



TR-4983：在NetApp ASA上使用 iSCSI 实现简化、自动化的 Oracle 部署

NetApp database solutions

NetApp
August 18, 2025

目录

TR-4983：在NetApp ASA上使用 iSCSI 实现简化、自动化的 Oracle 部署	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
实验室环境中的 Oracle 数据库配置	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
自动化参数文件	4
参数文件配置	5
脚本执行	8
执行后验证	10
使用SnapCenter进行 Oracle 备份、恢复和克隆	16
在哪里可以找到更多信息	16

TR-4983：在NetApp ASA上使用 iSCSI 实现简化、自动化的 Oracle 部署

Allen Cao、Niyaz Mohamed，NetApp

该解决方案提供了在NetApp ASA阵列中自动部署和保护 Oracle 的概述和详细信息，该阵列作为使用 iSCSI 协议的主数据库存储，并在独立 ReStart 中使用 asm 作为卷管理器配置 Oracle 数据库。

目的

NetApp ASA系统为您的 SAN 基础架构提供现代解决方案。它们可以大规模简化并使您能够加速业务关键型应用程序（如数据库），确保您的数据始终可用（99.9999%的正常运行时间），并降低 TCO 和碳足迹。NetApp ASA系统包括专为性能要求最高的应用程序设计的 A 系列型号和针对经济高效、大容量部署进行优化的 C 系列型号。ASA A 系列和 C 系列系统共同提供卓越的性能，以改善客户体验并缩短获得结果的时间，保持关键业务数据的可用性、受保护性和安全性，并为任何工作负载提供更有效的容量，并以业界最有效的保证为后盾。

本文档演示了使用 Ansible 自动化在ASA系统构建的 SAN 环境中简化 Oracle 数据库的部署。Oracle 数据库部署在独立的 ReStart 配置中，使用 iSCSI 协议进行数据访问，并使用 Oracle ASM 进行ASA存储阵列上的数据库磁盘管理。它还提供有关使用NetApp SnapCenter UI 工具备份、恢复和克隆 Oracle 数据库的信息，以便在NetApp ASA系统中实现高效的存储数据库操作。

此解决方案适用于以下用例：

- 在NetApp ASA系统中自动部署 Oracle 数据库作为主数据库存储
- 使用NetApp SnapCenter工具在NetApp ASA系统中备份和恢复 Oracle 数据库
- 使用NetApp SnapCenter工具在NetApp ASA系统中克隆 Oracle 数据库以用于开发/测试或其他用例

受众

此解决方案适用于以下人群：

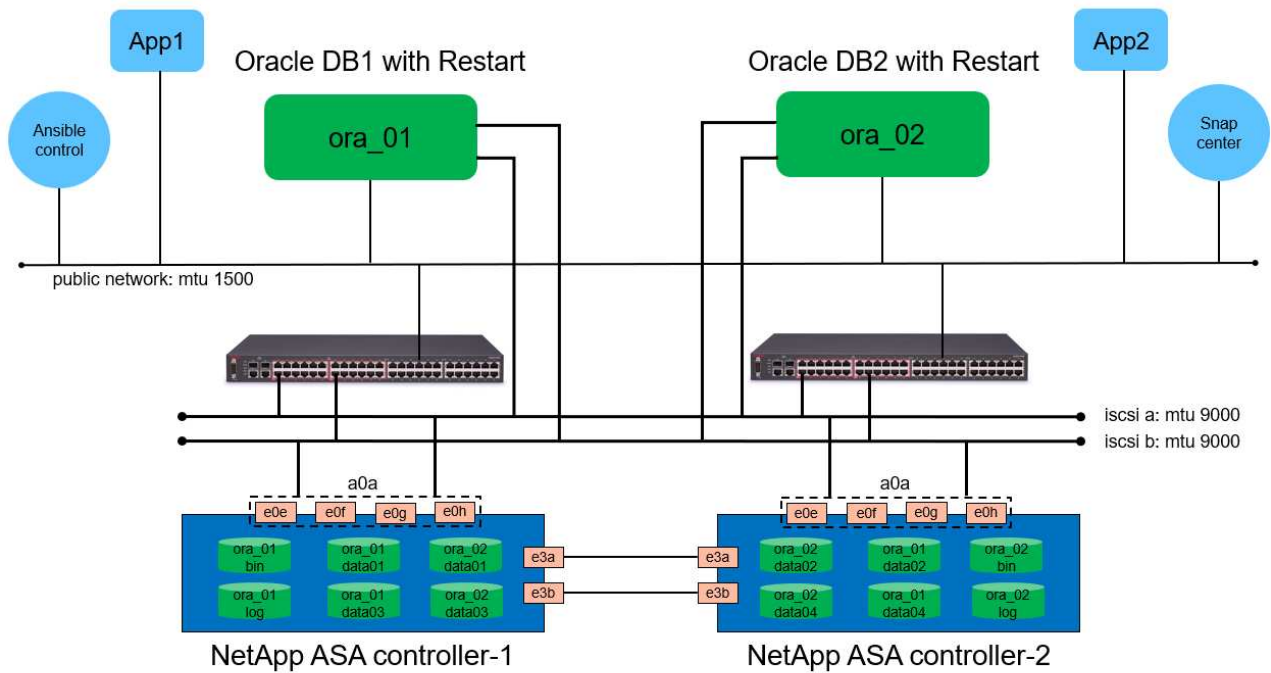
- 一位想要在NetApp ASA系统中部署 Oracle 的 DBA。
- 一位数据库解决方案架构师想要在NetApp ASA系统中测试 Oracle 工作负载。
- 一位存储管理员想要在NetApp ASA系统上部署和管理 Oracle 数据库。
- 希望在NetApp ASA系统中建立 Oracle 数据库的应用程序所有者。

解决方案测试和验证环境

该解决方案的测试和验证是在实验室环境中进行的，可能与最终部署环境不匹配。请参阅[部署考虑的关键因素](#)了解更多信息。

架构

Simplified, Automated Oracle Database Deployment on NetApp ASA with iSCSI



NetApp

硬件和软件组件

硬件		
NetApp ASA A400	版本 9.13.1P1	2 个 NS224 磁盘架、48 个 NVMe AFF驱动器，总容量为 69.3 TiB
UCSB-B200-M4	英特尔® 至强® CPU E5-2690 v4 @ 2.60GHz	4节点VMware ESXi集群
软件		
红帽Linux	RHEL-8.6, 4.18.0-372.9.1.el8.x86_64 内核	部署 RedHat 订阅进行测试
Windows Server	2022 标准版, 10.0.20348 内部版本 20348	托管SnapCenter服务器
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
Oracle OPatch	版本 12.2.0.1.36	最新补丁p6880880_190000_Linux-x86-64.zip
SnapCenter 服务器	版本 4.9P1	工作组部署

VMware vSphere 虚拟机管理程序	版本 6.5.0.20000	VMware Tools, 版本: 11365 - Linux, 12352 - Windows
打开 JDK	版本 java-1.8.0-openjdk.x86_64	数据库虚拟机上的SnapCenter插件要求

实验室环境中的 Oracle 数据库配置

服务器	数据库	数据库存储
ora_01	NTAP1(NTAP1_PDB1、NTAP1_PDB2、NTAP1_PDB3)	ASA A400上的 iSCSI LUN
ora_02	NTAP2(NTAP2_PDB1、NTAP2_PDB2、NTAP2_PDB3)	ASA A400上的 iSCSI LUN

部署考虑的关键因素

- Oracle 数据库存储布局。在此自动化 Oracle 部署中，我们默认配置四个数据库卷来托管 Oracle 二进制文件、数据和日志。然后，我们从数据和日志 LUN 创建两个 ASM 磁盘组。在 +DATA asm 磁盘组中，我们在每个ASA A400集群节点上的卷中配置两个数据 lun。在 +LOGS asm 磁盘组中，我们在单个ASA A400节点上的日志卷中创建两个 lun。ONTAP卷内布局的多个 LUN 通常可提供更好的性能。
- *多个数据库服务器部署。*自动化解决方案可以在单个 Ansible 剧本运行中将 Oracle 容器数据库部署到多个 DB 服务器。无论数据库服务器的数量有多少，剧本的执行都保持不变。如果发生多数据库服务器部署，剧本将使用算法构建，以最佳方式将数据库 LUN 放置在ASA A400的双控制器上。服务器主机索引中奇数 DB 服务器的二进制文件和日志 lun 位于控制器 1 上。服务器主机索引中偶数 DB 服务器的二进制文件和日志 lun 位于控制器 2 上。DB数据lun均匀分布到两个控制器上。Oracle ASM 将两个控制器上的数据 LUN 合并为一个 ASM 磁盘组，以充分利用两个控制器的处理能力。
- iSCSI 配置。数据库虚拟机通过 iSCSI 协议连接到ASA存储以进行存储访问。您应该在每个控制器节点上配置双路径以实现冗余，并在 DB 服务器上设置 iSCSI 多路径以实现多路径存储访问。在存储网络上启用巨型帧以最大化性能和吞吐量。
- *您创建的每个 Oracle ASM 磁盘组使用的 Oracle ASM 冗余级别。*由于ASA A400在RAID DP中配置存储以在集群磁盘级别进行数据保护，因此您应该使用 External Redundancy，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SnapCenter software套件，用于数据库备份、恢复和克隆，并具有用户友好的 UI 界面。NetApp建议实施这样的管理工具，以实现快速（一分钟内）的 SnapShot 备份、快速（几分钟内）的数据库恢复和数据库克隆。

解决方案部署

以下部分提供了在NetApp ASA A400中自动部署和保护 Oracle 19c 的分步过程，其中通过 iSCSI 将数据库 lun 直接安装到单节点重新启动配置中的 DB VM，并使用 Oracle ASM 作为数据库卷管理器。

部署先决条件

部署需要以下先决条件。

1. 假设已经安装并配置了NetApp ASA存储阵列。这包括 iSCSI 广播域、两个控制器节点上的 LACP 接口组 a0a、两个控制器节点上的 iSCSI VLAN 端口 (a0a-<iscsi-a-vlan-id>、a0a-<iscsi-b-vlan-id>)。如果需要帮助，以下链接提供了详细的分步说明。["详细指南 - ASA A400"](#)
2. 将 Linux VM 配置为 Ansible 控制器节点，并安装最新版本的 Ansible 和 Git。详细信息请参考以下链接：["NetApp解决方案自动化入门"](#)在第 - Setup the Ansible Control Node for CLI deployments on RHEL / CentOS`或者`Setup the Ansible Control Node for CLI deployments on Ubuntu / Debian。
3. 克隆适用于 iSCSI 的NetApp Oracle 部署自动化工具包的副本。

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

4. 配置 Windows 服务器以运行最新版本的NetApp SnapCenter UI 工具。详细信息请参考以下链接：["安装SnapCenter服务器"](#)
5. 构建两个 RHEL Oracle DB 服务器，可以是裸机，也可以是虚拟化 VM。在 DB 服务器上创建一个具有 sudo 权限且没有密码权限的管理员用户，并在 Ansible 主机和 Oracle DB 服务器主机之间启用 SSH 私钥/公钥认证。阶段遵循数据库服务器 /tmp/archive 目录中的 Oracle 19c 安装文件。

```
installer_archives:  
  - "LINUX.X64_193000_grid_home.zip"  
  - "p34762026_190000_Linux-x86-64.zip"  
  - "LINUX.X64_193000_db_home.zip"  
  - "p34765931_190000_Linux-x86-64.zip"  
  - "p6880880_190000_Linux-x86-64.zip"
```



确保您已在 Oracle VM 根卷中分配至少 50G，以便有足够的空间来存放 Oracle 安装文件。

6. 观看以下视频：

[使用 iSCSI 在NetApp ASA上简化和自动化 Oracle 部署](#)

自动化参数文件

Ansible playbook 使用预定义参数执行数据库安装和配置任务。对于此 Oracle 自动化解决方案，有三个用户定义参数文件需要在剧本执行之前用户输入。

- 主机 - 定义自动化剧本运行的目标。
- vars/vars.yml - 定义适用于所有目标的变量的全局变量文件。
- host_vars/host_name.yml - 定义仅适用于本地目标的变量的本地变量文件。在我们的用例中，这些是 Oracle DB 服务器。

除了这些用户定义的变量文件之外，还有几个默认变量文件，其中包含默认参数，除非必要，否则不需要更改。以下部分展示了如何配置用户定义变量文件。

参数文件配置

1. Ansible 目标 `hosts` 文件配置:

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

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

2. 全球的 `vars/vars.yml` 文件配置

```
#####
#####
#####          Oracle 19c deployment global user
configurable variables          #####
#####          Consolidate all variables from ONTAP, linux
and oracle          #####
#####
#####

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

# Enter the supported ONTAP platform: on-prem, aws-fsx.
ontap_platform: on-prem

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

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

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

```

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

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

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

# Enter RHEL subscription to enable repo
redhat_sub_username: xxxxxxxx
redhat_sub_password: "xxxxxxxx"

#####
#####
###                               Oracle DB env specific config variables
###
#####
#####

# Enter Database domain name
db_domain: solutions.netapp.com

# Enter initial password for all required Oracle passwords. Change
them after installation.
initial_pwd_all: xxxxxxxx

```

3. 本地数据库服务器 `host\_vars/host\_name.yml` 配置

```
# User configurable Oracle host specific parameters

# Enter container database SID. By default, a container DB is
created with 3 PDBs within the CDB
oracle_sid: NTAP1

# Enter database shared memory size or SGA. CDB is created with SGA
at 75% of memory_limit, MB. The grand total of SGA should not exceed
75% available RAM on node.
memory_limit: 8192
```

剧本执行

自动化工具包中共有六个剧本。每个执行不同的任务块并服务于不同的目的。

```
0-all_playbook.yml - execute playbooks from 1-4 in one playbook run.
1-ansible_requirements.yml - set up Ansible controller with required
libs and collections.
2-linux_config.yml - execute Linux kernel configuration on Oracle DB
servers.
3-ontap_config.yml - configure ONTAP svm/volumes/luns for Oracle
database and grant DB server access to luns.
4-oracle_config.yml - install and configure Oracle on DB servers for
grid infrastructure and create a container database.
5-destroy.yml - optional to undo the environment to dismantle all.
```

有三个选项可以使用以下命令运行剧本。

1. 在一次组合运行中执行所有部署剧本。

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

2. 按照 1-4 的数字序列逐个执行剧本。

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

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

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

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

3. 使用标签执行 0-all_playbook.yml。

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

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

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

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

4. 撤消环境

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

执行后验证

剧本运行后，以 oracle 用户身份登录 Oracle DB 服务器，验证 Oracle 网格基础架构和数据库是否成功创建。以下是主机 ora_01 上的 Oracle 数据库验证的示例。

1. 验证创建的电网基础设施和资源。

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

```

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

racle/product/19.0.0

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

```



忽略 `Not All Endpoints Registered` 在州详细信息中。这是由于手动和动态数据库注册与监听器发生冲突而导致的，可以安全地忽略。

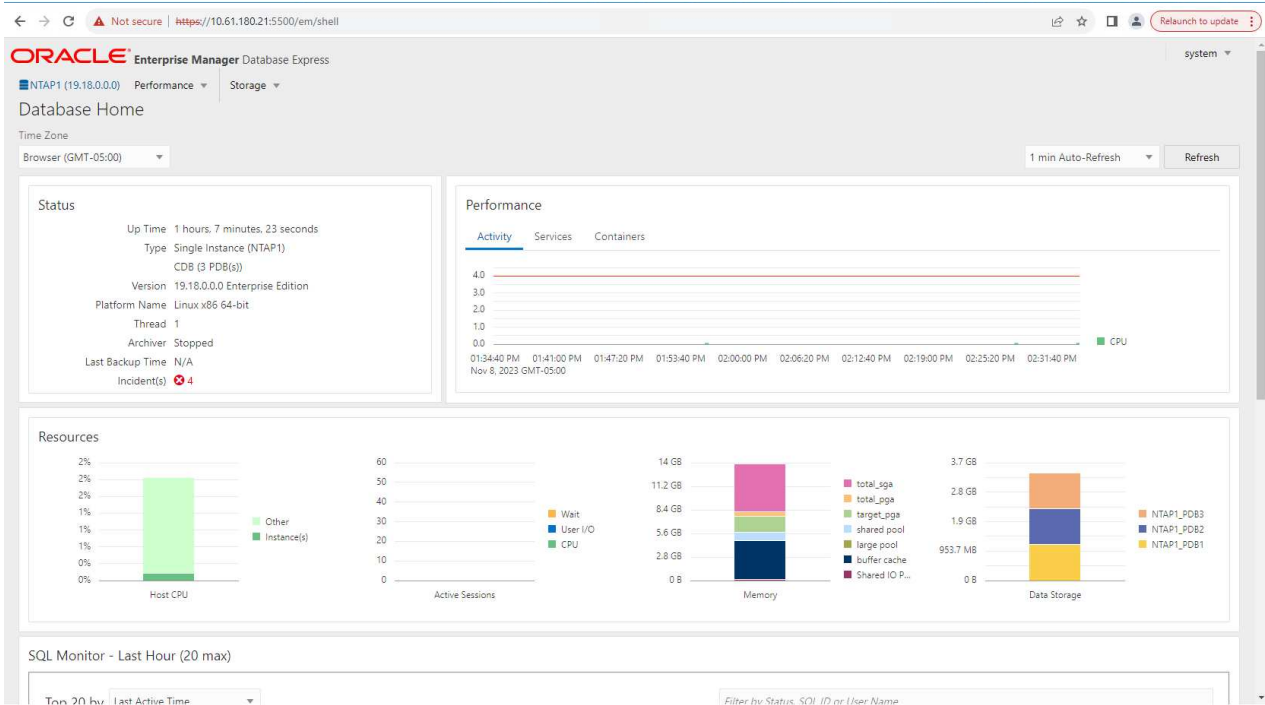
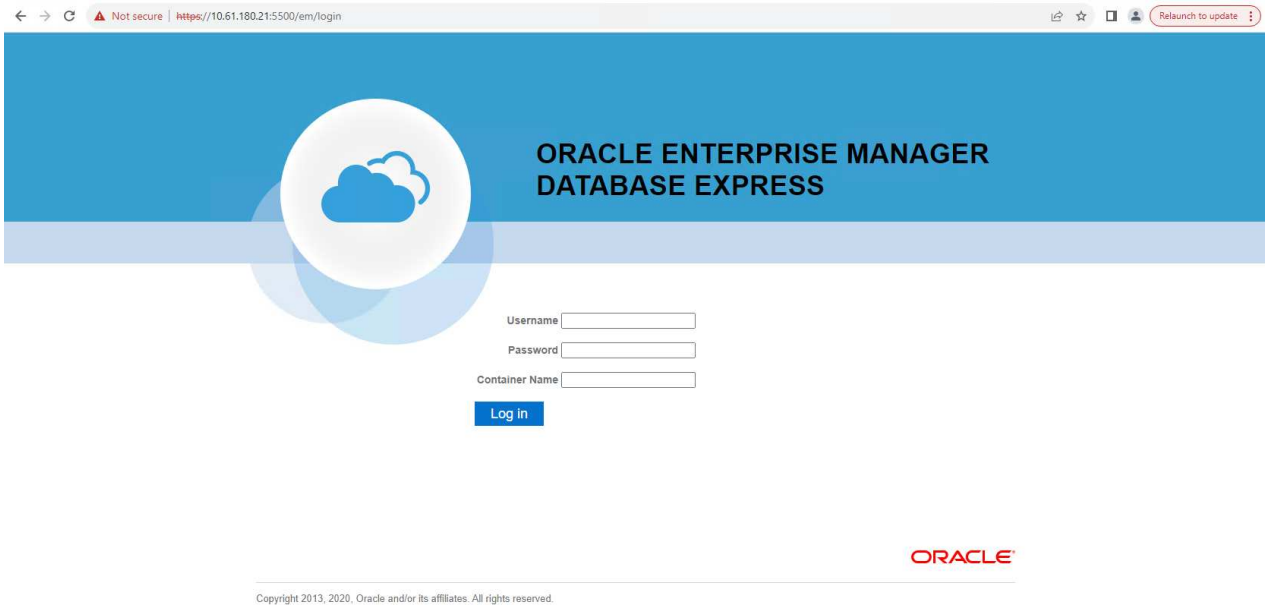
2. 验证 ASM 过滤器驱动程序是否按预期工作。

```

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

```

3. 登录 Oracle Enterprise Manager Express 来验证数据库。



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

```
SQL> show pdbs
```

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

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

Session altered.

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

```
DBMS_XDB_CONFIG.GETHTTPS()
-----
0
```

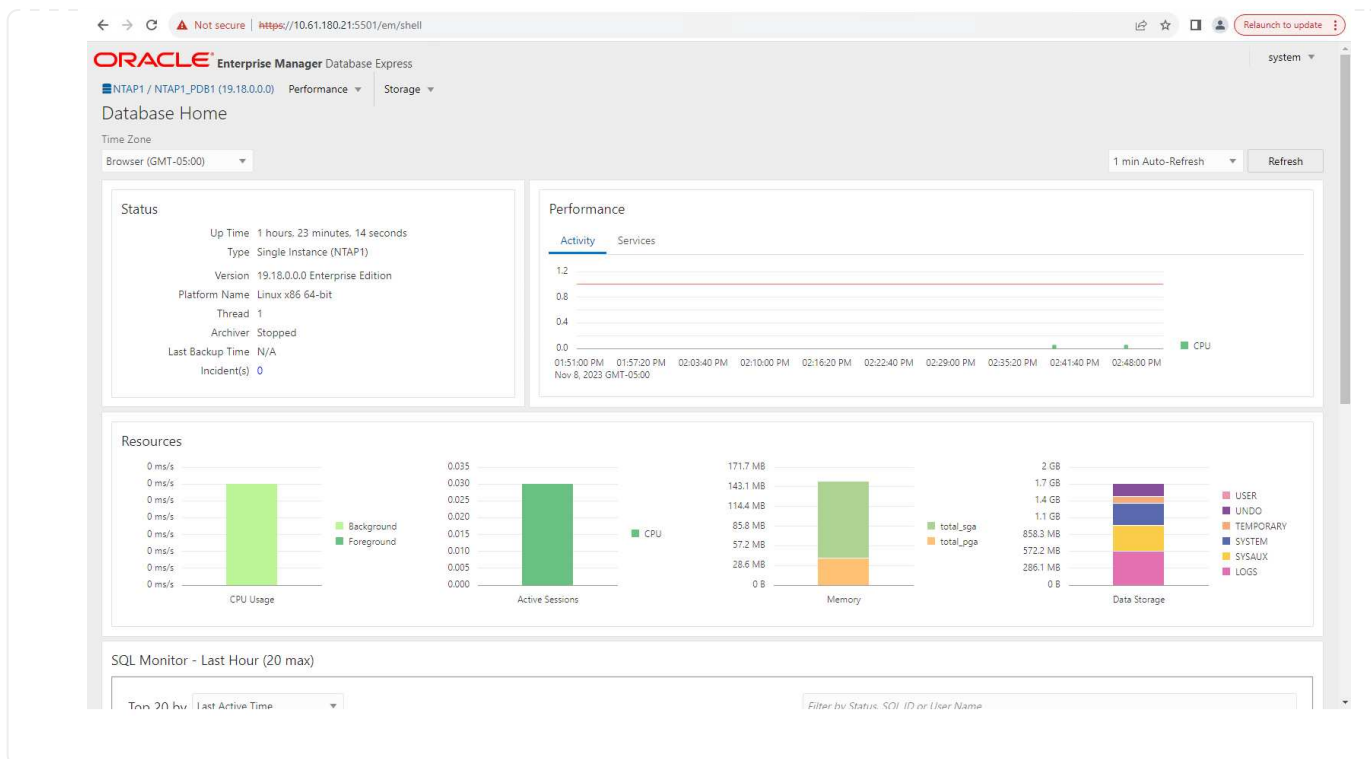
```
SQL> exec DBMS_XDB_CONFIG.SETHTTPS(5501);
```

PL/SQL procedure successfully completed.

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

```
DBMS_XDB_CONFIG.GETHTTPS()
-----
5501
```

login to NTAP1_PDB1 from port 5501.



使用SnapCenter进行 Oracle 备份、恢复和克隆

参考 TR-4979"VMware Cloud on AWS 中简化的自主管理 Oracle，带有客户机安装的 FSx ONTAP"部分`Oracle backup, restore, and clone with SnapCenter`有关设置SnapCenter和执行数据库备份、恢复和克隆工作流的详细信息。

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：

- NETAPP ASA：全闪存 SAN 阵列

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

- 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"](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"](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)

- 将 Red Hat Enterprise Linux 8.2 与ONTAP结合使用

["https://docs.netapp.com/us-en/ontap-sanhost/hu_rhel_82.html#all-san-array-configurations"](https://docs.netapp.com/us-en/ontap-sanhost/hu_rhel_82.html#all-san-array-configurations)

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。