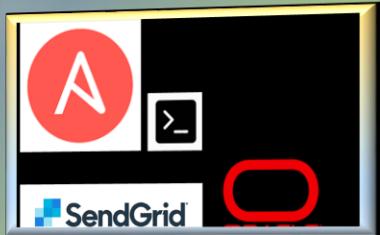




<https://linkedin.com/in/prafulpatel16>



<https://github.com/prafulpatel16>



## OCI BOOTCAMP PROJECT

### OCI PROJECT 1: DEPLOYING USERS, GROUPS AND POLICIES IN A

100% AUTOMATED WAY USING ANSIBLE AND CLOUD SHELL

### EMAIL AUTOMATION CHALLENGE USING ANSIBLE & SENDGRID

#### EMAIL API

IMPLEMENTED BY: PRAFUL PATEL

## PRAFUL PATEL

<https://linkedin.com/in/prafulpatel16>

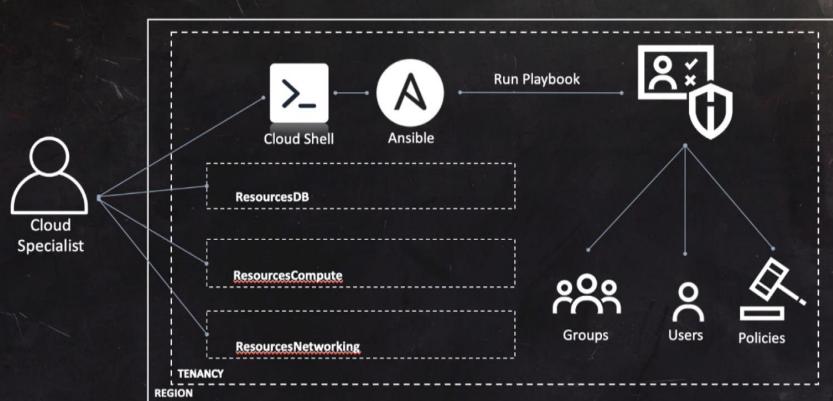
<https://github.com/prafulpatel16>



## SOLUTION ARCHITECTURE

## BOOTCAMP ORACLE CLOUD

DEPLOYING USERS, GROUPS AND POLICIES IN A 100% AUTOMATED WAY USING ANSIBLE AND CLOUD SHELL



Date: APR 18, 2022

➤ **Project Definition:**

**DEPLOYING USERS, GROUPS AND POLICIES IN A 100% AUTOMATED WAY USING ANSIBLE AND CLOUD SHELL**

+

**EMAIL AUTOMATION CHALLENGE USING ANSIBLE & SENDGRID EMAIL API**

An IT services provider company called **PRAfect Systems Inc.** is engaged into providing software development solutions. They have recently migrated their entire on-premises workload to the OCI (Oracle Cloud) and now they are facing a challenge to migrate and create a new users for all the departments. The IT manager has requested to utilize an automation and configuration management tool Ansible to create Users, Groups, Compartments and Policies so that the process can be speed up.

Also there was one more requirement which is when user is created automatically and email should be sent automatically to the respected users. And this challenge is supposed to solve by using and integrating an external Email API tool name sendGrid Email API.

➤ **Tools & Technologies covered:**

- OCI (Oracle Cloud)
- Ansible (Automation & Configuration Management Tool)
- sendGrid (Automated Email API)
- Oracle Cloud Services
  - Identity & Security
    - Identity
      - Users
      - Groups
    - Policies
    - Compartments
  - Oracle Cloud Shell

➤ **Solution Environment Preparation:**

- **Users, Groups, Policies and Compartments creation structure**

► Compartments Creation:

- resourcesCompute
- resourcesDatabase
- resourcesNetworking

- ▶ Groups Creation
    - ▶ ComputeAdmin
    - ▶ DBAsAdmin
    - ▶ NetworkingAdmin
  - ▶ Policies Creation
    - ▶ [ComputeAdminPolicy](#)
    - ▶ [DBAsAdminPolicy](#)
    - ▶ [NetworkingAdminPolicy](#)
  - ▶ Users Creation
- 
- ▶ Backend Environment:
    1. Ansible yaml file creation
      - a. tcb-bmc-iam-creating-groups-and-policies.yaml
      - b. tcb-bmc-iam-creating-users.yaml
      - c. tcb-bmc-iam-creating-users-cloud-admin.yaml
      - d. tcb-bmc-iam-creating-users-compute-admin.yaml
      - e. tcb-bmc-iam-creating-users-dba-admin.yaml
      - f. tcb-bmc-iam-creating-users-network-admin.yaml
      - g. tcb-bmc-iam-creating-users-operators.yaml
      - h. tcb-bmc-iam-creating-users-cloud-support.yaml
    2. Export Environment variables from Oracle Cloud Shell
      - a. export  
PARENT\_COMPARTMENT\_OCID=ocid1.tenancy.oc1..aaaaaaaaaygkyhfqjhggxm6jh  
g4ahphxb3dbrect3f6kcyjsjnamqntq
      - b. export  
NETWORKING\_COMPARTMENT\_OCID=ocid1.compartment.oc1..aaaaaaaaahb7dn  
otdmhhpr5xi36fah3e2ullcccevpemq
      - c. export  
COMPUTE\_COMPARTMENT\_OCID=ocid1.compartment.oc1..aaaaaaaaavrc4twkn  
ydsfa7ylu54gn2ne2mbtel2ioagfihibea
      - d. export  
DB\_COMPARTMENT\_OCID=ocid1.compartment.oc1..aaaaaaaaazictlmyngmmgl37  
zu75fzjmkpn5wbxdb3xbxaipiuj
    3. External Email API Integration : sendGrid Email API
      - a. Create an account with sendGrid
      - b. Create an Email API
      - c. Integrate that sendGrid API to Ansible Playbook

This project demonstrates an experience of 100% deploying and creating IAM Users, Groups, Policies and Compartments inside OCI (Oracle Cloud) completely automated way, and also integrating an external email API (sendGrid) through which automated email can be sent to the users after user creation process.

➤ **Solution:**

This project is based on an automation challenge using Ansible automation tool. The project requirement is to create an IAM Users, Groups and Policies completely automated way. And also there is one more additional automation challenge which is to send an email upon users creation by integrating external sendGrid email API.

In order to achieve and deploy the entire automation process it needs to create some required resources inside Oracle Cloud. First of all it will need to create required Compartments manually inside the Oracle Cloud which would be utilized with the user creation process through Ansible.

Now it needs to create all the required Ansible playbooks first which would run and create an IAM users including groups and all the necessary policies to access the web services inside Oracle Cloud.

Following Ansible Playbooks needs to be created and has to run from Oracle Cloud Shell

- a. tcb-bmc-iam-creating-groups-and-policies.yaml
- b. tcb-bmc-iam-creating-users.yaml
- c. tcb-bmc-iam-creating-users-cloud-admin.yaml
- d. tcb-bmc-iam-creating-users-compute-admin.yaml
- e. tcb-bmc-iam-creating-users-dba-admin.yaml
- f. tcb-bmc-iam-creating-users-network-admin.yaml
- g. tcb-bmc-iam-creating-users-operators.yaml
- h. tcb-bmc-iam-creating-users-cloud-support.yaml

Now it needs to prepare and environment from Oracle Cloud Shell in order to run ansible playbooks so that oracle resources can be accessed through Ansible playbooks.

Once the Environment is configured all the playbooks has been created it needs to run all playbooks in a sequential manner in order to create an IAM users.

Ansible playbooks < playbook name>

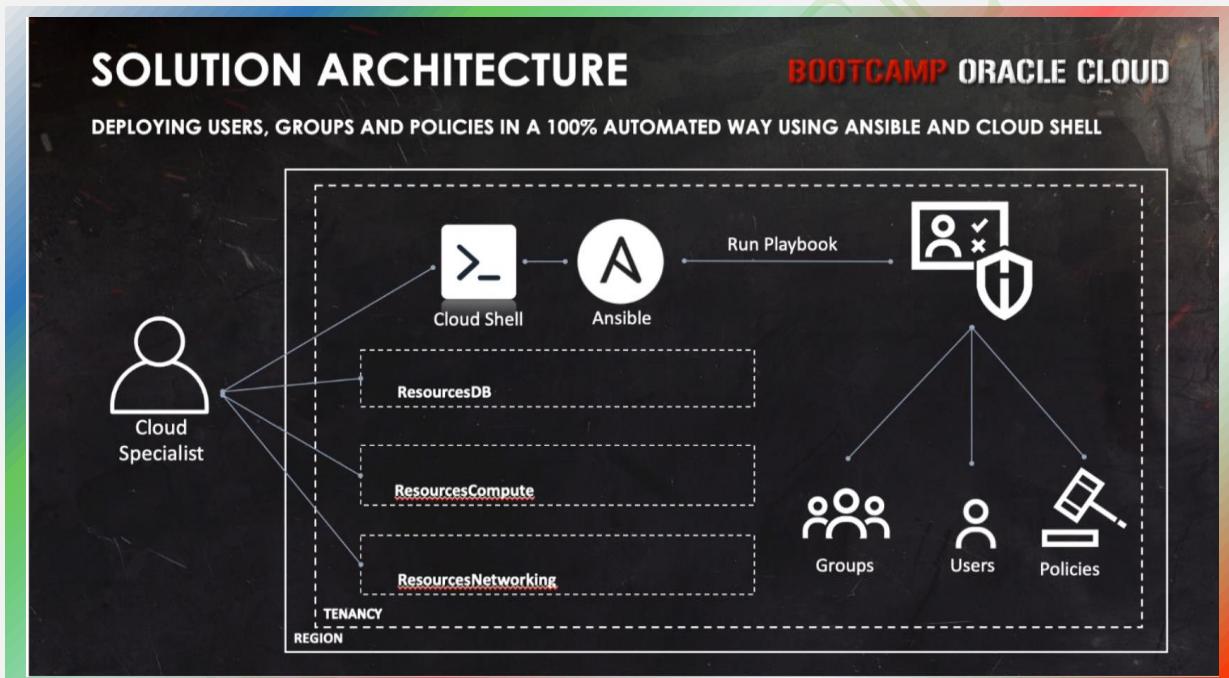
Next is to verify all those IAM resources have been created inside the Oracle Cloud and integrated the policies correctly with each users.

For automated email sent functionality, create a new account on sendGrid web portal, create a new web api key and integrate that to the Ansible playbook and run the Users creation process and verify that email has been sent successfully using sendGrid Email API.

➤ **Project Cost Estimation:**

(Note: This cost is Not any actual cost, it's just an estimation based on high level requirement. Price may be vary based on adding and removing services based on requirement.)

➤ **Solution Architecture:**



This migration project will be completed in 6 implementation phases.

➤ **Project implementation Phases:**

- Phase 1: Create account in OCI (Oracle Cloud).
- Phase 2: Create Compartments manually.
- Phase 3: Prepare Ansible environment and create a playbooks.
- Phase 4: Run all required playbooks from Oracle Cloud Shell to create users automation.
- Phase 5: Configure external API & integrate sendGrid web email API.

➤ **Implementation:**

- Phase 1: Create account in OCI (Oracle Cloud).
  1. Go to Oracle Cloud and create account.

**⊕ Phase 2: Create Compartments manually.**

1. Go to Identity & Security and create Compartments.
  - i. resourcesCompute
  - ii. resourcesDatabase
  - iii. resourcesNetworking

**⊕ Phase 3: Prepare Ansible environment and create a playbooks.**

1. tcb-bmc-iam-creating-groups-and-policies.yaml
2. tcb-bmc-iam-creating-users.yaml
3. tcb-bmc-iam-creating-users-cloud-admin.yaml
4. tcb-bmc-iam-creating-users-compute-admin.yaml
5. tcb-bmc-iam-creating-users-dba-admin.yaml
6. tcb-bmc-iam-creating-users-network-admin.yaml
7. tcb-bmc-iam-creating-users-operators.yaml
8. tcb-bmc-iam-creating-users-cloud-support.yaml

**⊕ Phase 4: Run all required playbooks from Oracle Cloud Shell to create users automation.**

- Run all playbooks in a sequence to create resources automated way.
- Go to Oracle Cloud shell and export the compartments variables
- Run playbook - Ansible playbook < playbook name >
- Verify that all resources have been created inside Oracle Cloud.

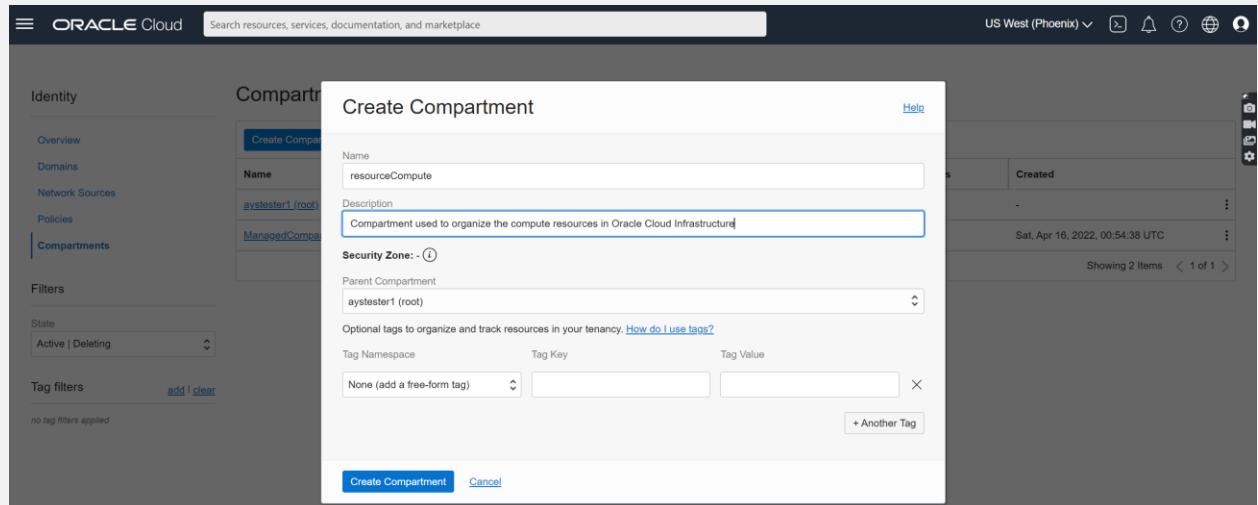
**⊕ Phase 5: Configure sendGrid external Email API & integrate with Ansible.**

1. Go to sendGrid web portal - create an account.
2. Access Web Email API - create a sendGrid API key
3. Add a sendGrid api key to Ansible playbook.
4. Run a playbook to create a user using ansible.
5. Verify that automated email has sent using sendGrid email api from ansible.

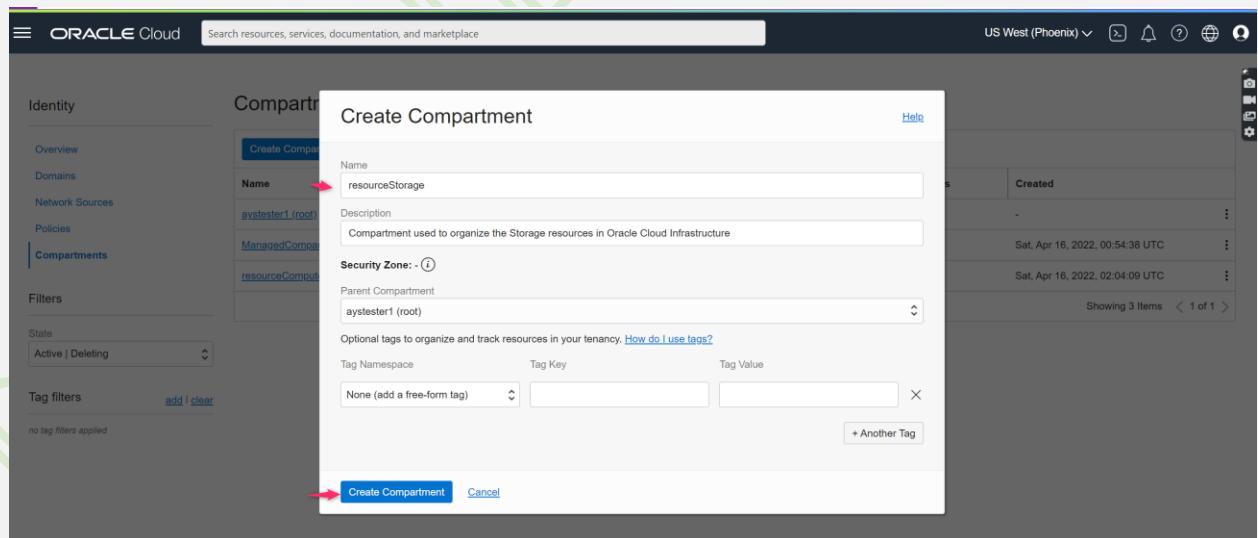
✓ **Solution Implementation in Action:**

1. Create a Compartment

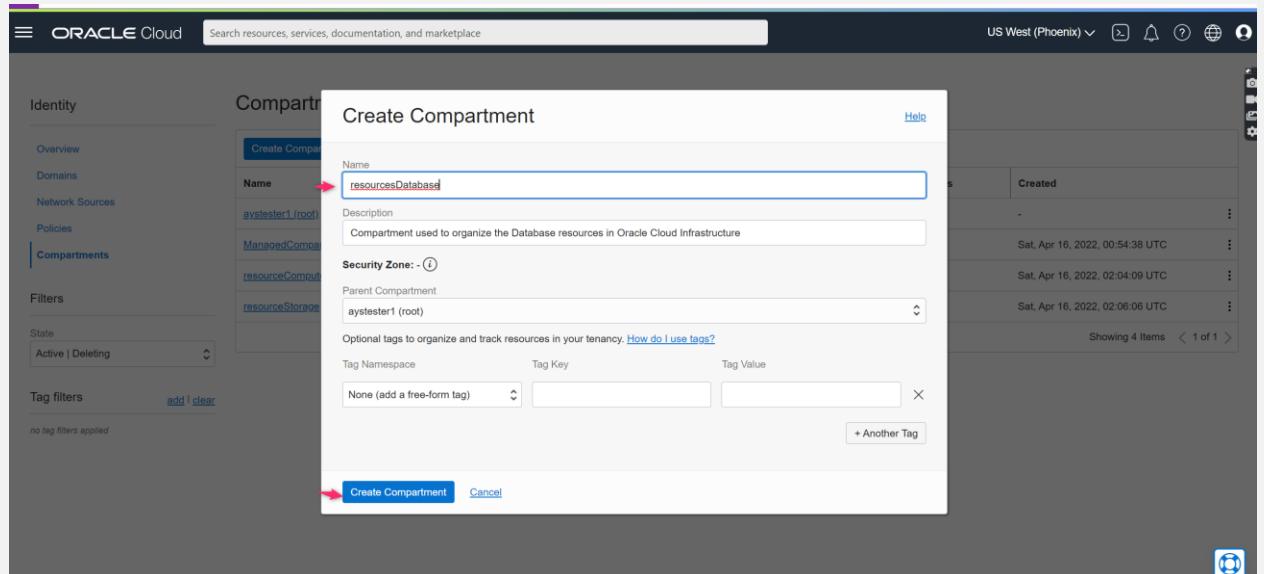
a. Resource compute



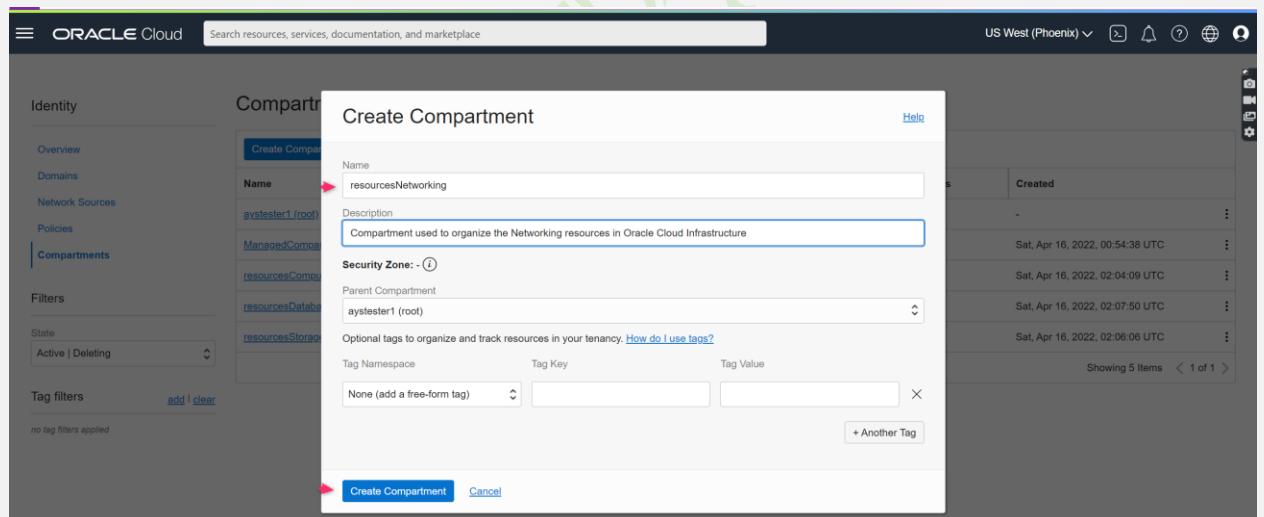
b. Resource storage



c. Resource database



#### d. Resources Networking



Compartments created

**Compartments**

Name	Status	OCID	Authorized	Security Zone	Subcompartments	Created
aysteller1 (root)	Active	...amontig	Yes	-	5	Sat, Apr 16, 2022, 00:54:38 UTC
ManagedCompartmentForPaaS	Active	...v3lmea	Yes	-	0	Sat, Apr 16, 2022, 02:04:09 UTC
resourcesCompute	Active	...ihbeaa	Yes	-	0	Sat, Apr 16, 2022, 02:07:50 UTC
resourcesDatabase	Active	...aiplua	Yes	-	0	Sat, Apr 16, 2022, 02:10:31 UTC
resourcesNetworking	Active	...evpema	Yes	-	0	Sat, Apr 16, 2022, 02:10:31 UTC
resourcesStorage	Active	...4cfzfa	Yes	-	0	Sat, Apr 16, 2022, 02:06:06 UTC

Showing 6 items < 1 of 1 >

**Create Groups**

**Users in Default Domain**

Username	Status	Email	Last access	Created
aysteller1@gmail.com	Active	aysteller1@gmail.com	Sat, Apr 16, 2022, 01:09:47 UTC	Sat, Apr 16, 2022, 00:22:55 UTC

Showing 1 item < Page 1 >

**Groups in Default Domain**

Name	Description	Created
All Domain Users	A group representing all users.	Sat, Apr 16, 2022, 00:22:58 UTC
Administrators	Administrators	Sat, Apr 16, 2022, 00:22:55 UTC

Showing 2 groups < Page 1 >

- Groups:

## CloudAdminExample

Users with Administrator privileges in Oracle Cloud Infrastructure

## DatabaseAdminExample

Users with DBA privileges in Oracle Cloud Infrastructure

### 1. Group : CloudAdminExample

The screenshot shows the Oracle Cloud Identity Groups creation interface. On the left, there's a sidebar with 'Identity domain' navigation. The 'Groups' option is selected. The main area shows a list of groups under 'Groups in Default Domain' with options to 'Create group' or 'More actions'. The 'Create group' dialog is open, showing the following fields:

- Name:** CloudAdminExample (highlighted with a red circle)
- Description:** Users with Administrator privileges in Oracle Cloud Infrastructure (highlighted with a red circle)
- User can request access:** Unchecked checkbox
- Users:** Optional (text)
- Select users to assign this group:** Search bar and table showing one user assigned: Praful Patel (aysteller1@gmail.com)
- Show Advanced Options:** Link

At the bottom of the dialog, there are 'Create' and 'Cancel' buttons, with a red arrow pointing to the 'Create' button.

The screenshot shows the Oracle Cloud Identity Groups page. At the top, there's a navigation bar with 'ORACLE Cloud' and a search bar. Below it, the URL is 'Identity » Domains » Default domain » Groups » CloudAdminExample'. The main content area has a large blue circular icon with a white 'G'. The title 'CloudAdminExample' is displayed above a table with columns for 'Edit group', 'Add Tags', and 'Delete'. Below the table, tabs for 'Group information' and 'Tags' are visible. Under 'Group information', details like OCID, Name (CloudAdminExample), Description (Users with Administrator privileges in Oracle Cloud Infrastructure), and User can request access (No) are shown. On the left sidebar, 'Resources' is expanded, showing 'Users' (selected) and 'Applications'. The 'Users' section lists a table with columns for 'Username', 'Display name', 'Title', and 'Email'. The table is empty with the message 'No items found.' and shows '0 Selected'. At the bottom right of the table, it says 'Showing 0 Items < Page 1 >'.

## 2. DatabaseAdminExample

Users with DBA privileges in Oracle Cloud Infrastructure

The screenshot shows the Oracle Cloud Identity Groups page. The left sidebar shows 'Identity domain' with 'Groups' selected. The main area shows 'Groups in Default Domain' with a search bar and a 'Create group' button. A modal window titled 'Create group' is open. It has two numbered steps: 1. 'Name' (DatabaseAdminExample) and 2. 'Description' (Users with DBA privileges in Oracle Cloud Infrastructure). There's a checkbox for 'User can request access' which is unchecked. Below the description, it says 'Optional' and 'Select users to assign this group.' A search bar and a table for selecting users are shown. The table has columns for 'First name', 'Last name', and 'Email'. One user, 'Praful Patel' (email: aystester1@gmail.com), is selected. At the bottom of the modal, there are 'Show Advanced Options' and 'Create' and 'Cancel' buttons. A red arrow points to the 'Create' button. The background shows a partially visible 'Create Users' section.

Create Users

The screenshot shows the Oracle Cloud Identity domain users page. On the left, there's a sidebar with options like Overview, Users (which is selected), Groups, Dynamic groups, Applications, Oracle Cloud Services, Jobs, Reports, and Security. The main area is titled "Users in Default Domain" and contains a search bar and a table with one row. The table columns are Username, Status, Email, Last access, and Created. The single row shows "aystester1@gmail.com" as the Username, "Active" as the Status, "aystester1@gmail.com" as the Email, "Sat, Apr 16, 2022, 01:09:47 UTC" as the Last access, and "Sat, Apr 16, 2022, 00:22:55 UTC" as the Created date. A red arrow points to the "Create user" button at the top of the table.

- Users:

david

- Groups: Cloud Admin

The screenshot shows the "Create user" dialog box. It has fields for First name (david), Last name, Username / Email (david@gmail.com), and a checkbox for "Use the email address as the username". Below these, there's a section for "Assign cloud account administrator role" with a checkbox. The "Groups" section lists four groups: DatabaseAdminExample, CloudAdminExample (which is checked), All Domain Users, and Administrators. A red circle with the number 3 is on the "CloudAdminExample" checkbox, and another red circle with the number 4 is on the "Create" button at the bottom.

The screenshot shows the Oracle Cloud Identity User Management interface. At the top, it displays the navigation path: Identity > Domains > Default domain > Users > david@gmail.com. Below this, there's a large green circular profile picture with a white 'U' in the center, labeled 'ACTIVE'. The main content area shows the user details for 'david koz'.

User information	Tags	Capabilities
OCID: ...ffova Show Copy User name: david@gmail.com Prefix: - First name: david Middle name: - Last name: koz Suffix: -		Email: david@gmail.com Recovery email: david@gmail.com Instant messaging address: - Home phone number: - Mobile phone number: - Federated: No My Oracle Support Account: - Authenticated by: IDCSApp

Below the user information, there are sections for 'Work information' and 'Other information'. On the left, under 'Resources', there are links for Groups, Applications, API keys, and Auth tokens.

**Groups**

Assign user to group: Remove user from group

Name	Description
CloudAdminExample	Users with Administrator privileges in Oracle Cloud Infrastructure

raj:

### - Groups: DatabaseAdmin

The screenshot shows the Oracle Cloud Identity User Management interface for creating a new user. The navigation path is Identity > Domains > Default domain > Users. The left sidebar shows the 'Identity domain' section with 'Users' selected. The main area is titled 'Create user'.

Step-by-step process:

- First name: Praful
- Last name: Patel
- Username: praful@gmail.com
- Groups: DatabaseAdminExample (selected)
- Create button

The 'Groups' section shows the available groups:

Name	Description
DatabaseAdminExample	Users with DBA privileges in Oracle Cloud Infrastructure
CloudAdminExample	Users with Administrator privileges in Oracle Cloud Infrastructure
All Domain Users	A group representing all users.
Administrators	Administrators

The screenshot shows the Oracle Cloud Identity interface for a user named Praful Patel. The user has a green circular profile picture with a white 'U' and the word 'ACTIVE' below it. The user information section includes:

- OCID:** ...2m7pq... (with Show and Copy options)
- User name:** praful@gmail.com
- Prefix:** -
- First name:** Praful
- Middle name:** -
- Last name:** Patel
- Suffix:** -
- Email:** praful@gmail.com
- Recovery email:** praful@gmail.com
- Instant messaging address:** -
- Home phone number:** -
- Mobile phone number:** -
- Federated:** No
- My Oracle Support Account:** -
- Authenticated by:** IDCSApp

Below the user information, there are sections for **Work information** and **Other information**. On the left, under **Resources**, there is a sidebar with links for Groups, Applications, API keys, Auth tokens, and a link to a client marketplace.

**Groups** section shows a table with one entry:

Assign user to groups		Remove user from group
<input type="checkbox"/>	Name	Description
<input type="checkbox"/>	DatabaseAdminExample	Users with DBA privileges in Oracle Cloud Infrastructure
0 Selected		

Showing 1 item < Page 1 >

## Create Policy

The screenshot shows the Oracle Cloud Identity Policies page for the compartment 'aysteller1 (root)'. The sidebar on the left shows links for Overview, Domains, Network Sources, Policies (which is selected), Compartments, List Scope, and Compartment (set to 'aysteller1 (root)').

The main area displays a table of policies:

Policies in aystester1 (root) Compartment			
<input type="checkbox"/>	Name	Description	Statements
<input type="checkbox"/>	PSM-root-policy	PSM managed compartment root policy	7 Sat, Apr 16, 2022, 00:54:38 UTC
<input type="checkbox"/>	Tenant Admin Policy	Tenant Admin Policy	1 Sat, Apr 16, 2022, 00:23:23 UTC
<input type="checkbox"/>	BootstrapProvisioning	Temporary policy for /bootstrapDefaultDomain API's whitelisted services. Gives access to IAM work request APIs and ability to delete these policy when access is no longer needed.	2 Sat, Apr 16, 2022, 00:21:53 UTC
0 Selected			

Displaying 3 Policies < Page 1 >

A red arrow points to the 'Create Policy' button at the top of the table.

- Policy:

TenancyAdminPolicyExample

Tenancy Admin policy.

The screenshot shows the Oracle Cloud Policy Builder interface. The policy is being created with the following details:

- Name:** TenancyAdminPolicyExample (marked with red circle 1)
- Description:** Tenancy Admin policy (marked with red circle 2)
- Compartment:** aystester1 (root) (marked with red circle 3)
- Policy use cases:** Compartment Management (marked with red circle 4)
- Identity domain:** Default (marked with red circle 5)
- Groups:** CloudAdminExample (marked with red circle 6)
- Location:** aystester1 (root) (marked with red circle 7)
- Policy Statements:** Allow group 'Default'/'CloudAdminExample' to manage all-resources in tenancy aystester1 (root) (highlighted with a red box).

The screenshot shows the details of the 'TenancyAdminPolicyExample' policy:

- Policy Information:**
  - OCID: ...6byhufng
  - Compartment: aystester1 (root)
  - Description: Tenancy Admin policy
  - Created: Sat, Apr 16, 2022, 03:06:38 UTC
- Statements:**
  - Allow group 'Default'/'CloudAdminExample' to manage all-resources in tenancy

### DatabaseAdminPolicyExample

DBA Admin policy.

allow group DatabaseAdminExample to manage database-family in compartment resourcesDatabase

Create Policy

Name: DatabaseAdminPolicyExample  
No spaces. Only letters, numerals, hyphens, periods, or underscores.

Description: DBA Admin policy.

Compartment: aysteller1 (root)

Policy Builder Show manual editor

allow group DatabaseAdminExample to manage database-family in compartment resourcesDatabase

Show Advanced Options

Create Cancel Create Another Policy

Identity Policies Policy Detail

DatabaseAdminPolicyExample

ACTIVE

Edit Policy Add Tags Delete

Policy Information Tags

OCID: ...wfqvqdpmq Show Copy  
Compartment: aysteller1 (root)  
Description: DBA Admin policy.  
Created: Sat, Apr 16, 2022, 03:10:17 UTC

Resources Statements

Edit Policy Statements

allow group DatabaseAdminExample to manage database-family in compartment resourcesDatabase

Showing 1 Item

### Create USER's Groups and Policies using Ansible automation

- You should have the below compartments already in place:

**networkingResources**

**computeResources**

**databaseResources**

The screenshot shows the Oracle Cloud Identity interface. On the left, there's a sidebar with 'Identity' selected, followed by 'Overview', 'Domains', 'Network Sources', 'Policies', 'Compartments' (which is highlighted with a red circle), 'Filters', 'State' (with 'Active | Deleting'), and 'Tag filters'. The main area is titled 'Compartments' and contains a table with the following data:

Name	Status	OCID	Authorized	Security Zone	Subcompartments	Created
aysteller1/root	Active	...amontq	Yes	-	5	-
ManagedCompartmentForPaaS	Active	...y3tmea	Yes	-	0	Sat, Apr 16, 2022, 00:54:38 UTC
<b>1 resourcesCompute</b>	Active	...ihbeaa	Yes	-	0	Sat, Apr 16, 2022, 02:04:09 UTC
<b>2 resourcesDatabase</b>	Active	...ajpuuq	Yes	-	0	Sat, Apr 16, 2022, 02:07:50 UTC
<b>3 resourcesNetworking</b>	Active	...expemq	Yes	-	0	Sat, Apr 16, 2022, 02:10:31 UTC
resourcesStorage	Active	...4cfzfq	Yes	-	0	Sat, Apr 16, 2022, 02:06:06 UTC

Showing 6 Items < 1 of 1 >

## # Steps using the Cloud Shell in OCI

- Setting up the environment variable

Get and replace the OCID of the Tenancy and the compartments:

NetworkingResources, ComputeResources and databaseResources:

```
export PARENT_COMPARTMENT_OCID=<insert-TENANCY-ocid>
```

The screenshot shows the Oracle Cloud Shell terminal. It displays a welcome message and information about Java runtimes. The user runs the command `export PARENT_COMPARTMENT_OCID=ocid1.user.oc1...`. The terminal window has a dark background with white text.

```
export NETWORKING_COMPARTMENT_OCID=<insert-NETWORKING_COMPARTMENT-ocid>
```

```
export COMPUTE_COMPARTMENT_OCID=<insert-COMPUTE_COMPARTMENT-ocid>
```

```
export DB_COMPARTMENT_OCID=<insert-DB_COMPARTMENT-ocid>
```

The screenshot shows the Oracle Cloud Shell terminal. It displays a welcome message and information about Java runtimes. The user runs three commands to set environment variables for Networking, Compute, and DB compartments. The terminal window has a dark background with white text.

```
export PARENT_COMPARTMENT_OCID=ocid1.user.oc1..aaaaaaaaay4kh3e
export NETWORKING_COMPARTMENT_OCID=ocid1.compartment.oc1..aaaaaaa
export COMPUTE_COMPARTMENT_OCID=ocid1.compartment.oc1..aaaaaaa
export DB_COMPARTMENT_OCID=ocid1.compartment.oc1..aaaaaaaaazict
```

- Creating and accessing a folder to download the code

```
mkdir tcb-bmc-iam
```

```
cd tcb-bmc-iam
```

- Downloading the file Ansible Playbooks

```
wget https://objectstorage.us-ashburn-1.oraclecloud.com/p/eRJ8pKclDBA_MZeuaqj75jkfpq3yvuqe4TslzNjqY7Y1RKTLMGipnbwfPO7cnp5F/n/idqfa2zz2mift/b/bootcamp-oci/o/EN/oci-handson-module-iam.zip
```



The screenshot shows a terminal window in the Oracle Cloud Cloud Shell interface. The user has run the command `wget https://objectstorage.us-ashburn-1.oraclecloud.com/p/eRJ8pKclDBA_MZeuaqj75jkfpq3yvuqe4TslzNjqY7Y1RKTLMGipnbwfPO7cnp5F/n/idqfa2zz2mift/b/bootcamp-oci/o/EN/oci-handson-module-iam.zip`. The output shows the progress of the download, indicating it's 100% complete at 5,098 bytes. The URL is highlighted with a red arrow.

```
unzip oci-handson-module-iam.zip
```

- Running the Ansible Playbooks from Cloud Shell

```
ansible-playbook tcb-bmc-iam-creating-groups-and-policies.yaml
```

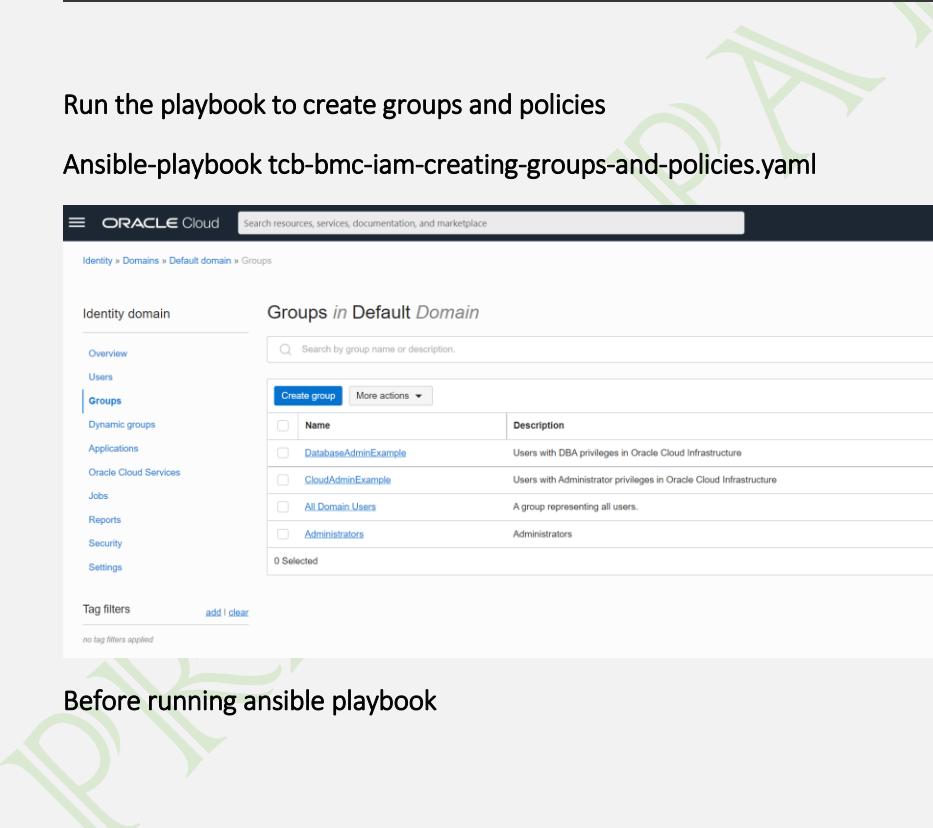
```
ansible-playbook tcb-bmc-iam-creating-users-cloud-admin.yaml
```

```
ansible-playbook tcb-bmc-iam-creating-users-dba-admin.yaml
```

```
ansible-playbook tcb-bmc-iam-creating-users-network-admin.yaml
```

```
ansible-playbook tcb-bmc-iam-creating-users-compute-admin.yaml
```

```
ansible-playbook tcb-bmc-iam-creating-users-operators.yaml
```



ORACLE Cloud Search resources, services, documentation, and marketplace US West (Phoenix) +

Cloud Shell

```

aystester1@cloudshell:~ (us-phoenix-1)$ mkdir tcb-bmc-iam
aystester1@cloudshell:~ (us-phoenix-1)$ cd tcb-bmc-iam/
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ wget https://objectstorage.us-ashburn-1.oraclecloud.com/p/eRJ8pKclDBA_MZeuaqj75jkfpq3yvuqe4TslzNjqY71RKTLGipnbwfP07cnp5F/n/idqfa2z2mift/b/boomcamp-oci/o/EN/oci-handson-module-iam.zip
--2022-04-17 23:30:20-- https://objectstorage.us-ashburn-1.oraclecloud.com/p/eRJ8pKclDBA_MZeuaqj75jkfpq3yvuqe4TslzNjqY71RKTLGipnbwfP07cnp5F/n/idqfa2z2mift/b/bootcamp-oci/o/EN/oci-handson-module-iam.zip
Resolving objectstorage.us-ashburn-1.oraclecloud.com (objectstorage.us-ashburn-1.oraclecloud.com)... 134.70.32.1, 134.70.24.1, 134.70.28.1
Connecting to objectstorage.us-ashburn-1.oraclecloud.com (objectstorage.us-ashburn-1.oraclecloud.com)|134.70.32.1|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5098 (5.0K) [application/zip]
Saving to: "oci-handson-module-iam.zip"

100%[=====] 5,098 --.-K/s in 0s

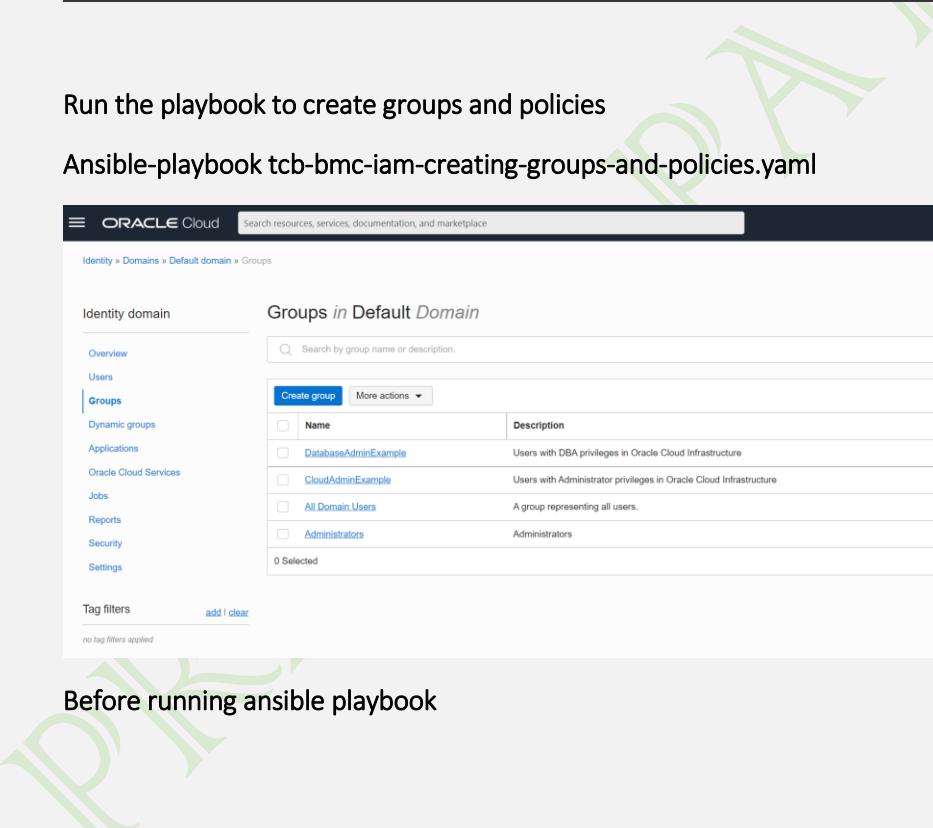
2022-04-17 23:30:20 (78.8 MB/s) - "oci-handson-module-iam.zip" saved [5098/5098]

aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ls -ltr
total 8
-rw-r--r-- 1 aystester1 oci 5098 Sep 25 2021 oci-handson-module-iam.zip
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ unzip oci-handson-module-iam.zip
Archive: oci-handson-module-iam.zip
  inflating: tcb-bmc-iam-creating-groups-and-policies.yaml
  inflating: tcb-bmc-iam-creating-users-cloud-admin.yaml
  inflating: tcb-bmc-iam-creating-users-compute-admin.yaml
  inflating: tcb-bmc-iam-creating-users-dba-admin.yaml
  inflating: tcb-bmc-iam-creating-users-network-admin.yaml
  inflating: tcb-bmc-iam-creating-users-operators.yaml
  inflating: tcb-bmc-iam-creating-users.yaml
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$

```

Run the playbook to create groups and policies

Ansible-playbook tcb-bmc-iam-creating-groups-and-policies.yaml



ORACLE Cloud Search resources, services, documentation, and marketplace US West (Phoenix) +

Identity » Domains » Default domain » Groups

Identity domain Groups in Default Domain

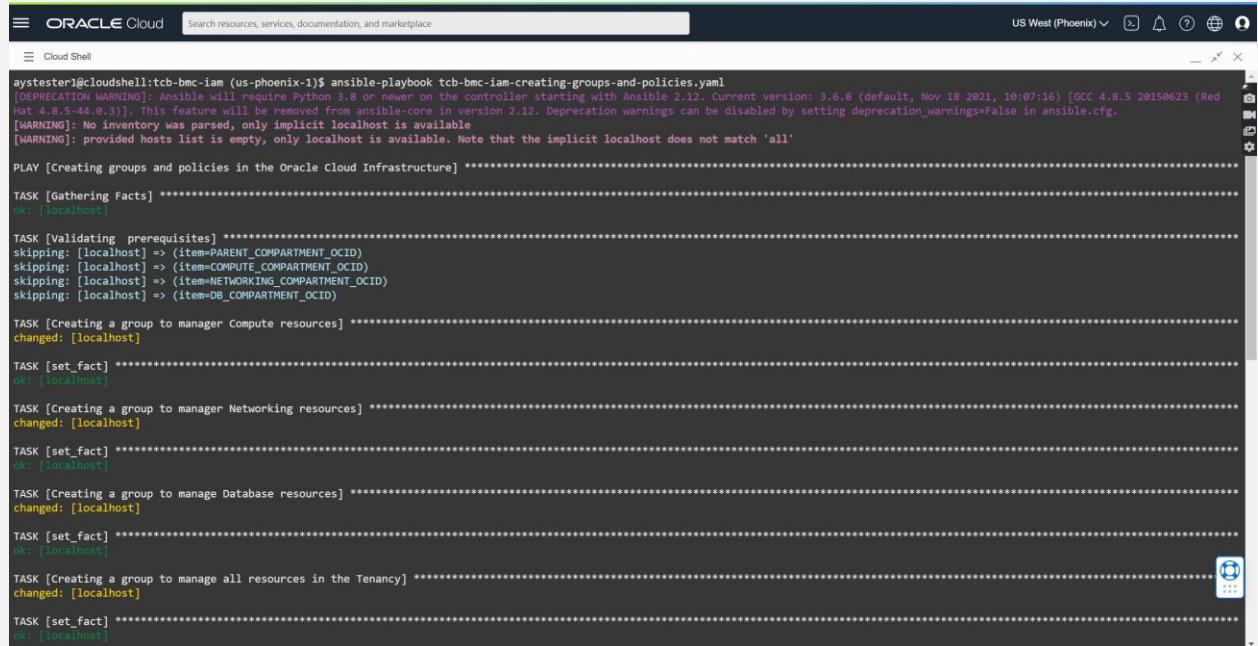
Overview Users Groups Dynamic groups Applications Oracle Cloud Services Jobs Reports Security Settings

Create group More actions ▾

<input type="checkbox"/> Name	Description	Created
<input type="checkbox"/> DatabaseAdminExample	Users with DBA privileges in Oracle Cloud Infrastructure	Sat, Apr 16, 2022, 02:19:30 UTC
<input type="checkbox"/> CloudAdminExample	Users with Administrator privileges in Oracle Cloud Infrastructure	Sat, Apr 16, 2022, 02:17:05 UTC
<input type="checkbox"/> AllDomain.Users	A group representing all users.	Sat, Apr 16, 2022, 00:22:58 UTC
<input type="checkbox"/> Administrators	Administrators	Sat, Apr 16, 2022, 00:22:55 UTC
0 Selected		Showing 4 groups < Page 1 >

Tag filters add | clear no tag filters applied

Before running ansible playbook



```

ayesterer@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-playbook tcb-bmc-iam-creating-groups-and-policies.yaml
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Creating groups and policies in the Oracle Cloud Infrastructure] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Validating prerequisites] *****
skipping: [localhost] => (item=PARENT_COMPARTMENT_OCID)
skipping: [localhost] => (item=COMPUTE_COMPARTMENT_OCID)
skipping: [localhost] => (item=NETWORKING_COMPARTMENT_OCID)
skipping: [localhost] => (item=DB_COMPARTMENT_OCID)

TASK [Creating a group to manager Compute resources] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Creating a group to manager Networking resources] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

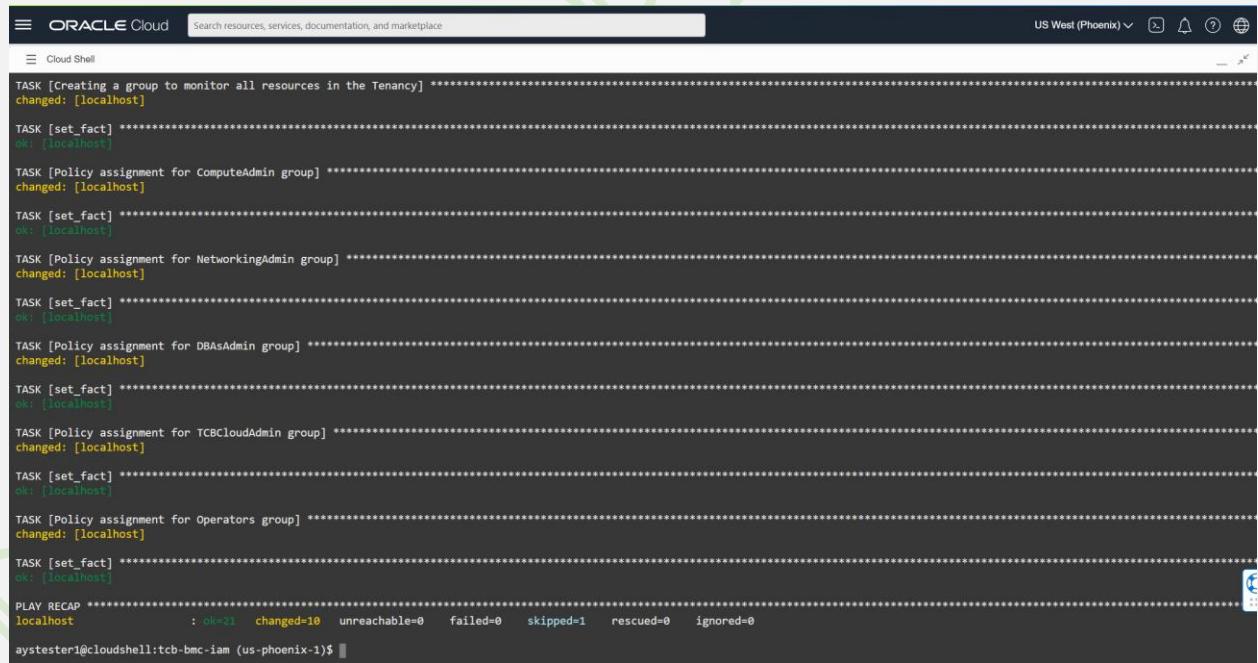
TASK [Creating a group to manage Database resources] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Creating a group to manage all resources in the Tenancy] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

```



```

TASK [Creating a group to monitor all resources in the Tenancy] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Policy assignment for ComputeAdmin group] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Policy assignment for NetworkingAdmin group] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Policy assignment for DBAsAdmin group] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Policy assignment for TCBCloudAdmin group] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Policy assignment for Operators group] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

PLAY RECAP *****
localhost          : ok=21   changed=10   unreachable=0   failed=0    skipped=1   rescued=0   ignored=0
ayesterer@cloudshell:tcb-bmc-iam (us-phoenix-1)$ 

```

Groups created by ansible automation script

The screenshot shows the Oracle Cloud Identity Groups page. The left sidebar has 'Identity domain' selected under 'Groups'. The main area is titled 'Groups in Default Domain' with a search bar. A table lists nine groups, each with a checkbox, name, description, and creation date. Some rows are highlighted with red circles containing numbers 1 through 4.

	Name	Description	Created
3	TCBCloudAdmin	Group in charge to monitor all running resources in the Tenancy	Sun, Apr 17, 2022, 23:58:31 UTC
4	DBAaAdmin	Group allowed to manage DB resources in the compartment dbResources	Sun, Apr 17, 2022, 23:58:27 UTC
1	NetworkingAdmin	Group allowed to manage Networking resources in the compartment networkingResources	Sun, Apr 17, 2022, 23:58:19 UTC
2	ComputeAdmin	Group allowed to manage Compute resources in the compartment computeResources	Sun, Apr 17, 2022, 23:58:15 UTC
	DatabaseAdminExample	Users with DBA privileges in Oracle Cloud Infrastructure	Sat, Apr 16, 2022, 02:19:39 UTC
	CloudAdminExample	Users with Administrator privileges in Oracle Cloud Infrastructure	Sat, Apr 16, 2022, 02:17:05 UTC
	All Domain Users	A group representing all users.	Sat, Apr 16, 2022, 00:22:58 UTC
	Administrators	Administrators	Sat, Apr 16, 2022, 00:22:55 UTC
0 Selected		Showing 9 groups < Page 1 >	

### Create users using ansible automation script

#### Before running user script

The screenshot shows the Oracle Cloud Identity Users page. The left sidebar has 'Identity domain' selected under 'Users'. The main area is titled 'Users in Default Domain' with a search bar. A table lists three users, each with a checkbox, status, email, last access, and creation date.

	Username	Status	Email	Last access	Created
	praful@gmail.com	Active	praful@gmail.com	-	Sat, Apr 16, 2022, 02:58:21 UTC
	david@gmail.com	Active	david@gmail.com	-	Sat, Apr 16, 2022, 02:58:58 UTC
	aysteller1@gmail.com	Active	aysteller1@gmail.com	Sun, Apr 17, 2022, 23:01:37 UTC	Sat, Apr 16, 2022, 00:22:55 UTC
0 Selected		Showing 3 items < Page 1 >			

### Create cloud admin users

```
ansible-playbook tcb-bmc-iam-creating-users-cloud-admin.yaml
```

ORACLE Cloud Search resources, services, documentation, and marketplace US West (Phoenix) ▾

Cloud Shell

```
-rw-r--r--. 1 aystester1 oci 5851 Sep 13 2021 tcb-bmc-iam-creating-groups-and-policies.yaml
-rw-r--r--. 1 aystester1 oci 1027 Sep 13 2021 tcb-bmc-iam-creating-users.yaml
-rw-r--r--. 1 aystester1 oci 5098 Sep 25 2021 oci-handson-module-iam.zip
aysteller1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-playbook tcb-bmc-iam-creating-users-cloud-admin.yaml → [DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Creating cloud Admin user in the OCI] ****
TASK [Gathering Facts] ****
ok: [localhost]

TASK [Validating prerequisites] ****
skipping: [localhost] => (item=PARENT_COMPARTMENT_OCID)

TASK [Getting Group ID] ****
ok: [localhost]

TASK [set_fact] ****
ok: [localhost]

TASK [Creating users, adding to a group and defining the password] ****
included: /home/aysteller1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => (item=jane.doe) ①
included: /home/aysteller1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => (item=john.paul) ②

TASK [Creating user] ****
changed: [localhost]

TASK [set_fact] ****
ok: [localhost]

TASK [debug] ****
ok: [localhost] => {
  "msg": "User jane.doe created"
}

TASK [Adding user to a group] ****
changed: [localhost]
```

Verify from console that users are created

ORACLE Cloud Search resources, services, documentation, and marketplace US West (Phoenix) ▾

Identity » Domains » Default domain » Users

Identity domain

Overview

**Users**

Groups

Dynamic groups

Applications

Oracle Cloud Services

Jobs

Reports

Security

Settings

Tag filters add | clear

no tag filters applied

Users in Default Domain

Create user More actions ▾

Username	Status	Email	Last access	Created
john.paul	Active	john.paul@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:39 UTC
jane.doe	Active	jane.doe@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:24 UTC
praful@gmail.com	Active	praful@gmail.com	-	Sat, Apr 16, 2022, 02:58:21 UTC
david@gmail.com	Active	david@gmail.com	-	Sat, Apr 16, 2022, 02:56:58 UTC
aysteller1@gmail.com	Active	aysteller1@gmail.com	Sun, Apr 17, 2022, 23:01:37 UTC	Sat, Apr 16, 2022, 00:22:55 UTC

0 Selected Showing 5 Items < Page 1 >

Create dbs users

ansible-playbook tcb-bmc-iam-creating-users-dba-admin.yaml

```

aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ls -ltr
total 40
-rw-r--r--. 1 aystester1 oci 929 Sep 13 2021 tcb-bmc-iam-creating-users-cloud-admin.yaml
-rw-r--r--. 1 aystester1 oci 931 Sep 13 2021 tcb-bmc-iam-creating-users-compute-admin.yaml
-rw-r--r--. 1 aystester1 oci 929 Sep 13 2021 tcb-bmc-iam-creating-users-dba-admin.yaml
-rw-r--r--. 1 aystester1 oci 934 Sep 13 2021 tcb-bmc-iam-creating-users-operators.yaml
-rw-r--r--. 1 aystester1 oci 945 Sep 13 2021 tcb-bmc-iam-creating-users-network-admin.yaml
-rw-r--r--. 1 aystester1 oci 5851 Sep 13 2021 tcb-bmc-iam-creating-groups-and-policies.yaml
-rw-r--r--. 1 aystester1 oci 1027 Sep 13 2021 tcb-bmc-iam-creating-users.yaml
-rw-r--r--. 1 aystester1 oci 5098 Sep 25 2021 oci-handson-module-iam.zip

aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-playbook tcb-bmc-iam-creating-users-dba-admin.yaml
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current Version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Creating DBAs Admin users in the OCI] ****
  TASK [Gathering Facts] ****
    ok: [localhost]
  TASK [Validating prerequisites] ****
    skipping: [localhost] => {item=COMPARTMENT_OCID}
  TASK [Getting Group ID] ****
    ok: [localhost]
  TASK [Set fact] ****
    ok: [localhost]
  TASK [Creating users, adding to a group and defining the password] ****
    included: /home/aytester1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => {item=bryan.brown} ①
    included: /home/aytester1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => {item=zyan.white} ②
  TASK [Creating user] ****
    changed: [localhost]
  TASK [Set fact] ****
    ok: [localhost]

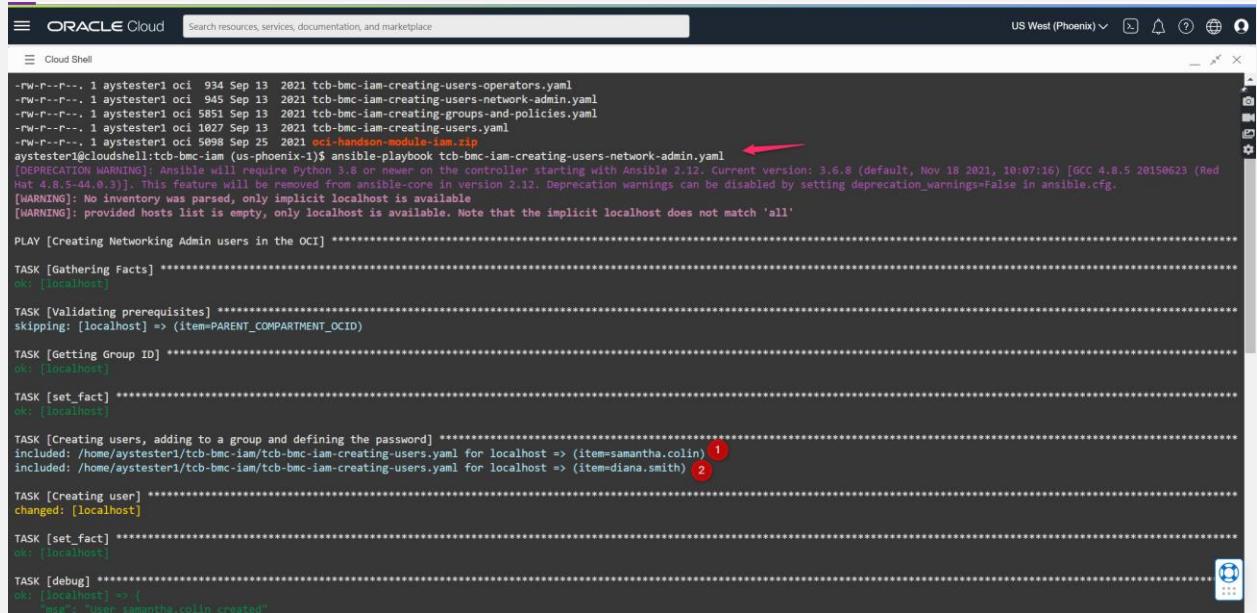
```

Verify that users are created

Username	Status	Email	Last access	Created
zyan.white	Active	zyan.white@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:20:19 UTC
bryan.brown	Active	bryan.brown@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:20:05 UTC
john.paul	Active	john.paul@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:39 UTC
jane.doe	Active	jane.doe@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:24 UTC
praful@gmail.com	Active	praful@gmail.com	-	Sat, Apr 16, 2022, 02:58:21 UTC
david@gmail.com	Active	david@gmail.com	-	Sat, Apr 16, 2022, 02:56:58 UTC
aytester1@gmail.com	Active	aytester1@gmail.com	Sun, Apr 17, 2022, 23:01:37 UTC	Sat, Apr 16, 2022, 00:22:55 UTC

Create network admin users

ansible-playbook tcb-bmc-iam-creating-users-network-admin.yaml



```

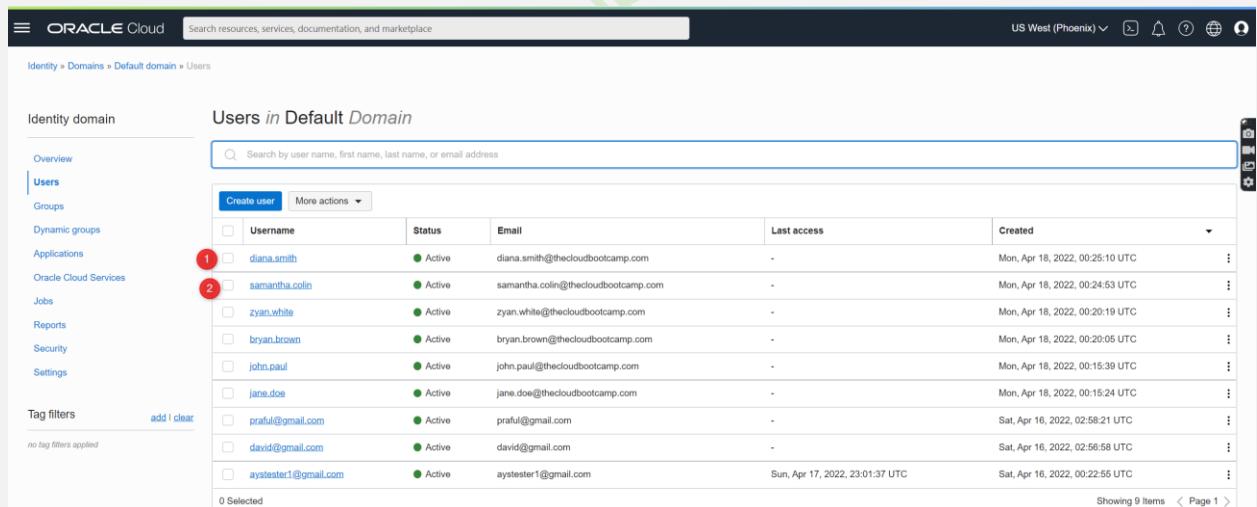
-rw-r--r--. 1 aystester1 oci 934 Sep 13 2021 tcb-bmc-iam-creating-users/operators.yaml
-rw-r--r--. 1 aystester1 oci 945 Sep 13 2021 tcb-bmc-iam-creating-users-network-admin.yaml
-rw-r--r--. 1 aystester1 oci 5851 Sep 13 2021 tcb-bmc-iam-creating-groups-and-policies.yaml
-rw-r--r--. 1 aystester1 oci 1827 Sep 13 2021 tcb-bmc-iam-creating-users.yaml
-rw-r--r--. 1 aystester1 oci 5098 Sep 25 2021 evci-hands-on-modular-iam.zip

aytester1@cloudshell:tcb-bmc-iam [us-phoenix-1]$ ansible-playbook tcb-bmc-iam-creating-users-network-admin.yaml
[DEPRECATION WARNING]: Ansible will require Python 3.6 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.3.el7)]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Creating Networking Admin users in the OCI] ****
TASK [Gathering Facts] ****
ok: [localhost]
TASK [Validating prerequisites] ****
skipping: [localhost] => {item=PARENT_COMPARTMENT_OCID}
TASK [Getting Group ID] ****
ok: [localhost]
TASK [set_fact] ****
ok: [localhost]
TASK [Creating users, adding to a group and defining the password] ****
included: /home/aytester1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => {item=samantha.colin} 1
included: /home/aytester1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => {item=diana.smith} 2
TASK [Creating user] ****
changed: [localhost]
TASK [set_fact] ****
ok: [localhost]
TASK [debug] ****
ok: [localhost] => {msg="User samantha.colin created"}

```

Verify from console that users are created



Username	Status	Email	Last access	Created
diana.smith	Active	diana.smith@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:25:10 UTC
samantha.colin	Active	samantha.colin@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:24:53 UTC
zyan.white	Active	zyan.white@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:20:19 UTC
bryan.brown	Active	bryan.brown@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:20:05 UTC
john.paul	Active	john.paul@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:39 UTC
jane.doe	Active	jane.doe@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:24 UTC
praful@gmail.com	Active	praful@gmail.com	-	Sat, Apr 16, 2022, 02:58:21 UTC
david@gmail.com	Active	david@gmail.com	-	Sat, Apr 16, 2022, 02:56:58 UTC
aytester1@gmail.com	Active	aytester1@gmail.com	Sun, Apr 17, 2022, 23:01:37 UTC	Sat, Apr 16, 2022, 00:22:55 UTC

Create compute admin users

Ansible-playbook

ansible-playbook tcb-bmc-iam-creating-users-compute-admin.yaml

```

-rw-r--r--. 1 aystester1 oci 5851 Sep 13 2021 tcb-bmc-iam-creating-groups-and-policies.yaml
-rw-r--r--. 1 aystester1 oci 1027 Sep 13 2021 tcb-bmc-iam-creating-users.yaml
-rw-r--r--. 1 aystester1 oci 5098 Sep 25 2021 oci-handson-module-iam.zip
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-playbook tcb-bmc-iam-creating-users-compute-admin.yaml
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.el8.3)]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Creating Compute Admin in the OCI] ****
TASK [Gathering Facts] ****
ok: [localhost]
TASK [Validating prerequisites] ****
skipping: [localhost] => {item=PARENT_COMPARTMENT_OCID}
TASK [Getting Group ID] ****
ok: [localhost]
TASK [Set fact] ****
ok: [localhost]
TASK [Creating users, adding to a group and define the password] ****
included: /home/aysteller1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => {item=denny.smith} 1
included: /home/aysteller1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => {item=juan.francisco} 2
TASK [Creating user] ****
changed: [localhost]
TASK [Set fact] ****
ok: [localhost]
TASK [Debug] ****
ok: [localhost] => {
    msg: "User denny.smith created"
}
TASK [Adding user to a group] ****
changed: [localhost]
TASK [Set fact] ****

```

Identity domain

Users in Default Domain

Username	Status	Email	Last access	Created
juan.francisco	Active	juan.francisco@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:34:21 UTC
denny.smith	Active	denny.smith@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:34:03 UTC
diana.smith	Active	diana.smith@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:25:10 UTC
samantha.colin	Active	samantha.colin@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:24:53 UTC
zyan.white	Active	zyan.white@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:20:19 UTC
bryan.brown	Active	bryan.brown@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:20:05 UTC
john.paul	Active	john.paul@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:39 UTC
jane.doe	Active	jane.doe@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:24 UTC
praful@gmail.com	Active	praful@gmail.com	-	Sat, Apr 16, 2022, 02:58:21 UTC
david@gmail.com	Active	david@gmail.com	-	Sat, Apr 16, 2022, 02:56:58 UTC
aysteller1@gmail.com	Active	aysteller1@gmail.com	Sun, Apr 17, 2022, 23:01:37 UTC	Sat, Apr 16, 2022, 00:22:55 UTC

0 Selected

Show 11 items < Page 1 >

Create operator users

ansible-playbook tcb-bmc-iam-creating-users-operators.yaml

```

-rw-r--r--. 1 aystester1 oci 5851 Sep 13 2021 tcb-bmc-iam-creating-groups-and-policies.yaml
-rw-r--r--. 1 aystester1 oci 1027 Sep 13 2021 tcb-bmc-iam-creating-users.yaml
-rw-r--r--. 1 aystester1 oci 5098 Sep 25 2021 oci-hands-on-module-iam.zip
aytester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-playbook tcb-bmc-iam-creating-users-operators.yaml
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Creating Operators users in the OCI] *****

TASK [Gathering Facts] *****
ok: [localhost]

TASK [Validating prerequisites] *****
skipping: [localhost] => {item=PARENT_COMPARTMENT_OCID}

TASK [Getting Group ID] *****
ok: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Creating users, adding to a group and defining the password] *****
included: /home/aytester1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => {item=petter.oliver} 1
included: /home/aytester1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => {item=jonne.delucca} 2

TASK [Creating user] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [debug] *****
ok: [localhost] => {
    "msg": "User petter.oliver created"
}

TASK [Adding user to a group] *****
changed: [localhost]

TASK [set_fact] *****

```

Verify from console that operator users created

Username	Status	Email	Last access	Created
jonne.delucca	Active	jonne.delucca@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:39:35 UTC
petter.oliver	Active	petter.oliver@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:39:19 UTC
juan.francisco	Active	juan.francisco@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:34:21 UTC
denny.smith	Active	denny.smith@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:34:03 UTC
diana.smith	Active	diana.smith@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:25:10 UTC
samantha.colin	Active	samantha.colin@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:24:53 UTC
zyan.white	Active	zyan.white@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:20:19 UTC
bryan.brown	Active	bryan.brown@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:20:05 UTC
john.paul	Active	john.paul@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:39 UTC
jane.doe	Active	jane.doe@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:24 UTC
praful@gmail.com	Active	praful@gmail.com	-	Sat, Apr 16, 2022, 02:58:21 UTC

## USERS

Verify that all user created using Ansible

**Identity domain**

**Users**

Username	Status	Email	Last access	Created
jonne.delucca	Active	jonne.delucca@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:39:35 UTC
petter.oliver	Active	petter.oliver@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:38:19 UTC
juan.francisco	Active	juan.francisco@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:34:21 UTC
denny.smith	Active	denny.smith@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:34:03 UTC
diana.smith	Active	diana.smith@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:25:10 UTC
samantha.colin	Active	samantha.colin@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:24:53 UTC
zyan.white	Active	zyan.white@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:20:19 UTC
bryan.brown	Active	bryan.brown@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:20:05 UTC
john.paul	Active	john.paul@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:39 UTC
jane.doe	Active	jane.doe@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:15:24 UTC
praful@gmail.com	Active	praful@gmail.com	-	Sat, Apr 16, 2022, 02:58:21 UTC
david@gmail.com	Active	david@gmail.com	-	Sat, Apr 16, 2022, 02:56:58 UTC
ayasteer1@gmail.com	Active	ayasteer1@gmail.com	Mon, Apr 18, 2022, 02:04:02 UTC	Sat, Apr 16, 2022, 00:22:55 UTC

Showing 13 items < Page 1

## GROUPS

Verify that all groups created using ansible automation script

**Identity domain**

**Groups**

Name	Description	Created
Operators	Group in charge to monitor all running resources in the Tenancy	Sun, Apr 17, 2022, 23:58:31 UTC
TCPCloudAdmin	Group of Cloud Specialist and Solution Architects	Sun, Apr 17, 2022, 23:58:27 UTC
DBAsAdmin	Group allowed to manage DB resources in the compartment dbResources	Sun, Apr 17, 2022, 23:58:23 UTC
NetworkingAdmin	Group allowed to manage Networking resources in the compartment networkingResources	Sun, Apr 17, 2022, 23:58:19 UTC
ComputeAdmin	Group allowed to manage Compute resources in the compartment computeResources	Sun, Apr 17, 2022, 23:58:15 UTC
DatabaseAdminExample	Users with DBA privileges in Oracle Cloud Infrastructure	Sat, Apr 16, 2022, 02:19:39 UTC
CloudAdminExample	Users with Administrator privileges in Oracle Cloud Infrastructure	Sat, Apr 16, 2022, 02:17:05 UTC
AllDomain.Users	A group representing all users.	Sat, Apr 16, 2022, 00:22:58 UTC
Administrators	Administrators	Sat, Apr 16, 2022, 00:22:55 UTC

Showing 9 groups < Page 1

## POLICIES

Verify that policies are created using Ansible automation script

The screenshot shows the Oracle Cloud Identity Policies page. The left sidebar has 'Identity' selected under 'Policies'. The main area displays a table of policies with columns for Name, Description, Statements, and Created. Red arrows point from the text 'Policies in aystester1 (root) Compartment' to the first two rows of the table.

Name	Description	Statements	Created
OperatorsPolicy	Permission to view and inspect all resources in the Tenancy	1	Sun, Apr 17, 2022, 23:58:49 UTC
TCBCloudAdminPolicy	Permission to manage all resources in the tenancy	1	Sun, Apr 17, 2022, 23:58:46 UTC
DBAsAdminPolicy	Permission to manage DB resources	1	Sun, Apr 17, 2022, 23:58:42 UTC
NetworkingAdminPolicy	Permission to manage Networking resources	2	Sun, Apr 17, 2022, 23:58:39 UTC
ComputeAdminPolicy	Permission to manage Compute resources	1	Sun, Apr 17, 2022, 23:58:35 UTC
DatabaseAdminPolicyExample	DBA Admin policy.	1	Sat, Apr 16, 2022, 03:10:17 UTC
TenancyAdminPolicyExample	Tenancy Admin policy	1	Sat, Apr 16, 2022, 03:06:38 UTC
PSM-root-policy	PSM managed compartment root policy	7	Sat, Apr 16, 2022, 00:54:38 UTC
TenantAdminPolicy	Tenant Admin Policy	1	Sat, Apr 16, 2022, 00:23:23 UTC
BootstrapProvisioning	Temporary policy for bootstrapDefaultDomain API's whitelisted services. Gives access to IAM work request APIs and ability to delete these policy when access is no longer needed.	2	Sat, Apr 16, 2022, 00:21:53 UTC

Verify that Users have been added to the Groups using ansible automation script

Group: ComputeAdmin

The screenshot shows the Oracle Cloud Identity Groups page. The left sidebar has 'Groups' selected under 'ComputeAdmin'. The main area shows group details like OCID, Name, and Description. Below is a table of users assigned to the group, with red circles numbered 1 and 2 pointing to specific rows.

Username	Display name	Title	Email
denny.smith	denny.smith	-	denny.smith@thecloudbootcamp.com
juan.francisco	juan.francisco	-	juan.francisco@thecloudbootcamp.com

Group: DBAsAdmin

**DBAsAdmin**

**Resources**

**Users**

Username	Display name	Title	Email
bryan.brown	bryan.brown	-	bryan.brown@thecloudbootcamp.com
zyan.white	zyan.white	-	zyan.white@thecloudbootcamp.com

Showing 2 Items < Page 1 >

**Group: NetworkingAdmin**

**NetworkingAdmin**

**Resources**

**Users**

Username	Display name	Title	Email
diana.smith	diana.smith	-	diana.smith@thecloudbootcamp.com
samantha.colin	samantha.colin	-	samantha.colin@thecloudbootcamp.com

Showing 2 Items < Page 1 >

**Group: TCBCloudADmin**

The screenshot shows the Oracle Cloud Identity Groups page for the 'TCBCloudAdmin' group. The group has a blue circular icon with a white 'G'. The 'Group information' tab is selected, showing the OCID, Name (TCBCloudAdmin), Description (Group of Cloud Specialist and Solution Architects), and User can request access (No). Below this, the 'Users' section lists two users: 'jane.doe' and 'john.paul', both assigned to the group.

Username	Display name	Title	Email
jane.doe	jane.doe	-	jane.doe@thecloudbootcamp.com
john.paul	john.paul	-	john.paul@thecloudbootcamp.com

### Group: Operators

The screenshot shows the Oracle Cloud Identity Groups page for the 'Operators' group. The group has a blue circular icon with a white 'G'. The 'Group information' tab is selected, showing the OCID, Name (Operators), Description (Group in charge to monitor all running resources in the Tenancy), and User can request access (No). Below this, the 'Users' section lists two users: 'jonne.delucca' and 'petter.oliver', both assigned to the group.

Username	Display name	Title	Email
jonne.delucca	jonne.delucca	-	jonne.delucca@thecloudbootcamp.com
petter.oliver	petter.oliver	-	petter.oliver@thecloudbootcamp.com

Project Implementation Complete.

Congratulations!!!! 🎉

Next: Solve the Automation Challenge using Ansible-sendGrid Email API



## ANSIBLE AUTOMATION EMAIL CHALLENGE USING SENDGRID EMAIL API

**Objective:** To send an automated email after creating each OCI - IAM user.

**Requirement:** It should send Log in credentials (Username & Password) to all the respected users into email body.

**Automation Tool:** Ansible

**Email Automation Tool:** sendGrid Email Web API

1. Go to sendGrid web portal

2. Create an account

3. Create email API key

NAME	API KEY	ACTION
ansible-key API Key ID: ObgzcCVzRb2RBbWUJo4pwg	*****	
sendemail API Key ID: WidaM9VATg6fICLEQRoESQ	*****	

4. Create a Playbook to create a “Cloud Support” user in OCI

`tcb-bmc-iam-creating-users-cloud-support.yaml`

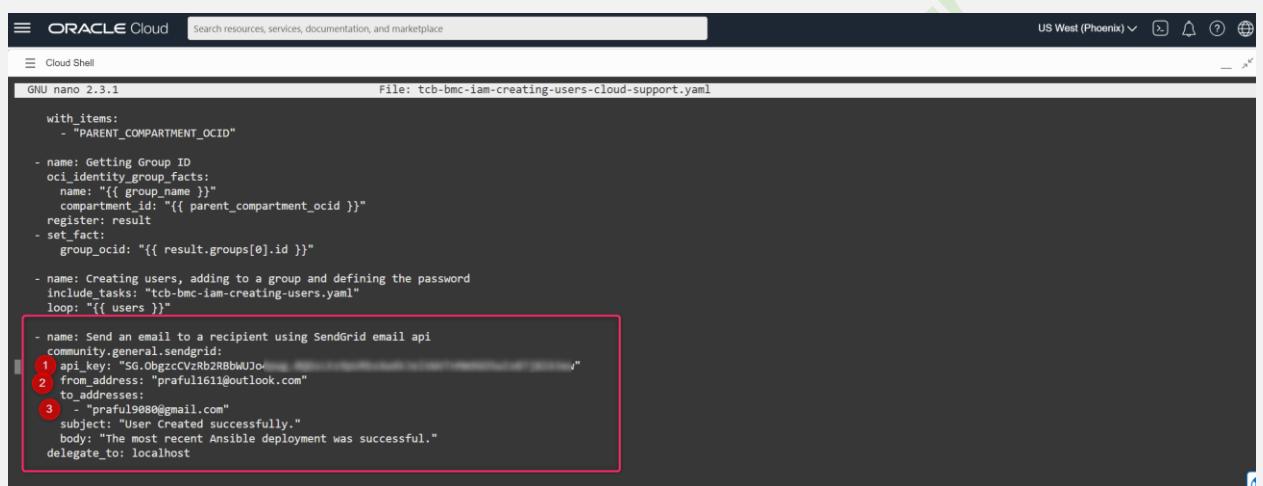
5. Add Task into playbook for sending an email using sendGrid Email API

```

- name: Send an email to a single recipient that the deployment was successful
  - sendgrid:
    community.general.sendgrid:
      api_key: "SG.ObgzcCVzRb2RBbWUJo4pwg.0QGsLKs9pURbsXwXkJ"
  
```

```

from_address: "praful1611@outlook.com"
to_addresses:
  - "praful9080@gmail.com"
subject: "OCI User Created Successfully."
body: "Please Log in using your user name in OCI."
delegate_to: localhost
  
```



```

GNU nano 2.3.1
File: tcb-bmc-iam-creating-users-cloud-support.yaml

with_items:
  - "PARENT_COMPARTMENT_OCID"

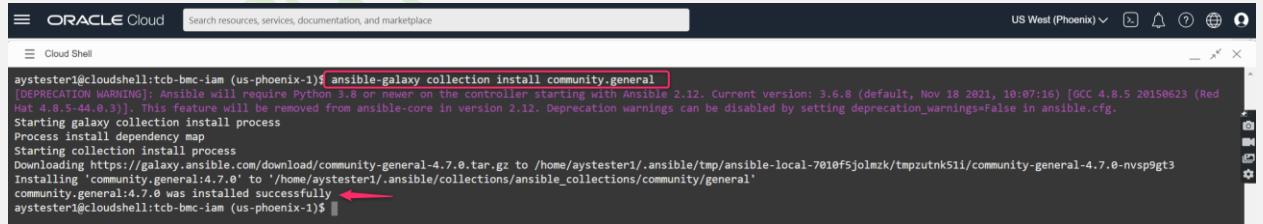
- name: Getting Group ID
  oci_identity_group_facts:
    name: "{{ group_name }}"
    compartment_id: "{{ parent_compartment_ocid }}"
  register: result
- set_fact:
    group_ocid: "{{ result.groups[0].id }}"

- name: Creating users, adding to a group and defining the password
  include_tasks: "tcb-bmc-iam-creating-users.yaml"
  loop: "{{ users }}"

- name: Send an email to a recipient using SendGrid email api
  community.general.sendgrid:
    1 api_key: "SG.ObgzccVzRbRBBwUJo"
    2 from_address: "praful1611@outlook.com"
    3 to_addresses:
      - "praful9080@gmail.com"
    subject: "User Created successfully."
    body: "The most recent Ansible deployment was successful."
    delegate_to: localhost
  
```

## 6. Install dependency module

ansible-galaxy collection install community.general.



```

aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-galaxy collection install community.general
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0)]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
Starting galaxy collection install process
Process install dependency map
Starting collection install process
Downloading collection https://galaxy.ansible.com/download/community-general-4.7.0.tar.gz to /home/aystester1/.ansible/tmp/ansible-local-7010f5jolmzk/tmpzutnk5ii/community-general-4.7.0-nvsp9gt3
Installing 'community.general:4.7.0' to '/home/aystester1/.ansible/collections/ansible_collections/community/general'
community.general:4.7.0 was installed successfully
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$
  
```

To use it in a playbook, specify: `community.general.sendgrid`.

Error

```

TASK [set_fact] *****
ok: [localhost]

TASK [Creating users, adding to a group and defining the password] *****
included: /home/aystester1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => (item=praful9088)
changed: [localhost]

TASK [Creating user] *****
changed: [localhost]

ok: [localhost]
ok: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [debug] *****
ok: [localhost] => {
    "msg": "User praful9088 created"
}

TASK [Adding user to a group] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Creating user password] *****
changed: [localhost]

ok: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [debug] *****
ok: [localhost] => {
    "msg": "Password of praful9088 is Obj0f920wAfHc"
}

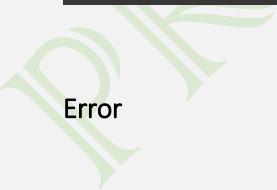
TASK [sendgrid] *****
fatal: [localhost]: FAILED! => {"changed": false, "msg": "Unsupported parameters for (sendgrid) module: community.general.sendgrid, delegate_to. Supported parameters include: html_body, username, to_addresses, bcc, cc, from_address, from_name, subject, headers, attachments, body, password, api_key."}

PLAY RECAP *****
localhost : ok=12  changed=3    unreachable=0   failed=1   skipped=1   rescued=0   ignored=0
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$

```

## Solution:

### Install sendgrid



aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)\$ pip install sendgrid  
Defaulting to user installation because normal site-packages is not writable  
Collecting sendgrid  
 Downloading sendgrid-6.9.7-py3-none-any.whl (101 kB)  
Collecting starkbank-ecdsa>=2.0.1  
 Downloading starkbank-ecdsa-2.0.3.tar.gz (12 kB)  
 Preparing metadata (setup.py)... done  
Collecting python-http-client>3.2.1  
 Downloading python\_http\_client-3.3.7-py3-none-any.whl (8.4 kB)  
Using legacy 'setup.py install' for starkbank-ecdsa, since package 'wheel' is not installed.  
Installing collected packages: starkbank-ecdsa, python-http-client, sendgrid  
 Running setup.py install for starkbank-ecdsa ... done  
Successfully installed python-http-client-3.3.7 sendgrid-6.9.7 starkbank-ecdsa-2.0.3  
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)\$

Error



ORACLE Cloud Search resources, services, documentation, and marketplace US West (Phoenix) ▾

Cloud Shell

```

TASK [set_fact] *****
ok: [localhost]

TASK [Creating users, adding to a group and defining the password] *****
included: /home/aystester1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => (item=praful9080)
changed: [localhost]

TASK [Creating user] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [debug] *****
ok: [localhost] => {
    "msg": "User praful9080 created"
}

TASK [Adding user to a group] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Creating user password] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [debug] *****
ok: [localhost] => {
    "msg": "Password of praful9080 is s38xShgPteuX"
}

TASK [Send an email to a recipient using SendGrid email api] *****
fatal: [localhost -> localhost]: FAILED! => {"changed": false, "msg": "Please install sendgrid==1.6.22 or lower since module uses Sendgrid V2 APIs."}

PLAY RECAP *****
localhost : ok=12  changed=3  unreachable=0  failed=1  skipped=1  rescued=0  ignored=0
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$

```

**Solution:**

Check sendGrid module version



ORACLE Cloud Search resources, services, documentation, and marketplace US West (Phoenix) ▾

Cloud Shell

```

TASK [Creating user password] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [debug] *****
ok: [localhost] => {
    "msg": "Password of praful9080 is s38xShgPteuX"
}

TASK [Send an email to a recipient using SendGrid email api] *****
fatal: [localhost -> localhost]: FAILED! => {"changed": false, "msg": "Please install sendgrid==1.6.22 or lower since module uses Sendgrid V2 APIs."}

PLAY RECAP *****
localhost : ok=12  changed=3  unreachable=0  failed=1  skipped=1  rescued=0  ignored=0
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible --version
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
ansible [core 2.11.10]
  config file = None
  configured module search path = ['/home/aystester1/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/local/lib/python3.6/site-packages/ansible
  ansible collection location = /home/aystester1/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/local/bin/ansible
  python version = 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]
  jinja version = 3.0.3
  libyaml = True
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ pip3 show sendgrid
Name: sendgrid
Version: 6.9.7
Summary: Twilio SendGrid library for Python
Home-page: https://github.com/sendgrid/sendgrid-python/
Author: Elmer Thomas, Yamil Asusta
Author-email: help@twilio.com
License: MIT
Location: /home/aystester1/.local/lib/python3.6/site-packages
Requires: python-http-client, starkbank-ecdsa
Required-by:
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$

```

Install required version

```
pip install sendgrid==1.6.22
```

```
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible --version
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
ansible [core 2.11.10]
  config file = None
  configured module search path = ['/home/aystester1/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/local/lib/python3.6/site-packages/ansible
  executable location = /usr/local/bin/ansible
  python version = 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]
  jinja version = 3.0.3
  libyaml = True
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ pip3 show sendgrid
Name: sendgrid
Version: 6.9.7
Summary: Twilio SendGrid library for Python
Home-page: https://github.com/sendgrid/sendgrid-python/
Author: Elmer Thomas, Yamil Asusta
Author-email: help@twilio.com
License: MIT
Location: /home/aystester1/.local/lib/python3.6/site-packages
Requires: python-http-client, starkbank-ecdsa
Required-by:
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ pip install sendgrid==1.6.22
Defaulting to user installation because normal site-packages is not writable
Collecting sendgrid==1.6.22
  Downloading sendgrid-1.6.22.tar.gz (15 kB)
    Preparing metadata (setup.py) ... done
  Collecting smpapi==0.3.1
    Downloading smpapi-0.3.1.tar.gz (3.0 kB)
      Preparing metadata (setup.py) ... done
Using legacy 'setup.py install' for sendgrid, since package 'wheel' is not installed.
Using legacy 'setup.py install' for smpapi, since package 'wheel' is not installed.
Installing collected packages: smpapi, sendgrid
  Running setup.py install for smpapi ... done
  Attempting uninstall: sendgrid
  Found existing installation: sendgrid 6.9.7
  Uninstalling sendgrid-6.9.7:
    Successfully uninstalled sendgrid-6.9.7
    Running setup.py install for sendgrid ... done
Successfully installed sendgrid-1.6.22 smpapi-0.3.1
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$
```

## Error

```
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-playbook -i hosts main.yml
PLAY [Adding user] *****
  TASK [set_fact] *****
  ok: [localhost]

  TASK [debug] *****
  ok: [localhost] => {
    "msg": "User praful9088 created"
  }

  TASK [Adding user to a group] *****
  changed: [localhost]

  TASK [set_fact] *****
  ok: [localhost]

  TASK [Creating user password] *****
  changed: [localhost]

  TASK [set_fact] *****
  ok: [localhost]

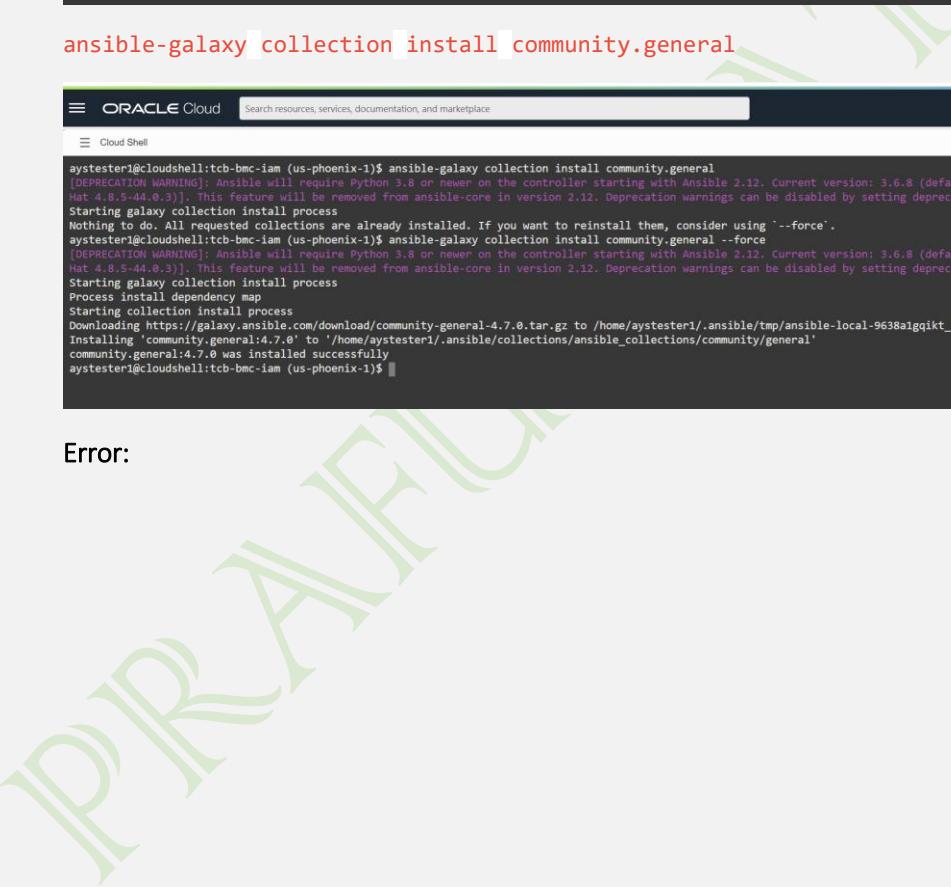
  TASK [debug] *****
  ok: [localhost] => {
    "msg": "Password of praful9088 is G8hdBCEbj7zh"
  }

  TASK [Send an email to a recipient using SendGrid email api] *****
fatal: [localhost -> localhost]: FAILED! => {"changed": false, "module_stderr": "Traceback (most recent call last):\n  File \"/home/aystester1/.ansible/tmp/ansible-tmp-1650310079.9474623-88\n  6538352332652/AnsiballZ_sendgrid.py\", line 100, in <module>\n    _ansiballz_main()\n  File \"/home/aystester1/.ansible/tmp/ansible-tmp-1650310079.9474623-8899-16538352332652/AnsiballZ_se\n  id.py\", line 92, in _ansiballz_main\n    invoke_module(zipped_mod, temp_path, ANSIBALLZ_PARAMS)\n  File \"/home/aystester1/.ansible/tmp/ansible-tmp-1650310079.9474623-8899-165383352332652/AnsiballZ_\n  sendgrid.py\", line 41, in invoke_module\n    run_name='__main__', alter_sys=True)\n  File \"\"/usr/lib64/python3.6/runpy.py\", line 205, in run_module\n    return _run_module_code(code, in\nit_globals, run_name, mod_spec)\n  File \"\"/usr/lib64/python3.6/runpy.py\", line 96, in _run_module_code\n    mod_name, mod_spec, pkg_name, script_name)\n  File \"\"/usr/lib64/python3.6/runpy.py\", line 85, in _run_code\n    exec(code, run_globals)\n  File \"\"/tmp/ansible_community.general.sendgrid_payload.eb800rf6/ansible_community.general.sendgrid_payload.zip/ansible_collections/commu\n  nity/general/plugins/modules/sendgrid.py\", line 272, in <module>\n  File \"\"/tmp/ansible_community.general.sendgrid_payload.eb800rf6/ansible_community.general.sendgrid_payload.zip/ansible_collections/commu\n  nity/general/plugins/modules/sendgrid.py\", line 266, in main\nTypeError: byte indices must be integers or slices, not str\n  module_stdout": "", "msg": "MODULE FAILURE\nSee stdout\n/stderr for the exact error", "rc": 1}

PLAY RECAP *****
localhost : ok=12  changed=3  unreachable=0  failed=1  skipped=1  rescued=0  ignored=0
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$
```

## Solution:

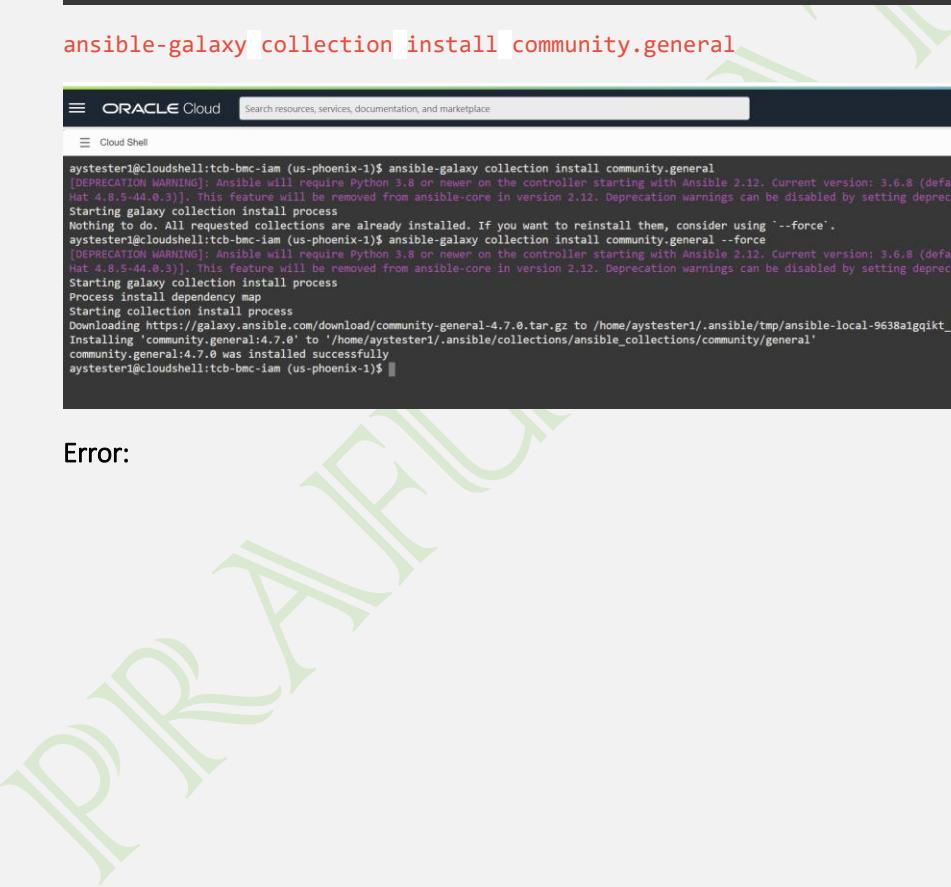
```
pip install sendgrid==2.2.1
```



Cloud Shell

```
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ pip install sendgrid==2.2.1
Defaulting to user installation because normal site-packages is not writeable
Collecting sendgrid==2.2.1
  Downloading sendgrid-2.2.1.tar.gz (8.3 kB)
    Preparing metadata (setup.py) ... done
Requirement already satisfied: smtplib==0.3.1 in /home/aystester1/.local/lib/python3.6/site-packages (from sendgrid==2.2.1) (0.3.1)
Collecting python_http_client==1.2.3
  Downloading python_http_client-1.2.3.tar.gz (3.6 kB)
    Preparing metadata (setup.py) ... done
Using legacy 'setup.py install' for sendgrid, since package 'wheel' is not installed.
Using legacy 'setup.py install' for python-http-client, since package 'wheel' is not installed.
Installing collected packages: python-http-client, sendgrid
  Attempting uninstall: python-http-client
    Found existing installation: python-http-client 3.3.7
    Uninstalling python-http-client-3.3.7...
      Successfully uninstalled python-http-client-3.3.7
    Running setup.py install for python-http-client ... done
  Attempting uninstall: sendgrid
    Found existing installation: sendgrid 1.6.22
    Uninstalling sendgrid-1.6.22...
      Successfully uninstalled sendgrid-1.6.22
    Running setup.py install for sendgrid ...
      Successfully installed python-http-client-1.2.3 sendgrid-2.2.1
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$
```

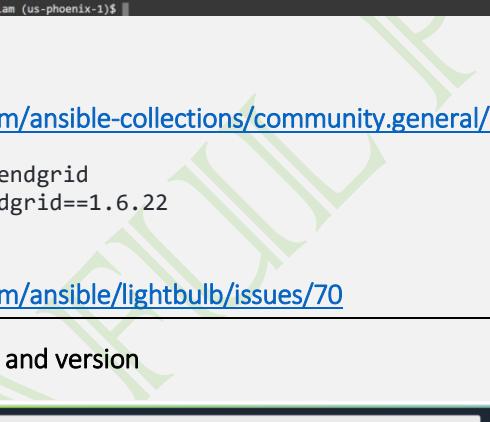
```
ansible-galaxy collection install community.general
```



Cloud Shell

```
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-galaxy collection install community.general
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]. This feature will be removed from ansible-core in version 2.12. Deprecation Warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
Starting galaxy collection install process
Nothing to do. All requested collections are already installed. If you want to reinstall them, consider using '--force'.
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-galaxy collection install community.general --force
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]. This feature will be removed from ansible-core in version 2.12. Deprecation Warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
Starting galaxy collection install process
Process install dependency map
Starting collection install process
Downloading https://galaxy.ansible.com/download/community-general-4.7.0.tar.gz to /home/aystester1/.ansible/tmp/ansible-local-9638aigqikt_/_tmpgpyboxy_z/community-general-4.7.0-wb8r_asy
Installing 'community.general:4.7.0' to '/home/aystester1/.ansible/collections/ansible_collections/community/general'
community.general:4.7.0 was installed successfully
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$
```

Error:



ORACLE Cloud Search resources, services, documentation, and marketplace US West (Phoenix) ▾

Cloud Shell

```

TASK [set_fact] *****
ok [localhost]
TASK [Creating users, adding to a group and defining the password] *****
included: /home/aystester1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => (item=praful988)
TASK [Creating user] *****
changed: [localhost]
TASK [set_fact] *****
ok [localhost]
TASK [debug] *****
ok [localhost] => {
    "msg": "User praful988 created"
}
TASK [Adding user to a group] *****
changed: [localhost]
TASK [set_fact] *****
ok [localhost]
TASK [Creating user password] *****
changed: [localhost]
TASK [set_fact] *****
ok [localhost]
TASK [debug] *****
ok [localhost] => {
    "msg": "Password of praful988 is jELBfEK0J7Wu"
}
TASK [Send an email to a recipient using SendGrid email api] *****
fatal: [localhost -> localhost]: FAILED! => { "changed": false, "msg": "Please install sendgrid==1.6.22 or lower since module uses Sendgrid V2 APIs." }
PLAY RECAP *****
localhost : ok=12   changed=3   unreachable=0   failed=1   skipped=1   rescued=0   ignored=0
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$

```

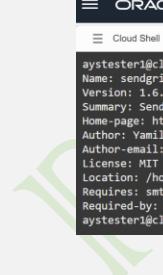
### Solution:

<https://github.com/ansible-collections/community.general/issues/1553>

```
pip uninstall sendgrid
pip install sendgrid==1.6.22
```

<https://github.com/ansible/lightbulb/issues/70>

### Working Solution and version



ORACLE Cloud Search resources, services, documentation, and marketplace US West (Phoenix) ▾

Cloud Shell

```

aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ pip3 show sendgrid
Name: sendgrid
Version: 1.6.22
Summary: SendGrid library for Python
Home-page: https://github.com/sendgrid/sendgrid-python/
Author: Yamill Asusta
Author-email: yamill@sendgrid.com
License: MIT
Location: /home/aystester1/.local/lib/python3.6/site-packages
Requires: smtplib
Required-by:
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$

```

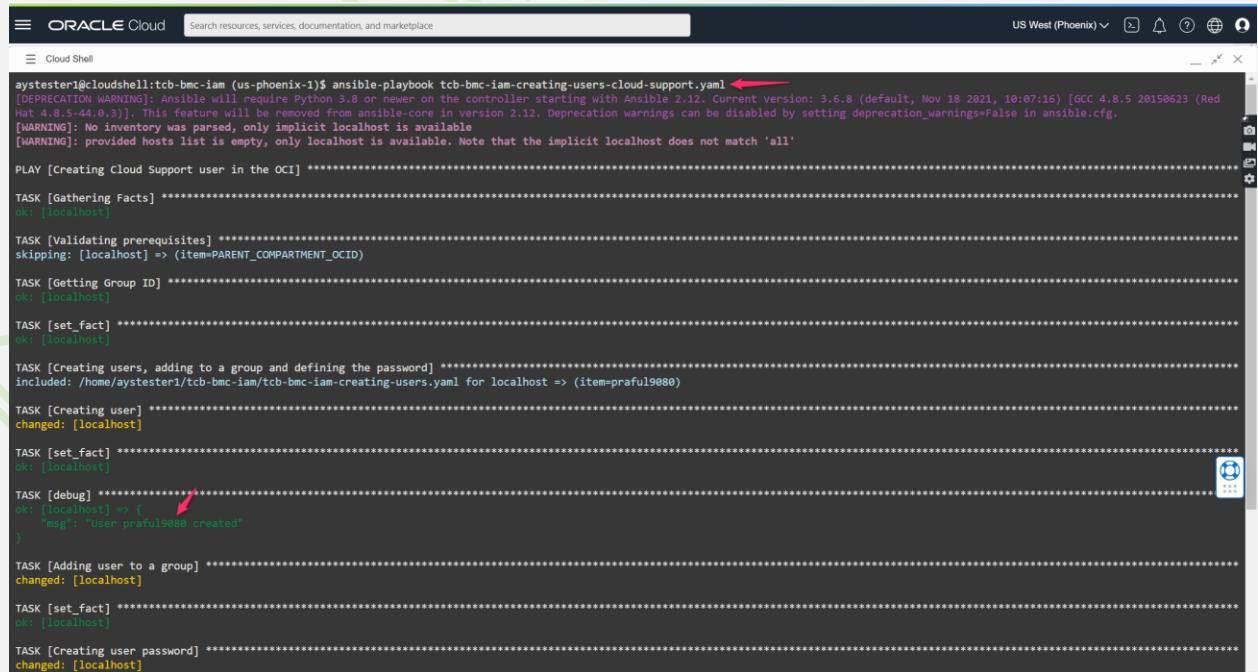
### Static Solution 1:

1. Create a new user creation yaml file: tcb-bmc-iam-creating-users-cloud-support.yaml
2. Update the code with below sendGrid Email.

```
- name: Send an email to a single recipient that the deployment was
  successful
  - sendgrid:
    community.general.sendgrid:
      api_key: "SG.0bgzcCVzRb2RBbWUJo4pwg.0QGsLKs9pURbsXwXkJ"
      from_address: "praful1611@outlook.com"
      to_addresses:
        - "praful19080@gmail.com"
      subject: "OCI User Created Successfully."
      body: "Please Log in using your user name in OCI."
      delegate_to: localhost
```

### 3.Run Playbook

Ansible-playbook tcb-bmc-iam-creating-users-cloud-support.yaml



```
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-playbook tcb-bmc-iam-creating-users-cloud-support.yaml
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 3.6.8 (default, Nov 18 2021, 10:07:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44.0.3)]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
PLAY [Creating Cloud Support user in the OCI] ****
TASK [Gathering Facts] ****
ok: [localhost]
TASK [Validating prerequisites] ****
skipping: [localhost] => {item=PARENT_COMPARTMENT_OCID}
TASK [Getting Group ID] ****
ok: [localhost]
TASK [set_fact] ****
ok: [localhost]
TASK [creating users, adding to a group and defining the password] ****
included: /home/aystester1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => {item=praful19080}
TASK [creating user] ****
changed: [localhost]
TASK [set_fact] ****
ok: [localhost]
TASK [debug] ****
ok: [localhost] => {
  "msg": "User praful19080 created"
}
TASK [Adding user to a group] ****
changed: [localhost]
TASK [set_fact] ****
ok: [localhost]
TASK [creating user password] ****
changed: [localhost]
```

### 4. Sending email successful

OCI BOOTCAMP PROJECT

OCI PROJECT 1: DEPLOYING USERS, GROUPS AND POLICIES

IN A 100% AUTOMATED WAY USING ANSIBLE AND CLOUD SHELL

EMAIL AUTOMATION CHALLENGE USING ANSIBLE & SENDGRID EMAIL API

IMPLEMENTED BY: PRAFUL PATEL

ORACLE Cloud Search resources, services, documentation, and marketplace US West (Phoenix) ▾

Cloud Shell

```
TASK [set_fact] ****
ok: [localhost]

TASK [Creating users, adding to a group and defining the password] ****
included: /home/aysteller1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => (item=praful9088)

TASK [Creating user] ****
changed: [localhost]

TASK [set_fact] ****
ok: [localhost]

TASK [debug] ****
ok: [localhost] => {
    "msg": "User praful9088 created"
}

TASK [Adding user to a group] ****
changed: [localhost]

TASK [set_fact] ****
ok: [localhost]

TASK [Creating user password] ****
changed: [localhost]

TASK [set_fact] ****
ok: [localhost]

TASK [debug] ****
ok: [localhost] => {
    "msg": "Password of praful9088 is 8AT1lpSMcdr6"
}

[TASK [Send an email to a recipient using SendGrid email api] ****
ok: [localhost -> localhost]

PLAY RECAP ****
localhost : ok=13   changed=3   unreachable=0   failed=0   skipped=1   rescued=0   ignored=0

aysteller1@cloudshell:tcb-bmc-iam (us-phoenix-1)$
```

5. Let's verify if user email has been sent successful.

The screenshot shows the Oracle Cloud Identity domain users page. The left sidebar has 'Identity domain' selected under 'Users'. The main area title is 'Users in Default Domain'. A search bar at the top says 'Search resources, services, documentation, and marketplace'. Below it, a button says 'Create user' and a dropdown says 'More actions'. A table lists users: praful9080 (selected), jonne.delUcca, and petter.oliver. Each row includes checkboxes for selection, columns for 'Username', 'Status', 'Email', 'Last access', and 'Created'. The 'praful9080' row is highlighted with a red box.

	Username	Status	Email	Last access	Created
<input checked="" type="checkbox"/>	praful9080	Active	praful9080@gmail.com	-	Mon, Apr 18, 2022, 23:21:54 UTC
<input type="checkbox"/>	jonne.delUcca	Active	jonne.delUcca@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:39:35 UTC
<input type="checkbox"/>	petter.oliver	Active	petter.oliver@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:39:19 UTC

#### **6.Go to Gmail and verify the inbox**

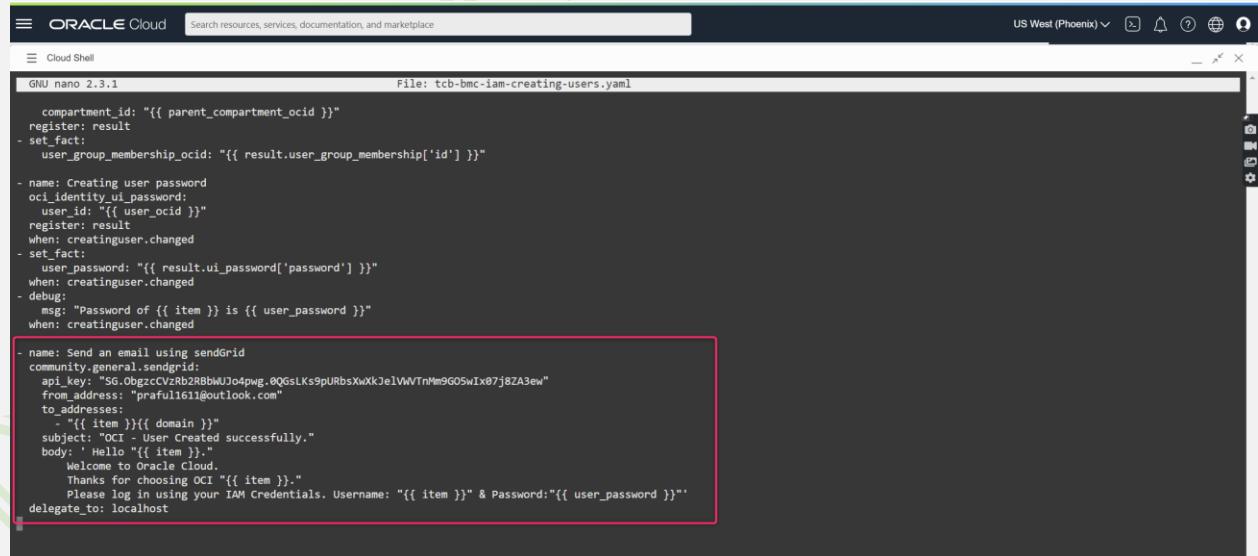
A screenshot of a Gmail inbox. The top navigation bar includes the Gmail logo, a search bar labeled "Search mail", and various account settings icons. Below the bar, a "Compose" button is on the left, followed by a series of icons for reply, forward, delete, etc. The main area shows an "Inbox" folder with 13,061 messages. A red arrow points to a specific email from "praful1611@outlook.com via sendgrid.net" with the subject "OCI - User Created successfully.". The message body says "Thanks for choosing OCI - Please log in using your user name." To the right of the message are standard email controls like print and reply. The bottom of the screen shows other inbox categories: Starred, Snoozed, Sent, and Drafts.

Email has been received successfully using automation ansible-sendGrid Email API

## Dynamic Solution 2:

1. Go to file: tcb-bmc-iam-creating-users.yaml
2. Update the code with below sendGrid Email

```
- name: Send an email using sendGrid
  community.general.sendgrid:
    api_key: "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"
    from_address: "praful1611@outlook.com"
    to_addresses:
      - "{{ item }}{{ domain }}"
    subject: "OCI - User Created successfully."
    body: ' Hello "{{ item }}."'
      Welcome to Oracle Cloud.
      Thanks for choosing OCI "{{ item }}."
      Please log in using your IAM Credentials. Username: "{{ item }}" &
Password:"{{ user_password }}"
    delegate_to: localhost
```



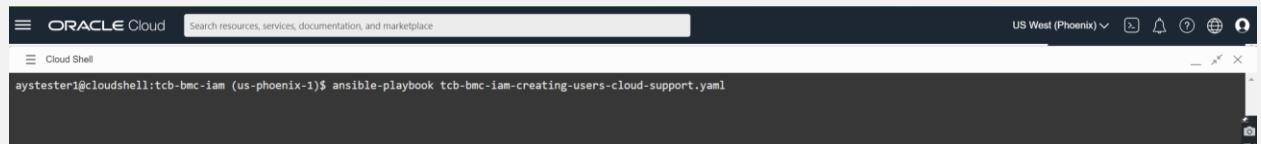
```
File: tcb-bmc-iam-creating-users.yaml
compartment_id: "{{ parent_compartment_ocid }}"
register: result
- set_fact:
  user_group_membership_ocid: "{{ result.user_group_membership['id'] }}"

- name: Creating user password
  oci_identity_ui_password:
    user_id: "{{ user_ocid }}"
  register: result
  when: creatinguser.changed
- set_fact:
  user_password: "{{ result.ui_password['password'] }}"
  when: creatinguser.changed
- debug:
  msg: "Password of {{ item }} is {{ user_password }}"
  when: creatinguser.changed

- name: Send an email using sendGrid
  community.general.sendgrid:
    api_key: "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"
    from_address: "praful1611@outlook.com"
    to_addresses:
      - "{{ item }}{{ domain }}"
    subject: "OCI - User Created successfully."
    body: ' Hello "{{ item }}."
      Welcome to Oracle Cloud.
      Thanks for choosing OCI "{{ item }}."
      Please log in using your IAM Credentials. Username: "{{ item }}" & Password:"{{ user_password }}"
    delegate_to: localhost
```

### 3. Run Playbook

- Ansible-playbook < tcb-bmc-iam-creating-users-cloud-support.yaml >



```
aystester1@cloudshell:tcb-bmc-iam (us-phoenix-1)$ ansible-playbook tcb-bmc-iam-creating-users-cloud-support.yaml
```

#### 4. Playbook executed successfully.

```

TASK [set_fact] *****
ok: [localhost]

TASK [Creating users, adding to a group and defining the password] *****
included: /home/aysteller1/tcb-bmc-iam/tcb-bmc-iam-creating-users.yaml for localhost => (item=praful9080)

TASK [Creating user] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [debug] *****
ok: [localhost] => {
    "msg": "User praful9080 created"
}

TASK [Adding user to a group] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [Creating user password] *****
changed: [localhost]

TASK [set_fact] *****
ok: [localhost]

TASK [debug] *****
ok: [localhost] => {
    "msg": "Password of praful9080 is kIgv0Ezq5h4y"
}

TASK [Send an email using sendGrid] *****
ok: [localhost -> localhost]

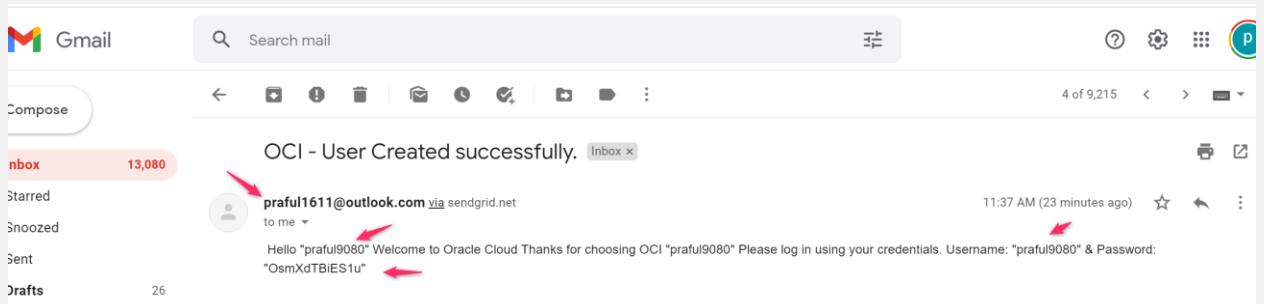
PLAY RECAP *****
localhost          : ok=13   changed=3   unreachable=0   failed=0    skipped=1   rescued=0   ignored=0
aysteller1@cloudshell:tcb-bmc-iam (us-phoenix-1)$

```

#### 5. Verify from console that user is created

	Username	Status	Email	Last access	Created
<input type="checkbox"/>	praful9080	● Active	praful9080@gmail.com	-	Tue, Apr 19, 2022, 17:58:06 UTC
<input type="checkbox"/>	jonne.delucca	● Active	jonne.delucca@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:39:35 UTC
<input type="checkbox"/>	petter.oliver	● Active	petter.oliver@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:39:19 UTC
<input type="checkbox"/>	juan.francisco	● Active	juan.francisco@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:34:21 UTC
<input type="checkbox"/>	donna.mitch	● Active	donna.mitch@thecloudbootcamp.com	-	Mon, Apr 18, 2022, 00:34:02 UTC

#### 6. Verify that email has been received successfully.



Congratulations!!!! 🎉