

# Hands On Lab 0: Configuring ATP in OCI and First Data Import



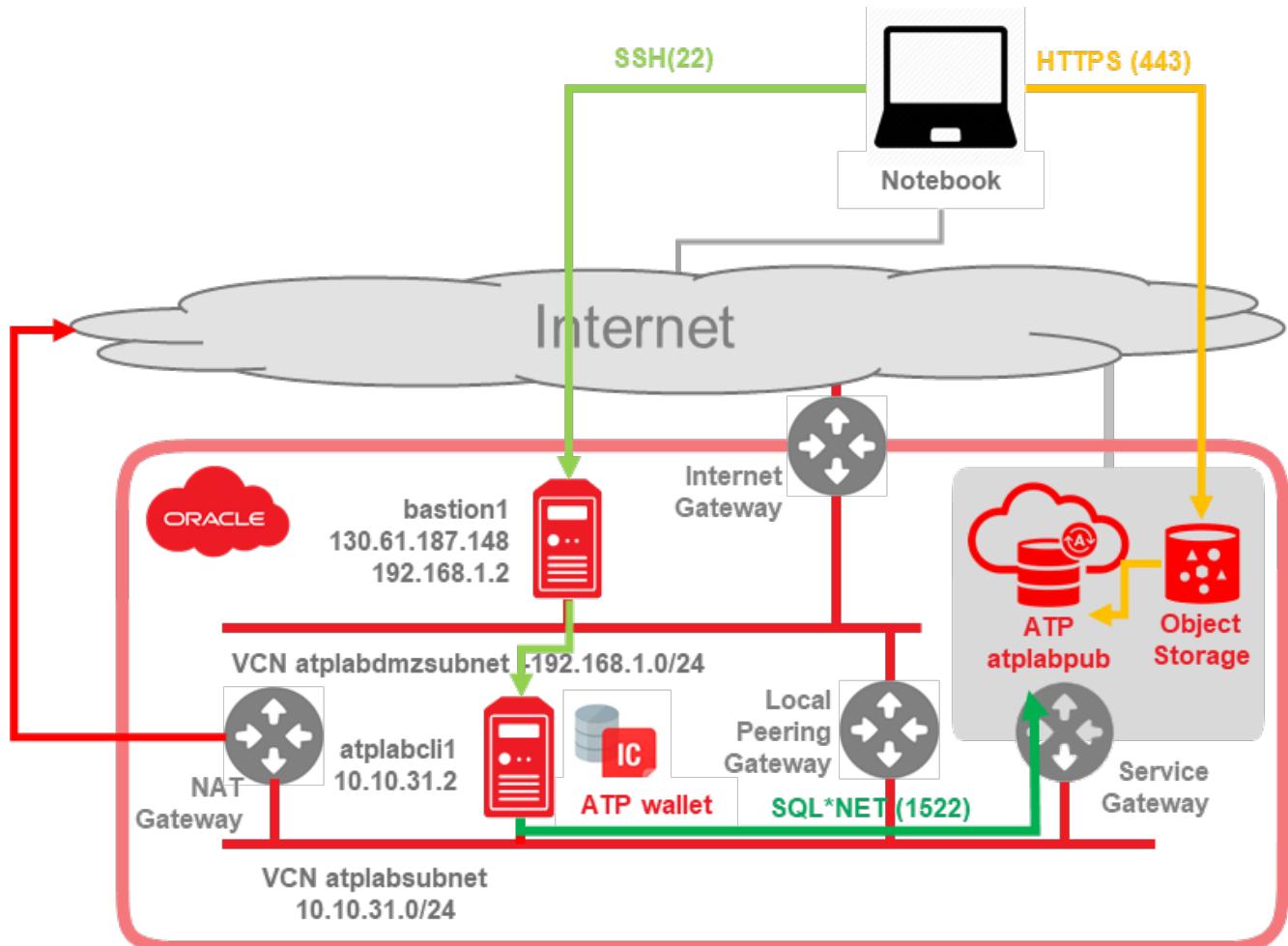
# Index

<b>HANDS ON LAB 0: CONFIGURING ATP IN OCI AND FIRST DATA IMPORT .....</b>	<b>1</b>
<b>LABORATORY OBJECTIVE .....</b>	<b>3</b>
<b>REVIEW OF THE FIRST RESOURCES PROVIDED.....</b>	<b>4</b>
<b>CREATING THE AUTONOMOUS TRANSACTION PROCESSING DATABASE.....</b>	<b>10</b>
<b>CREATING OBJECT STORAGE.....</b>	<b>15</b>
CREATE PRE-AUTHENTICATED REQUEST.....	18
<b>CONTINUING WITH ATP, SERVICES REVIEW .....</b>	<b>21</b>
HOW TO GENERATE AND DOWNLOAD THE CREDENTIAL FILE FROM THE ATP ADMINISTRATION CONSOLE .....	24
<b>CONFIGURATION OF ATP ACCESS MACHINES (BASTION AND CLIENT).....</b>	<b>25</b>
<b>DOWNLOAD THE CLIENT SOFTWARE TO ACCESS THE ATP .....</b>	<b>33</b>
<b>CHECK THE CONNECTION WITH THE AUTONOMOUS DATABASE (ATP).....</b>	<b>46</b>
<b>LAB SUMMARY.....</b>	<b>48</b>



# Laboratory Objective

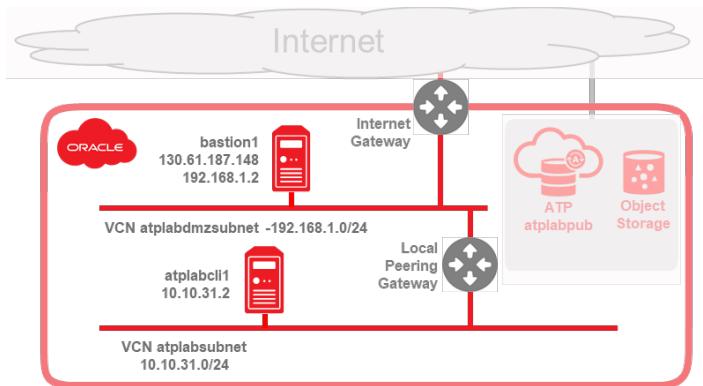
The objective of this laboratory is to verify that the necessary prerequisites are correctly provisioned, after which we will proceed to create our first Autonomous Transaction Processing Database (ATP), and then all the necessary resources to import data into this database for the first time.



# Review of the first resources provided

## What am I going to do?

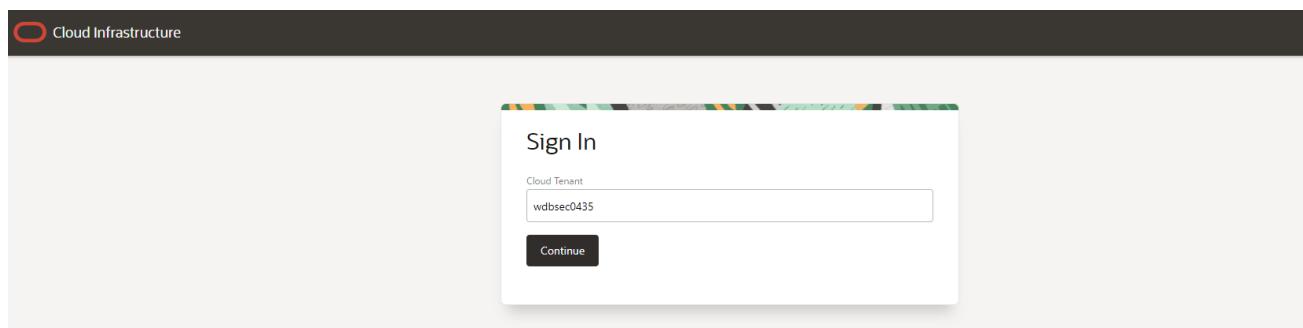
You will review that all the resources needed for the laboratory are properly provisioned.

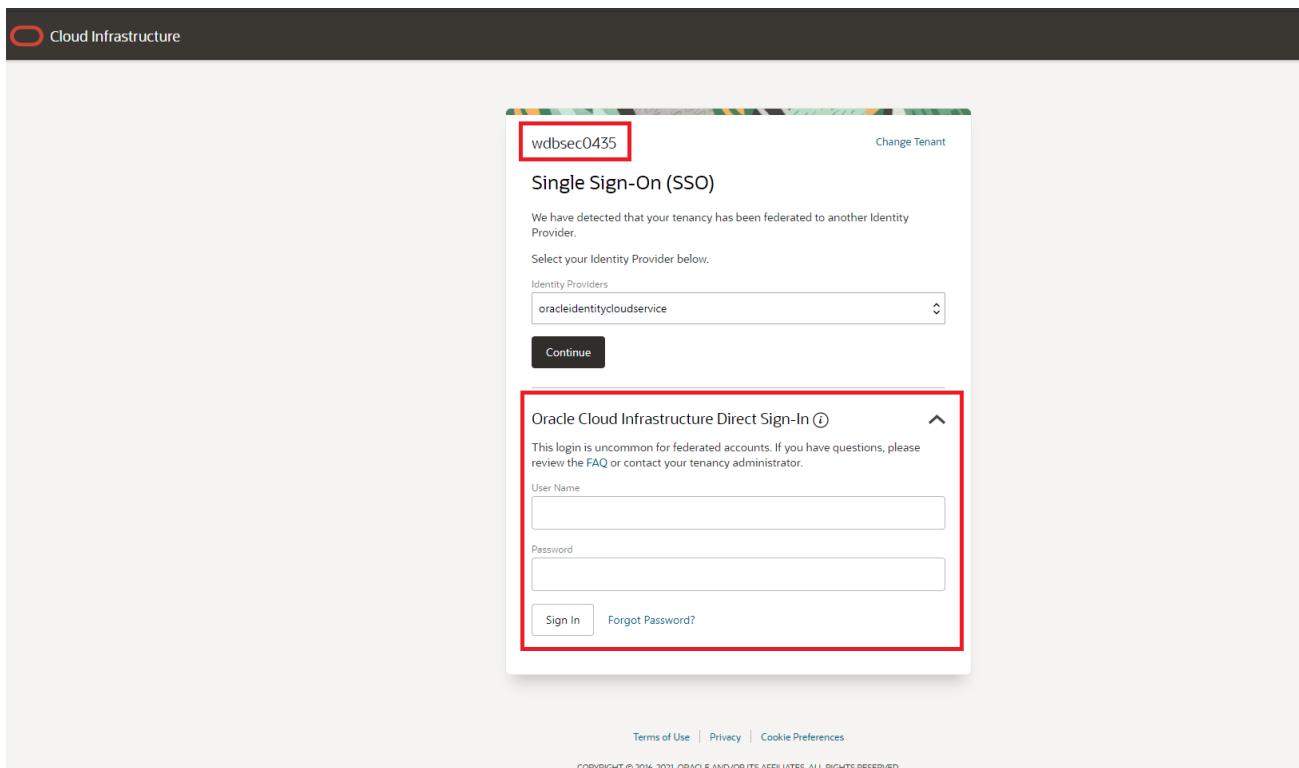


Data for the connection in this section:

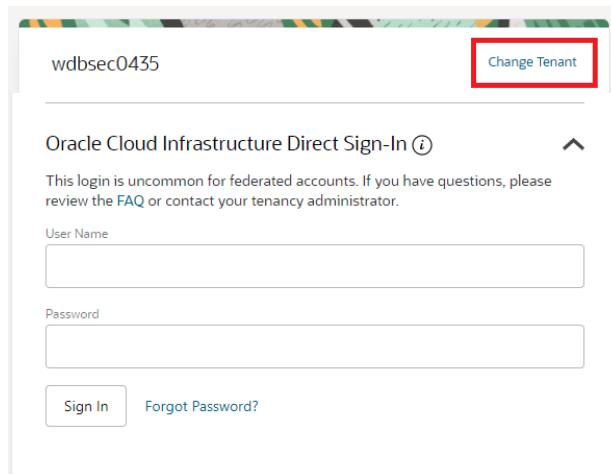
<b>OCI Console URL</b>	<a href="https://console.eu-frankfurt-1.oraclecloud.com/">https://console.eu-frankfurt-1.oraclecloud.com/</a>
<b>Tenant</b>	Use the credentials you have been assigned
<b>User name</b>	Use the credentials you have been assigned
<b>Password</b>	Use the credentials you have been assigned

In order to access the OCI management console, you must access it from the URL indicated above, make sure that your **cloud tenant** is the one marked as Tenant in the table above and click on the **Continue** icon. Deploy the **Oracle Cloud Infrastructure Direct Sign-In** option, enter your login and password received for this lab and click on the **Sign In** button.



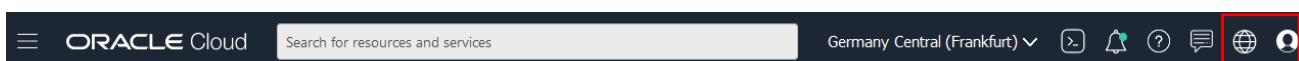


If your tenant is not the one marked in the table or you need to change tenants, click the **Change tenant** link to the right of the current tenant name and enter your correct tenant name.

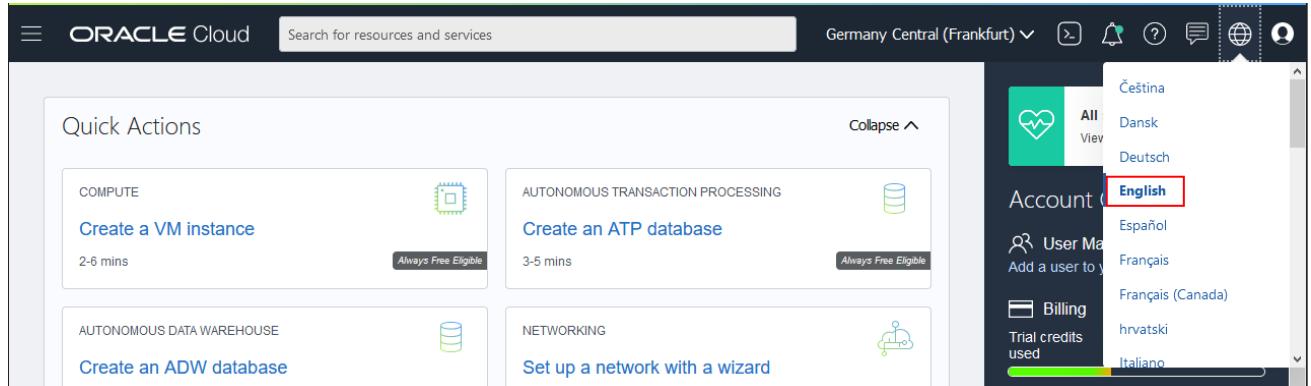


Then enter the assigned user and password to access the OCI Admin console.

All the manuals of the different laboratories have the screenshots in English, so we recommend selecting English language in the OCI management console to facilitate the use of the screenshots. To select the English language, if it is not already selected, click the world ball icon right next to the profile icon in the upper-right corner of the console.

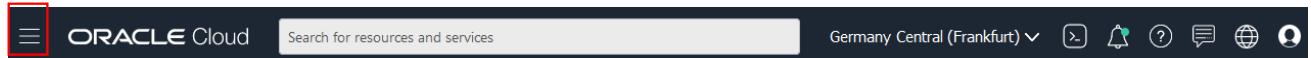


Then select the English language to change the language of the OCI Management Console.



Now you can continue with the laboratory, in which the first thing you will do is verify that the prerequisites are correctly created to be able to perform the rest of the workshop.

Once in the main OCI console, you can open the menu at the top left of the screen marked with the hamburger icon or three parallel horizontal lines.

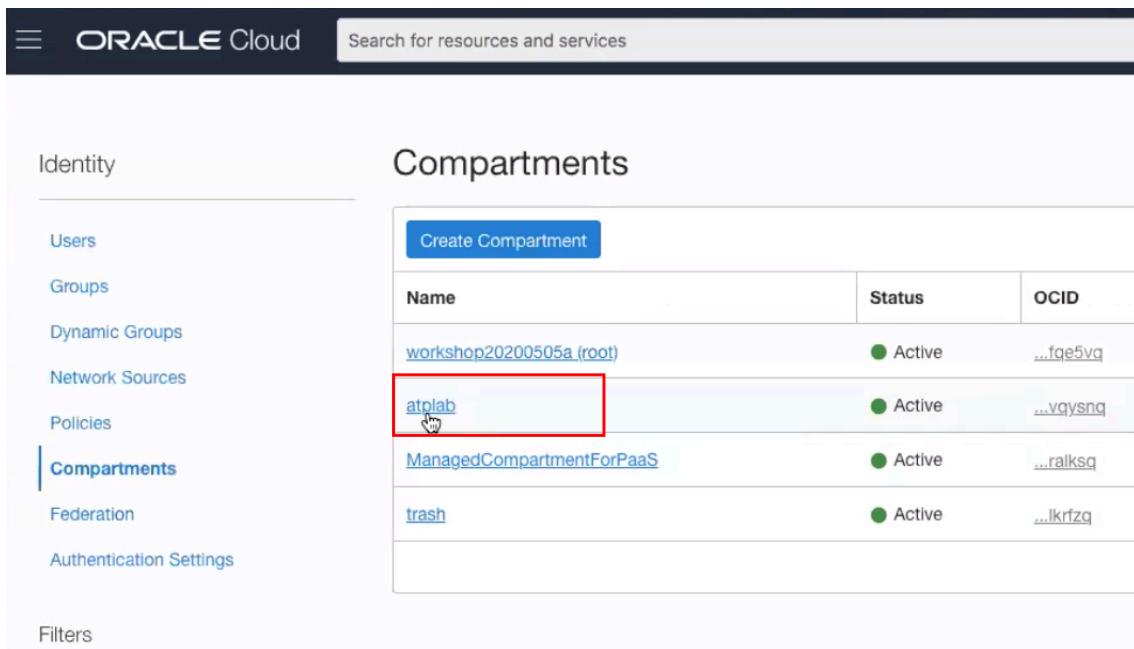


Select **Identity & Security** and within the *Identity* section select **Compartments**



Once in the compartments section, we can identify the compartment that we will use for this workshop, called **atplab**

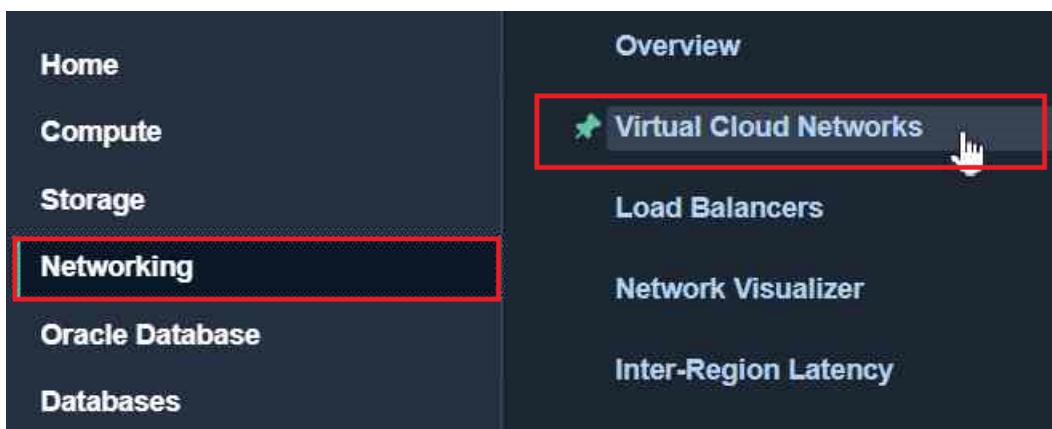




The screenshot shows the Oracle Cloud Identity interface. On the left, there's a sidebar with a hamburger menu icon, followed by the "ORACLE Cloud" logo and a search bar. Below the search bar is a list of navigation items: "Identity", "Users", "Groups", "Dynamic Groups", "Network Sources", "Policies", "Compartments" (which is highlighted in blue), "Federation", and "Authentication Settings". Under "Compartments", there's a "Filters" section. The main content area is titled "Compartments" and contains a table. At the top of the table is a blue button labeled "Create Compartment". The table has columns for "Name", "Status", and "OCID". There are five rows of data:

Name	Status	OCID
workshop20200505a (root)	Active	...fqe5vg
atplab	Active	...vgysnq
ManagedCompartmentForPaaS	Active	...ralksq
trash	Active	...lkrfzq

Then open the OCI main menu again (hamburger icon), and select **Networking -> Virtual Cloud Networks**.



The screenshot shows the Oracle Cloud Infrastructure (OCI) main menu. On the left is a sidebar with the following items: "Home", "Compute", "Storage", "Networking" (which is highlighted with a red box), "Oracle Database", and "Databases". To the right is the "Overview" section, which includes links for "Virtual Cloud Networks" (also highlighted with a red box and a cursor icon pointing at it), "Load Balancers", "Network Visualizer", and "Inter-Region Latency".

***IMPORTANT NOTE: Always make sure that you are using the correct compartment [atplab] on all OCI consoles.***



☰ ORACLE Cloud Search for resources and services Germany Central (Frankfurt) ⓘ 🔔 ? 📣 🌐

## Networking

### Virtual Cloud Networks

Overview

**Virtual Cloud Networks**

- Dynamic Routing Gateways
- Customer-Premises Equipment
- IPSec Connections
- Load Balancers
- FastConnect
- Public IPs
- DNS Zone Management
- TSIG Keys
- Traffic Management Steering Policies
- HTTP Redirects

List Scope

COMPARTMENT

Search compartments

- workshop20200505a (root)
  - atplab
  - ManagedCompartmentForPaaS
  - trash

Pick a Compartment

Oracle Cloud Infrastructure uses compartments to organize your resources.

View and manage your resources: pick a compartment and resource type using the filter on the left

[Learn more about compartments](#)

Verify that two networks are created:

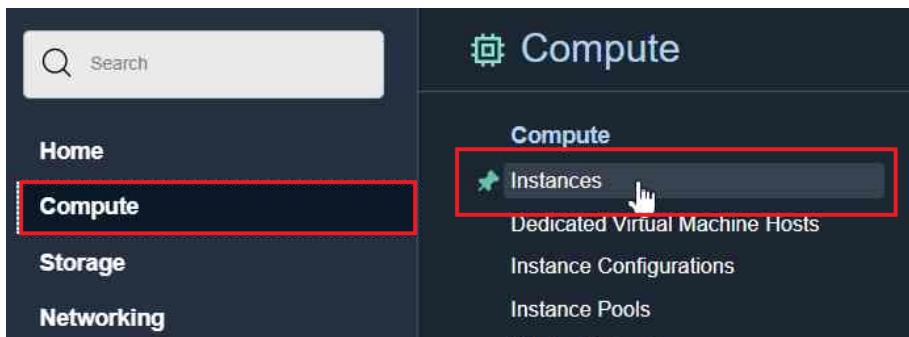
- One is the network for **named DMZ [atplabdmznet]**
  - The other **[atplabnet]**, is a private network that we will use to connect all the servers that will have access to the ATP Database. This network is the one we will also use to connect with Equinix and the other cloud providers.

Virtual Cloud Networks *in* atplab *Compartments*

Name	State	CIDR Block	Default Route Table	DNS Domain Name	Created
<a href="#">atplabdmnet</a>	<span>● Available</span>	192.168.1.0/24	<a href="#">Default Route Table for atplabdmnet</a>	atplabdmnet.oraclevcn.com	Wed, Apr 22, 2020, 01:45:00
<a href="#">atplabnet</a>	<span>● Available</span>	10.10.31.0/24	<a href="#">Default Route Table for atplabnet</a>	atplabnet.oraclevcn.com	Wed, Apr 22, 2020, 01:45:00

Then select from the OCI main menu (hamburger icon) **Compute -> Instances**.





Check that there are two servers created and booted (**Running** state, if they are not in running state, please boot the machines by clicking on the three-dot icon to the right of each instance and selecting **Start**):

- **[bastion1]**, which will be the entry point with the outside (internet or other network external to OCI). This server will have a public IP
- **[atplabcli1]**, which is located in the private network mentioned above (**atplab**), and which will contain the necessary software client, to be able to connect with the ATP (Autonomous Transaction Processing Database).

#### Instances *in atplab Compartiment*

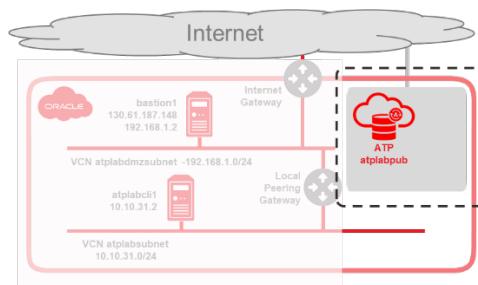
Create Instance						
Name	Status	Public IP	Shape	Availability Domain	Fault Domain	Created
<a href="#">bastion1</a>	● Running	130.61.225.5	VM.Standard2.1	AD-1	FD-3	Wed, Apr 22, 2020, 07:03:02 UTC
<a href="#">atplabcli1</a>	● Running	-	VM.Standard2.1	AD-1	FD-3	Wed, Apr 22, 2020, 07:03:02 UTC
Showing 2 Items						<a href"="">&lt;</a> <a href"="">Page 1</a> <a href"="">&gt;</a>



# Creating the Autonomous Transaction Processing Database

## What am I going to do?

You will create the autonomous database (ATP) within Oracle Cloud Infrastructure.



Select from the OCI main menu (hamburger icon) **Oracle Database -> , Autonomous Transaction Processing (ATP)..**

The screenshot shows the OCI navigation bar with the following items: Home, Compute, Storage, Networking, Oracle Database (highlighted with a red box), and Databases. On the right, there's an 'Overview' section with links to Autonomous Database, Autonomous Data Warehouse, Autonomous JSON Database, and Autonomous Transaction Processing (which is also highlighted with a red box and has a cursor icon pointing to it).

Click the **Create Autonomous Database** button to create a new Autonomous Database

The screenshot shows the 'Autonomous Database' creation interface. On the left, there are tabs for Autonomous Database (selected), Dedicated Infrastructure (with a help icon), Autonomous Container Database, and Autonomous Exadata Infrastructure. On the right, it displays 'Autonomous Databases in atplab Compartiment' with a table header: Display Name, State, Dedicated, OCPUs, and Storage (TB). The table body shows 'No items'. At the bottom left, there's a 'List Scope' link.



Fill in the data needed to create the database (you can refer to the screenshots included after the table below).

Field	Value
<b>Compartment</b>	atplab
<b>Display name</b>	atplabpub
<b>Database name</b>	atplabpub
<b>Workload type</b>	Transaction processing
<b>Deployment type</b>	Shared infrastructure
<b>Database version</b>	19c
<b>OCPUs</b>	1
<b>Storage</b>	1TB
<b>Auto scaling</b>	Disabled

## Create Autonomous Database

Provide basic information for the Autonomous Database

**Compartment**  

workshop2020101401 (root)/atplab

**Display name**  

A user-friendly name to help you easily identify the resource.

**Database name**  

The name must contain only letters and numbers, starting with a letter. Maximum of 14 characters.

Choose a workload type

**Data Warehouse**  
Built for decision support and data warehouse workloads. Fast queries over large volumes of data.

**Transaction Processing**  
Built for transactional workloads. High concurrency for short-running queries and transactions.

**JSON**  
Built for JSON-centric application development. Developer-friendly document APIs and native JSON storage.

Choose a deployment type

**Shared Infrastructure**  
Run Autonomous Database on shared Exadata infrastructure.

**Dedicated Infrastructure**  
Run Autonomous Database on dedicated Exadata infrastructure.



#### Configure the database

Always Free ⓘ  
 Show only Always Free configuration options

Choose database version  
19c

OCPU count      Storage (TB)

1      1

The number of OCPU cores to enable. Available cores are subject to your tenancy's service limits.

Auto scaling  
Allows system to use up to three times the provisioned number of cores as the workload increases. [Learn more.](#)

In the *password* field enter: **Autonomous#2020**

Create administrator credentials ⓘ

Username READ-ONLY  
ADMIN  
Admin username cannot be edited.

Password  
.....

Confirm password  
.....

In this phase, select **Secure access from allowed IPs and VCNs only**. A new form appears, to input access rules for the database. (*Currently this configuration can be accessed from a new dedicated option*).

#### Choose network access

Access Type

Secure access from everywhere  
Allow users with database credentials to access the database from the internet.

Secure access from allowed IPs and VCNs only  
Restrict access to specified IP addresses and VCNs.

Private endpoint access only  
Restrict access to a private endpoint within an OCI VCN.

IP notation type  
CIDR Block      Values  
0.0.0.0/0

IP notation type  
CIDR Block      Values  
240.0.0.0/4

+ Access Control Rule

Require mutual TLS (mTLS) authentication ⓘ  
If you select this option, mTLS will be required to authenticate connections to your Autonomous Database.



Select **CIDR Block** from the IP Notation type box on the left side. Enter values **0.0.0.0/0** to do the **Hands On Lab 5** later. You should really put the IP address of your laptop or local desktop, but it could be behind a company VPN or a firewall, and to avoid problems that is why we will use the indicated value.

Click on the **[+ Acess Control Rule]** button and also enter in the Access Control List, the **CIDR Block 240.0.0.0/4**, which corresponds to the CIDR of the **Service Gateway** that we will discuss later.

Also select the "Bring Your Own License" option.

There are two licensing models in Oracle's standalone databases:

**Bring your own license:** If you own an Oracle license, and you want to use it in Oracle Cloud, you will only have to pay for the cloud resources consumed, and not for the use of your license again (since you own one).

**License Included:** You will be charged for the use of cloud resources, as well as the possession of a new database license in the Oracle cloud.

For the purpose of this demo, led by Oracle, and since it is a demo environment, you can choose the "**Bring Your Own License**" option, which will consume much less credits.

Once all the fields have been filled in, click on the **Create Autonomous Database** button to create the database.

Choose a license type

Bring Your Own License (BYOL)  
Bring my organization's Oracle Database software licenses to the Database service.  
[Learn more](#)

License Included  
Subscribe to new Oracle Database software licenses and the Database service.

Provide up to 10 maintenance contacts

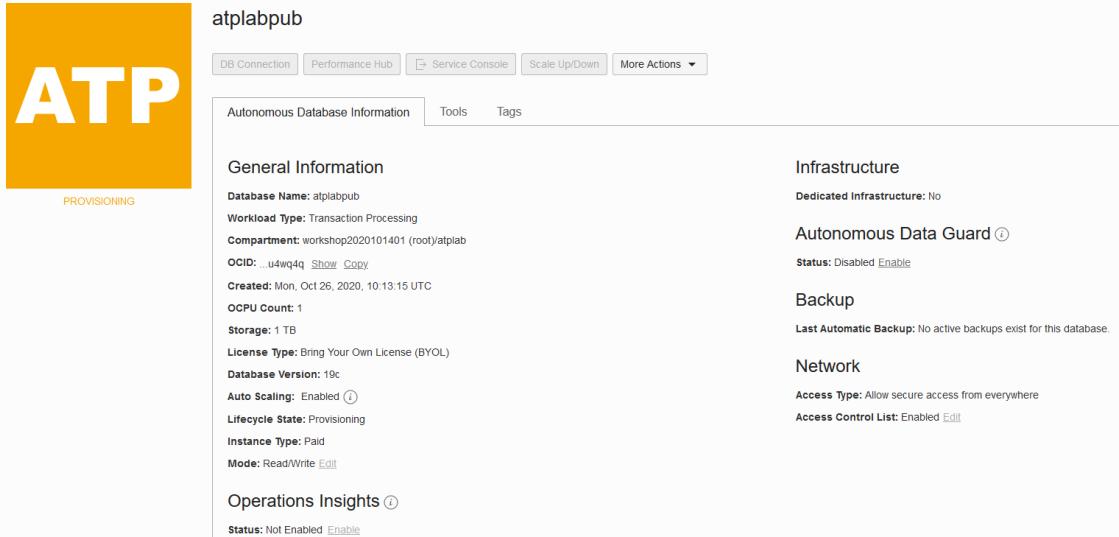
Add Contact

Show Advanced Options

Create Autonomous Database Cancel

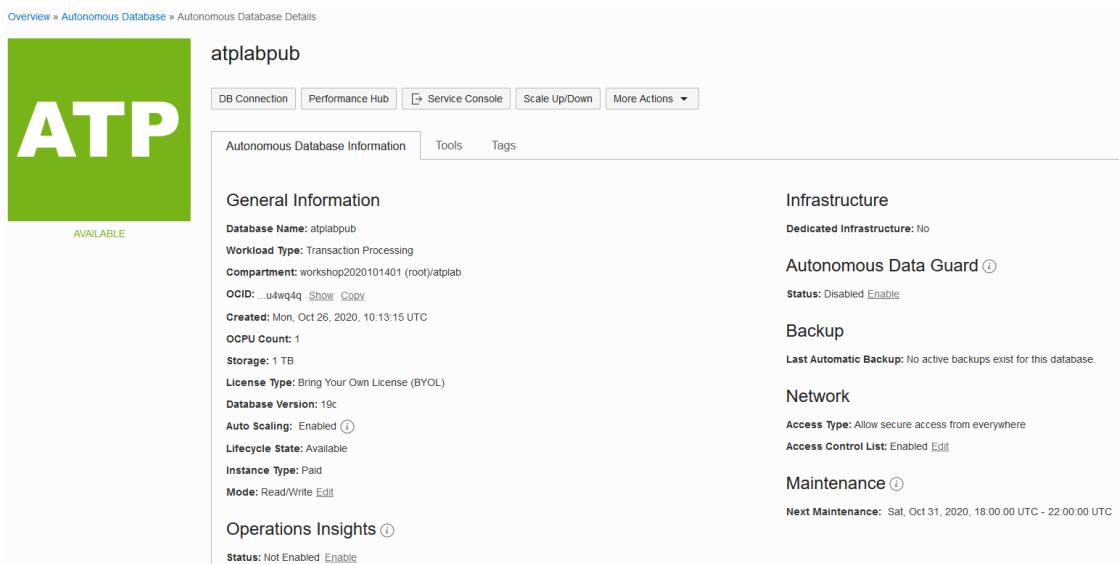
It will show in **[provisioning]** state for a few minutes, until it finishes provisioning in OCI. You can continue in the meantime with the laboratory.





The screenshot shows the Oracle Cloud Infrastructure (OCI) Autonomous Database details page. The database name is 'atplabpub'. The status is 'PROVISIONING'. The General Information section includes details like Workload Type (Transaction Processing), Compartment (workshop2020101401 (root)/atplab), and License Type (Bring Your Own License (BYOL)). The Infrastructure section shows 'Dedicated Infrastructure: No'. The Network section shows 'Access Type: Allow secure access from everywhere' and 'Access Control List: Enabled'. The Operations Insights section shows 'Status: Not Enabled'.

It will then become Available in Green.



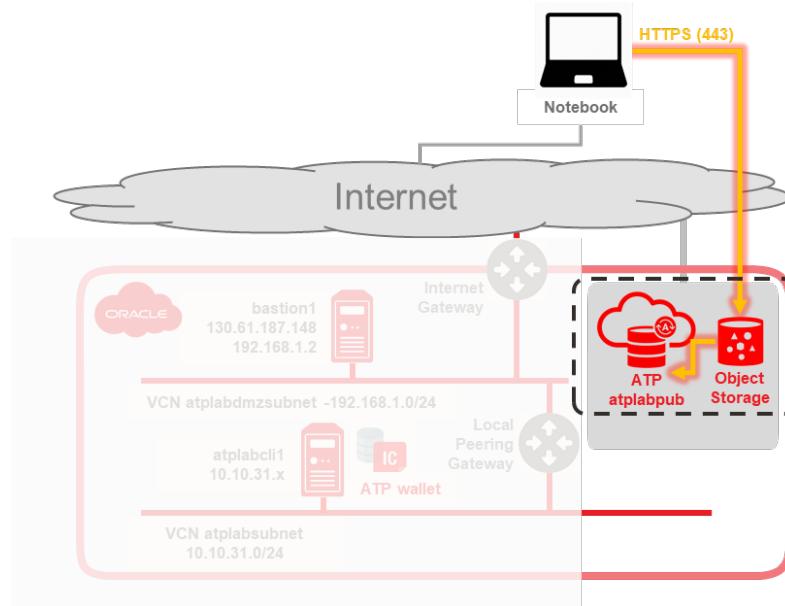
The screenshot shows the Oracle Cloud Infrastructure (OCI) Autonomous Database details page for the same database 'atplabpub'. The status is now 'AVAILABLE'. The General Information section remains the same. The Infrastructure section shows 'Dedicated Infrastructure: No'. The Network section shows 'Access Type: Allow secure access from everywhere' and 'Access Control List: Enabled'. The Operations Insights section shows 'Status: Not Enabled'.



# Creating Object Storage

## What am I going to do?

You are going to create cloud storage with the Object Storage/bucket element, to store a file with the information to be uploaded to the autonomous database that you created in the previous section.



While the database is being created, you can create an Object Storage bucket (object/file oriented storage) to upload the data you want to import into the database.

From the OCI main menu (hamburger icon), select **Storage -> Object Storage**.



Make sure that the selected compartment is [**atplab**], and then click the [**Create Bucket**] button to create a new Bucket.



Object Storage

### Buckets in atplab Compartments

You can use 10 GiB of Object Storage and 10 GiB of Archive Storage for free in your home region. You are currently using 0 GiB of Object Storage and 0 GiB of Archive Storage. You have not upgraded when your Free Trial ends, your data is deleted. [Show details](#)

**Create Bucket**

Name	Storage Tier
	No item

**COMPARTMENT**

- atplab

workshop20200505a (root)/atplab

Fill in the fields needed to create this Bucket (you can refer to the screenshots included after the table below to guide you through the creation).

<b>Bucket Name</b>	atplab_bucket
<b>Storage Tier</b>	STANDARD
<b>Encryption</b>	Encrypt using ORACLE managed Keys

**Create Bucket** [Help](#) [Cancel](#)

**BUCKET NAME**  
atplab\_bucket

**STORAGE TIER**  
Storage tier for a bucket can only be specified during creation. Once set, you cannot change the storage tier in which a bucket resides.

STANDARD  
 ARCHIVE

**OBJECT EVENTS** [i](#)  
 EMIT OBJECT EVENTS

**OBJECT VERSIONING** [i](#)  
 ENABLE OBJECT VERSIONING

**ENCRYPTION**

ENCRYPT USING ORACLE MANAGED KEYS  
Leaves all encryption-related matters to Oracle.  
 ENCRYPT USING CUSTOMER-MANAGED KEYS  
Requires a valid key from a vault that you have access to. ([Learn More](#))

**TAGS**  
Tagging is a metadata system that allows you to organize and track resources within your tenancy. Tags are composed of keys and values that can be attached to resources.  
[Learn more about tagging](#)

TAG NAMESPACE	TAG KEY	VALUE
None (add a free-form tag)		

+ Additional Tag

**Create** **Cancel**



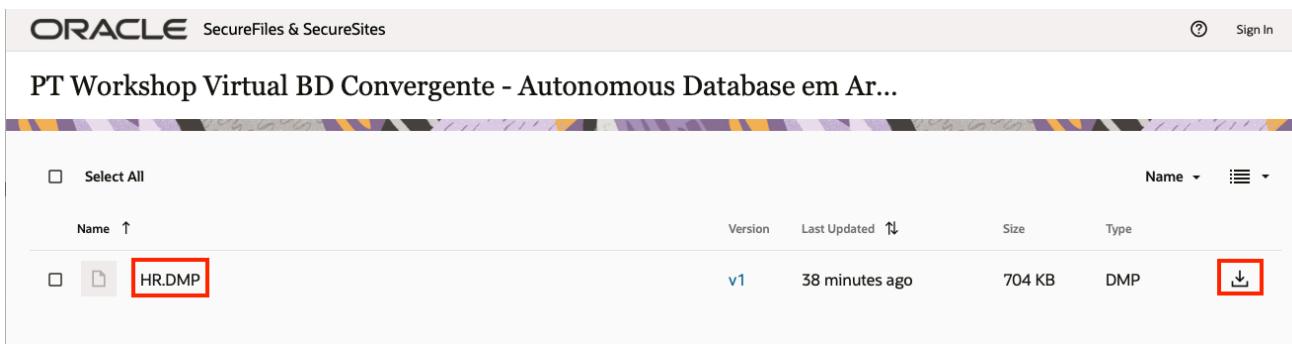
When finished, press the [**Create**] button to create a new Object Storage Bucket, and verify that it was created successfully.

Access the new bucket by clicking on its name.

Create Bucket				
Name	Storage Tier	Visibility	Created	⋮
<a href="#">atplab_bucket</a>	Standard	Private	Tue, May 5, 2020, 09:56:41 UTC	⋮
Showing 1 Item < 1 of 1 >				

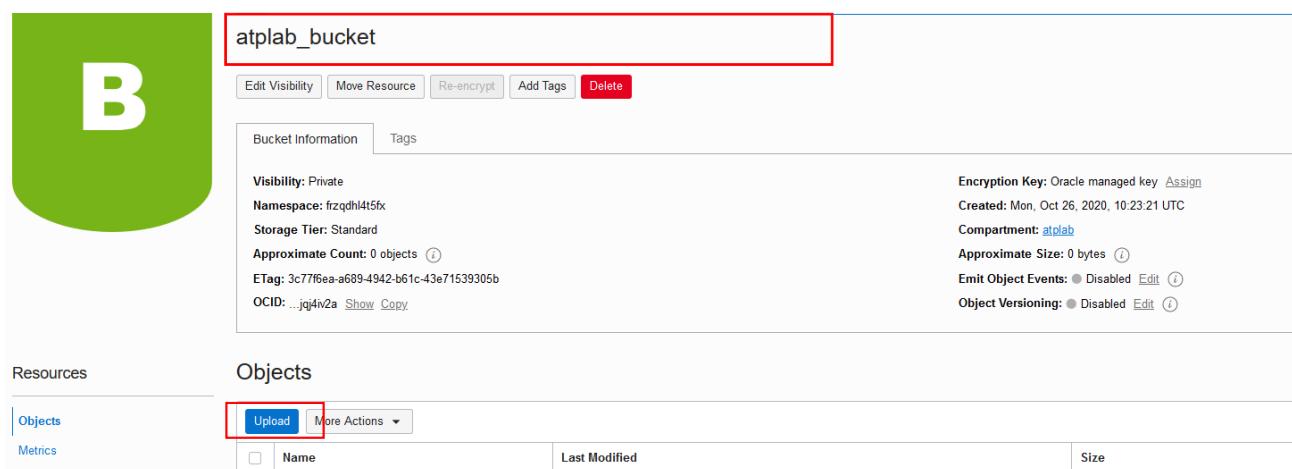
Download the **HR.DMP** file from the Oracle SecureFiles repository:

<https://bit.ly/PTWkshopMulticloudFev2022>



The screenshot shows the Oracle SecureFiles & SecureSites interface. At the top, there's a navigation bar with the Oracle logo, a search bar, and a sign-in link. Below the header, a banner reads "PT Workshop Virtual BD Convergente - Autonomous Database em Ar...". The main content area displays a list of files. A file named "HR.DMP" is highlighted with a red box around its name. To the right of the file list are columns for Version, Last Updated, Size, and Type. The "Type" column shows "DMP" and the "Size" column shows "704 KB". A red box also surrounds the download icon (a downward arrow) for the "HR.DMP" file.

Click the [**Upload**] button to upload a new object (in this case the **HR.DMP** file) to this bucket.



The screenshot shows the Oracle Object Storage Bucket details page for "atplab\_bucket". The top section displays bucket metadata: Visibility: Private, Namespace: frzqdh14t5fx, Storage Tier: Standard, Approximate Count: 0 objects, ETag: 3c77f6ea-a689-4942-b61c-43e71539305b, OCID: ...joj4n2a, and links for Show and Copy. Below this is a "Resources" section with tabs for Objects and Metrics. Under the Objects tab, there's a table with columns for Name, Last Modified, and Size. A red box highlights the "Upload" button at the top left of the table. The "Metrics" tab is also visible.

You can upload the file by dragging it to this window. Then press the [**Upload**] button to upload the previously uploaded file.



## Upload Objects

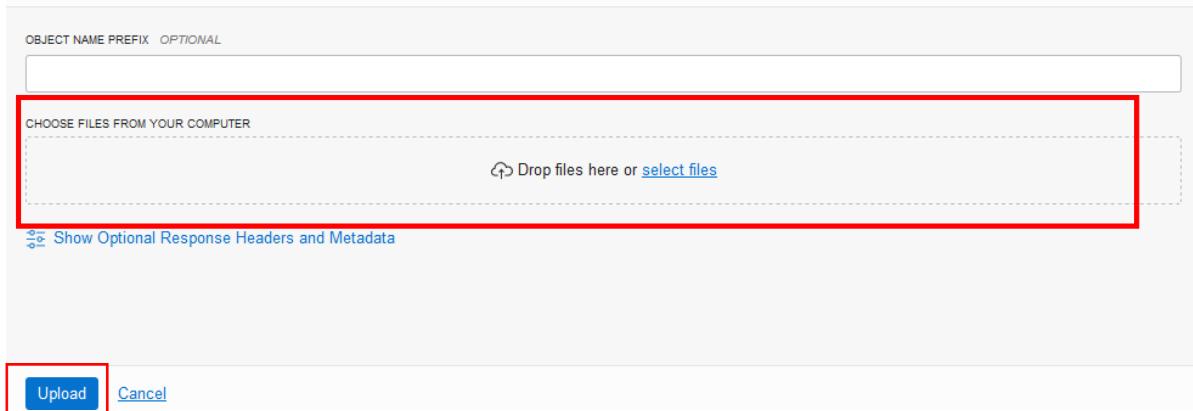
[Help](#)

OBJECT NAME PREFIX OPTIONAL

CHOOSE FILES FROM YOUR COMPUTER

[Show Optional Response Headers and Metadata](#)

[Cancel](#)



Then press the **Close** button to continue.

Once loaded into OCI you will notice that it appears inside the Objects table of the Bucket you previously created.

## Objects

Objects				
<input type="button" value="Upload"/>	<a href="#">More Actions</a>	<input type="text" value="Search by prefix"/>		
<input type="checkbox"/>	Name	Last Modified	Size	Status
<input type="checkbox"/>	HR.DMP	Mon, Oct 26, 2020, 10:27:40 UTC	704 KiB	Available

In this case we have uploaded an Export Datapump of the HR user, that is included in the workshop material.

## Create Pre-Authenticated Request

Objects in an Object Storage can be downloaded in two ways:

- **Authentication of the cloud account:** Having a user with access to the Object Storage bucket, the object can be accessed through its associated URL, with username and password, which is located in the details of the object.
- **Pre-Authenticated token:** A pre-authenticated URL is created, that will not ask for username and password, and it will uniquely identify that object for a predefined period of time. This second method is the one we will use to import the data from this backup into our ATP.

The next step will therefore be to create a **Pre-Authenticated request**, for the HR file.dmp that we imported into the bucket in the previous step.

Click on the menu on the right (above the three-dot icon) of the object and select the **Create Pre-Authenticated Request** option.



## Objects

Name	Last Modified	Size
HR.DMP	Mon, Oct 26, 2020, 10:27:40 UTC	704 KiB

View Object Details  
Download  
Copy  
Restore  
**Create Pre-Authenticated Request**  
Re-encrypt  
Rename  
Delete

Rename it or leave the default name. Click on the **Create Pre-Authenticated Request** button to create the URL with the access token.

Name: par-object-20210427-2019

Pre-Authenticated Request Target:

- Bucket**: Create a pre-authenticated request that applies to all objects in the bucket.
- Object**: Create a pre-authenticated request that applies to a specific object.
- Objects with prefix**: Create a pre-authenticated request that applies to all objects with a specific prefix.

Object Name: HR.DMP

Access Type:

- Permit object reads
- Permit object writes
- Permit object reads and writes

Expiration: May 31, 2021 18:19 UTC

**Create Pre-Authenticated Request** **Cancel**

You can then copy the URL with the token by clicking on the copy icon to the right of the same URL. Keep it in a safe place, you will need it later and it cannot be recovered once this window is closed. Once the URL has been copied, click on the **Close** button.

**Important note! If the access URL is lost, a new one will have to be created.**



### Pre-Authenticated Request Details

[Close](#)

NAME READ-ONLY

par-object-HR.DMP-20200422-1816

PRE-AUTHENTICATED REQUEST URL READ-ONLY

https://objectstorage.eu-frankfurt-1.oraclecloud.com/p/bErnpSW1I9-To\_sDILb8GpICCdGblooYNSeP6yLxeYk/n/fr692z7uvgw:



Copy this URL for your records. It will not be shown again.

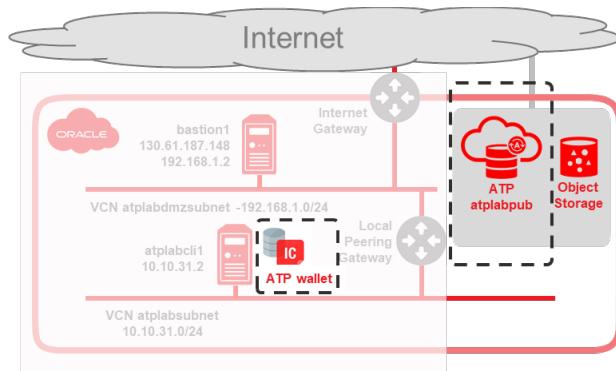
[Close](#)



# Continuing with ATP, services review

## What am I going to do?

You will continue with the ATP configuration and generate the configuration files and keys (wallet) necessary for the rest of the laboratory.



Select **Oracle Database -> Autonomous Transaction processing** again from the OCI main menu (hamburger icon).

The screenshot shows the OCI navigation bar with the following menu items: Home, Compute, Storage, Networking, Oracle Database (highlighted with a red box), and Databases. On the right, there is an 'Overview' section with links to Autonomous Database, Autonomous Data Warehouse, Autonomous JSON Database, and Autonomous Transaction Processing (also highlighted with a red box and a cursor icon).

Verify that the standalone database has already been created and is available.

Click on the database name to access its control panel.

Autonomous Databases *in* atplab Compartiment

Create Autonomous Database							
Display Name	State	Dedicated	OCPUs	Storage (TB)	Workload Type	Autonomous Data Guard	Created
atplabpub	Available	No	1	1	Transaction Processing	Disabled	Mon, Oct 26, 2020, 10:13:15 UTC

Displaying 1 Autonomous Database < 1 of 1 >

Here you can explore the different sections that make up this database control panel. You can select the tools tab to access some tools that come included with your database, such as Database Actions, Oracle Machine Learning, or Oracle Application Express.



The screenshot shows the Oracle Autonomous Database Details page. On the left is a green box with 'ATP' and 'AVAILABLE' text. The top navigation bar includes 'DB Connection', 'Performance Hub', 'Service Console', 'Scale Up/Down', and 'More Actions'. The 'Tools' tab is highlighted with a red box. Below the tabs, there's a section titled 'Autonomous Database Information' with 'Tools', 'Tags', and 'Database administration and developer tools for Autonomous Database' sections. The 'Database Actions' section contains a 'Open Database Actions' button. The 'Oracle Application Express' section contains a 'Open APEX' button. The 'Oracle ML User Administration' section contains a 'Open Oracle ML User Administration' button. The 'SODA Drivers' section contains a 'Download SODA Drivers' button.

You can also explore the **performance hub** section, where you can see the queries plans running, as well as an overview of database usage and performance.

The screenshot shows the Oracle Autonomous Database Details page with the 'Performance Hub' tab highlighted with a red box. The rest of the interface is identical to the previous screenshot, showing the 'Tools' tab selected and the various developer tools available for the Autonomous Database.

## Performance Hub

The screenshot shows the Oracle Performance Hub page. At the top, there are filters for 'QUICK SELECT' (Last Hour), 'TIME RANGE' (Oct 26, 2020 9:34:26 AM - 10:34:26 AM), and 'TIME ZONE' (UTC). There are 'Reports' and 'Refresh' buttons. Below this is a chart titled 'Activity' showing thread activity over time from 09:40 AM to 10:20 AM. The chart includes a legend for 'MAXIMUM THREADS' (orange), 'Wait' (blue), 'User I/O' (green), and 'CPU' (light blue). The 'ASH Analytics' tab is selected, showing a bar chart of Average Active Sessions over time. The chart has 'ASH DIMENSIONS' (Consumer Group) set to 'None'. There are 'VIEW OPTIONS' and 'SAMPLE RESOLUTION' dropdowns. The bottom of the page features a large red 'O' logo and a footer with copyright information.

Next, go to the **Service Console** in the main menu of your ATP. A new tab will open in your browser. If the new tab does not appear, please check your browser settings because it could be blocking it.

Overview » Autonomous Database » Autonomous Database Details

atplabpub

DB Connection Performance Hub **Service Console** Scale Up/Down More Actions ▾

Autonomous Database Information Tools Tags

Database administration and developer tools for Autonomous Database

Database Actions

Load, explore, transform, model, and catalog your data. Use an SQL worksheet, build REST interfaces and low-

On the new tab, select **Administration** from the menu on the left.

Autonomous Transaction Processing

Overview

Activity

**Administration**

Development

DATABASE  
ATPLABPUB

Here you can see some of the administration tasks that can be executed. In this case we will show you how to download the credential file of the Oracle client (Wallet)

Autonomous Transaction Processing

Overview

Activity

**Administration**

Development

DATABASE  
ATPLABPUB

**Download Client Credentials (Wallet)**

Connections to Autonomous Transaction Processing use a secure connection. Your existing tools and applications will need to use this wallet file to connect to your Autonomous Transaction Processing instance. If you are familiar with using an Oracle Database within your own data center, you may not have previously used these secure connections.

Set Resource Management Rules

Set resource management rules to allocate CPU/I/O shares to consumer groups and to cancel SQL statements based on their runtime and amount of I/O.

Set Administrator Password

Set or reset your database administrator user's (ADMIN) password and when locked unlock your administrator user account on Autonomous Transaction Processing.

Manage Oracle ML Users

Create new Oracle Machine Learning user accounts and manage the credentials for existing Oracle Machine Learning users.

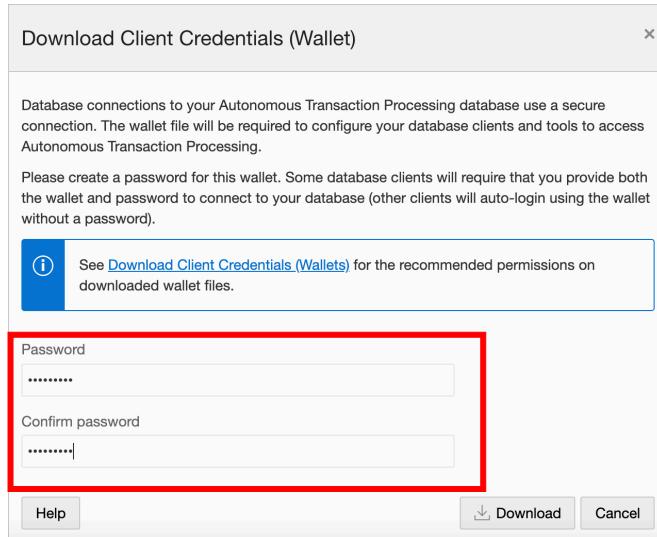
Send Feedback to Oracle

Use our Cloud Customer Connect forum to provide feedback about the service to Oracle, post questions, connect with experts, and share your thoughts and ideas. Click here to link to the forum.

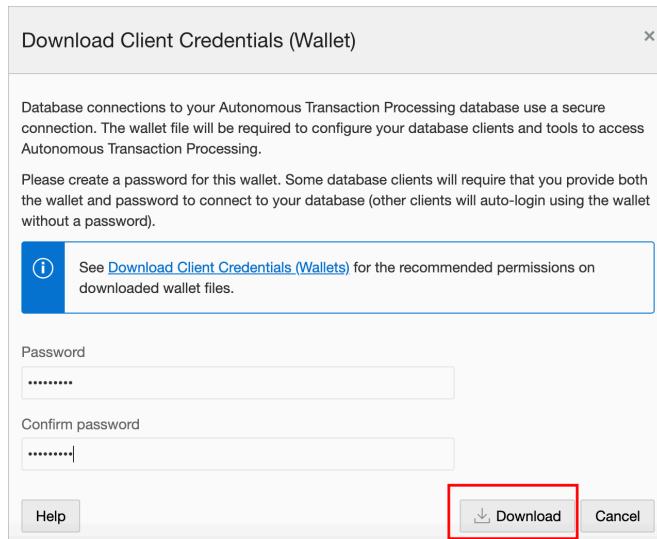


# How to generate and download the credential file from the ATP administration console

Provide a password (you can use the same password you have been using so far **Autonomous#2020** or one of your choice, with at least 8 characters.



Once this is done, press the **Download** button and download the file containing the Wallet to your local machine.



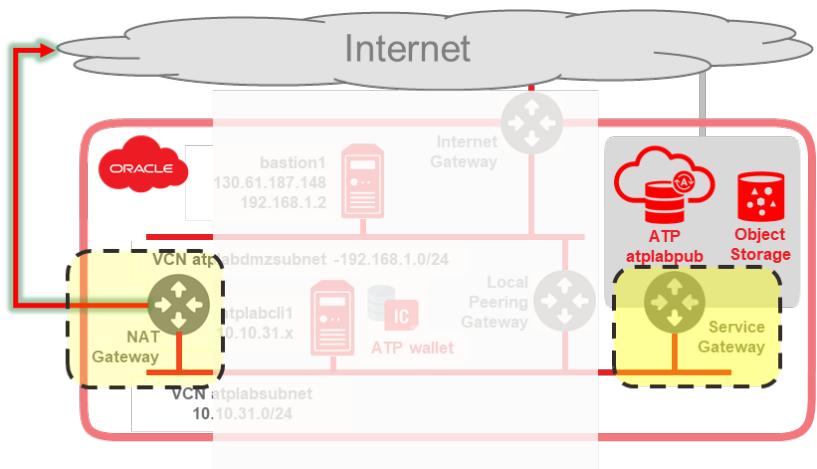
There are other methods to download the wallet, such as using the command line, which will be used later in the lab to download the credential file (wallet) of your ATP.



# Configuration of ATP access machines (Bastion and Client)

## What am I going to do?

You will install the client software required to access the ATP and create the necessary network resources (NAT Gateway and Service Gateway) to make the necessary connections.



To download the pieces of software necessary to connect with the ATP database, we will first access the servers from where we will access our ATP (Bastion and Client machine).

For this workshop we need different pieces of Software related to the Oracle client:

- Oracle Instant Client Basic
- SQLPlus
- Tools

The first thing to do is to allow the Client machine to access the internet, to download the software described. To do this, you have to create access rules in the virtual networks. Return to the OCI tab in your browser if you are not in it and select from the main menu of OCI (hamburger icon) **Networking -> Virtual Cloud Networks**.

A screenshot of the Oracle Cloud Infrastructure (OCI) console. On the left is a sidebar with navigation links: Home, Compute, Storage, Networking (which is highlighted with a red box), Oracle Database, and Databases. On the right is the 'Overview' page for Virtual Cloud Networks, which includes links for Virtual Cloud Networks (also highlighted with a red box), Load Balancers, Network Visualizer, and Inter-Region Latency. A cursor is hovering over the 'Virtual Cloud Networks' link.



Next, select the **atplabnet** network by clicking on its name within the virtual network table of your atplab compartment.

### Virtual Cloud Networks *in atplab Compartment*

Name	State	CIDR Block	Default
atplabdmznet	Available	192.168.1.0/24	<a href="#">Default</a>
<a href="#">atplabnet</a>	Available	10.10.31.0/24	<a href="#">Default</a>

Select **NAT Gateway** in the *Resources* menu on the left and create a NAT Gateway by clicking on the **Create NAT Gateway** button, this will allow us to access the internet from the private network where the machines with which we will connect to the ATP are located. Internet access only occurs from these machines to the outside, and not in the opposite direction, from the outside to the machines on the private network. See the documentation for detailed information about NAT Gateway:

<https://docs.cloud.oracle.com/en-us/iaas/Content/Network/Tasks/NATgateway.htm>

### Resources

### NAT Gateways *in atplab Compartment*

Name	State

**Create NAT Gateway**

Subnets (1)  
Route Tables (1)  
Internet Gateways (0)  
Dynamic Routing Gateways (0)  
Network Security Groups (1)  
Security Lists (1)  
DHCP Options (1)  
Local Peering Gateways (1)  
**NAT Gateways (0)**  
Service Gateways (0)

Enter the name **[atplabnatg]** and the atplab compartment for the NAT Gateway. Then finish creating the NAT Gateway by clicking on the **Create NAT Gateway** button. Afterwards, in the next window press the **Close** button to finish the process.



## Create NAT Gateway

[Help](#)

A NAT gateway lets instances that don't have public IP addresses access the internet.

NAME

atplabnatg

CREATE IN COMPARTMENT

atplab

workshop2020101401 (root)/atplab

### Ephemeral Public IP Address

The public IP address' lifetime is bound to the lifetime of the NAT Gateway.

### Reserved Public IP Address

You control the public IP address' lifetime. You can unassign it or reassign it to another resource in the same region.

Oracle will generate an IP address for you.

[Show Advanced Options](#)

[Create NAT Gateway](#)

[Cancel](#)

Next, enter **Route Tables** in the left menu of *Resources* and select the **Default Route Table for atplabnet** from the route table on the right.

The screenshot shows the Oracle Cloud Infrastructure Resources page. On the left, there is a sidebar with links: Subnets (1), **Route Tables (1)**, Internet Gateways (0), Dynamic Routing Gateways (0), Network Security Groups (1), and Security Lists (1). The 'Route Tables (1)' link is highlighted with a red box. On the right, the main content area is titled 'Route Tables in atplab Compartment'. It features a 'Create Route Table' button and a table with two columns: 'Name' and 'State'. A single row is listed: 'Default Route Table for atplabnet' with a status of 'Available'. This row is also highlighted with a red box.

The next step will be to add the rule that allows access to the machines of the private network to the Internet. To do this, click on the **Add Route Rules** button and a new window will appear on the right, to add the necessary data to create the new rule.

In *TARGET TYPE* select **NAT Gateway** from the drop-down menu. New configuration options will appear such as:

- *Destination CIDR Block*, to be filled in with the value **0.0.0.0/0**
  - *Target NAT Gateway*, in ATPLAB you must select the previously created NAT Gateway **[atplabnatg]**
  - *Description*, which is an optional value and could include a descriptive comment.
- Finally, press the **Add Route Rules** button to create the new route.



In addition to accessing the Internet to download the software necessary to access the ATP, the machines of the private network need to have access to the network where the OCI services are, in this case the ATP. To be able to access the network containing these services, you have to add a new route in the default route table through another network element that is the **Service Gateway**.

To create the Service Gateway, go to the **atplabnet** network menu. You can do this by directly selecting the name in the top navigation string.

Then select **Service Gateway** from the Resources menu on the left. And then click on the **Create Service Gateway** button to create the new Service Gateway. A context menu appears on the right to configure the Service Gateway.



## Resources

- [Subnets \(1\)](#)
- [Route Tables \(3\)](#)
- [Internet Gateways \(0\)](#)
- [Dynamic Routing Gateways \(1\)](#)
- [Network Security Groups \(0\)](#)
- [Security Lists \(1\)](#)
- [DHCP Options \(1\)](#)
- [Local Peering Gateways \(1\)](#)
- [NAT Gateways \(1\)](#)
- [Service Gateways \(0\)](#)

## Service Gateways *in atplab Compartment*

[Create Service Gateway](#)

Name	State	Services
No items found		



Enter the following values to create the Service Gateway:

<b>NAME</b>	atplabsrg
<b>COMPARTMENT</b>	atplab
<b>SERVICES</b>	All FRA Services in Oracle Services Network

The screenshot shows the Oracle Cloud interface for creating a Service Gateway. On the left, there's a sidebar with 'VCN' and 'atplabnet' selected. The main area shows 'Service Gateways in atplab Compartment'. A 'Create Service Gateway' button is visible. On the right, a 'Create Service Gateway' dialog box is open. It has three main sections: 'NAME' (containing 'atplabsrg'), 'CREATE IN COMPARTMENT' (set to 'atplab'), and 'SERVICES' (set to 'All FRA Services in Oracle Services Network'). The 'SERVICES' section and the entire dialog box are highlighted with a red box. At the bottom of the dialog, there are 'Create Service Gateway' and 'Cancel' buttons, with the 'Create Service Gateway' button also highlighted with a red box.

Press the **Create Service Gateway** button, and then in the next window the **Close** button to finish the process.

Once the Service Gateway is created, a new rule must be added to the default route table in order to make effective use of the Oracle Services network. So far, you have made the network connections and with the new rule you will enable the transfer of information through them.

Select **Route Tables** again from the left menu Resources of the VCN atplab and click on the name of the route table **Default Route Table for atplabnet**.



Resources

### Route Tables in atplab Compartment

Create Route Table	
Name	State
Default Route Table for atplabnet	Available

- Subnets (1)
- Route Tables (1)**
- Internet Gateways (0)
- Dynamic Routing Gateways (0)
- Network Security Groups (1)
- Security Lists (1)

Add another rule to access Frankfurt OCI services through the **Service Gateway**, with the following data:

<b>TARGET TYPE</b>	Service Gateway
<b>DESTINATION SERVICE</b>	All FRA Services in Oracle Services Network
<b>TARGET SERVICE GATEWAY</b>	atplabsrg

The screenshot shows the Oracle Cloud interface for managing route tables. On the left, there's a sidebar with 'Route Tables (1)' selected. The main area shows the 'Default Route Table for atplabnet' with its details. On the right, a modal window titled 'Add Route Rules' is open, containing fields for setting up a new route rule. The 'TARGET TYPE' dropdown is set to 'Service Gateway'. The 'DESTINATION SERVICE' dropdown is set to 'All FRA Services In Oracle Services Network'. The 'TARGET SERVICE GATEWAY IN ATPLAB' dropdown is set to 'atplabsrg'. At the bottom of the modal, the 'Add Route Rules' button is highlighted with a red box.

Finally, click on the **Add Route Rules** button to create the new rule that will allow access and transfer of information between your private network and the private network of native OCI services.

Once the routes are created, we can see the following entries:



Route Rules			
		Add Route Rules	Edit
	Destination	Target Type	Target
<input type="checkbox"/>	0.0.0.0/0	NAT Gateway	<a href="#">atplabnatg</a>
<input type="checkbox"/>	192.168.1.0/24	Local Peering Gateway	<a href="#">atplabnetlg</a>
<input type="checkbox"/>	All FRA Services In Oracle Services Network	Service Gateway	<a href="#">atplabsg</a>

0 Selected

- **The first** rule is to be able to go out to the internet through the **NAT Gateway**, so the **CIDR Block 0.0.0.0/0** has been selected, which means that there are no IP restrictions regarding the exit to the internet. See the documentation for detailed information about NAT Gateway: <https://docs.cloud.oracle.com/en-us/iaas/Content/Network/Tasks/NATgateway.htm>
- **The second** rule is to access our **private network (192.168.1.0/24)**, through the **Local Peering Gateway**, which is the gateway that will connect this subnet.

A **Local Peering Gateway** is a gateway that connects different networks within the same tenant (or cloud account). For more information, go to the following link in the documentation: <https://docs.cloud.oracle.com/en-us/iaas/Content/Network/Tasks/localVCNpeering.htm>

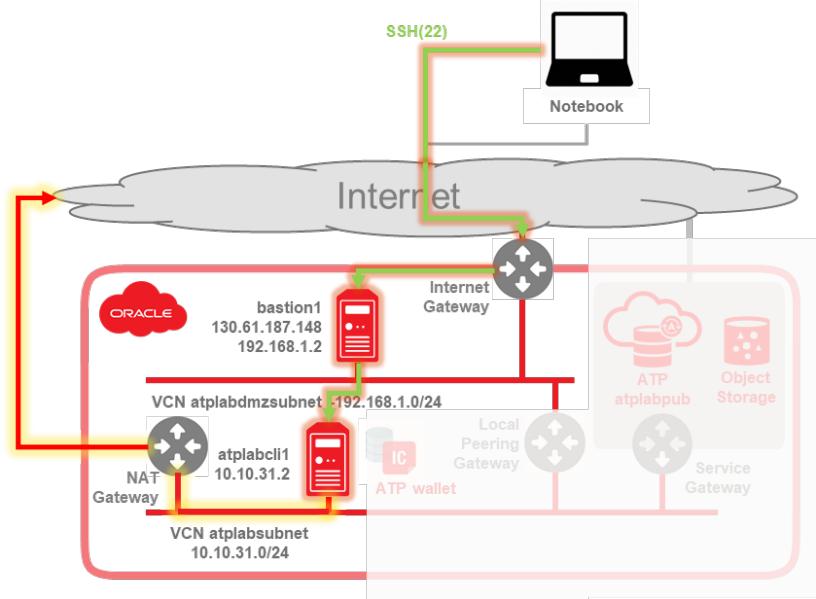
- **The third** rule is to **connect our private network to the native Oracle Cloud Infrastructure services** within this region (FRA – Frankfurt). To do this, these services are linked through a gateway called **service gateway**. For more information about **Service Gateway**, see the following link to the documentation: <https://docs.cloud.oracle.com/en-us/iaas/Content/Network/Tasks/servicegateway.htm>



# Download the client software to access the ATP

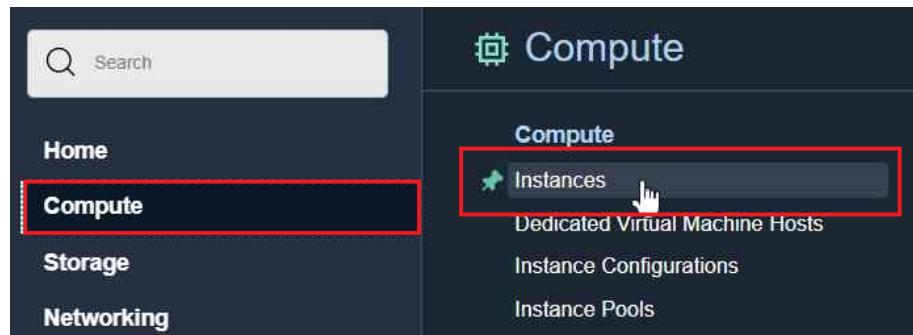
## What am I going to do?

You will download the client software needed to access the ATP from the client machines. To do this, you will use the NAT Gateway previously created to have access to the Internet from the private network.



Once all the gateways and routes have been created, the next thing to do is to download the necessary software to access the ATP Database.

Go to the OCI main menu (hamburger icon) and select **Compute -> Instances**.



There you can see the virtual machines of the atplab compartment and specifically the **bastion1** machine, which will be used to connect from outside the Oracle Cloud network. There is also the **atplabcli1** client machine, which is within the private network, and will be used to connect to the ATP.



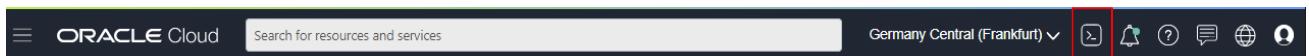
Next, take note of the **public IP address** of the **bastion1** machine, you will need it to be able to connect later by ssh.

### Instances *in atlab Compartment*

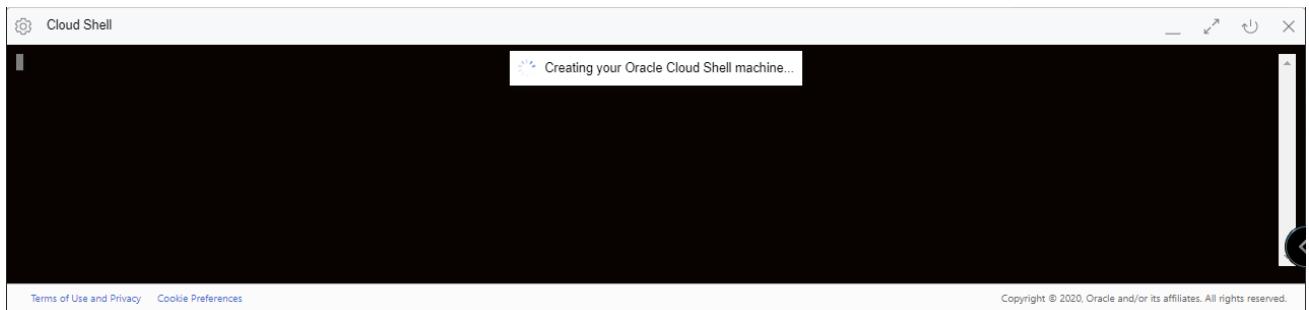
Create Instance						
Name	Status	Public IP	Shape	Availability Domain	Fault Domain	Created
bastion1	● Running	130.61.225.5	VM.Standard2.1	AD-1	FD-3	Wed, Apr 22, 2020, 07:03:02 UTC
atplabcli1	● Running	-	VM.Standard2.1	AD-1	FD-3	Wed, Apr 22, 2020, 07:03:02 UTC
Showing 2 Items					Page 1	>

To connect by ssh you can use any local computer, which already has an ssh client installed. You can also use the preconfigured Linux cloud **Shell** machine, which is available on all tenants and for all users with permissions in OCI. More information about [Cloud Shell](#) can be found on its documentation website.

To access the Cloud Shell click on the **terminal icon** at the top right of the main OCI panel right next to the Region selector.



Cloud Shell will open at the bottom of the main OCI dashboard. It will take a few seconds for the system's command prompt to appear.



The Linux Cloud Shell machine has an internet connection and you can use it to connect via ssh to other machines with internet access, in the case of this workshop with the **bastion machines** of the different cloud providers, for which you will have to use the private key files, which have been provided at the beginning of the workshop.

Although we recommend using Cloud Shell to make all connections, to avoid having to download programs to your laptop or desktop, you can also use your computer by downloading and using any of the following applications.



**Note:** You can use any ssh client to connect to the bastion machine, preferably **Xterm moba**, or **putty + winscp**.

<https://mobaxterm.mobatek.net/>

<https://www.putty.org/>

<https://winscp.net/eng/download.php>

Connect to the **bastion1** machine via ssh, using the private key provided at the beginning of the workshop [**atplab\_rsa**].

If you use Cloud Shell click on the Cloud Shell menu, hamburger icon or three parallel lines, and select the **Upload** option.



Drag the [**atplab\_rsa**] file to the window that appears or click on the **select from your computer** link and select the file stored on your laptop. Press the **Upload** button and the file will be uploaded to the Cloud Shell

File Upload to your Home Directory

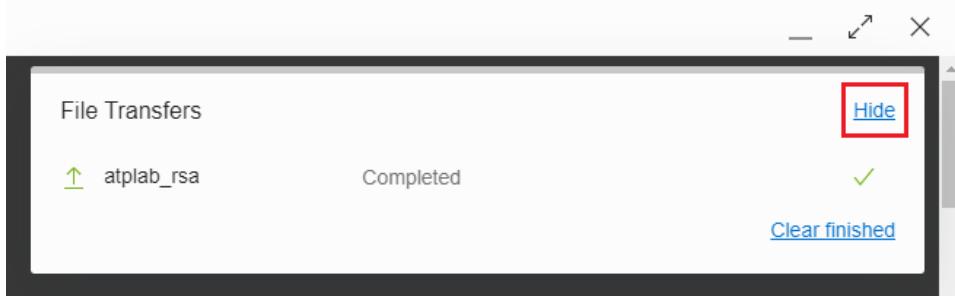
Drag and drop or select a file to upload to your home (~/) directory.

Drop a file or [select from your computer](#)

atplab\_rsa

A pop-up window will appear indicating the status of the file transfer, click on the **Hide** link to hide the window.





If you have access permission problems from a local **Linux** machine, please change the file permissions **atplab\_rsa** with the command **chmod 400 atplab\_rsa**.

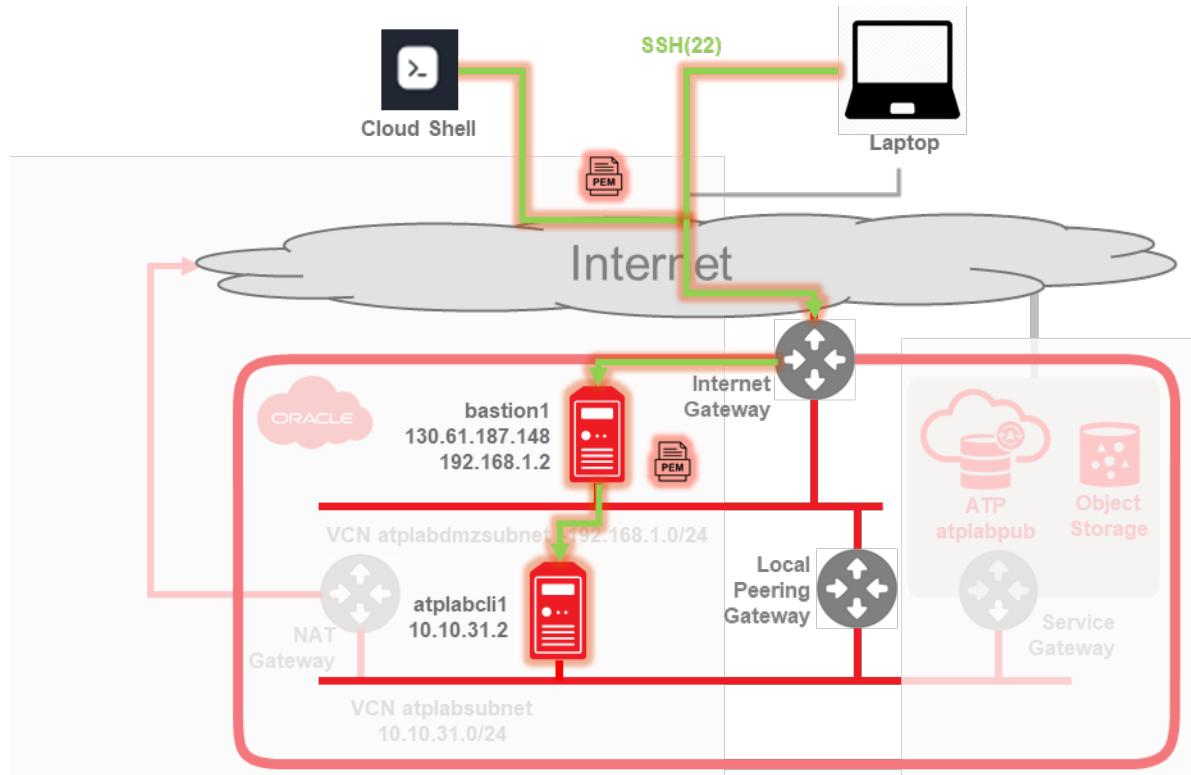
Type the following commands to connect to the **bastion1** machine.

```
$ chmod 400 atplab_rsa
$ ssh -i atplab_rsa opc@<ip_publica_bastion1>
[opc@bastion1 ~]$
```

For example:

```
$ chmod 400 atplab_rsa
$ ssh -i atplab_rsa opc@130.61.225.5
[opc@bastion1 ~]$
```

Once you have verified that you can connect from your local machine or from Cloud Shell to the **bastion1** machine, you have to pass the private key **[atplab\_rsa]** to that bastion1 machine, in the **.ssh** directory, in order to access the **atplabcli1** client machine later.



Back in Cloud Shell run the following command to transfer the ssh private key atplab\_rsa:

```
[opc@bastion1 ~]$exit  
$ scp -i atplab_rsa atplab_rsa opc@<bastion1_public_ip>:/home/opc/.ssh  
atplab_rsa 100% 1843 35.9KB/s 00:00  
$
```

Access the **atplabcli** machine from the **bastion1** machine via ssh. To get the private IP of the **atplabcli1** machine, go to the compute panel, instances menu and to the **atplabcli1** machine. (Currently, Private IP can be seen directly on the instances list.)

### Instances *in* atplab Compartmet

Create Instance						
Name	Status	Public IP	Shape	Availability Domain	Fault Domain	Created
bastion1	● Running	130.61.225.5	VM.Standard2.1	AD-1	FD-3	Wed, Apr 22, 2020, 07:03:02 UTC
atplabcli1	● Running	-	VM.Standard2.1	AD-1	FD-3	Wed, Apr 22, 2020, 07:03:02 UTC
Showing 2 Items					<	Page 1 >

In the primary VNIC section of the instance information you will be able to view the private IP.

### Primary VNIC

**Private IP Address:** 10.10.31.4

**Network Security Groups:** None [Edit](#)

**Internal FQDN:** atplabcli11... [Show](#) [Copy](#)

**Subnet:** [atplabsubnet](#)

To be able to access **atplabcli** through your Private IP, you must first change the permissions of the key file that you copied to the **bastion1** machine, with the command **chmod 400 atplab\_rsa**

```
# Jump to the bastion machine1  
$ ssh -i atplab_rsa opc@<public_ip_bastion1>  
  
# Jump to atplabcli machine  
[opc@bastion1 ~]$ chmod 600 .ssh/atplab_rsa  
[opc@bastion1 ~]$ ssh -i .ssh/atplab_rsa opc@<private_ip_atplabcli1>  
[opc@atplabcli1 ~]$
```

Once on the **atplabcli1** machine, you have to download the client software to access the ATP database.



The files are in zip format and to facilitate their download we have added the following **wget** commands that you will have to execute in **atplibcli1**.

```
 wget https://download.oracle.com/otn_software/linux/instantclient/199000/instantclient-basic-linux.x64-19.9.0.0.0dbru.zip
```

```
[opc@atplibcli1 ~]$ wget https://download.oracle.com/otn_software/linux/instantclient/199000/instantclient-basic-linux.x64-19.9.0.0.0dbru.zip
--2020-10-26 11:18:28-- https://download.oracle.com/otn_software/linux/instantclient/199000/instantclient-basic-linux.x64-19.9.0.0.0dbru.zip
Resolving download.oracle.com (download.oracle.com)... 172.227.12.117
Connecting to download.oracle.com (download.oracle.com)|172.227.12.117|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 75702778 (72M) [application/zip]
Saving to: 'instantclient-basic-linux.x64-19.9.0.0.0dbru.zip'

100%[=====] 75,782,778 220MB/s in 0.3s

2020-10-26 11:18:28 (220 MB/s) - 'instantclient-basic-linux.x64-19.9.0.0.0dbru.zip' saved [75702778/75702778]

[opc@atplibcli1 ~]$ ]
```

```
 wget https://download.oracle.com/otn_software/linux/instantclient/199000/instantclient-sqlplus-linux.x64-19.9.0.0.0dbru.zip
```

```
[opc@atplibcli1 ~]$ wget https://download.oracle.com/otn_software/linux/instantclient/199000/instantclient-sqlplus-linux.x64-19.9.0.0.0dbru.zip
--2020-10-26 11:19:56-- https://download.oracle.com/otn_software/linux/instantclient/199000/instantclient-sqlplus-linux.x64-19.9.0.0.0dbru.zip
Resolving download.oracle.com (download.oracle.com)... 172.227.12.117
Connecting to download.oracle.com (download.oracle.com)|172.227.12.117|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 911248 (896K) [application/zip]
Saving to: 'instantclient-sqlplus-linux.x64-19.9.0.0.0dbru.zip'

100%[=====] 911,248 --.-K/s in 0.05s

2020-10-26 11:19:57 (18.2 MB/s) - 'instantclient-sqlplus-linux.x64-19.9.0.0.0dbru.zip' saved [911248/911248]

[opc@atplibcli1 ~]$ ]
```

```
 wget https://download.oracle.com/otn_software/linux/instantclient/199000/instantclient-tools-linux.x64-19.9.0.0.0dbru.zip
```

```
[opc@atplibcli1 ~]$ wget https://download.oracle.com/otn_software/linux/instantclient/199000/instantclient-tools-linux.x64-19.9.0.0.0dbru.zip
--2020-10-26 11:21:39-- https://download.oracle.com/otn_software/linux/instantclient/199000/instantclient-tools-linux.x64-19.9.0.0.0dbru.zip
Resolving download.oracle.com (download.oracle.com)... 172.227.12.117
Connecting to download.oracle.com (download.oracle.com)|172.227.12.117|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1084110 (1.0M) [application/zip]
Saving to: 'instantclient-tools-linux.x64-19.9.0.0.0dbru.zip'

100%[=====] 1,084,110 --.-K/s in 0.09s

2020-10-26 11:21:39 (12.0 MB/s) - 'instantclient-tools-linux.x64-19.9.0.0.0dbru.zip' saved [1084110/1084110]

[opc@atplibcli1 ~]$ ]
```

You also have them available for manual download in different versions at the following URL: <https://www.oracle.com/database/technologies/instant-client/linux-x86-64-downloads.html>

Basic Client:



Version 19.9.0.0.0(Requires glibc 2.14)		
Base - one of these packages is required		
Name	Download	Description
Basic Package (ZIP)	<a href="#"> instantclient-basic-linux.x64-19.9.0.0.0dbru.zip</a>	All files required to run OCI, OCCI, and JDBC-OCI applications (75,702,778 bytes) (cksum - 1093795493)
Basic Package (RPM)	<a href="#"> oracle-instantclient19.9-basic-19.9.0.0-1.x86_64.rpm</a>	All files required to run OCI, OCCI, and JDBC-OCI applications (54,276,396 bytes) (cksum - 3150152145)

SQL Plus and also with the tools package:

Tools - optional packages		
Name	Download	Description
SQL*Plus Package (ZIP)	<a href="#"> instantclient-sqlplus-linux.x64-19.9.0.0.0dbru.zip</a>	The SQL*Plus command line tool for SQL and PL/SQL queries (911,248 bytes) (cksum - 4044886519)
SQL*Plus Package (RPM)	<a href="#"> oracle-instantclient19.9-sqlplus-19.9.0.0-1.x86_64.rpm</a>	The SQL*Plus command line tool for SQL and PL/SQL queries (702,872 bytes) (cksum - 2166935388)
Tools Package (ZIP)	<a href="#"> instantclient-tools-linux.x64-19.9.0.0.0dbru.zip</a>	Includes Data Pump, SQL*Loader and Workload Replay Client (1,084,110 bytes) (cksum - 893787797)
Tools Package (RPM)	<a href="#"> oracle-instantclient19.9-tools-19.9.0.0-1.x86_64.rpm</a>	Includes Data Pump, SQL*Loader and Workload Replay Client (835,796 bytes) (cksum - 2657462313)

Unzip all the packages you downloaded with the **unzip** command followed by the name of the downloaded file.

```
$ unzip instantclient-basic-linux.x64-19.9.0.0.0dbru.zip
$ unzip instantclient-sqlplus-linux.x64-19.9.0.0.0dbru.zip
$ unzip instantclient-tools-linux.x64-19.9.0.0.0dbru.zip
```

Then generate the wallet credential file of your ATP. You can use two options:

If you are using Cloud Shell, you can use the OCI CLI to download the configuration file using command line ([https://docs.cloud.oracle.com/en-us/iaas/tools/oci-cli/2.10.3/oci\\_cli\\_docs/cmdref/db/autonomous-database/generate-wallet.html](https://docs.cloud.oracle.com/en-us/iaas/tools/oci-cli/2.10.3/oci_cli_docs/cmdref/db/autonomous-database/generate-wallet.html)):

```
Cloud Shell Machine $ oci db autonomous-database generate-wallet --autonomous-database-id <ATP_OCID> --file <nombre_fichero_wallet>.zip --password <wallet_password>
```

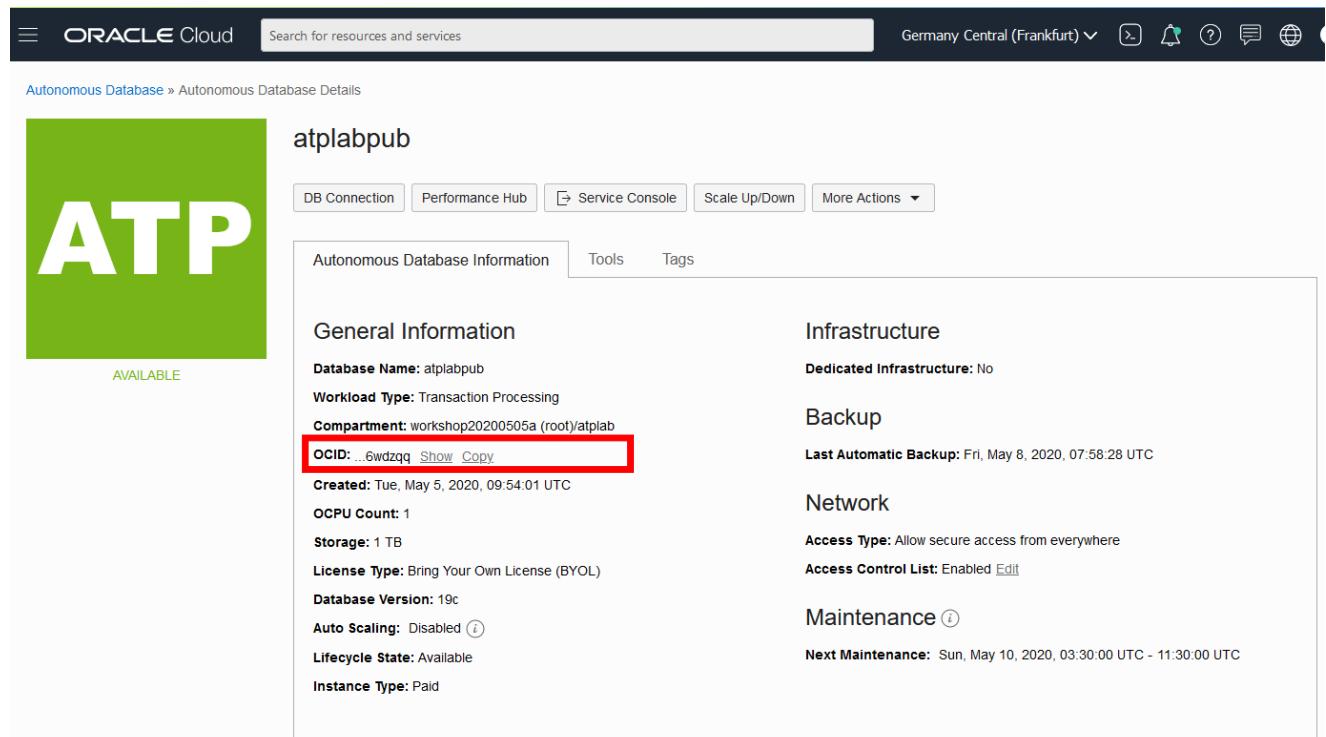
**ATP\_OCID**

OCID of your ATP DATABASE

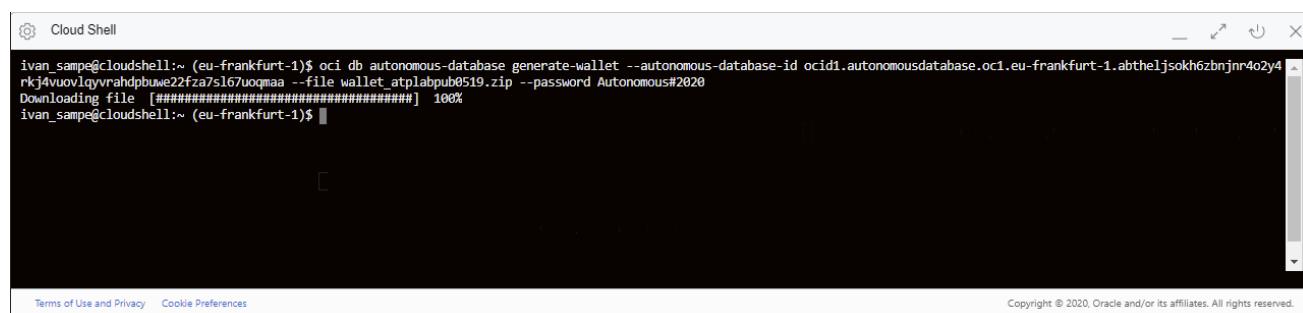


<b>NOMBRE_FICHERO_WALLET</b>	wallet_<atp_name>.zip EX: wallet_atplabpub.zip
<b>PASSWORD</b>	Autonomous#2020

The **OCID of your ATP database** can be found on the information screen of your ATP, accessing the main menu of OCI (hamburger icon), **Oracle Database -> Autonomous Transaction Processing** and selecting your ATP database.



The screenshot shows the Oracle Cloud interface for an Autonomous Database named 'atplabpub'. The OCID field, which contains the value 'oci1.autonomousdatabase.oc1.eu-frankfurt-1.abthe1jsokh6zbnjnr4o2y4...', is highlighted with a red box. Other visible details include the database name, workload type (Transaction Processing), compartment, creation date (Tue, May 5, 2020, 09:54:01 UTC), and various infrastructure and maintenance settings.

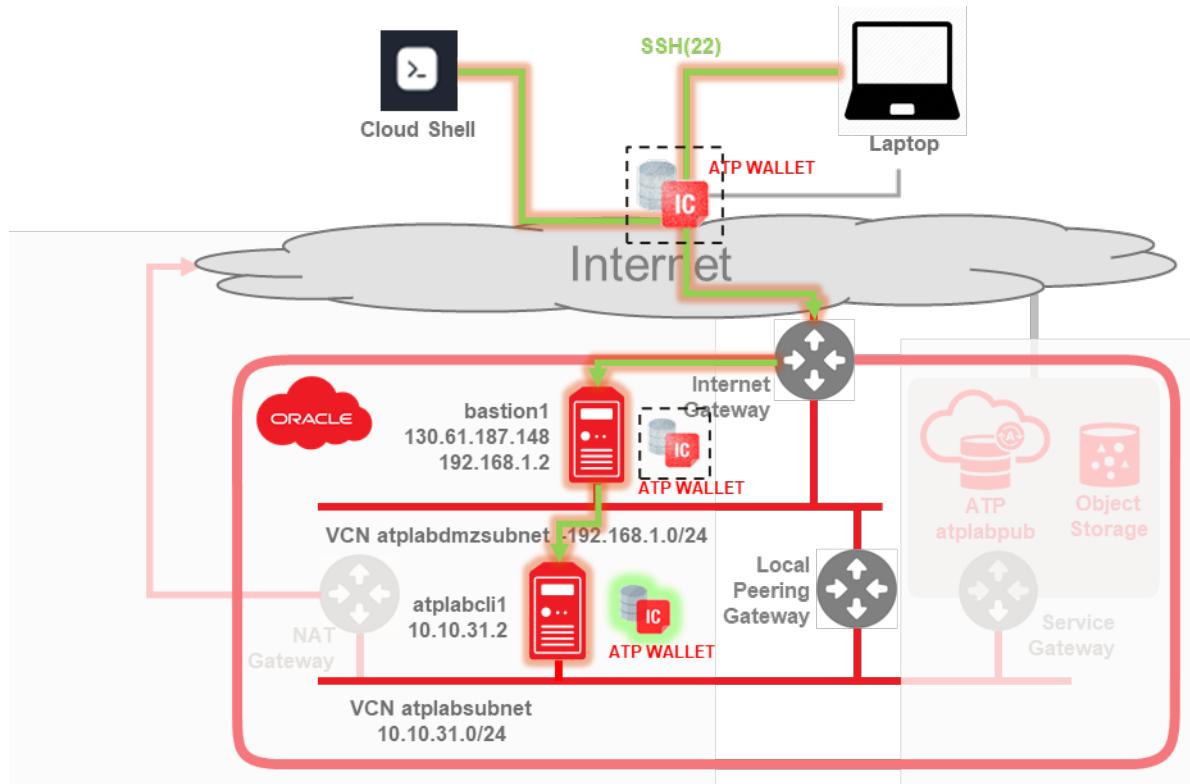


```
Cloud Shell
ivan_sampe@cloudshell:~ (eu-frankfurt-1)$ oci db autonomous-database generate-wallet --autonomous-database-id oci1.autonomousdatabase.oc1.eu-frankfurt-1.abthe1jsokh6zbnjnr4o2y4...
rkj4vuvolgyvrhdpbuwe22fza7s167uoqmaa --file wallet_atplabpub0519.zip --password Autonomous#2020
Downloading file [########################################] 100%
ivan_sampe@cloudshell:~ (eu-frankfurt-1)$
```

If you are using your local machine (laptop or desktop), generate and download the wallet file as indicated in the section **How to generate and download the credential file from the ATP administration console**, of this manual.

Once generated and downloaded copy it first to the **bastion1** server, and then to the **atplabcli1** client server.





```
Laptop/CloudShell$ scp -i atplab_rsa wallet_<ATP_NAME>.zip opc@<public_ip_bastion1>/home/opc/
wallet_<ATP_NAME>.zip
100% 20KB 143.0KB/s 00:00
```

```
Laptop/CloudShell$ ssh -i atplab_rsa opc@<public_ip_bastion1>
```

```
[opc@bastion1 ~]$ scp -i .ssh/atplab_rsa wallet_<ATP_NAME>.zip opc@<private_ip_atplab cli1>/home/opc/
wallet_<ATP_NAME>.zip
100% 20KB 8.6MB/s 00:00
```

```
[opc@bastion1 ~]$ ssh -i .ssh/atplab_rsa opc@<private_ip_atplab cli1>
Last login: Tue Apr 28 12:16:47 2020 from 192.168.1.3
```

Modify the file **.bash\_profile** on the **atplabcli1** server to make it look as shown in the following example:

```
[opc@atplabcli1 ~]$ cat .bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

ORACLE_HOME=$HOME/instantclient_19_9
export ORACLE_HOME
LD_LIBRARY_PATH=$ORACLE_HOME
export LD_LIBRARY_PATH
TNS_ADMIN=$ORACLE_HOME/network/admin
export TNS_ADMIN
PATH=$PATH:$HOME/.local/bin:$HOME/bin:$ORACLE_HOME

export PATH
```



To make the changes effective, load the environment

```
[opc@atplabcli1 ~]$ . .bash_profile
```

To verify that the changes have been made, you can run the following command:

```
[opc@atplabcli1 ~]$ echo $TNS_ADMIN  
/home/opc/instantclient_19_9/network/admin
```

Move the wallet file to the client's TNS\_ADMIN directory and unzip it there.

```
[opc@atplabcli1 ~]$ mv wallet_<ATP_NAME>.zip $TNS_ADMIN  
[opc@atplabcli1 ~]$ cd $TNS_ADMIN  
[opc@atplabcli1 admin] $ unzip wallet_<ATP_NAME>.zip
```

```
[opc@atplabcli1 admin]$ unzip wallet_atplabpub.zip  
Archive: wallet_atplabpub.zip  
replace README? [y]es, [n]o, [A]ll, [N]one, [r]ename: y  
  inflating: README  
  inflating: cwallet.sso  
  inflating: tnsnames.ora  
  inflating: truststore.jks  
  inflating: ojdbc.properties  
  inflating: sqlnet.ora  
  inflating: ewallet.p12  
  inflating: keystore.jks  
[opc@atplabcli1 admin]$
```

Once the environment is loaded and the wallet has been unzipped within the Oracle client, you should be able to connect to **atp** (if you have put the name indicated in the laboratory it will be **atplabpub**) from the **atplabcli1** client machine.

When connecting by SQLPlus, the suffix **[\_medium]** with the name of the ATP is used, but other connection suffixes could be used. For more information you can consult the following link: <https://docs.oracle.com/en/cloud/paas/autonomous-database/adbsa/connect-predefined-generic.html#GUID-E49773B3-6C07-4F6F-906B-42705D237523>

<b>ATP_NAME</b>	atplabpub
-----------------	-----------

```
[opc@atplabcli11 ~]$ sqlplus admin/Autonomous#2020@<ATP_NAME>_medium
```

SQL\*Plus: Release 19.0.0.0.0 - Production on Tue Apr 28 12:40:51 2021  
Version 19. 9.0.0.0

Copyright (c) 1982, 2020, Oracle. All rights reserved.



Last Successful login time: Wed Apr 22 2021 16:44:02 +00:00

Connected to:  
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production  
Version 19.5.0.0.0

SQL>

Once this is done, you will have to *import data into the ATP database*. To do this go to the Service Console menu of your **ATP -> Service Console -> Development**.

Overview » Autonomous Database » Autonomous Database Details

The screenshot shows the Oracle Cloud Infrastructure console for an Autonomous Transaction Processing database named 'atplabpub'. The 'Service Console' tab is selected and highlighted with a red box. Below the tabs, there are three categories: 'Autonomous Database Information', 'Tools', and 'Tags'. Under 'Tools', there is a section titled 'Database Actions' with a callout pointing to the 'Importing Data Using Oracle Data Pump' link.

Go to the quick link of the documentation that tells us how to import data

The screenshot shows the Oracle Autonomous Transaction Processing documentation page. The 'Development' section is selected. A red box highlights the 'Importing Data Using Oracle Data Pump' link under the 'Download Oracle Instant Client' section. Other sections visible include 'SQL Developer Web' and 'Oracle Machine Learning Notebooks'.

Now we go to the part of the documentation that tells us the **impdp** command that we must execute to be able to import data from object storage.

2. Run Data Pump Import with the *dumpfile* parameter set to the list of file URLs on your Cloud Object Storage and the *credential* parameter set to the name of the credential you created in the previous step. For example:

```
impdp admin/password@ATPG1_high \
directory=data_pump_dir \
credential=def_cred_name \
dumpfile=https://objectstorage.us-ashburn-1.oraclecloud.com/n/namespace-string/b/bucketname/o/export \
parallel=16 \
encryption_pwd_prompt=yes \
transform=segment_attributes:n \
transform=dwcs_cvt_iots:y transform=constraint_use_default_index:y \
exclude=cluster,db_link
```



The command would be as follows and would have to be executed from the machine on the private network **[atplabcli1]**. Replace the parts marked in red with your own values to launch the command correctly.

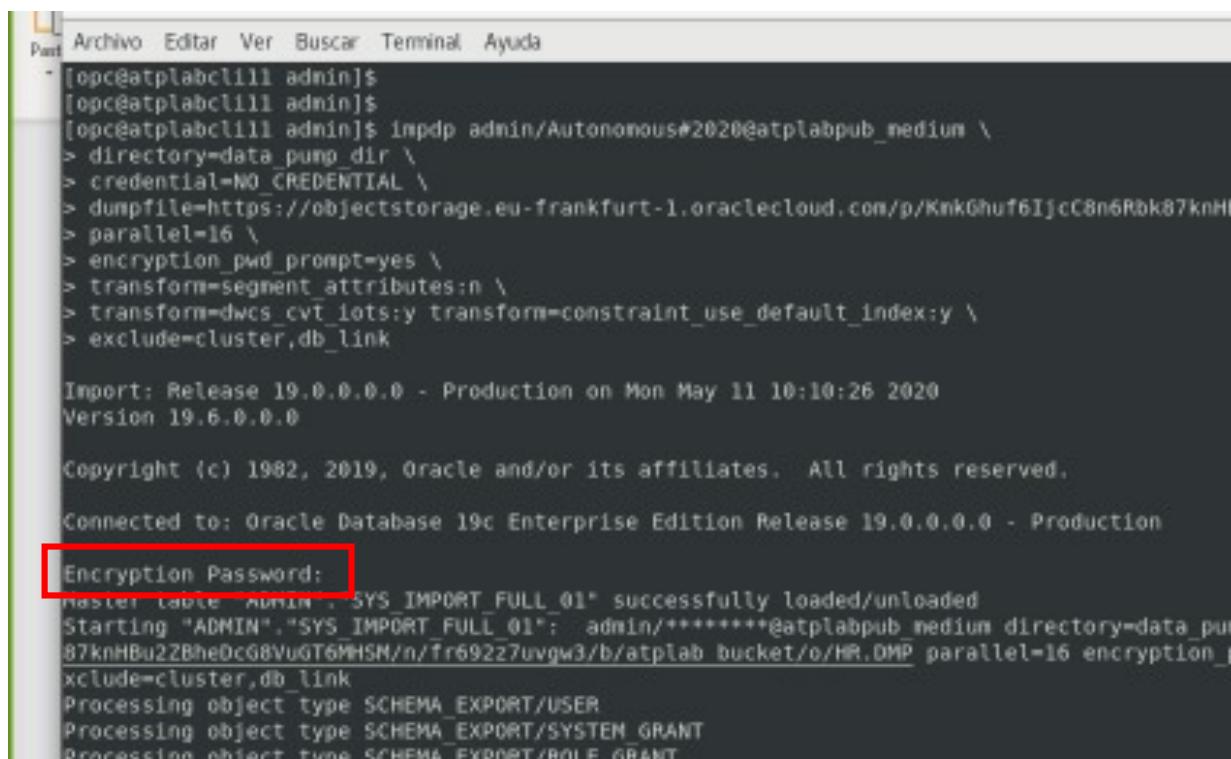


Dumpfile corresponds to the **pre-authenticated key** that you created earlier with the **HR.DMP** file. Be careful when replacing the dumpfile variable and putting (\) at the end of the pre-authenticated key you created earlier in the Object Storage section.

<b>Password</b>	Autonomous#2020
<b>ATP_NAME</b>	atplabpub
<b>Pre-authenticated_key</b>	Preatuthenticated key previously created in your bucket for the HR file.DMP

```
impdp admin/password@<ATP_NAME>_medium \
directory=data_pump_dir \
credential=NO_CREDENTIAL \
dumpfile=preatuthenticated-key-hr.dmp \
parallel=16 \
encryption_pwd_prompt=yes \
transform=segment_attributes:n \
transform=dwcs_cvt_iots:y \
transform=constraint_use_default_index:y \
exclude=cluster,db_link
```

When asked for the encryption key (*Encryption password*) you must enter the key you input for your **wallet** file (if you used the one indicated by the laboratory it should be **Autonomous#2020**).



```
[opc@atplabcllll admin]$ [opc@atplabcllll admin]$ [opc@atplabcllll admin]$ impdp admin/Autonomous#2020@atplabpub_medium \
> directory=data_pump_dir \
> credential=NO_CREDENTIAL \
> dumpfile=https://objectstorage.eu-frankfurt-1.oraclecloud.com/p/KnkGhuf6Ijcc8n6Rbk87knHB
> parallel=16 \
> encryption_pwd_prompt=yes \
> transform=segment_attributes:n \
> transform=dwcs_cvt_iots:y transform=constraint_use_default_index:y \
> exclude=cluster,db_link

Import: Release 19.0.0.0.0 - Production on Mon May 11 10:10:26 2020
Version 19.6.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.

Connected to: Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production

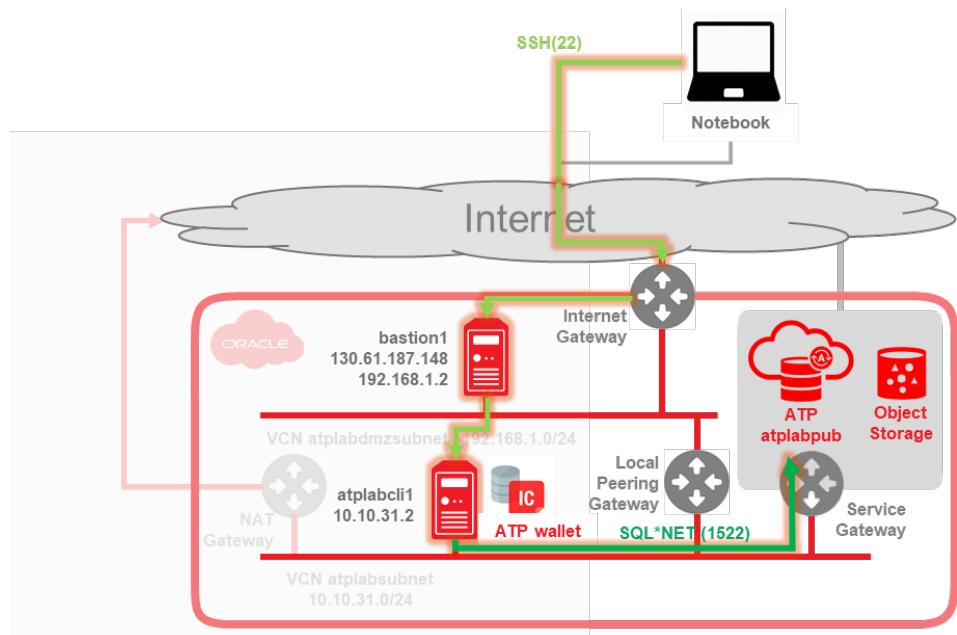
Encryption Password:
```



# Check the connection with the Autonomous Database (ATP).

## What am I going to do?

You will make the first connection against the ATP from the atplabcli server by launching SQL commands from SQLPlus.



Once the data is imported, we can connect to the ATP, and start querying data by launching the following SQL commands after logging in with SQLPlus and user **hr**.

```
[opc@atplabcli11 ~]$ sqlplus hr/hr@<ATP_NAME>_medium  
SQL*Plus: Release 19.0.0.0.0 - Production on Tue Apr 28 12:48:02 2020  
Version 19. 9.0.0.0  
Copyright (c) 1982, 2019, Oracle. All rights reserved.  
Last Successful login time: Mon Apr 27 2020 18:32:54 +00:00  
Connected to:  
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production  
Version 19.5.0.0.0
```

```
select * from employees where rownum < 2;
```

```
SQL> select * from employees where rownum < 2;  
EMPLOYEE_ID FIRST_NAME LAST_NAME  
-----  
EMAIL PHONE_NUMBER HIRE_DATE JOB_ID SALARY  
-----  
COMMISSION_PCT MANAGER_ID DEPARTMENT_ID  
-----  
100 Steven King  
SKING 515.123.4567 17-JUN-03 AD_PRES 24000  
90
```



```
select json_object(*) from employees where rownum < 2;
```

```
SQL> select json_object(*) from employees where rownum < 2;
```

```
JSON_OBJECT(*)
```

```
{"EMPLOYEE_ID":100,"FIRST_NAME":"Steven","LAST_NAME":"King","EMAIL":"SKING","PHONE_NUMBER":"515.123.4567","HIRE_DATE":"2003-06-17T00:00:00","JOB_ID":"AD_PRES","SALARY":24000,"COMMISSION_PCT":null,"MANAGER_ID":null,"DEPARTMENT_ID":90}
```

Finally, create a table as a select from the lineorder table. **This table will be used later in lab 4.**

```
create table lineorder as select * from ssb.lineorder where to_char(lo_orderdate,'YYYY') = '1994';
```

This operation may take several minutes, you can continue in the meantime with the next laboratory.



# Lab Summary

1. In this lab, you provisioned and configured an **ATP standalone** database in the Oracle Cloud Infrastructure (OCI) console.
2. That ATP has been loaded with information from an HR.DMP file that you have previously uploaded to a bucket within OCI **Object Storage**.
3. You have created a **Nat Gateway** to be able to have an internet connection from the private network where the atplabcli machine is.
4. You have created a **Service Gateway** to be able to connect from the private network where atplabcli is to the Oracle Cloud SaaS services network.
5. You have created the necessary **rules** in the **routing tables** to be able to direct traffic between the machines and the internet and the machines and the OCI service network.
6. You have used **Cloud Shell** to execute operations for access, upload, and download of files.
7. You have downloaded the client software from the internet with **wget**, to be able to access the ATP from the client machine on the private network. And you've unzipped the software with **unzip**.
8. You have verified that the **SQLPlus** tool has access to atm by launching several **SQL commands** from it.
9. You have **created a new table** in the ATP for testing in the rest of the laboratories.

