



# 在 **NFS** 上自動部署 **Oracle 19c for ONTAP**

## NetApp database solutions

NetApp  
August 18, 2025

This PDF was generated from <https://docs.netapp.com/zh-tw/netapp-solutions-databases/automation/marketing-overview.html> on August 18, 2025. Always check docs.netapp.com for the latest.

# 目錄

|                                        |    |
|----------------------------------------|----|
| 在 NFS 上自動部署 Oracle 19c for ONTAP ..... | 1  |
| 解決方案概述 .....                           | 1  |
| 在 NFS 上自動部署 Oracle19c for ONTAP .....  | 1  |
| 入門 .....                               | 2  |
| AWX/塔 .....                            | 2  |
| 透過 Ansible 控制主機的 CLI .....             | 2  |
| 要求 .....                               | 2  |
| 自動化詳細信息 .....                          | 3  |
| 預設參數 .....                             | 3  |
| 部署說明 .....                             | 3  |
| 執照 .....                               | 4  |
| 逐步部署流程 .....                           | 4  |
| AWX/Tower 部署 Oracle 19c 資料庫 .....      | 4  |
| 逐步部署流程 .....                           | 13 |
| CLI 部署 Oracle 19c 資料庫 .....            | 13 |

# 在 NFS 上自動部署 Oracle 19c for ONTAP

## 解決方案概述

本頁介紹在NetApp ONTAP儲存上部署 Oracle19c 的自動化方法。

### 在 NFS 上自動部署 Oracle19c for ONTAP

組織正在實現其環境自動化，以提高效率、加速部署並減少人工工作量。Ansible 等組態管理工具正被用來簡化企業資料庫操作。在此解決方案中，我們示範如何使用 Ansible 自動設定和設定具有NetApp ONTAP的 Oracle 19c。透過讓儲存管理員、系統管理員和 DBA 能夠一致、快速地部署新儲存、配置資料庫伺服器和安裝 Oracle 19c 軟體，您可以獲得以下好處：

- 消除設計複雜性和人為錯誤，並實施可重複的一致部署和最佳實踐
- 減少儲存配置、資料庫主機配置和 Oracle 安裝的時間
- 提高資料庫管理員、系統和儲存管理員的工作效率
- 輕鬆實現儲存和資料庫的擴展

NetApp為客戶提供經過驗證的 Ansible 模組和角色，以加速 Oracle 資料庫環境的部署、設定和生命週期管理。此解決方案提供了說明和 Ansible 劇本程式碼，以協助您：

- 為 Oracle 資料庫建立和配置ONTAP NFS 存儲
- 在 RedHat Enterprise Linux 7/8 或 Oracle Linux 7/8 上安裝 Oracle 19c
- 在ONTAP NFS 儲存上配置 Oracle 19c

欲了解更多詳細資訊或開始，請觀看下面的概述影片。

#### AWX/塔式部署

第 1 部分：入門、需求、自動化詳細資訊和初始 AWX/Tower 配置

##### [AWX 部署](#)

第 2 部分：變數和運行劇本

##### [AWX 劇本運行](#)

#### CLI部署

第 1 部分：入門、需求、自動化詳細資訊和 Ansible 控制主機設置

##### [CLI部署](#)

第 2 部分：變數和運行劇本

##### [CLI Playbook 運行](#)

# 入門

此解決方案設計為在 AWX/Tower 環境中運作或透過 Ansible 控制主機上的 CLI 運作。

## AWX/塔

對於 AWX/Tower 環境，將引導您建立ONTAP叢集管理和 Oracle 伺服器（IP 和主機名稱）的清單、建立憑證、設定從NetApp Automation Github 中提取 Ansible 程式碼的專案以及啟動自動化的作業範本。

1. 填寫特定於您的環境的變量，然後將其複製並貼上到作業範本中的 Extra Vars 欄位中。
2. 將額外的變數新增至您的作業範本後，您可以啟動自動化。
3. 作業範本透過為 ontap\_config、linux\_config 和 oracle\_config 指定標籤分三個階段運作。

## 透過 Ansible 控制主機的 CLI

1. 設定 Linux 主機，以便將其用作 Ansible 控制主機["按此處查看詳細說明"](#)
2. 設定 Ansible 控制主機後，您可以 git clone Ansible Automation 儲存庫。
3. 使用ONTAP叢集管理的 IP 和/或主機名稱以及 Oracle 伺服器的管理 IP 編輯 hosts 檔案。
4. 填寫特定於您的環境的變量，然後將其複製並貼上到 `vars.yml` 文件。
5. 每個 Oracle 主機都有一個由其主機名稱識別的變數文件，其中包含特定於主機的變數。
6. 所有變數檔案完成後，您可以透過指定標籤分三個階段運行劇本 ontap\_config， linux\_config，和 oracle\_config。

## 要求

| 環境         | 要求                                             |
|------------|------------------------------------------------|
| Ansible 環境 | AWX/Tower 或 Linux 主機作為 Ansible 控制主機            |
|            | Ansible v.2.10 及更高版本                           |
|            | Python 3                                       |
|            | Python 函式庫 - netapp-lib - xmltodict - jmespath |
| ONTAP      | ONTAP版本 9.3 - 9.7                              |
|            | 兩個數據聚合                                         |
|            | NFS vlan 和 ifgrp 創建                            |
| Oracle 伺服器 | RHEL 7/8                                       |
|            | Oracle Linux 7/8                               |
|            | NFS、公共和可選管理的網路接口                               |
|            | Oracle 伺服器上的 Oracle 安裝文件                       |

## 自動化詳細信息

此自動化部署採用由三個獨立角色組成的單一 Ansible 劇本設計。這些角色適用於ONTAP、Linux 和 Oracle 配置。下表描述了正在自動化的任務。

| 角色           | 任務                      |
|--------------|-------------------------|
| ontap_config | ONTAP環境的預檢查             |
|              | 為 Oracle 建立基於 NFS 的 SVM |
|              | 制定出口政策                  |
|              | 為 Oracle 建立卷            |
|              | 建立 NFS LIF              |
| linux_配置     | 建立掛載點並掛載 NFS 卷          |
|              | 驗證 NFS 掛載               |
|              | 作業系統特定配置                |
|              | 建立 Oracle 目錄            |
|              | 配置大頁面                   |
|              | 停用 SELinux 和防火牆守護進程     |
|              | 啟用並啟動 chronyd 服務        |
|              | 增加檔案描述符的硬限制             |
|              | 建立 pam.d 會話文件           |
| oracle_配置    | Oracle軟體安裝              |
|              | 建立 Oracle 監聽器           |
|              | 建立 Oracle 資料庫           |
|              | Oracle環境配置              |
|              | 保存 PDB 狀態               |
|              | 啟用實例存檔模式                |
|              | 啟用 DNFS 用戶端             |
|              | 在作業系統重新啟動之間啟用資料庫自動啟動和關閉 |

## 預設參數

為了簡化自動化，我們預設了許多必要的 Oracle 部署參數的預設值。對於大多數部署來說，通常不需要更改預設參數。更高級的用戶可以謹慎地更改預設參數。預設參數位於 defaults 目錄下的每個角色資料夾中。

## 部署說明

在開始之前，請下載以下 Oracle 安裝和修補程式文件，並將它們放在 `/tmp/archive` 每個要部署的資料庫伺服器上的所有使用者均具有讀取、寫入和執行存取權限的目錄。自動化任務在該特定目錄中尋找指定的安裝檔案以進行 Oracle 安裝和設定。

```
LINUX.X64_193000_db_home.zip -- 19.3 base installer
p31281355_190000_Linux-x86-64.zip -- 19.8 RU patch
p6880880_190000_Linux-x86-64.zip -- opatch version 12.2.0.1.23
```

## 執照

您應該閱讀 Github 儲存庫中所述的許可證資訊。透過存取、下載、安裝或使用此儲存庫中的內容，您同意所列出的授權條款["這裡"](#)。

請注意，製作和/或共享此儲存庫中的內容的任何衍生作品都受到某些限制。請務必閱讀["執照"](#)在使用內容之前。如果您不同意所有條款，請不要存取、下載或使用此儲存庫中的內容。

準備好後，點擊["點擊此處查看 AWX/Tower 部署的詳細步驟"](#)或者["此處用於 CLI 部署"](#)。

## 逐步部署流程

本頁介紹在NetApp ONTAP儲存上部署 Oracle19c 的自動化方法。

### AWX/Tower 部署 Oracle 19c 資料庫

#### 1. 為您的環境建立清單、群組、主機和憑證

本節介紹 AWX/Ansible Tower 中的庫存、群組、主機和存取憑證的設置，為使用NetApp自動化解決方案準備環境。

1. 配置庫存。
  - a. 導覽至資源 → 庫存 → 新增，然後按一下新增庫存。
  - b. 提供姓名和組織詳細信息，然後按一下「儲存」。
  - c. 在庫存頁面上，按一下建立的庫存。
  - d. 如果有任何庫存變量，請將其貼到變數欄位中。
  - e. 導航至“群組”子選單並點擊“新增”。
  - f. 提供ONTAP的群組名稱，貼上群組變數（如果有），然後按一下「儲存」。
  - g. 對 Oracle 的另一個群組重複此程序。
  - h. 選擇已建立的ONTAP群組，前往「主機」子選單並點選「新增主機」。
  - i. 提供ONTAP叢集管理 IP 的 IP 位址，貼上主機變數（如果有），然後按一下「儲存」。
  - j. 必須對 Oracle 群組和 Oracle 主機管理 IP/主機名稱重複此程序。
2. 建立憑證類型。對於涉及ONTAP的解決方案，您必須設定憑證類型以符合使用者名稱和密碼項目。
  - a. 導覽至管理 → 憑證類型，然後按一下新增。
  - b. 提供名稱和描述。
  - c. 在輸入配置中貼上以下內容：

```
fields:
  - id: username
    type: string
    label: Username
  - id: password
    type: string
    label: Password
    secret: true
  - id: vsadmin_password
    type: string
    label: vsadmin_password
    secret: true
```

1. 將以下內容貼到注入器配置中：

```
extra_vars:
  password: '{{ password }}'
  username: '{{ username }}'
  vsadmin_password: '{{ vsadmin_password }}'
```

1. 配置憑證。

- a. 導覽至資源 → 憑證，然後按一下新增。
- b. 輸入ONTAP的名稱和組織詳細資料。
- c. 選擇您為ONTAP建立的自訂憑證類型。
- d. 在類型詳細資料下，輸入使用者名稱、密碼和 vsadmin\_password。
- e. 按一下“返回憑證”，然後按一下“新增”。
- f. 輸入 Oracle 的名稱和組織詳細資訊。
- g. 選擇機器憑證類型。
- h. 在類型詳細資料下，輸入 Oracle 主機的使用者名稱和密碼。
- i. 選擇正確的權限提升方法，並輸入使用者名稱和密碼。

## 2. 創建專案

1. 前往資源 → 項目，然後按一下新增。

- a. 輸入姓名和組織詳細資料。
- b. 在原始碼控制憑證類型欄位中選擇 Git。
- c. 進入 `https://github.com/NetApp-Automation/na\_oracle19c\_deploy.git` 作為原始碼控制 URL。
- d. 按一下「Save（儲存）」。
- e. 當原始程式碼發生變化時，專案可能需要偶爾同步。

### 3.配置 Oracle host\_vars

本節定義的變數適用於每個單獨的 Oracle 伺服器 and 資料庫。

1. 在下列嵌入式 Oracle 主機變數或 host\_vars 表單中輸入特定環境的參數。



必須更改藍色項目以符合您的環境。

#### 主機 VARS 配置

```
#####
#####          Host Variables Configuration          #####
#####

# Add your Oracle Host
ansible_host: "10.61.180.15"

# Oracle db log archive mode: true - ARCHIVELOG or false - NOARCHIVELOG
log_archive_mode: "true"

# Number of pluggable databases per container instance identified by sid.
Pdb_name specifies the prefix for container database naming in this case
cdb2_pdb1, cdb2_pdb2, cdb2_pdb3
oracle_sid: "cdb2"
pdb_num: "3"
pdb_name: "{{ oracle_sid }}_pdb"

# CDB listener port, use different listener port for additional CDB on
same host
listener_port: "1523"

# CDB is created with SGA at 75% of memory_limit, MB. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB. The grand total SGA should not exceed 75% available RAM on node.
memory_limit: "5464"

# Set "em_configuration: DBEXPRESS" to install enterprise manager express
and choose a unique port from 5500 to 5599 for each sid on the host.
# Leave them blank if em express is not installed.
em_configuration: "DBEXPRESS"
em_express_port: "5501"

# {{groups.oracle[0]}} represents first Oracle DB server as defined in
Oracle hosts group [oracle]. For concurrent multiple Oracle DB servers
deployment, [0] will be incremented for each additional DB server. For
example, {{groups.oracle[1]}} represents DB server 2,
```

```
"{{groups.oracle[2]}}" represents DB server 3 ... As a good practice and the default, minimum three volumes is allocated to a DB server with corresponding /u01, /u02, /u03 mount points, which store oracle binary, oracle data, and oracle recovery files respectively. Additional volumes can be added by click on "More NFS volumes" but the number of volumes allocated to a DB server must match with what is defined in global vars file by volumes_nfs parameter, which dictates how many volumes are to be created for each DB server.
```

```
host_datastores_nfs:
  - {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}
```

1. 在藍色欄位中填寫所有變數。
2. 完成變數輸入後，按一下表單上的複製按鈕，複製所有要傳輸到 AWX 或 Tower 的變數。
3. 導覽回 AWX 或 Tower，前往資源 → 主機，然後選擇並開啟 Oracle 伺服器設定頁面。
4. 在“詳細資料”標籤下，按一下“編輯”，然後將步驟 1 中複製的變數貼上到 YAML 標籤下的“變數”欄位中。
5. 按一下「Save（儲存）」。
6. 對系統中的任何其他 Oracle 伺服器重複此程序。

#### 4.配置全域變數

本節定義的變數適用於所有 Oracle 主機、資料庫和ONTAP叢集。

1. 在以下嵌入式全域變數或 vars 形式中輸入特定於環境的參數。



必須更改藍色項目以符合您的環境。

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

#####
### Ontap env specific config variables ###
#####

#Inventory group name
#Default inventory group name - 'ontap'
#Change only if you are changing the group name either in inventory/hosts
file or in inventory groups in case of AWX/Tower
```

```

hosts_group: "ontap"

#CA_signed_certificates (ONLY CHANGE to 'true' IF YOU ARE USING CA SIGNED
CERTIFICATES)
ca_signed_certs: "false"

#Names of the Nodes in the ONTAP Cluster
nodes:
  - "AFF-01"
  - "AFF-02"

#Storage VLANs
#Add additional rows for vlans as necessary
storage_vlans:
  - {vlan_id: "203", name: "infra_NFS", protocol: "NFS"}
More Storage VLANsEnter Storage VLANs details

#Details of the Data Aggregates that need to be created
#If Aggregate creation takes longer, subsequent tasks of creating volumes
may fail.
#There should be enough disks already zeroed in the cluster, otherwise
aggregate create will zero the disks and will take long time
data_aggregates:
  - {aggr_name: "aggr01_node01"}
  - {aggr_name: "aggr01_node02"}

#SVM name
svm_name: "ora_svm"

# SVM Management LIF Details
svm_mgmt_details:
  - {address: "172.21.91.100", netmask: "255.255.255.0", home_port: "e0M"}

# NFS storage parameters when data_protocol set to NFS. Volume named after
Oracle hosts name identified by mount point as follow for oracle DB server
1. Each mount point dedicates to a particular Oracle files: u01 - Oracle
binary, u02 - Oracle data, u03 - Oracle redo. Add additional volumes by
click on "More NFS volumes" and also add the volumes list to corresponding
host_vars as host_datastores_nfs variable. For multiple DB server
deployment, additional volumes sets needs to be added for additional DB
server. Input variable "{{groups.oracle[1]}}_u01",
 "{{groups.oracle[1]}}_u02", and "{{groups.oracle[1]}}_u03" as vol_name for
second DB server. Place volumes for multiple DB servers alternately
between controllers for balanced IO performance, e.g. DB server 1 on
controller node1, DB server 2 on controller node2 etc. Make sure match lif
address with controller node.

```

```

volumes_nfs:
  - {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}

#NFS LIFs IP address and netmask

nfs_lifs_details:
  - address: "172.21.94.200" #for node-1
    netmask: "255.255.255.0"
  - address: "172.21.94.201" #for node-2
    netmask: "255.255.255.0"

#NFS client match

client_match: "172.21.94.0/24"

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

#NFS Mount points for Oracle DB volumes

mount_points:
  - "/u01"
  - "/u02"
  - "/u03"

# Up to 75% of node memory size divided by 2mb. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB.
# Leave it blank if hugepage is not configured on the host.

hugepages_nr: "1234"

# RedHat subscription username and password

redhat_sub_username: "xxx"
redhat_sub_password: "xxx"

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

```

```
db_domain: "your.domain.com"

# Set initial password for all required Oracle passwords. Change them
after installation.

initial_pwd_all: "netapp123"
```

1. 在藍色欄位中填寫所有變數。
2. 完成變數輸入後，點選表單上的複製按鈕，將所有要傳輸到 AWX 或 Tower 的變數複製到下列作業範本中。

## 5. 配置並啟動作業範本。

1. 建立作業模板。
  - a. 導覽至資源 → 範本 → 新增，然後按一下新增作業範本。
  - b. 輸入名稱和描述
  - c. 選擇作業類型；運行根據劇本配置系統，檢查執行劇本的試運行而不實際配置系統。
  - d. 為劇本選擇相應的庫存、項目、劇本和憑證。
  - e. 選擇 all\_playbook.yml 作為要執行的預設劇本。
  - f. 將從步驟 4 複製的全域變數貼上到 YAML 標籤下的範本變數欄位中。
  - g. 勾選「作業標籤」欄位中的「啟動時提示」複選框。
  - h. 按一下「Save（儲存）」。
2. 啟動作業模板。
  - a. 導航至資源 → 範本。
  - b. 按一下所需的模板，然後按一下啟動。
  - c. 啟動時提示輸入作業標籤時，輸入 requirements\_config。您可能需要按一下 requirements\_config 下方的「建立作業標籤」行來輸入作業標籤。



requirements\_config 確保您擁有正確的程式庫來執行其他角色。

1. 按一下“下一步”，然後按一下“啟動”以開始作業。
2. 按一下檢視 → 作業來監控作業輸出和進度。
3. 啟動時提示輸入作業標籤時，輸入 ontap\_config。您可能需要點擊 ontap\_config 正下方的建立「作業標籤」行來輸入作業標籤。
4. 按一下“下一步”，然後按一下“啟動”以開始作業。
5. 點選查看 → 作業來監控作業輸出和進度
6. ontap\_config 角色完成後，再次為 linux\_config 執行此程序。
7. 導航至資源 → 範本。
8. 選擇所需的模板，然後按一下啟動。
9. 當啟動時提示在 linux\_config 中輸入作業標籤類型時，您可能需要選擇 linux\_config 正下方的建立「作業標

籤」行來輸入作業標籤。

10. 按一下“下一步”，然後按一下“啟動”以開始作業。
11. 選擇檢視 → 作業來監控作業輸出和進度。
12. linux\_config 角色完成後，再次為 oracle\_config 執行此程序。
13. 前往資源 → 模板。
14. 選擇所需的模板，然後按一下啟動。
15. 啟動時提示輸入作業標籤時，請輸入 oracle\_config。您可能需要選擇 oracle\_config 正下方的建立「作業標籤」行來輸入作業標籤。
16. 按一下“下一步”，然後按一下“啟動”以開始作業。
17. 選擇檢視 → 作業來監控作業輸出和進度。

## 6.在同一 Oracle 主機上部署附加資料庫

劇本的 Oracle 部分每次執行都會在 Oracle 伺服器上建立一個 Oracle 容器資料庫。若要在同一伺服器上建立其他容器資料庫，請完成下列步驟。

1. 修改 host\_vars 變數。
  - a. 返回步驟 2 - 設定 Oracle host\_vars。
  - b. 將 Oracle SID 變更為不同的命名字串。
  - c. 將偵聽器連接埠變更為不同的號碼。
  - d. 如果您正在安裝 EM Express，請將 EM Express 連接埠變更為其他號碼。
  - e. 將修改後的主機變數複製並貼上到「主機配置詳細資料」標籤中的「Oracle 主機變數」欄位。
2. 僅使用 oracle\_config 標籤啟動部署作業範本。
3. 以oracle用戶登入Oracle伺服器，執行以下命令：

```
ps -ef | grep ora
```



如果安裝如預期完成且 Oracle DB 已啟動，這將列出 Oracle 進程

4. 登入資料庫，檢查資料庫配置設定和使用以下命令集建立的 PDB。

```

[oracle@localhost ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 12:52:51 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL>

SQL> select name, log_mode from v$database;
NAME          LOG_MODE
-----
CDB2          ARCHIVELOG

SQL> show pdbs

          CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
          2 PDB$SEED                                READ ONLY  NO
          3 CDB2_PDB1                                READ WRITE NO
          4 CDB2_PDB2                                READ WRITE NO
          5 CDB2_PDB3                                READ WRITE NO

col svrname form a30
col dirname form a30
select svrname, dirname, nfsversion from v$dnfs_servers;

SQL> col svrname form a30
SQL> col dirname form a30
SQL> select svrname, dirname, nfsversion from v$dnfs_servers;

SVRNAME                                DIRNAME                                NFSVERSION
-----
172.21.126.200                        /rhelora03_u02                        NFSv3.0
172.21.126.200                        /rhelora03_u03                        NFSv3.0
172.21.126.200                        /rhelora03_u01                        NFSv3.0

```

這證實 dNFS 運作正常。

5. 使用下列命令透過監聽器連接到資料庫來檢查 Oracle 監聽器配置。變更為適當的監聽連接埠和資料庫服務名稱。

```
[oracle@localhost ~]$ sqlplus
system@//localhost:1523/cdb2_pdb1.cie.netapp.com

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 13:19:57 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Enter password:
Last Successful login time: Wed May 05 2021 17:11:11 -04:00

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL> show user
USER is "SYSTEM"
SQL> show con_name
CON_NAME
CDB2_PDB1
```

這證實 Oracle 監聽器正常運作。

去哪裡尋求協助？

如果您需要工具包的協助，請加入["NetApp解決方案自動化社群支援 Slack 頻道"](#)並尋找解決方案自動化管道來發布您的問題或詢問。

## 逐步部署流程

本文檔詳細介紹了使用自動化命令列介面 (cli) 部署 Oracle 19c。

### CLI 部署 Oracle 19c 資料庫

本節介紹使用 CLI 準備和部署 Oracle19c 資料庫所需的步驟。確保您已查看["入門和要求部分"](#)並相應地準備您的環境。

#### 下載 Oracle19c 倉庫

1. 從您的 ansible 控制器執行以下命令：

```
git clone https://github.com/NetApp-Automation/na_oracle19c_deploy.git
```

2. 下載儲存庫後，將目錄變更為 na\_oracle19c\_deploy <cd na\_oracle19c\_deploy>。

## 編輯 **hosts** 文件

部署前請完成以下操作：

1. 編輯您的主機檔案 `na_oracle19c_deploy` 目錄。
2. 在 `[ontap]` 下，將 IP 位址變更為您的叢集管理 IP。
3. 在 `[oracle]` 群組下，新增 oracle 主機名稱。必須透過 DNS 或 `hosts` 檔案將主機名稱解析為其 IP 位址，或必須在主機中指定。
4. 完成這些步驟後，請儲存所有變更。

以下範例描述了一個主機檔案：

```
#ONTAP Host

[ontap]

"10.61.184.183"

#Oracle hosts

[oracle]

"rtpora01"

"rtpora02"
```

此範例執行劇本並在兩個 Oracle DB 伺服器上同時部署 Oracle 19c。您也可以只使用一個資料庫伺服器進行測試。在這種情況下，您只需要設定一個主機變數檔案。



無論您部署多少個 Oracle 主機和資料庫，劇本的執行方式都是相同的。

## 編輯 **host\_vars** 下的 **host\_name.yml** 文件

每個 Oracle 主機都有一個由其主機名稱識別的主機變數文件，其中包含特定於主機的變數。您可以為您的主機指定任何名稱。編輯並複製 `host_vars` 從主機 VARS 設定部分並將其貼上到您想要的 `host_name.yml` 文件。



必須更改藍色項目以符合您的環境。

## 主機 **VARS** 配置

```
#####
##### Host Variables Configuration #####
#####

# Add your Oracle Host
```

```

ansible_host: "10.61.180.15"

# Oracle db log archive mode: true - ARCHIVELOG or false - NOARCHIVELOG
log_archive_mode: "true"

# Number of pluggable databases per container instance identified by sid.
Pdb_name specifies the prefix for container database naming in this case
cdb2_pdb1, cdb2_pdb2, cdb2_pdb3
oracle_sid: "cdb2"
pdb_num: "3"
pdb_name: "{{ oracle_sid }}_pdb"

# CDB listener port, use different listener port for additional CDB on
same host
listener_port: "1523"

# CDB is created with SGA at 75% of memory_limit, MB. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB. The grand total SGA should not exceed 75% available RAM on node.
memory_limit: "5464"

# Set "em_configuration: DBEXPRESS" to install enterprise manager express
and choose a unique port from 5500 to 5599 for each sid on the host.
# Leave them blank if em express is not installed.
em_configuration: "DBEXPRESS"
em_express_port: "5501"

# {{groups.oracle[0]}} represents first Oracle DB server as defined in
Oracle hosts group [oracle]. For concurrent multiple Oracle DB servers
deployment, [0] will be incremented for each additional DB server. For
example, {{groups.oracle[1]}} represents DB server 2,
"{{groups.oracle[2]}}" represents DB server 3 ... As a good practice and
the default, minimum three volumes is allocated to a DB server with
corresponding /u01, /u02, /u03 mount points, which store oracle binary,
oracle data, and oracle recovery files respectively. Additional volumes
can be added by click on "More NFS volumes" but the number of volumes
allocated to a DB server must match with what is defined in global vars
file by volumes_nfs parameter, which dictates how many volumes are to be
created for each DB server.
host_datastores_nfs:
  - {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}

```

## 編輯 vars.yml 文件

這 `vars.yml` 檔案整合了 Oracle 部署的所有環境特定變數（ONTAP、Linux 或 Oracle）。

1. 編輯並複製 VARS 部分中的變量，並將這些變量貼到您的 `vars.yml` 文件。

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

#####
### Ontap env specific config variables ###
#####

#Inventory group name
#Default inventory group name - 'ontap'
#Change only if you are changing the group name either in inventory/hosts
file or in inventory groups in case of AWX/Tower
hosts_group: "ontap"

#CA_signed_certificates (ONLY CHANGE to 'true' IF YOU ARE USING CA SIGNED
CERTIFICATES)
ca_signed_certs: "false"

#Names of the Nodes in the ONTAP Cluster
nodes:
  - "AFF-01"
  - "AFF-02"

#Storage VLANs
#Add additional rows for vlans as necessary
storage_vlans:
  - {vlan_id: "203", name: "infra_NFS", protocol: "NFS"}
More Storage VLANsEnter Storage VLANs details

#Details of the Data Aggregates that need to be created
#If Aggregate creation takes longer, subsequent tasks of creating volumes
may fail.
#There should be enough disks already zeroed in the cluster, otherwise
aggregate create will zero the disks and will take long time
data_aggregates:
  - {aggr_name: "aggr01_node01"}
  - {aggr_name: "aggr01_node02"}

#SVM name
svm_name: "ora_svm"
```

```

# SVM Management LIF Details
svm_mgmt_details:
  - {address: "172.21.91.100", netmask: "255.255.255.0", home_port: "e0M"}

# NFS storage parameters when data_protocol set to NFS. Volume named after
Oracle hosts name identified by mount point as follow for oracle DB server
1. Each mount point dedicates to a particular Oracle files: u01 - Oracle
binary, u02 - Oracle data, u03 - Oracle redo. Add additional volumes by
click on "More NFS volumes" and also add the volumes list to corresponding
host_vars as host_datastores_nfs variable. For multiple DB server
deployment, additional volumes sets needs to be added for additional DB
server. Input variable "{{groups.oracle[1]}}_u01",
 "{{groups.oracle[1]}}_u02", and "{{groups.oracle[1]}}_u03" as vol_name for
second DB server. Place volumes for multiple DB servers alternately
between controllers for balanced IO performance, e.g. DB server 1 on
controller node1, DB server 2 on controller node2 etc. Make sure match lif
address with controller node.

volumes_nfs:
  - {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}

#NFS LIFs IP address and netmask

nfs_lifs_details:
  - address: "172.21.94.200" #for node-1
    netmask: "255.255.255.0"
  - address: "172.21.94.201" #for node-2
    netmask: "255.255.255.0"

#NFS client match

client_match: "172.21.94.0/24"

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

#NFS Mount points for Oracle DB volumes

mount_points:

```

```

- "/u01"
- "/u02"
- "/u03"

# Up to 75% of node memory size divided by 2mb. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB.
# Leave it blank if hugepage is not configured on the host.

hugepages_nr: "1234"

# RedHat subscription username and password

redhat_sub_username: "xxx"
redhat_sub_password: "xxx"

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

db_domain: "your.domain.com"

# Set initial password for all required Oracle passwords. Change them
after installation.

initial_pwd_all: "netapp123"

```

## 運行劇本

完成所需的環境前提條件並將變數複製到 `vars.yml` 和 `your_host.yml`，您現在就可以部署劇本了。



必須更改 `<username>` 以符合您的環境。

1. 透過傳遞正確的標籤和ONTAP叢集使用者名稱來運行ONTAP劇本。出現提示時，填寫ONTAP集群和vsadmin 的密碼。

```

ansible-playbook -i hosts all_playbook.yml -u username -k -K -t
ontap_config -e @vars/vars.yml

```

2. 運行 Linux 劇本來執行部署的 Linux 部分。輸入管理員 ssh 密碼以及 sudo 密碼。

```

ansible-playbook -i hosts all_playbook.yml -u username -k -K -t
linux_config -e @vars/vars.yml

```

3. 執行 Oracle 劇本來執行部署的 Oracle 部分。輸入管理員 ssh 密碼以及 sudo 密碼。

```
ansible-playbook -i hosts all_playbook.yml -u username -k -K -t  
oracle_config -e @vars/vars.yml
```

在同一 **Oracle** 主機上部署附加資料庫

劇本的 Oracle 部分每次執行都會在 Oracle 伺服器上建立一個 Oracle 容器資料庫。若要在同一台伺服器上建立額外的容器資料庫，請完成以下步驟：

1. 修改 host\_vars 變數。
  - a. 返回步驟 3 - 編輯 host\_name.yml 文件下 `host\_vars`。
  - b. 將 Oracle SID 變更為不同的命名字串。
  - c. 將偵聽器連接埠變更為不同的號碼。
  - d. 如果您已經安裝了 EM Express，請將 EM Express 連接埠變更為其他號碼。
  - e. 將修改後的主機變數複製並貼上到 Oracle 主機變數檔案下 host\_vars。
2. 使用以下方式執行劇本 `oracle\_config` 標籤如上所示[\[運行劇本\]](#)。

驗證 **Oracle** 安裝

1. 以 oracle 用戶登入 Oracle 伺服器，執行以下命令：

```
ps -ef | grep ora
```



如果安裝如預期完成且 Oracle DB 已啟動，這將列出 Oracle 進程

2. 登入資料庫，檢查資料庫配置設定和使用以下命令集建立的 PDB。

```
[oracle@localhost ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 12:52:51 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL>

SQL> select name, log_mode from v$database;
NAME          LOG_MODE
-----
CDB2          ARCHIVELOG

SQL> show pdbs

      CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
          2 PDB$SEED                                READ ONLY  NO
          3 CDB2_PDB1                                READ WRITE NO
          4 CDB2_PDB2                                READ WRITE NO
          5 CDB2_PDB3                                READ WRITE NO

col svrname form a30
col dirname form a30
select svrname, dirname, nfsversion from v$dnfs_servers;

SQL> col svrname form a30
SQL> col dirname form a30
SQL> select svrname, dirname, nfsversion from v$dnfs_servers;

SVRNAME                                DIRNAME                                NFSVERSION
-----
172.21.126.200                        /rhelora03_u02                        NFSv3.0
172.21.126.200                        /rhelora03_u03                        NFSv3.0
172.21.126.200                        /rhelora03_u01                        NFSv3.0
```

這證實 dNFS 運作正常。

3. 使用下列命令透過監聽器連接到資料庫來檢查 Oracle 監聽器配置。變更為適當的監聽連接埠和資料庫服務名稱。

```
[oracle@localhost ~]$ sqlplus
system@//localhost:1523/cdb2_pdb1.cie.netapp.com

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 13:19:57 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Enter password:
Last Successful login time: Wed May 05 2021 17:11:11 -04:00

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL> show user
USER is "SYSTEM"
SQL> show con_name
CON_NAME
CDB2_PDB1
```

這證實 Oracle 監聽器正常運作。

去哪裡尋求協助？

如果您需要工具包的協助，請加入["NetApp解決方案自動化社群支援 Slack 頻道"](#)並尋找解決方案自動化管道來發布您的問題或詢問。

## 版權資訊

Copyright © 2025 NetApp, Inc. 版權所有。台灣印製。非經版權所有人事先書面同意，不得將本受版權保護文件的任何部分以任何形式或任何方法（圖形、電子或機械）重製，包括影印、錄影、錄音或儲存至電子檢索系統中。

由 NetApp 版權資料衍伸之軟體必須遵守下列授權和免責聲明：

此軟體以 NETAPP「原樣」提供，不含任何明示或暗示的擔保，包括但不限於有關適售性或特定目的適用性之擔保，特此聲明。於任何情況下，就任何已造成或基於任何理論上責任之直接性、間接性、附隨性、特殊性、懲罰性或衍生性損害（包括但不限於替代商品或服務之採購；使用、資料或利潤上的損失；或企業營運中斷），無論是在使用此軟體時以任何方式所產生的契約、嚴格責任或侵權行為（包括疏忽或其他）等方面，NetApp 概不負責，即使已被告知有前述損害存在之可能性亦然。

NetApp 保留隨時變本文所述之任何產品的權利，恕不另行通知。NetApp 不承擔因使用本文所述之產品而產生的責任或義務，除非明確經過 NetApp 書面同意。使用或購買此產品並不會在依據任何專利權、商標權或任何其他 NetApp 智慧財產權的情況下轉讓授權。

本手冊所述之產品受到一項（含）以上的美國專利、國外專利或申請中專利所保障。

有限權利說明：政府機關的使用、複製或公開揭露須受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中的「技術資料權利 - 非商業項目」條款 (b)(3) 小段所述之限制。

此處所含屬於商業產品和 / 或商業服務（如 FAR 2.101 所定義）的資料均為 NetApp, Inc. 所有。根據本協議提供的所有 NetApp 技術資料和電腦軟體皆屬於商業性質，並且完全由私人出資開發。美國政府對於該資料具有非專屬、非轉讓、非轉授權、全球性、有限且不可撤銷的使用權限，僅限於美國政府為傳輸此資料所訂合約所允許之範圍，並基於履行該合約之目的方可使用。除非本文另有規定，否則未經 NetApp Inc. 事前書面許可，不得逕行使用、揭露、重製、修改、履行或展示該資料。美國政府授予國防部之許可權利，僅適用於 DFARS 條款 252.227-7015(b)（2014 年 2 月）所述權利。

## 商標資訊

NETAPP、NETAPP 標誌及 <http://www.netapp.com/TM> 所列之標章均為 NetApp, Inc. 的商標。文中所涉及的所有其他公司或產品名稱，均為其各自所有者的商標，不得侵犯。