Contents

[OCI Configuration 1](#_Toc201352343)

[Screenshots 4](#_Toc201352344)

[Lessons learned from my installation process: 7](#_Toc201352345)

# OCI Configuration

To configure the **OCI configuration file** (~/.oci/config), you'll need to gather some specific information from your **Oracle Cloud Infrastructure (OCI)** account. Here's a step-by-step guide to help you get this information:

**Step 1: Sign in to Oracle Cloud**

Go to the [**Oracle Cloud Console**](https://cloud.oracle.com/) and log in with your credentials.

**Step 2: Find Your OCID Information**

You will need several pieces of information to fill out your config file.

The steps below may vary depending on whether you are using Classic or Redwood in the OCI console. Look at the screenshots section for a visual aid in the Redwood UI.

**a) User OCID**

1. In the Oracle Cloud Console, click on your **profile icon** (usually at the top right).
2. Click **"My Profile"**.
3. Look for the **User OCID** under your profile details. It should look something like this:
4. ocid1.user.oc1..xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

**b) Tenancy OCID**

1. In the Oracle Cloud Console, go to **"Identity & Security"** > **"Tenancy"**.
2. You will find the **Tenancy OCID** under the **General Information** section. It will look like this:
3. ocid1.tenancy.oc1..xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

**c) Compartment OCID**

1. In the Oracle Cloud Console, navigate to **"Identity & Security"** > **"Compartments"**.
2. Find the compartment you are working with (where the AI service and other resources are located). Click on it, and you will see the **Compartment OCID**, which looks like:
3. ocid1.compartment.oc1..xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

**d) API Key and Private Key**

For authentication, you'll need to create an **API key** (a private key) and associate it with your Oracle user account.

**How to Generate a Private Key:**

1. Go to **"Identity & Security"** > **"Users"**.
2. Click on your **user name**.
3. In the **"API Keys"** section, click **"Add API Key"**.
4. Choose to either upload an existing public key or generate a new one. If you generate a new one, Oracle will provide you with the **public key** and you'll need to store the **private key** securely on your local machine.

Make sure to save the **private key** in a file, such as ~/.oci/oci\_api\_key.pem.

**Step 3: Populate the Config File**

Once you have gathered all the information, you can now populate the OCI config file (~/.oci/config). Here's a template you can follow:

[DEFAULT]

user=ocid1.user.oc1..xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

fingerprint=xx:xx:xx:xx:xx:xx:xx:xx

key\_file=/path/to/your/private-key.pem

tenancy=ocid1.tenancy.oc1..xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

region=us-phoenix-1

* **user**: Your **User OCID**.
* **fingerprint**: This is the **fingerprint** of the public key you uploaded when creating the API key. You can find it by running this command (assuming you have OpenSSL installed and have the public key file):
* openssl rsa -in /path/to/public-key.pem -pubin -noout -fingerprint
* **key\_file**: The full path to the **private key** file you saved earlier (e.g., ~/.oci/oci\_api\_key.pem).
* **tenancy**: Your **Tenancy OCID**.
* **region**: The **OCI region** where your resources are hosted.

**Step 4: Verify the Config File**

Once your config file is populated, you can verify it's set up correctly by running a simple OCI CLI command to check the connection:

oci os ns get

If the config is correct, this should return the Object Storage namespace for your tenancy.

**Final Thoughts**

The OCI configuration file is crucial for authenticating and using the OCI SDK. Once it’s properly set up, the Python script you have should be able to authenticate and make requests to Oracle's Generative AI service, read the other documentation to get more information on this.

**Note**:

* Alternatively, after the “add” step when creating an API key for your user, it gives you the entire string except compartment ID, and you can copy that configuration.
* To install the OIC CLI go to the next link: <https://docs.oracle.com/en-us/iaas/Content/API/SDKDocs/cliinstall.htm>
* You will need Python on your machine, if you are using windows, the CLI installation may fail if you have not installed Python from the official site and added it to the env path.
* Once you install it, run this command to test connectivity: oci iam compartment list –all as an alternate to the “oci os ns get” command
* OCI CLI may ask you to create a config file if it can’t find one, and it will ask you for the details we gathered above at that time.

## Screenshots

**Screenshots of several of the steps described above:**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

# Lessons learned from my installation process:

**1. Ensure Python Dependencies Are Installed:**

* **distutils** is a crucial package required by the OCI CLI. If missing, it can cause errors during installation.
* **setuptools** is another key package that helps in managing Python packages, and its absence can also lead to issues.

**2. Correct Python Version:**

* **OCI CLI** relies on Python 3. Ensure you're using a compatible version of Python (e.g., Python 3.7+). You can verify this using:
* python --version

**3. Install Missing Dependencies:**

* When you encounter errors about missing modules (distutils or setuptools), you can install them manually using:
* python -m pip install setuptools
* python -m pip install distutils
* If **distutils** is missing, it might require installation through a legacy version (distutils-legacy) or by ensuring it's bundled with your Python version.

**4. Virtual Environments:**

* It's best practice to install the **OCI CLI** inside a **virtual environment (venv)**. This ensures isolation and prevents conflicts with other Python packages installed globally.
* To set up a virtual environment:
* python -m venv oci-venv
* .\oci-venv\Scripts\activate # For Windows

**5. Ensure pip is Up-to-Date:**

* The **pip** package manager must be up-to-date to avoid issues while installing dependencies.
* Upgrade it using:
* python -m pip install --upgrade pip

**6. Handling Private Key for Authentication:**

* When setting up authentication, ensure the **path to your private key** is correct and enclosed in quotes if there are spaces in the directory name.
* The CLI might prompt you for the private key file during initial configuration, and you must ensure it’s accessible.

**7. OCI Configuration File:**

* After installation, use the **OCI CLI setup** command (oci setup config) to configure the CLI with necessary information such as your **tenancy OCID**, **user OCID**, and **private key path**.
* If you encounter errors about missing configuration details, it may be due to incorrectly provided OCID values or paths.

**8. Dealing with Errors:**

* If you face errors related to **missing dependencies** or other issues, refer to the error message details for guidance. Many issues can be resolved by installing missing packages or fixing file paths.
* Always check the logs or error message stack traces to understand which component failed.

**9. Version Compatibility:**

* Ensure that the **OCI CLI version** you are installing is compatible with your Python version and any other libraries you may have installed.

**10. Environment Variables:**

* **Environment Variables** (such as OCI\_CONFIG or OCI\_PRIVATE\_KEY\_PATH) may need to be set in some cases if the CLI doesn’t pick up the configuration file automatically.
* This is important for setting the path to your **configuration file** or **private key**.

**11. Common OCI CLI Commands:**

* After installation, ensure you can run basic commands like oci --version and oci iam compartment list --all to confirm the CLI is set up correctly and able to interact with OCI.

**12. CLI Documentation and Resources:**

* It’s helpful to consult the official **OCI CLI documentation** to understand common errors, installation guidelines, and troubleshooting tips.
* Link to documentation: <https://docs.oracle.com/en-us/iaas/Content/API/SDKDocs/cliinstall.htm>