# Execute PODBA using Ansible Automation Platform 2 (AAP2)

**Overview:** Ansible Automation Platform Version 2 offers new features and components. This readme outlines the process of executing the PODBA playbooks using AAP2 GUI.

Reference to AAP2: https://www.ansible.com/blog/introducing-ansible-automation-platform-2

AAP2 can be installed by referring this link:

Chapter 3. Installing Red Hat Ansible Automation Platform Red Hat Ansible Automation Platform 2.3 | Red Hat Customer Portal

## Prepare the Execution Environment:

1. Requires python version 3.8 or later.
2. Install podman using dnf [# dnf install podman]
3. Install ansible-builder [$ pip install ansible-builder]
4. Install ansible-navigator [$ pip install ansible-navigator]
5. In any present working directory, create a directory named "context".
6. Inside the "context" directory place the extracted oracle client software directory with the name "oracle_client".
7. Create a file (example: create_podba.yml) with the following content.

```
# Execution Environment
---
version: 3

images:
  base_image:
    name: registry.redhat.io/ansible-automation-platform-24/ee-minimal-rhel8:latest
options:
  package_manager_path: /usr/bin/microdnf
additional_build_steps:
  append_base:
   - RUN microdnf install gcc python39-devel libnsl* libaio* find* which* sudo dnf
   - RUN pip3 install wheel
   - RUN python3.9 -m pip install cx_Oracle --upgrade
   - RUN ln -s /usr/lib64/libnsl.so.2 /usr/lib64/libnsl.so.1
   - COPY oracle_client /oracle_client_sw   #Oracle Instant Client Path in the EE.
   - COPY ansible-automation-platform-managed-ca-cert.crt /etc/pki/ca-trust/source/anchors
```

8. Run the following command to build the execution environment image.

```
$ ansible-builder build -t powerodba -f create_aap2_ee.yml
Running command:
  podman build -f context/Containerfile -t powerodba context
Complete! The build context can be found at: /var/lib/awx/aap2/context
```

9. List the created image:

```
$ podman images
REPOSITORY                                                    TAG        IMAGE ID      CREATED           SIZE
localhost/powerodba                                                latest     e04948d6013a  About a minute ago  908 MB
<none>                                                    <none>      1f47c496159f  2 minutes ago       908 MB
registry.redhat.io/ansible-automation-platform-24/ee-supported-rhel8  latest     b2d26de2d8de  4 months ago
1.79 GB
registry.redhat.io/ansible-automation-platform-24/ee-minimal-rhel8    latest     c239714e9480  4 months ago
380 MB
```

Once the Execution Environment is created, there's NO need to do it again.

### Execution of playbooks from CLI using ansible-navigator:

1. Create a file called ansible-navigator.yml inside the {{ collection_name }}/playbooks directory with the following content.

```
$ cat ansible-navigator.yml
---
ansible-navigator:
  execution-environment:
  enabled: True
  image: powerodba:latest       # Name of the REPOSITORY:TAG
```

2. Follow the readme files under "docs" to understand how to update the required variables for each task.
3. Run the following command to execute the playbooks. The following example shows execution of manage-db-directories.yml playbook.

```
ansible-navigator run <playbook name> --pp=missing --m stdout -i <name of inventory file>
Example:
$ ansible-navigator run manage-db-directories.yml --pp=missing --m stdout -i hosts.yml
```

To use escalated privileges, please use "--playbook-artifact-enable false" at the end of the command.

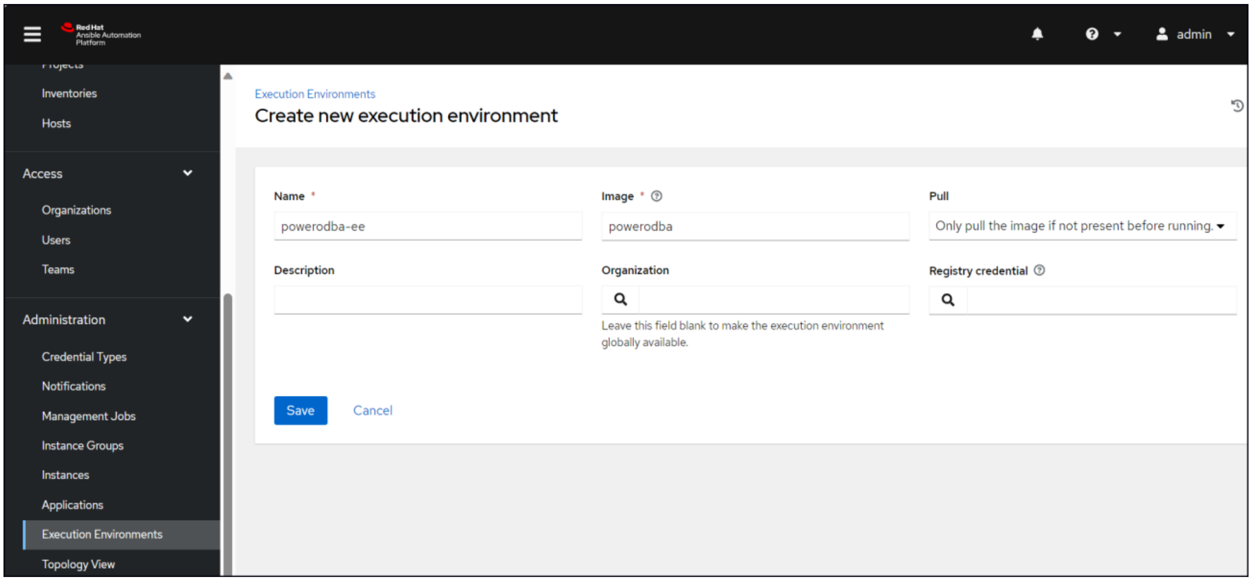
**Example**:

ansible-navigator run db-opatch.yml --pp=missing --m stdout -i hosts.yml --ask-become-pass --playbookartifact-enable false

**Execution of playbooks from GUI:**

We're going to setup the project in AAP2 and show one example template. In this example we're going to execute "manage-users.yml" playbook which will create two users testuser1 and testuser2 in a NON-Container database "atsdb".

Step 1: Login to AAP2 and "Create new execution environment".



Execution Environments

Create new execution environment

Name * powerodba-ee

Image * powerodba

Pull: Only pull the image if not present before running.

Description

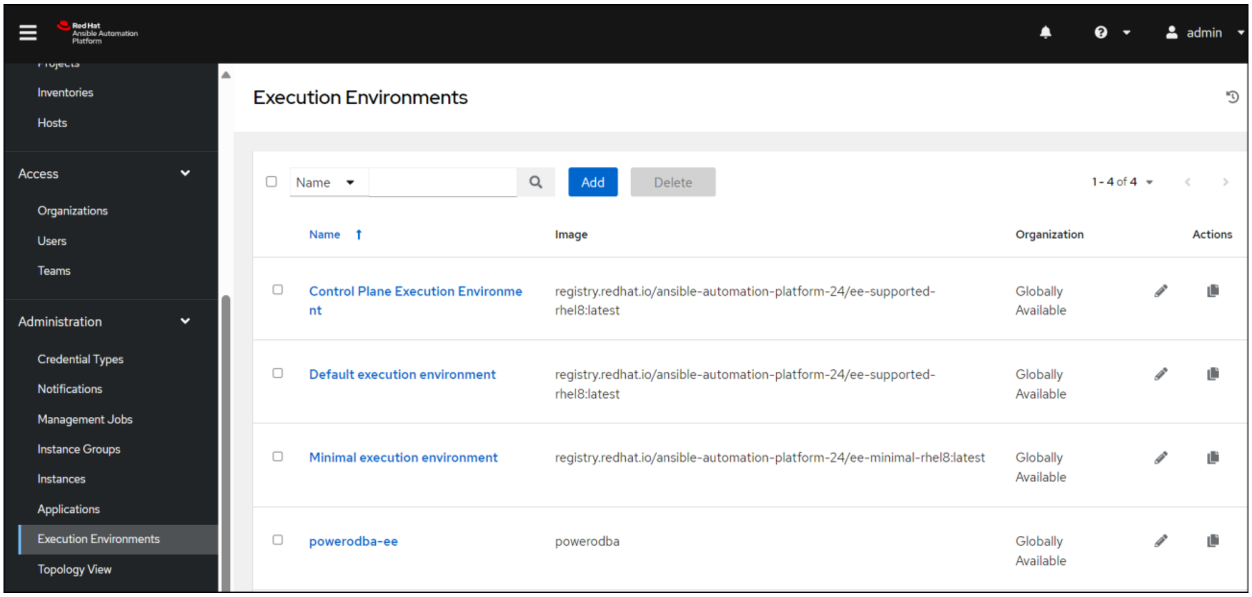
Organization

Leave this field blank to make the execution environment globally available.

Registry credential

Name: Any desired name for identification.

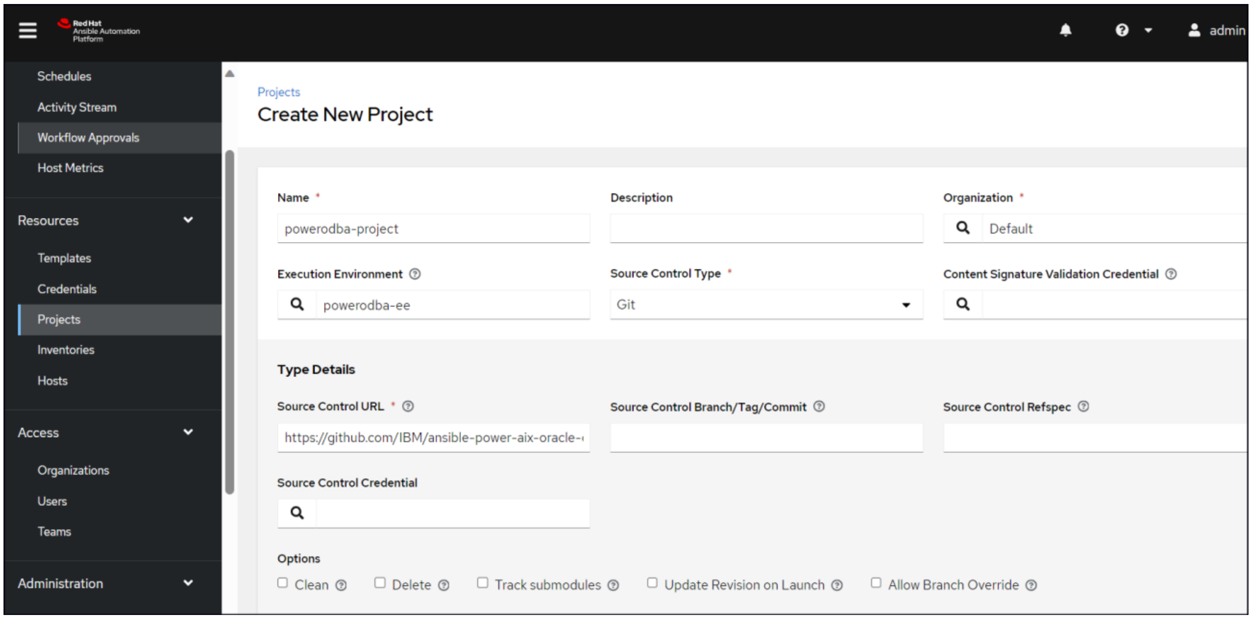
Image: Provide the name of the image which was created earlier in this document using "ansible-builder".



Execution Environments

| Name | Image | Organization |
|---|---|---|
| Control Plane Execution Environment | registry.redhat.io/ansible-automation-platform-24/ee-supported-rhel8:latest | Globally Available |
| Default execution environment | registry.redhat.io/ansible-automation-platform-24/ee-supported-rhel8:latest | Globally Available |
| Minimal execution environment | registry.redhat.io/ansible-automation-platform-24/ee-minimal-rhel8:latest | Globally Available |
| powerodba-ee | powerodba | Globally Available |

Step 2: Create a new project.



Projects

Create New Project

Name * powerodba-project

Description

Organization * Default

Execution Environment powerodba-ee

Source Control Type * Git

Content Signature Validation Credential

Type Details

Source Control URL * https://github.com/IBM/ansible-power-aix-oracle-

Source Control Branch/Tag/Commit

Source Control Refspec

Source Control Credential

Options

□ Clean □ Delete □ Track submodules □ Update Revision on Launch □ Allow Branch Override

Name: Any desired name for identification.

Organization: Your existing Organization name or leave it “Default”

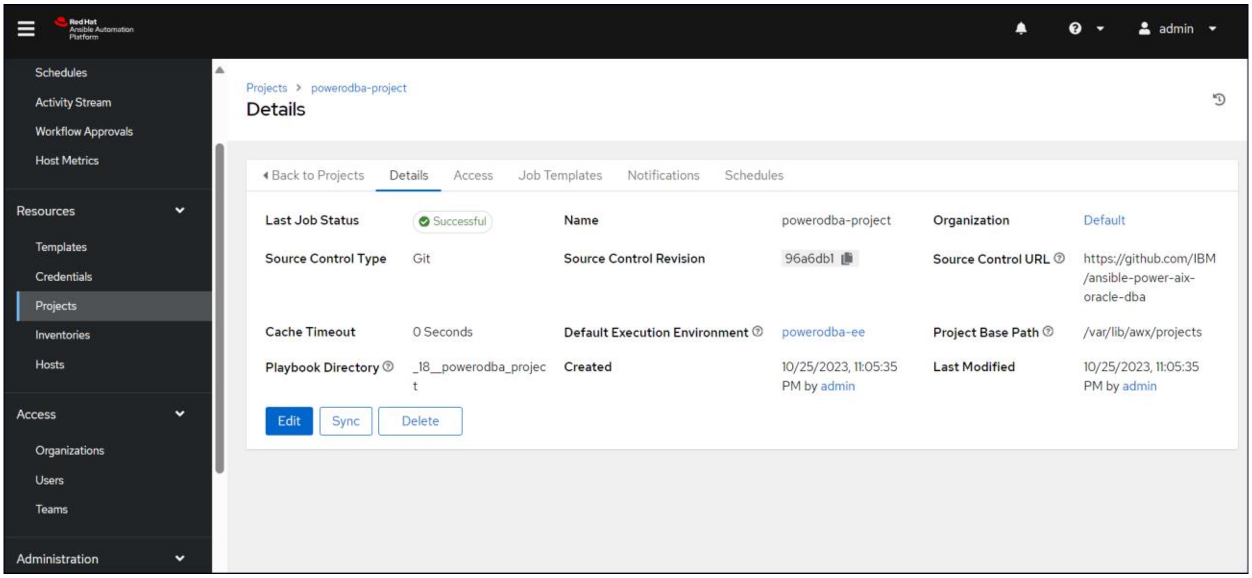
Execution Environment: Name of the Execution Environment created in Step 1.

Source Control Type: Git

Source Control URL: https://github.com/IBM/ansible-power-aix-oracle-dba

Click “save”

The “Last Job Status” must show Successful as shown below.



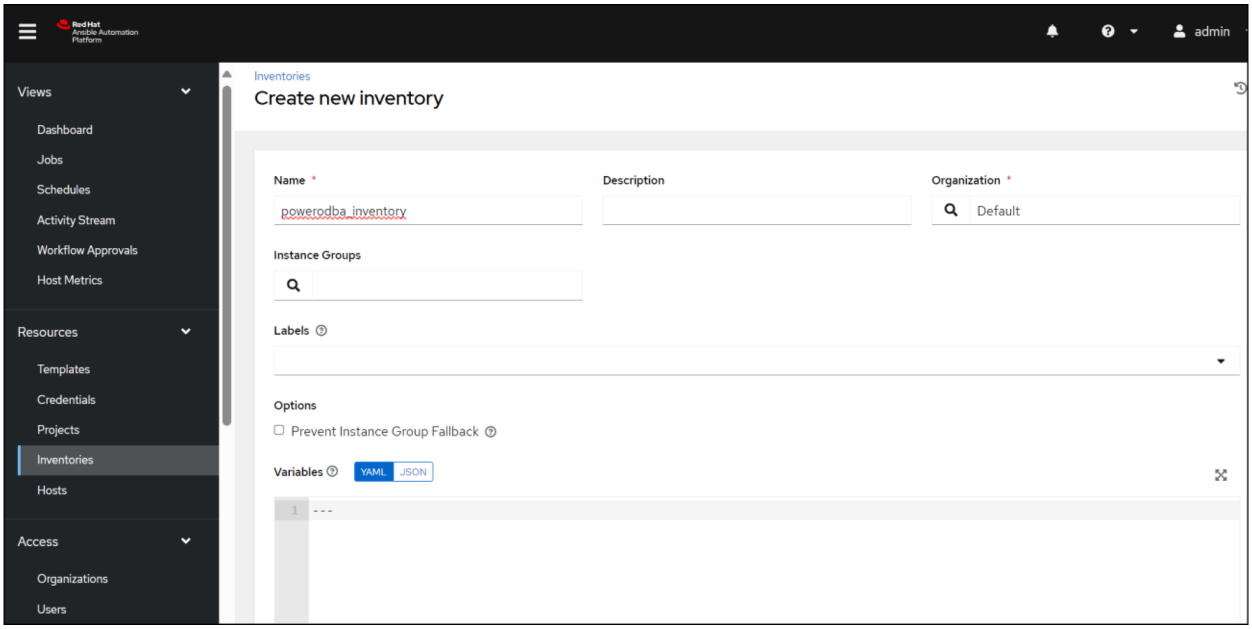
Projects > powerodba-project

Details

| | | | | | |
|---|---|---|---|---|---|
| Last Job Status | Successful | Name | powerodba-project | Organization | Default |
| Source Control Type | Git | Source Control Revision | 96a6db1 | Source Control URL | https://github.com/IBM/ansible-power-aix-oracle-dba |
| Cache Timeout | 0 Seconds | Default Execution Environment | powerodba-ee | Project Base Path | /var/lib/awx/projects |
| Playbook Directory | _18__powerodba_project | Created | 10/25/2023, 11:05:35 PM by admin | Last Modified | 10/25/2023, 11:05:35 PM by admin |

From the above fig, the collection is installed in “/var/lib/awx/projects”

Step 3: Create an inventory with a desired name and save it.



Inventories

Create new inventory

Name * powerodba_inventory

Description

Organization * Default

Instance Groups

Labels

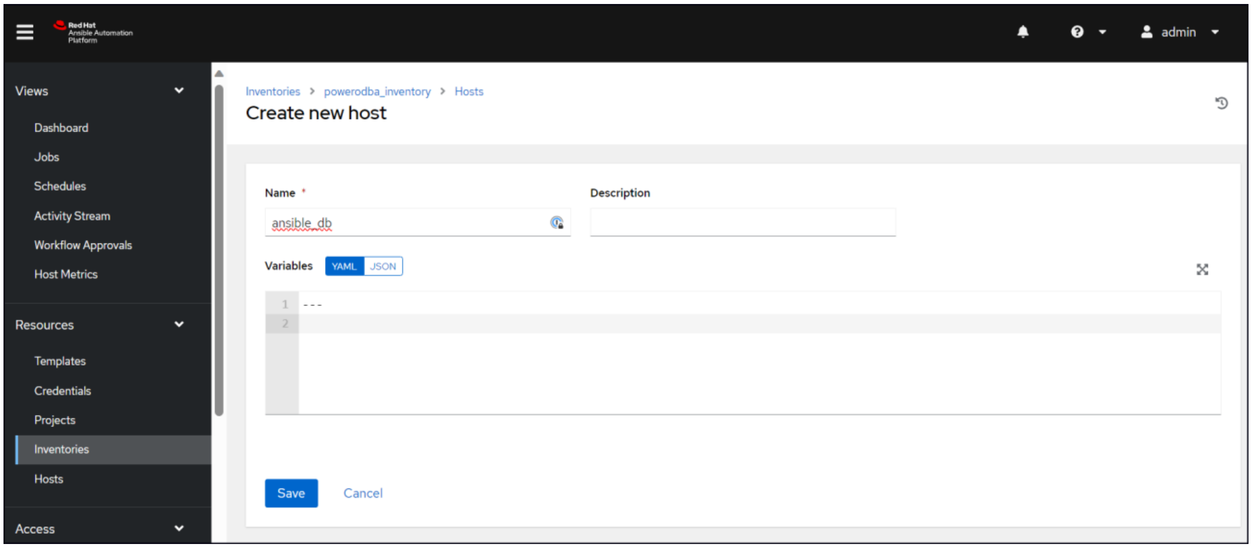
Options

□ Prevent Instance Group Fallback

Variables YAML JSON

```
---
```

Step 4: Add the hostname where the database is running in the inventory and save it.



Inventories > powerodba_inventory > Hosts

Create new host

Name * ansible_db

Description

Variables YAML JSON

```
---
```

The Name should match the host defined in your inventory.

Example: Go to this location - /var/lib/awx/projects/_18__powerodba_project/playbooks and create an inventory.yml file as shown below. Here ansible_db is the host which has an IP 192.168.10.3.

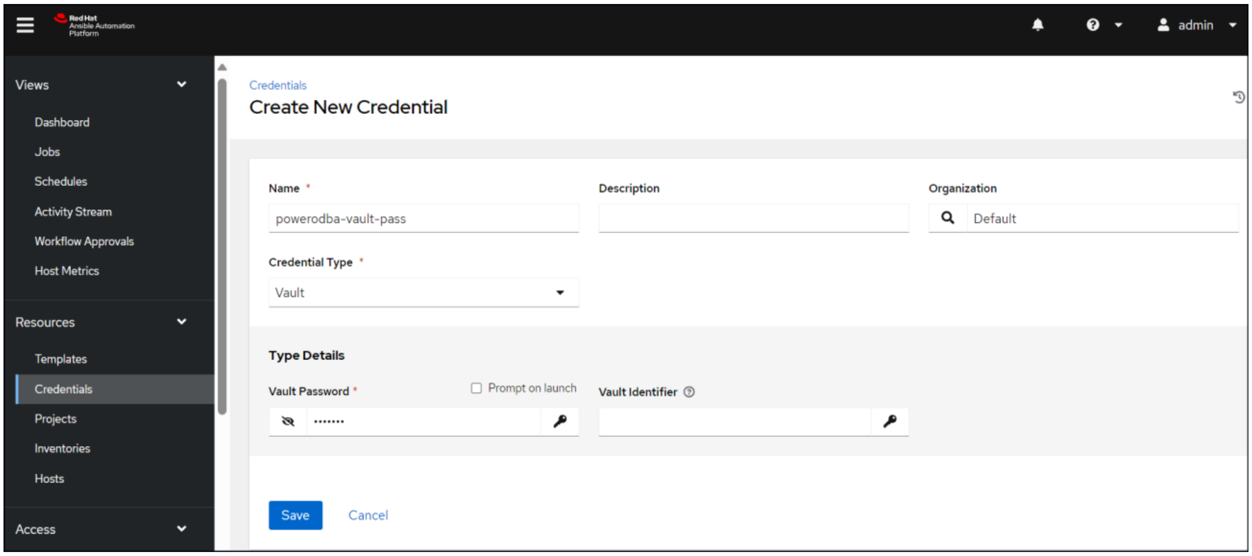
```
[awx@p208n149 playbooks]$ pwd
/var/lib/awx/projects/_18__powerodba_project/playbooks
[awx@p208n149 playbooks]$ cat inventory.yml
ansible_db
[192.168.10.3]
```

Step 5: Set SYS user password in the vault.yml file for “default_dbpass” variable and encrypt it.

```
[awx@p208n149 vars]$ pwd
/var/lib/awx/projects/_18__powerodba_project/playbooks/vars
[awx@p208n149 vars]$ cat vault.yml
default_gipass: Oracle4u      # ASM sys user password
default_dbpass: Oracle4u          # Sys user password
```

```
[awx@p208n149 vars]$ ansible-vault encrypt vault.yml
New Vault password:
Confirm New Vault password:
Encryption successful
```

Step 6: Set the vault password in “Create New Credentials” tab.



Create New Credential

Name * powerodba-vault-pass

Description

Organization Default

Credential Type * Vault

Type Details

Vault Password * ☐ Prompt on launch •••••••

Vault Identifier

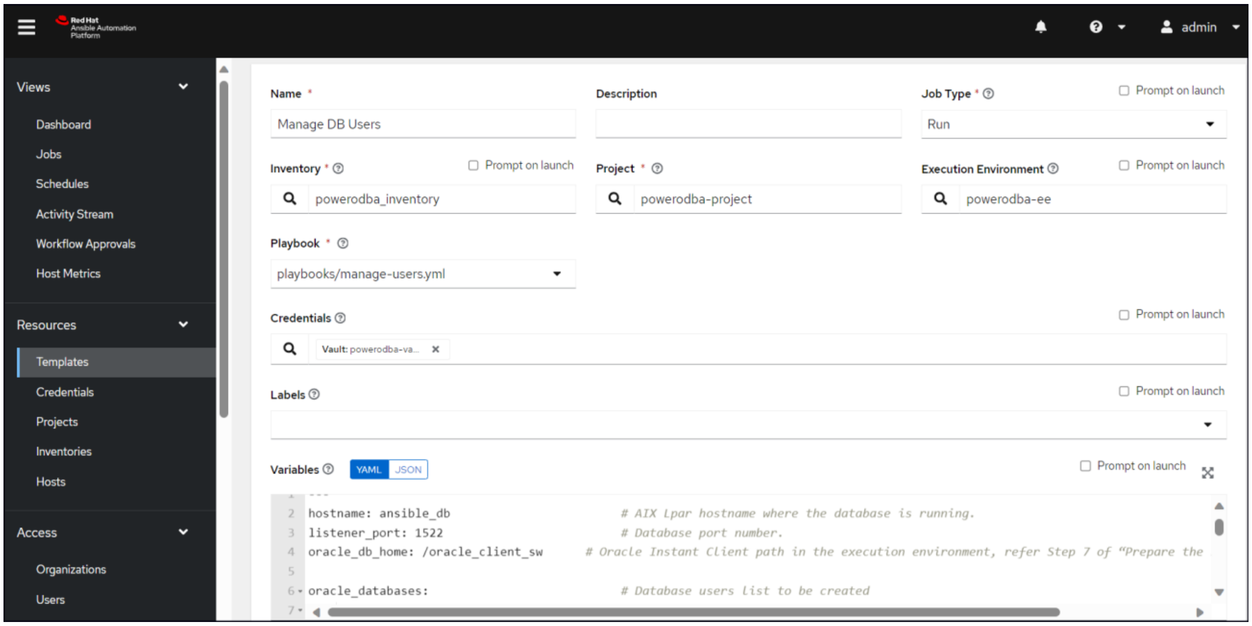
Name: Any desired name for identification.

Credential Type: vault

Vault Password: Update the password used in Step 5.

Step 7: Each playbook must have its own template. In this example we're going to create a template for manage-users.yml playbook which will create database users and grant privileges to it.

Create New Job Template → Add job template.



Name * : Manage DB Users

Description :

Job Type * : Run — □ Prompt on launch

Inventory * : powerodba_inventory — □ Prompt on launch

Project * : powerodba-project

Execution Environment : powerodba-ee — □ Prompt on launch

Playbook * : playbooks/manage-users.yml

Credentials : Vault: powerodba-va... — □ Prompt on launch

Labels : — □ Prompt on launch

Variables (YAML | JSON) — □ Prompt on launch

```
---
hostname: ansible_db                    # AIX Lpar hostname where the database is running.
listener_port: 1522                     # Database port number.
oracle_db_home: /oracle_client_sw     # Oracle Instant Client path in the execution environment, refer Step 7 of "Prepare the

oracle_databases:                       # Database users list to be created
```

Name: Manage DB Users

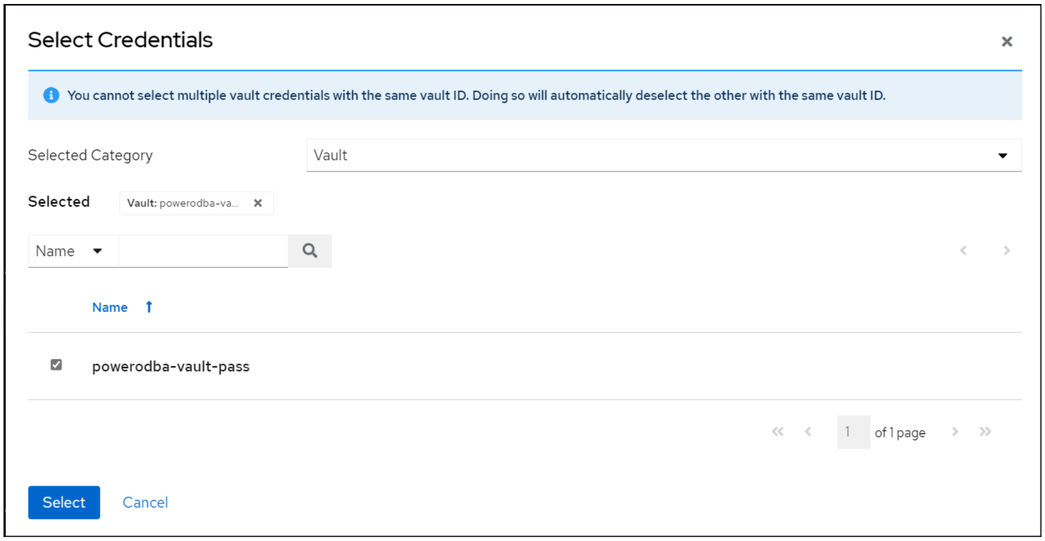
Inventory: Select the inventory created in Step 3.

Project: Select the project created in Step 2.

Execution Environment: Select the Execution environment used in Step 1.

Playbook: From the drop down, select manage-users.yml

Credentials: Select the credentials created in Step 6.



Select Credentials

You cannot select multiple vault credentials with the same vault ID. Doing so will automatically deselect the other with the same vault ID.

Selected Category: Vault

Selected: Vault: powerodba-va...

| Name ↑ |
|---|
| ☑ powerodba-vault-pass |

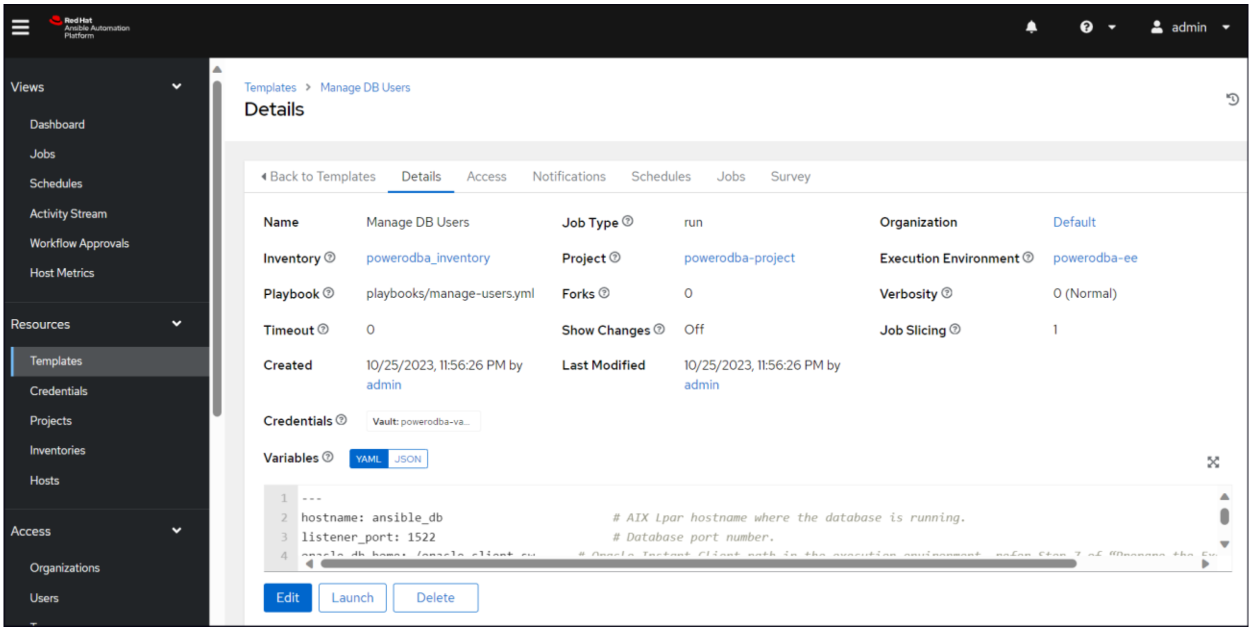
Variables:

```
---
hostname: ansible_db                    # AIX Lpar hostname where the database is running.
listener_port: 1522                   # Database port number.
oracle_db_home: /oracle_client_sw      # Oracle Instant Client path in the execution environment, refer Step 7 of “Prepare
the Execution Environment”

oracle_databases:                     # Database users list to be created
    - users:
      - schema: testuser1           # Username to be created.
     default_tablespace: users         # Default tablespace to be assigned to the user.
     service_name: atsdb             # Database service name.
     schema_password: oracle3          # Password for the user.
     grants_mode: enforce             # enforce|append.
     grants:
      - connect                     # Provide name of the privilege as a list to grant to the user.
      - resource
     state: present                  # present|absent|locked|unlocked [present: Creates user, absent: Drops user]
# Multiple users can be created with different attributes as shown below.
    - users:
      - schema: testuser2
     default_tablespace: users
     service_name: atsdb
     grants_mode: enforce
     grants:
      - connect
     schema_password: oracle4
     state: present
```

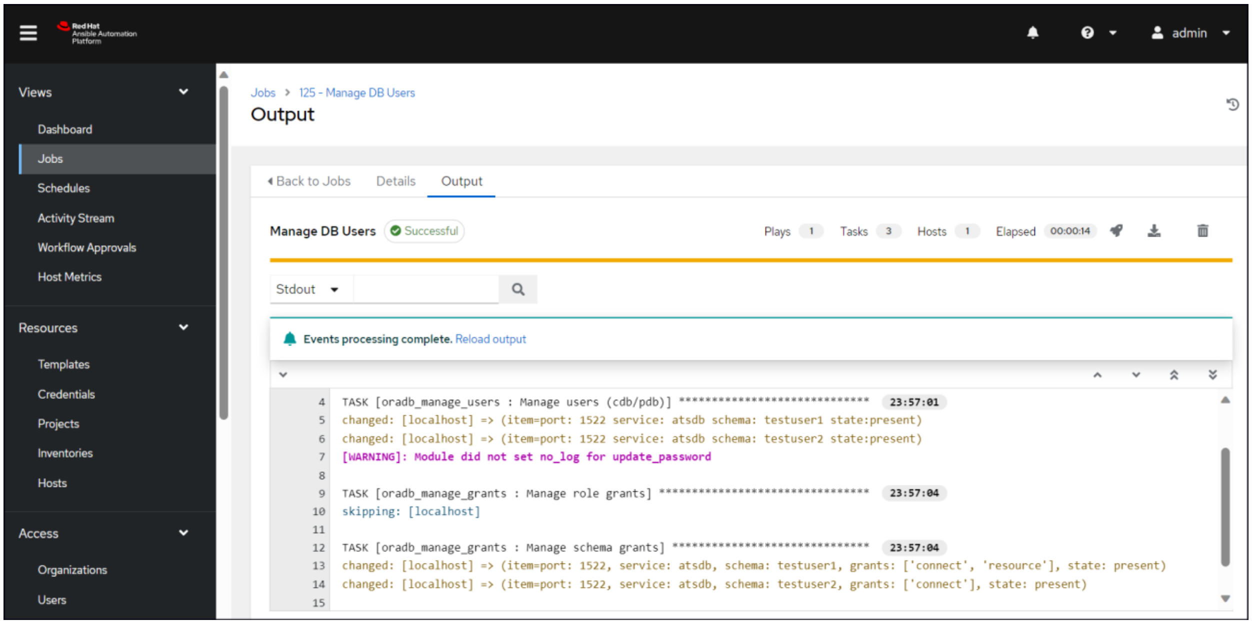
Save the template.

Step 8: Click Launch



Name: Manage DB Users
Job Type: run
Organization: Default
Inventory: powerodba_inventory
Project: powerodba-project
Execution Environment: powerodba-ee
Playbook: playbooks/manage-users.yml
Forks: 0
Verbosity: 0 (Normal)
Timeout: 0
Show Changes: Off
Job Slicing: 1
Created: 10/25/2023, 11:56:26 PM by admin
Last Modified: 10/25/2023, 11:56:26 PM by admin
Credentials: Vault: powerodba-va...
Variables:

```
---
hostname: ansible_db                    # AIX Lpar hostname where the database is running.
listener_port: 1522                     # Database port number.
oracle_db_home: /oracle_client_sw    # Oracle Instant Client path in the execution environment, refer Step 7 of “Prepare the Ex
```



Manage DB Users ✓ Successful

Plays 1 Tasks 3 Hosts 1 Elapsed 00:00:14

Events processing complete. Reload output

```
TASK [oradb_manage_users : Manage users (cdb/pdb)] ****************************** 23:57:01
changed: [localhost] => (item=port: 1522 service: atsdb schema: testuser1 state:present)
changed: [localhost] => (item=port: 1522 service: atsdb schema: testuser2 state:present)
[WARNING]: Module did not set no_log for update_password

TASK [oradb_manage_grants : Manage role grants] ********************************* 23:57:04
skipping: [localhost]

TASK [oradb_manage_grants : Manage schema grants] ******************************* 23:57:04
changed: [localhost] => (item=port: 1522, service: atsdb, schema: testuser1, grants: ['connect', 'resource'], state: present)
changed: [localhost] => (item=port: 1522, service: atsdb, schema: testuser2, grants: ['connect'], state: present)

```

We can see two users testuser1 and testuser2 have been created and granted privileges.