.diskDiscoveryString=/dev/mapper/*
oracle.install.asm.monitorPassword="SetPWD"
oracle.install.asm.configureAFD=true
```

12. 以 root 用户身份登录 EC2 实例并设置 `ORACLE_HOME`和 `ORACLE_BASE`。

```
export ORACLE_HOME=/u01/app/oracle/product/19.0.0/grid
export ORACLE_BASE=/tmp
cd /u01/app/oracle/product/19.0.0/grid/bin
```

13. 提供磁盘设备以供 Oracle ASM 过滤器驱动程序使用。

```
 ./asmcmd afd_label DATA01 /dev/mapper/ora_01_data_01 --init

 ./asmcmd afd_label DATA02 /dev/mapper/ora_01_data_02 --init

 ./asmcmd afd_label DATA03 /dev/mapper/ora_01_data_03 --init

 ./asmcmd afd_label DATA04 /dev/mapper/ora_01_data_04 --init

 ./asmcmd afd_label LOGS01 /dev/mapper/ora_01_logs_01 --init

 ./asmcmd afd_label LOGS02 /dev/mapper/ora_01_logs_02 --init
```

14. 安装 `cvuqdisk-1.0.10-1.rpm`。

```
rpm -ivh /u01/app/oracle/product/19.0.0/grid/cv/rpm/cvuqdisk-1.0.10-
1.rpm
```

15. 取消设置 `$ORACLE_BASE`。

```
unset ORACLE_BASE
```

16. 以 Oracle 用户身份登录 EC2 实例并提取补丁 `/tmp/archive`文件夹。

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

17. 从 grid home /u01/app/oracle/product/19.0.0/grid 并以 oracle 用户身份启动 `gridSetup.sh`用于电网基础设施安装。

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

忽略有关电网基础设施错误组的警告。我们使用单个 Oracle 用户来管理 Oracle Restart，因此这是预料之中的。

18. 以 root 用户身份执行以下脚本：

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

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

19. 以 root 用户身份重新加载 multipathd。

```
systemctl restart multipathd
```

20. 以Oracle用户执行以下命令完成配置：

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

21. 以 Oracle 用户身份，从 $GRID_HOME 创建 LOGS 磁盘组。

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

22. 以 Oracle 用户身份，在安装配置后验证网格服务。

```
bin/crsctl stat res -t
+
Name                 Target  State        Server
State details
Local Resources
ora.DATA.dg          ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.LISTENER.lsnr    ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.LOGS.dg          ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.asm              ONLINE  ONLINE       ip-172-30-15-58
Started,STABLE
ora.ons              OFFLINE OFFLINE      ip-172-30-15-58
STABLE
Cluster Resources
ora.cssd             ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.diskmon          OFFLINE OFFLINE
STABLE
ora.driver.afd       ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.evmd             ONLINE  ONLINE       ip-172-30-15-58
STABLE
```

23. 验证 ASM 过滤器驱动程序状态。

```
[oracle@ip-172-30-15-58 grid]$ export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/grid
[oracle@ip-172-30-15-58 grid]$ export ORACLE_SID=+ASM
[oracle@ip-172-30-15-58 grid]$ export PATH=$PATH:$ORACLE_HOME/bin
[oracle@ip-172-30-15-58 grid]$ asmcmd
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  1048576
81920    81847                0           81847              0
N  DATA/
MOUNTED  EXTERN  N         512             512   4096  1048576
81920    81853                0           81853              0
N  LOGS/
ASMCMD> afd_state
ASMCMD-9526: The AFD state is 'LOADED' and filtering is 'ENABLED' on
host 'ip-172-30-15-58.ec2.internal'
```

Oracle数据库安装

1. 以 Oracle 用户身份登录并取消设置 `$ORACLE_HOME`和 `$ORACLE_SID`如果已设置。

```
unset ORACLE_HOME
unset ORACLE_SID
```

2. 创建 Oracle DB 主目录并更改至该目录。

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

3. 解压缩 Oracle DB 安装文件。

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

4. 从数据库主目录删除 `OPatch`目录。

```
rm -rf OPatch
```

5. 从 DB home 解压缩 `p6880880_190000_Linux-x86-64.zip`。

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

6. 从 DB 主页修改 `cv/admin/cvu_config`，并取消注释并替换 `CV_ASSUME_DISTID=OEL5`和 `CV_ASSUME_DISTID=OL7`。

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

7. 从 `/tmp/archive`目录中，解压DB 19.18 RU补丁。

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

8. 更新 DB 静默安装标准 rsp 文件 `/tmp/archive/dbinstall.rsp`目录中的相关部分具有以下值：

```
oracle.install.option=INSTALL_DB_SWONLY
UNIX_GROUP_NAME=oinstall
INVENTORY_LOCATION=/u01/app/oraInventory
ORACLE_HOME=/u01/app/oracle/product/19.0.0/db1
ORACLE_BASE=/u01/app/oracle
oracle.install.db.InstallEdition=EE
oracle.install.db.OSDBA_GROUP=dba
oracle.install.db.OSOPER_GROUP=oper
oracle.install.db.OSBACKUPDBA_GROUP=oper
oracle.install.db.OSDGDBA_GROUP=dba
oracle.install.db.OSKMDBA_GROUP=dba
oracle.install.db.OSRACDBA_GROUP=dba
oracle.install.db.rootconfig.executeRootScript=false
```

9. 从 db1 home /u01/app/oracle/product/19.0.0/db1 执行静默纯软件 DB 安装。

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

10. 以 root 用户身份运行 `root.sh`仅安装软件后的脚本。

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

11. 以 Oracle 用户身份更新标准 `dbca.rsp`文件的相关部分包含以下条目：

```
gdbName=db1.demo.netapp.com
sid=db1
createAsContainerDatabase=true
numberOfPDBs=3
pdbName=db1_pdb
useLocalUndoForPDBs=true
pdbAdminPassword="yourPWD"
templateName=General_Purpose.dbc
sysPassword="yourPWD"
systemPassword="yourPWD"
dbsnmpPassword="yourPWD"
datafileDestination=+DATA
recoveryAreaDestination=+LOGS
storageType=ASM
diskGroupName=DATA
characterSet=AL32UTF8
nationalCharacterSet=AL16UTF16
listeners=LISTENER
databaseType=MULTIPURPOSE
automaticMemoryManagement=false
totalMemory=8192
```

12. 以 Oracle 用户身份，从 $ORACLE_HOME 目录，使用 dbca 启动 DB 创建。

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

output:
Prepare for db operation
7% complete
Registering database with Oracle Restart
11% complete
Copying database files
33% complete
Creating and starting Oracle instance
35% complete
38% complete
42% complete
45% complete
48% complete
Completing Database Creation
53% complete
55% complete
56% complete
Creating Pluggable Databases
60% complete
64% complete
69% complete
78% complete
Executing Post Configuration Actions
100% complete
Database creation complete. For details check the logfiles at:
 /u01/app/oracle/cfgtoollogs/dbca/db1.
Database Information:
Global Database Name:db1.demo.netapp.com
System Identifier(SID):db1
Look at the log file "/u01/app/oracle/cfgtoollogs/dbca/db1/db1.log"
for further details.
```

13. 以 Oracle 用户身份，在创建数据库后验证 Oracle Restart HA 服务。

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

Name            Target  State        Server                   State
details

Local Resources

ora.DATA.dg     ONLINE  ONLINE       ip-172-30-15-58          STABLE
ora.LISTENER.lsnr   ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.LOGS.dg     ONLINE  ONLINE       ip-172-30-15-58          STABLE
ora.asm     ONLINE  ONLINE       ip-172-30-15-58
Started,STABLE
ora.ons     OFFLINE OFFLINE      ip-172-30-15-58          STABLE

Cluster Resources

ora.cssd            ONLINE  ONLINE       ip-172-30-15-58
STABLE
ora.db1.db      ONLINE  ONLINE       ip-172-30-15-58
Open,HOME=/u01/app/oracle/product/19.0.0/db1,STABLE
ora.diskmon     OFFLINE OFFLINE                               STABLE
ora.driver.afd  ONLINE  ONLINE       ip-172-30-15-58          STABLE
ora.evmd        ONLINE  ONLINE       ip-172-30-15-58          STABLE
```

14. 设置 Oracle 用户 `.bash_profile`。

```
vi ~/.bash_profile
```

15. 添加以下条目：

```
export ORACLE_HOME=/u01/app/oracle/product/19.0.0/db1
export ORACLE_SID=db1
export PATH=$PATH:$ORACLE_HOME/bin
alias asm='export
ORACLE_HOME=/u01/app/oracle/product/19.0.0/grid;export
ORACLE_SID=+ASM;export PATH=$PATH:$ORACLE_HOME/bin'
```

16. 验证创建的 CDB/PDB。

```
source /home/oracle/.bash_profile

sqlplus / as sysdba
```

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

NAME      OPEN_MODE

DB1       READ WRITE

SQL> select name from v$datafile;

NAME

+DATA/DB1/DATAFILE/system.256.1132176177
+DATA/DB1/DATAFILE/sysaux.257.1132176221
+DATA/DB1/DATAFILE/undotbs1.258.1132176247
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/system.265.11321
77009
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/sysaux.266.11321
77009
+DATA/DB1/DATAFILE/users.259.1132176247
+DATA/DB1/86B637B62FE07A65E053F706E80A27CA/DATAFILE/undotbs1.267.113
2177009
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/system.271.11321
77853
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/sysaux.272.11321
77853
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/undotbs1.270.113
2177853
+DATA/DB1/F7852758DCD6B800E0533A0F1EAC1DC6/DATAFILE/users.274.113217
7871

NAME

+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/system.276.11321
77871
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/sysaux.277.11321
77871
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/undotbs1.275.113
2177871
+DATA/DB1/F785288BBCD1BA78E0533A0F1EACCD6F/DATAFILE/users.279.113217
7889
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/system.281.11321
77889
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/sysaux.282.11321
77889
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/undotbs1.280.113
2177889
```

```
+DATA/DB1/F78529A14DD8BB18E0533A0F1EACB8ED/DATAFILE/users.284.113217
7907

19 rows selected.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED

         2 PDB$SEED                       READ ONLY  NO
         3 DB1_PDB1                       READ WRITE NO
         4 DB1_PDB2                       READ WRITE NO
         5 DB1_PDB3                       READ WRITE NO
SQL>
```

17. 将 DB 恢复目标大小设置为 +LOGS 磁盘组大小。

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

18. 使用sqlplus登录数据库并启用存档日志模式。

```
sqlplus /as sysdba.

shutdown immediate;

startup mount;

alter database archivelog;

alter database open;
```

这完成了 Oracle 19c 版本 19.18 在Amazon FSx ONTAP和 EC2 计算实例上的重新启动部署。如果需要，NetApp建议将 Oracle 控制文件和联机日志文件重新定位到 +LOGS 磁盘组。

自动部署选项

参考"TR-4986：使用 iSCSI 在Amazon FSx ONTAP上简化、自动化 Oracle 部署"了解详情。

使用SnapCenter服务备份、恢复和克隆 Oracle 数据库

看"适用于 Oracle 的SnapCenter服务"有关使用NetApp BlueXP控制台备份、恢复和克隆 Oracle 数据库的详细信息。

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：

- 使用新数据库安装为独立服务器安装 Oracle Grid Infrastructure

  "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"

- 使用响应文件安装和配置 Oracle 数据库

  "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"

- Amazon FSx ONTAP

  "https://aws.amazon.com/fsx/netapp-ontap/"

- Amazon EC2

  https://aws.amazon.com/pm/ec2/?trk=36c6da98-7b20-48fa-8225-4784bced9843&sc_channel=ps&s_kwcid=AL!4422!3!467723097970!e!!g!!aws%20ec2&ef_id=Cj0KCQiA54KfBhCKARIsAJzSrdqwQrghn6I71jiWzSeaT9Uh1-vY-VfhJixF-xnv5rWwn2S7RqZOTQ0aAh7eEALw_wcB:G:s&s_kwcid=AL!4422!3!467723097970!e!!g!!aws%20ec2

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。