



Oracle データベースを導入しています NetApp Solutions

NetApp
April 10, 2024

This PDF was generated from <https://docs.netapp.com/ja-jp/netapp-solutionshttps://docs.netapp.com/us-en/ontap-apps-dbs/oracle/oracle-overview.html> on April 10, 2024. Always check docs.netapp.com for the latest.

目次

Oracle データベースを導入しています	1
解決策の概要	1

Oracle データベースを導入しています

解決策の概要

Oracle19c for ONTAP の NFS への自動導入

組織は環境を自動化して、効率を高め、導入を高速化し、手動作業を削減しています。Ansible などの構成管理ツールを使用して、エンタープライズデータベースの運用を合理化しています。この解決策では、Ansible を使用して、Oracle 19C のプロビジョニングと設定を NetApp ONTAP で自動化する方法を紹介します。ストレージ管理者、システム管理者、DBA は、新しいストレージの一貫した迅速な導入、データベースサーバの構成、Oracle 19C ソフトウェアのインストールを可能にすることで、次のようなメリットを得ることができます。

- 設計の複雑さと人為的ミスを排除し、繰り返し実行可能な一貫した導入とベストプラクティスを実装する
- ストレージのプロビジョニング、DB ホストの構成、Oracle のインストールにかかる時間を短縮
- データベース管理者、システム管理者、ストレージ管理者の生産性を向上
- ストレージとデータベースを簡単に拡張できます

ネットアップは、検証済みの Ansible モジュールとロールをお客様に提供し、Oracle データベース環境の導入、構成、ライフサイクル管理を迅速化します。この解決策では、以下の作業に役立つ Ansible の Playbook コードを提供しています。

- Oracle データベース用の ONTAP NFS ストレージを作成して設定します
- Oracle 19C を Red Hat Enterprise Linux 7/8 または Oracle にインストールします Linux 7/8.
- ONTAP NFS ストレージ上に Oracle 19C を設定します

詳細または概要については、以下の概要ビデオをご覧ください。

AWX / タワー型の導入

パート 1 : はじめに、要件、自動化の詳細、AWX/Tower の初期構成

[AWXの導入](#)

パート 2 : 変数とプレイブックの実行

[AWX Playbookの実行](#)

CLI の導入

パート 1 : はじめに、要件、自動化の詳細、Ansible Control Host Setup を確認する

[CLI の導入](#)

パート 2 : 変数とプレイブックの実行

[CLI Playbookの実行](#)

はじめに

この解決策は、AWX/Tower 環境または Ansible コントロールホストの CLI で実行されるように設計されています。

AWX / タワー

AWX / タワー環境の場合は、ONTAP クラスタ管理と Oracle サーバ（IP およびホスト名）のインベントリの作成、クレデンシャルの作成、NetApp Automation Github から Ansible コードを取得するプロジェクトの設定、および自動化を開始するジョブテンプレートの設定を案内されます。

1. 環境に固有の変数を入力し、ジョブテンプレートのその他の VAR フィールドにコピーして貼り付けます。
2. ジョブテンプレートに変数を追加したら、自動化を起動できます。
3. ジョブテンプレートは、ontap/config、linux_config、および ORACLE_config のタグを指定することで、3つのフェーズで実行されます。

Ansible コントロールホストを介して CLI に接続します

1. Ansible 制御ホストとして使用できるように Linux ホストを設定するには、["詳細な手順については、ここをクリックしてください"](#)
2. Ansible 制御ホストが設定されたら、Ansible Automation リポジトリのクローンを Git で作成できます。
3. ONTAP クラスタ管理 IP および Oracle サーバの管理 IP の IP またはホスト名を使用して hosts ファイルを編集してください。
4. 環境に固有の変数を入力し '変数 .yml ファイルにコピーして貼り付けます
5. 各 Oracle ホストには、ホスト固有の変数を含むホスト名で識別される変数ファイルがあります。
6. すべての変数ファイルが完了したら 'ONTAP_config' 'linux_config' および 'ORACLE_config' のタグを指定することで '3つのフェーズでプレイブックを実行できます

要件

環境	要件
* Ansible 環境 *	AWX/Tower または Linux ホストを Ansible コントロールホストにします
	Ansible v.2.10 以上
	Python 3.
	Python ライブラリ - NetApp-lib-xmltodict-jmespath
* ONTAP *	ONTAP バージョン 9.3-9.7
	データアグリゲート × 2
	NFS VLAN および ifgrp が作成されました

環境	要件
* Oracle サーバ *	RHEL 7/8
	Oracle Linux 7/8
	NFS、パブリック、オプションの管理用のネットワークインターフェイス
	Oracle サーバ上の Oracle インストールファイル

自動化の詳細

この自動導入は、3つのロールで構成される Ansible プレイブックを使用して設計されています。ロールは ONTAP、Linux、Oracle の各構成に対応しています。次の表に、自動化されるタスクを示します。

ロール	タスク
* ONTAP_CONFIG *	ONTAP 環境の事前チェック
	Oracle 用の NFS ベースの SVM の作成
	エクスポートポリシーが作成されました
	Oracle 用のボリュームの作成
	NFS LIF の作成
linux_config	マウントポイントを作成し、NFS ボリュームをマウント
	NFS マウントを確認
	OS 固有の設定
	Oracle ディレクトリを作成します
	hugepages を設定します
	SELinux とファイアウォールデーモンを無効にする
	サービスを有効にして開始します
	ファイル記述子のハードリミットを増やします
	pam.d セッションファイルを作成します
* ORACLE_CONFIG *	Oracle ソフトウェアのインストール
	Oracle リスナーを作成します
	Oracle データベースを作成します
	Oracle 環境構成
	PDB 状態を保存します
	インスタンスアーカイブモードを有効にします
	DNFS クライアントを有効にしてください
	OS のリブート間のデータベースの自動起動とシャットダウンを有効にします

デフォルトパラメータ

自動化を簡易化するために、必要な Oracle 導入パラメータが多数デフォルト値であらかじめ設定されています。通常、ほとんどの環境でデフォルトパラメータを変更する必要はありません。上級ユーザーは 'デフォルト・パラメータを変更する際に注意してくださいデフォルトのパラメータは、各ロールフォルダの defaults ディレクトリにあります。

導入手順

開始する前に '次の Oracle インストール・ファイルとパッチ・ファイルをダウンロードし 'tmp/archive' ディレクトリに配置しますこのディレクトリには '展開する各 DB サーバ上のすべてのユーザに対する読み取り '書き込み 'および実行のアクセス権が含まれます自動化タスクは、その特定のディレクトリにある指定されたインストールファイルを検索して、Oracle のインストールと構成を行います。

```
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
```

使用許諾

Github リポジトリに記載されているライセンス情報をお読みください。このリポジトリ内のコンテンツにアクセス、ダウンロード、インストール、または使用することにより、ライセンスの条項に同意したものとみなされます "[こちらをご覧ください](#)"。

このリポジトリ内のコンテンツの作成および / または派生著作物の共有に関しては、一定の制限事項があります。の条件を必ずお読みください "[使用許諾](#)" コンテンツを使用する前に。すべての条件に同意しない場合は、このリポジトリのコンテンツにアクセスしたり、コンテンツをダウンロードしたり、使用したりしないでください。

準備ができたなら、をクリックします "[AWX/Tower の導入手順の詳細については、こちらを参照してください](#)" または "[CLI の導入については、こちらをご覧ください](#)"。

ステップバイステップの導入手順

AWX/Tower の導入 Oracle 19C データベース

1. 環境のインベントリ、グループ、ホスト、およびクレデンシャルを作成します

このセクションでは、ネットアップの自動化ソリューションを使用する環境を準備するための AWX/Ansible タワーでのインベントリ、グループ、ホスト、アクセスクレデンシャルのセットアップについて説明します。

1. インベントリを設定します。
 - a. リソース→インベントリ→追加と進み、インベントリの追加をクリックします。
 - b. 名前と組織の詳細を入力し、[保存] をクリックします。
 - c. インベントリページで、作成されたインベントリをクリックします。
 - d. インベントリ変数がある場合は、その変数を変数フィールドに貼り付けます。
 - e. [グループ] サブメニューに移動し、[追加] をクリックします。

- f. ONTAP のグループの名前を入力し、グループ変数（ある場合）を貼り付けて、[保存] をクリックします。
 - g. Oracle の別のグループに対してこの手順を繰り返します。
 - h. 作成した ONTAP グループを選択し、Hosts サブメニューに移動して、Add New Host をクリックします。
 - i. ONTAP クラスタ管理 IP の IP アドレスを入力し、ホスト変数（存在する場合）を貼り付けて、[保存] をクリックします。
 - j. このプロセスは、Oracle グループおよび Oracle ホストの管理 IP / ホスト名に対して繰り返す必要があります。
2. クレデンシャルタイプを作成する。ONTAP を使用するソリューションでは、ユーザ名とパスワードのエントリを照合するようにクレデンシャルタイプを設定する必要があります。
 - a. [管理] → [資格情報の種類] に移動し、[追加] をクリックします。
 - b. 名前と概要を指定します。
 - c. 入力構成に次の内容を貼り付けます。

```
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. 次の内容をインジェクター設定に貼り付けます。

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

1. クレデンシャルを設定します。
 - a. [リソース] → [資格情報] に移動し、[追加] をクリックします。
 - b. ONTAP の名前と組織の詳細を入力します。
 - c. ONTAP 用に作成したカスタム資格情報タイプを選択します。
 - d. [タイプの詳細] で、ユーザー名、パスワード、および vsadmin-readonly を入力します。

- e. [資格情報に戻る] をクリックし、[追加] をクリックします
- f. Oracle の名前と組織の詳細を入力します。
- g. マシクレデンシャルタイプを選択します。
- h. Type Details （タイプの詳細）に、Oracle ホストのユーザー名とパスワードを入力します。
- i. 適切な特権昇格方式を選択し、ユーザ名とパスワードを入力します。

2. プロジェクトを作成します

1. [リソース] → [プロジェクト] に移動し、[追加] をクリックします。
 - a. 名前と組織の詳細を入力します
 - b. Source Control Credential Type フィールドで Git を選択します。
 - c. 入力するコマンド <https://github.com/NetApp-Automation/na_oracle19c_deploy.git> をソース管理 URL として指定します。
 - d. [保存] をクリックします。
 - e. ソースコードが変更されたときに、プロジェクトの同期が必要になることがあります。

3. Oracle host_vars を設定します

このセクションで定義した変数は、個々の Oracle サーバとデータベースに適用されます。

1. 次の組み込み Oracle ホスト変数または host_vars フォームに、環境固有のパラメータを入力します。



青の項目は、環境に合わせて変更する必要があります。

ホスト VAR 構成

```
#####
#####          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. 青のフィールドにすべての変数を入力します。
2. 変数の入力完了したら、フォームの [コピー] ボタンをクリックして、AWX またはタワーに転送されるすべての変数をコピーします。
3. AWX またはタワーに戻って、Resources (リソース) → Hosts (ホスト) に移動し、Oracle サーバ設定ページを選択して開きます。
4. [詳細] タブで、編集をクリックし、コピーした変数を手順 1 から YAML タブの [変数] フィールドに貼り付けます。
5. [保存] をクリックします。
6. システム内の他の Oracle サーバについても、この手順を繰り返します。

4. グローバル変数を設定します

このセクションで定義する変数は、すべての Oracle ホスト、データベース、および ONTAP クラスタに適用されます。

1. 次の組み込みグローバル変数または変数フォームに環境固有のパラメータを入力します。



青の項目は、環境に合わせて変更する必要があります。

```
#####
##### 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. すべての変数を青のフィールドに入力します。
 2. 変数の入力完了したら、フォームの [コピー] ボタンをクリックして、AWX またはタワーに転送されるすべての変数を次のジョブテンプレートにコピーします。
5. ジョブテンプレートを設定して起動します。
1. ジョブテンプレートを作成します。
 - a. [リソース] → [テンプレート] → [追加] に移動し、[ジョブテンプレートの追加] をクリックします。
 - b. 名前と概要を入力します
 - c. ジョブタイプを選択します。Run は、プレイブックに基づいてシステムを設定します。Check は、実際にシステムを設定することなく、プレイブックの事前チェックを実行します。
 - d. 対応するインベントリ、プロジェクト、プレイブック、およびクレデンシャルを選択します。
 - e. 実行するデフォルトのプレイブックとして、all_cplaybook.yml を選択します。

- f. 手順 4 からコピーしたグローバル変数を YAML タブの Template Variables フィールドに貼り付けます。
 - g. [ジョブタグ] フィールドの [起動時にプロンプトを表示する] チェックボックスをオンにします。
 - h. [保存] をクリックします。
2. ジョブテンプレートを起動します。
 - a. [リソース] → [テンプレート] に移動します。
 - b. 目的のテンプレートをクリックし、[起動] をクリックします。
 - c. ジョブタグの起動時にプロンプトが表示されたら、requires_config と入力します。requires_config の下にある Create Job Tag 行をクリックして、ジョブタグを入力する必要がある場合があります。



requirement_config により、他のロールを実行するための正しいライブラリが確保されます。

1. [次へ] をクリックし、[起動] をクリックしてジョブを開始します。
2. ジョブの出力と進行状況を監視するには、表示→ジョブをクリックします。
3. ジョブタグの起動を求めるプロンプトが表示されたら、「ONTAP_config」と入力します。ジョブタグを入力するには、ONTAP_config の下にある「ジョブタグの作成」行をクリックする必要があります。
4. [次へ] をクリックし、[起動] をクリックしてジョブを開始します。
5. ジョブ出力およびを監視するには、表示→ジョブをクリックします 進捗状況
6. ONTAP_CONFIG ロールの完了後、linux_config のプロセスを再度実行します。
7. [リソース] → [テンプレート] に移動します。
8. 目的のテンプレートを選択し、[起動] をクリックします。
9. linux_config でジョブタグタイプの起動時にプロンプトが表示されたら、linux_config のすぐ下にある「ジョブタグの作成」行を選択して、ジョブタグを入力する必要があります。
10. [次へ] をクリックし、[起動] をクリックしてジョブを開始します。
11. ジョブの出力と進行状況を監視するには、表示→ジョブを選択します。
12. linux_config ロールが完了したら、ORACLE_config のプロセスを再度実行します。
13. [リソース] → [テンプレート] に移動します。
14. 目的のテンプレートを選択し、[起動] をクリックします。
15. ジョブタグの起動時にプロンプトが表示されたら、ORACLE_config と入力します。ORACLE_config の直下にある「ジョブタグの作成」行を選択して、ジョブタグを入力する必要がある場合があります。
16. [次へ] をクリックし、[起動] をクリックしてジョブを開始します。
17. ジョブの出力と進行状況を監視するには、表示→ジョブを選択します。

6. 同じ Oracle ホストに追加のデータベースを配置します

このプレイブックの Oracle 部分では、1 回の実行につき Oracle サーバ上に Oracle コンテナデータベースが 1 つ作成されます。同じサーバ上に追加のコンテナデータベースを作成するには、次の手順を実行します。

1. host_vars 変数を改訂。
 - a. 手順 2 - Oracle host_vars の設定に戻ります。

- b. Oracle SID を別の名前文字列に変更します。
 - c. リスナーポートを別の番号に変更します。
 - d. EM Express をインストールする場合は、EM Express ポートを別の番号に変更します。
 - e. 改訂されたホスト変数を Host Configuration Detail タブの Oracle Host Variables フィールドにコピーして貼り付けます。
2. ORACLE_config タグのみを使用して、導入ジョブテンプレートを起動します。
 3. OracleユーザとしてOracleサーバにログインし、次のコマンドを実行します。

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



インストールが正常に完了した場合は、Oracle プロセスが一覧表示されます Oracle DB のサポートを開始しました

4. データベースにログインして、次のコマンドセットを使用して作成されたDB設定およびPDBを確認します。

```

[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

```

これにより、dNFSが正常に動作していることが確認されます。

5. 次のコマンドを使用して'リスナー経由でデータベースに接続し'Oracleリスナーの構成を確認します適切なリスナーポートとデータベースサービス名に変更します。

```
[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
```

これにより、Oracleリスナーが正常に動作していることが確認されます。

サポートが必要な場所

ツールキットに関するサポートが必要な場合は、にご参加ください["ネットアップの解決策自動化コミュニティでは、余裕期間のチャンネルがサポートさ"](#) また、ソリューション自動化チャンネルを検索して、質問や問い合わせを投稿しましょう。

ステップバイステップの導入手順

このドキュメントでは、自動コマンドラインインターフェイス（CLI）を使用したOracle 19Cの導入について詳しく説明します。

CLI による Oracle 19C データベースの導入

このセクションでは、CLI を使用して Oracle19c データベースを準備および導入するために必要な手順について説明します。を確認しておきます "[「はじめに」](#) および [「要件」セクション](#)" それに応じて環境の準備を整えます。

Oracle19c repo をダウンロードします

1. Ansibleコントローラで、次のコマンドを実行します。

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


2. リポジトリをダウンロードしたら、ディレクトリをna_oracle19c_deploy <cd na_oracle19c_deploy>に変更します。

hosts ファイルを編集します

導入前に、次の手順を実行します。

1. hosts ファイル na_oracle19c_deploy ディレクトリを編集します。
2. ONTAP で、IP アドレスをクラスタ管理 IP に変更します。
3. [Oracle] グループの下に、Oracle ホスト名を追加します。DNS または hosts ファイルを使用してホスト名を IP アドレスに解決しておくか、ホストで指定する必要があります。
4. これらの手順を完了したら、変更を保存します。

次の例は、ホストファイルを示しています。

```
#ONTAP Host

[ontap]

"10.61.184.183"

#Oracle hosts

[oracle]

"rtpora01"

"rtpora02"
```

この例では、Playbook を実行し、Oracle 19C を 2 台の Oracle DB サーバに同時に導入しています。1 つの DB サーバでテストすることもできます。この場合、設定が必要なホスト変数ファイルは 1 つだけです。



このプレイブックの内容は、導入する Oracle ホストとデータベースの数に関係なく同じです。

host_vars で **host_name.yml** ファイルを編集します

各 Oracle ホストには、ホスト固有の変数を含むホスト名で識別されるホスト変数ファイルがあります。ホストには任意の名前を指定できます。Host VAR Config セクションから「host_vars」を編集してコピーし、目的の「host_name.yml」ファイルに貼り付けます。



青の項目は、環境に合わせて変更する必要があります。

ホスト **VAR** 構成

```
#####
##### 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 black 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"}
```

vars.yml ファイルを編集します

変数 `.yml` ファイルは 'Oracle の導入に向けて' 環境固有のすべての変数（ONTAP 'Linux' Oracle）を統合します

1. 変数を VAR セクションから編集してコピーし、変数を自分の「vars.yml」ファイルに貼り付けます。

```
#####
##### 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"

```

プレイブックを実行します

必要な環境の前提条件を完了し '変数を vars.yml' および 'Your_host.yml' にコピーした後 'プレイブックを導入する準備が整いました



<username> は、環境に合わせて変更する必要があります。

1. 正しいタグとONTAPクラスタユーザ名を渡してONTAP Playbookを実行します。プロンプトが表示されたら、ONTAPクラスタのパスワードとvsadminを入力します。

```

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

```

2. Linux Playbookを実行して、導入のLinux部分を実行します。admin sshパスワードとsudoパスワードを入力します。

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

3. Oracle Playbookを実行して、導入のOracle部分を実行します。admin sshパスワードとsudoパスワードを入力します。

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

同じ **Oracle** ホストに追加のデータベースを導入します

このプレイブックの Oracle 部分では、1 回の実行につき Oracle サーバ上に Oracle コンテナデータベースが 1 つ作成されます。同じサーバ上に追加のコンテナデータベースを作成するには、次の手順を実行します。

1. host_vars 変数を改訂します。
 - a. ステップ 3 に戻ります - 'host_vars' の下の 'host_name.yml' ファイルを編集します
 - b. Oracle SID を別の名前文字列に変更します。
 - c. リスナーポートを別の番号に変更します。
 - d. EM Express をインストールしている場合は、EM Express ポートを別の番号に変更します。
 - e. 変更したホスト変数を 'host_vars' の下の Oracle ホスト変数ファイルにコピーして貼り付けます
2. 上記のように 'ORACLE_CONFIG' タグを使用してプレイブックを実行します インチ [\[プレイブックを実行します\]](#)。

Oracle のインストールを検証します

1. OracleユーザとしてOracleサーバにログインし、次のコマンドを実行します。

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



インストールが正常に完了した場合は、Oracle プロセスが一覧表示されます Oracle DB のサポートを開始しました

2. データベースにログインして、次のコマンドセットを使用して作成されたDB設定およびPDBを確認します。

```
[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
```

これにより、dNFSが正常に動作していることが確認されます。

3. 次のコマンドを使用して'リスナー経由でデータベースに接続し'Oracleリスナーの構成を確認します適切なリスナーポートとデータベースサービス名に変更します。

```
[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
```

これにより、Oracleリスナーが正常に動作していることが確認されます。

サポートが必要な場所

ツールキットに関するサポートが必要な場合は、にご参加ください["ネットアップの解決策自動化コミュニティでは、余裕期間のチャンネルがサポートさ"](#) また、ソリューション自動化チャンネルを検索して、質問や問い合わせを投稿しましょう。

著作権に関する情報

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および / または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータ ソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用权を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用权については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。