



# **Step-by-step deployment procedure**

## **NetApp Solutions**

Brandon, sufian ahmad, Allen Cao, Dorian Henderson  
May 19, 2021

# Table of Contents

Step-by-step deployment procedure .....	1
AWX/Tower deployment Oracle 19c Database .....	1

# Step-by-step deployment procedure

## AWX/Tower deployment Oracle 19c Database

### 1. Create the inventory, group, hosts, and credentials for your environment

This section describes the setup of inventory, groups, hosts, and access credentials in AWX/Ansible Tower that prepare the environment for consuming NetApp automated solutions.

1. Configure the inventory.
  - a. Navigate to Resources → Inventories → Add, and click Add Inventory.
  - b. Provide the name and organization details, and click Save.
  - c. On the Inventories page, click the inventory created.
  - d. If there are any inventory variables, paste them in the variables field.
  - e. Navigate to the Groups sub-menu and click Add.
  - f. Provide the name of the group for ONTAP, paste the group variables (if any) and click Save.
  - g. Repeat the process for another group for Oracle.
  - h. Select the ONTAP group created, go to the Hosts sub-menu and click Add New Host.
  - i. Provide the IP address of the ONTAP cluster management IP, paste the host variables (if any), and click Save.
  - j. This process must be repeated for the Oracle group and Oracle host(s) management IP/hostname.
2. Create credential types. For solutions involving ONTAP, you must configure the credential type to match username and password entries.
  - a. Navigate to Administration → Credential Types, and click Add.
  - b. Provide the name and description.
  - c. Paste the following content in Input Configuration:

```
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
```

- d. Paste the following content into Injector Configuration:

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

### 3. Configure the credentials.

- a. Navigate to Resources → Credentials, and click Add.
- b. Enter the name and organization details for ONTAP.
- c. Select the custom Credential Type you created for ONTAP.
- d. Under Type Details, enter the username, password, and vsadmin\_password.
- e. Click Back to Credential and click Add.
- f. Enter the name and organization details for Oracle.
- g. Select the Machine credential type.
- h. Under Type Details, enter the Username and Password for the Oracle hosts.
- i. Select the correct Privilege Escalation Method, and enter the username and password.

## 2. Create a project

1. Go to Resources → Projects, and click Add.
  - a. Enter the name and organization details.
  - b. Select Git in the Source Control Credential Type field.
  - c. enter [https://github.com/NetApp-Automation/na\\_oracle19c\\_deploy.git](https://github.com/NetApp-Automation/na_oracle19c_deploy.git) as the source control URL.
  - d. Click Save.
  - e. The project might need to sync occasionally when the source code changes.

## 3. Configure Oracle host\_vars

The variables defined in this section are applied to each individual Oracle server and database.

1. Input your environment-specific parameters in the following embedded Oracle hosts variables or host\_vars form.



The items in blue must be changed to match your environment.

Unresolved directive in <stdin> - include::ent-db/host\_vars.adoc[]

- a. Fill in all variables in the blue fields.
- b. After completing variables input, click the Copy button on the form to copy all variables to be transferred to AWX or Tower.
- c. Navigate back to AWX or Tower and go to Resources → Hosts, and select and open the Oracle server configuration page.
- d. Under the Details tab, click edit and paste the copied variables from step 1 to the Variables field under the YAML tab.

- e. Click Save.
- f. Repeat this process for any additional Oracle servers in the system.

## 4. Configure global variables

Variables defined in this section apply to all Oracle hosts, databases, and the ONTAP cluster.

1. Input your environment-specific parameters in following embedded global variables or vars form.



The items in blue must be changed to match your environment.

Unresolved directive in <stdin> - include::ent-db/vars.adoc[]

2. Fill in all variables in blue fields.
3. After completing variables input, click the Copy button on the form to copy all variables to be transferred to AWX or Tower into the following job template.

## 5. Configure and launch the job template.

1. Create the job template.
  - a. Navigate to Resources → Templates → Add and click Add Job Template.
  - b. Enter the name and description
  - c. Select the Job type; Run configures the system based on a playbook, and Check performs a dry run of a playbook without actually configuring the system.
  - d. Select the corresponding inventory, project, playbook, and credentials for the playbook.
  - e. Select the all\_playbook.yml as the default playbook to be executed.
  - f. Paste global variables copied from step 4 into the Template Variables field under the YAML tab.
  - g. Check the box Prompt on Launch in the Job Tags field.
  - h. Click Save.
2. Launch the job template.
  - a. Navigate to Resources → Templates.
  - b. Click the desired template and then click Launch.
  - c. When prompted on launch for Job Tags, type in requirements\_config. You might need to click the Create Job Tag line below requirements\_config to enter the job tag.



requirements\_config ensures that you have the correct libraries to run the other roles.

- d. Click Next and then Launch to start the job.
- e. Click View → Jobs to monitor the job output and progress.
- f. When prompted on launch for Job Tags, type in ontap\_config. You might need to click the Create "Job Tag" line right below ontap\_config to enter the job tag.
- g. Click Next and then Launch to start the job.
- h. Click View → Jobs to monitor the job output and progress

- i. After the `ontap_config` role has completed, run the process again for `linux_config`.
- j. Navigate to Resources → Templates.
- k. Select the desired template and then click Launch.
- l. When prompted on launch for the Job Tags type in `linux_config`, you might need to select the Create "job tag" line right below `linux_config` to enter the job tag.
- m. Click Next and then Launch to start the job.
- n. Select View → Jobs to monitor the job output and progress.
- o. After the `linux_config` role has completed, run the process again for `oracle_config`.
- p. Go to Resources → Templates.
- q. Select the desired template and then click Launch.
- r. When prompted on launch for Job Tags, type `oracle_config`. You might need to select the Create "Job Tag" line right below `oracle_config` to enter the job tag.
- s. Click Next and then Launch to start the job.
- t. Select View → Jobs to monitor the job output and progress.

## 6. Deploy additional database on same Oracle host

The Oracle portion of the playbook creates a single Oracle container database on an Oracle server per execution. To create additional container databases on the same server, complete the following steps.

1. Revise `host_vars` variables.
  - a. Go back to step 2 - Configure Oracle `host_vars`.
  - b. Change the Oracle SID to a different naming string.
  - c. Change the listener port to different number.
  - d. Change the EM Express port to a different number if you are installing EM Express.
  - e. Copy and paste the revised host variables to the Oracle Host Variables field in the Host Configuration Detail tab.
2. Launch the deployment job template with only the `oracle_config` tag.

Unresolved directive in <stdin> - include::ent-db/validation.adoc[]

## Copyright Information

Copyright © 2021 NetApp, Inc. All rights reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means-graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system-without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

## Trademark Information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.