

Operating System Management

# **Assignment-1**

**Nectar**

**Name: Elvan Alandi**

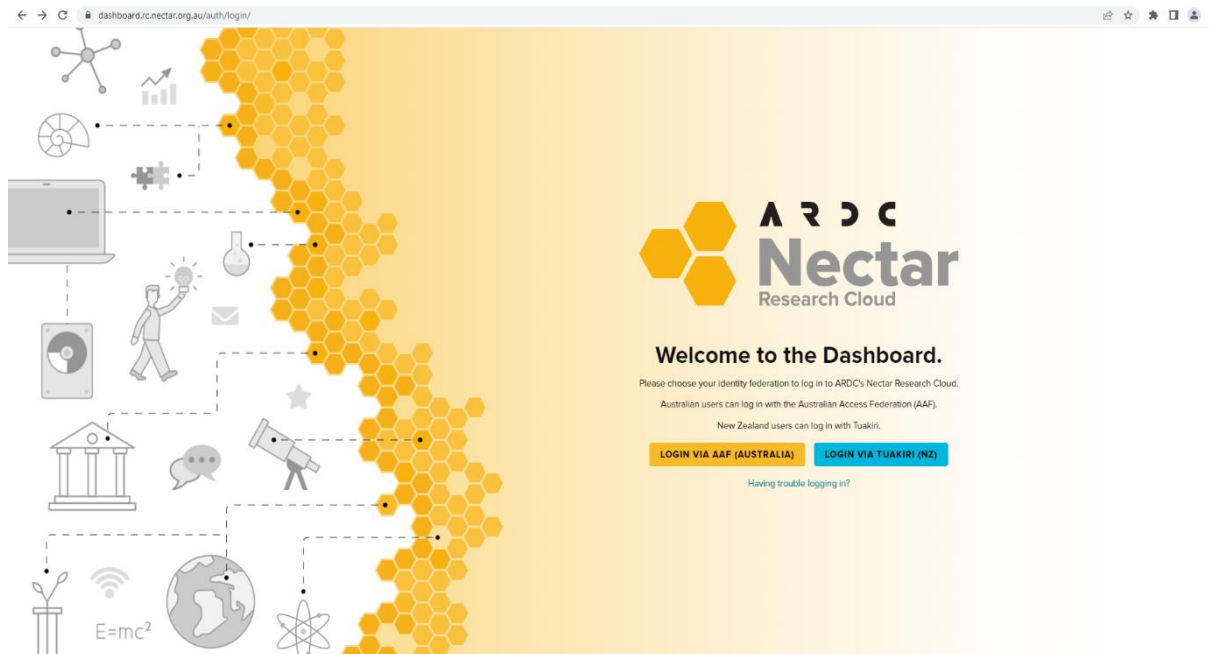
# Table of Contents

<b>Task 1. Nectar Walkthrough.....</b>	<b>3</b>
<b>1.1 Login to Nectar .....</b>	<b>3</b>
<b>1.2 Setup SSH Keys .....</b>	<b>5</b>
<b>1.2.1 Windows Users - PuTTY .....</b>	<b>5</b>
<b>1.2.2 Using Nectar Public Key (Alternative) .....</b>	<b>7</b>
<b>1.3 Creating Virtual Machine.....</b>	<b>8</b>
<b>1.4 Ping and Login .....</b>	<b>11</b>
<b>1.4.1 Ping.....</b>	<b>11</b>
<b>1.4.2 Login using Windows PuTTY.....</b>	<b>12</b>
<b>Task 2. Add Web server, DB server, and Website within an LXC Container.....</b>	<b>15</b>
<b>2.1 Adding LXC Web server .....</b>	<b>15</b>
<b>2.1.1 Login to Nectar.....</b>	<b>15</b>
<b>2.1.2 Update and upgrade packages .....</b>	<b>15</b>
<b>2.1.3 LXC and LXD .....</b>	<b>15</b>
<b>2.2 Adding Database Server .....</b>	<b>17</b>
<b>2.2.1 Install MySQL server.....</b>	<b>17</b>
<b>2.2.2 Open MySQL .....</b>	<b>17</b>
<b>2.2.3 Create root password .....</b>	<b>17</b>
<b>2.2.4 Login again using root as user .....</b>	<b>17</b>
<b>2.2.5 Granting the root user privileges.....</b>	<b>17</b>
<b>2.2.6 Create database.....</b>	<b>17</b>
<b>2.2.7 Create table for the website .....</b>	<b>17</b>
<b>2.3 Adding Simple Functional Website .....</b>	<b>18</b>
<b>2.3.1 Install PHP .....</b>	<b>18</b>
<b>2.3.2 Install php-mysqli to connect mysqli with php.....</b>	<b>18</b>
<b>2.3.3 Create an index.php file in /var/www/html folder.....</b>	<b>18</b>
<b>2.3.4 Test the web page .....</b>	<b>19</b>
<b>References .....</b>	<b>21</b>

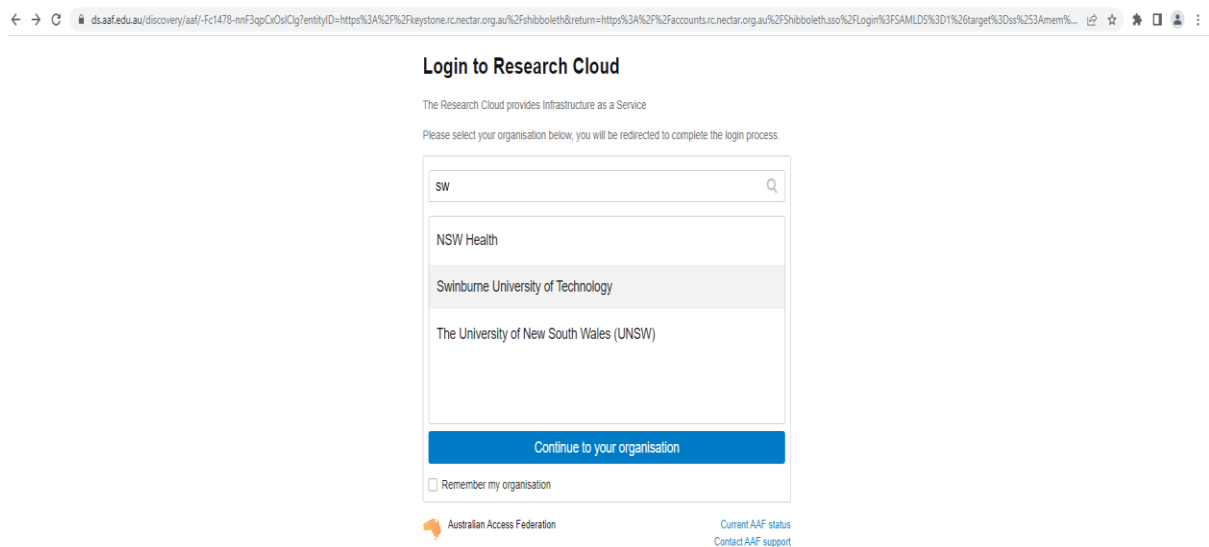
# Task 1. Nectar Walkthrough

## 1.1 Login to Nectar

1. Go to web browser and navigate to <https://dashboard.rc.nectar.org.au/auth/login>
2. Click the “LOGIN VIA AAF (AUSTRALIA)”



3. Search for your organisation (My organisation is Swinburne University of Technology)



4. Login using organisation's login credential (I have logged in before and I just need click on the approve button)

idp.ocawim.edu.au/idp/profile/SAML2/Redirect/SSO?execution=el2a2

SWINBURNE UNIVERSITY OF TECHNOLOGY

You are about to access the service:  
Research Cloud of unimelb.edu.au

This service will validate the attributes provided about you from your identity provider ensuring they are provided in a format that is suitable for consumption by AAF connected services.

Information to be Provided to Service:	
Display name	ELVAN ALANDI
Affiliation	member student
Given name	ELVAN
Home organization	swin.edu.au
Home organization type	
E-mail	
Organization name	SWINBURNE UNIVERSITY OF TECHNOLOGY
Surname	ALANDI

Do you agree to release this information to the service every time you access it?

☐ Ask me again at next login

☒ Ask me again if information to be provided to this service changes

☐ Do not ask me again

Approve

Reject

5. Login into nectar by aggregating the terms and conditions

Terms and Conditions

accounts.rc.nectar.org.au/login/

d. must notify Nectar Research Cloud support immediately if they notice any breach of security or unauthorised use of their allocation or user account.

3. Attribution

Users must

1. attribute ownership of the Service to ARDC when using the Service in accordance with [ARDC's Attribution and Acknowledgement Guidelines](#), and
2. do all things reasonably necessary to preserve the integrity and security of the Service and to protect the Service from any modification, delacement or disparaging treatment.

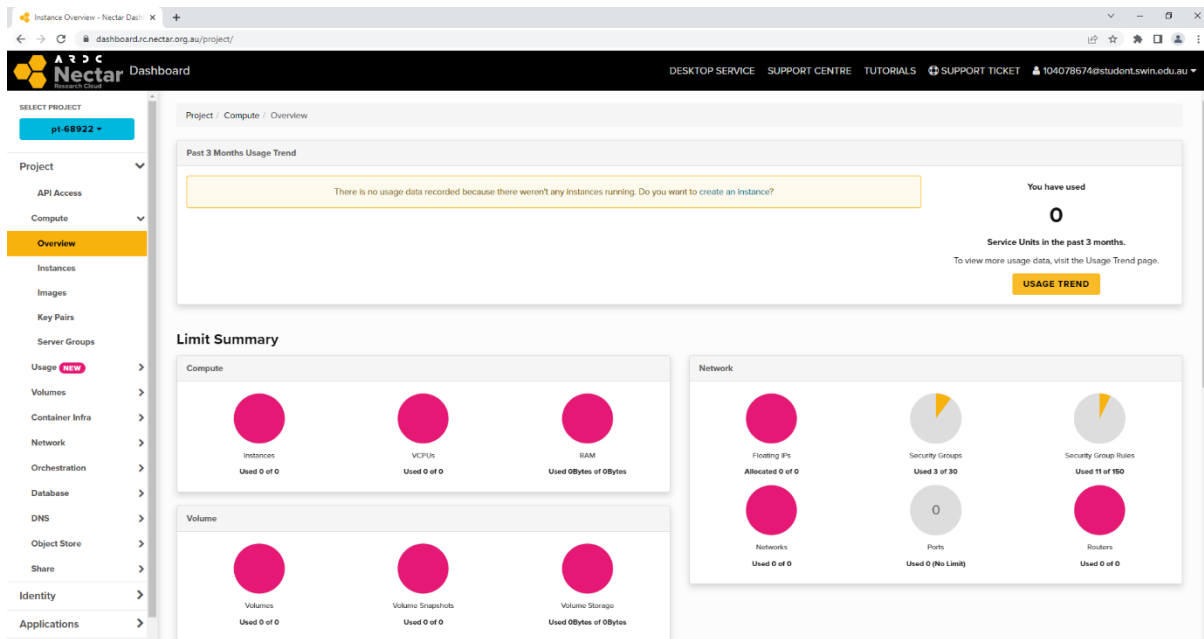
4. Expiry, Suspension, or Termination

1. Any data, virtual environments or other material created or stored in the Service by Users will be deleted 3 months after the expiry of the Term.
2. ARDC and its Node partners reserve the right to monitor, search for, and investigate violations to the Terms of Service and may remove, suspend, or terminate access if there are actions that violate the Terms of Service.
3. At any time prior to the expiry of the Term, ARDC may:
  - a. limit, attenuate, suspend or terminate Users' use of the Service at its discretion and without notice, or
  - b. immediately remove any material on the Service (including any virtual environment Users create or contribute to) in any of the following circumstances:
    - i. inappropriate use of the Services (including any use that breaches this agreement);
    - ii. a detected vulnerability or compromise in the security of the Services;
    - iii. a request or direction by a law enforcement agency, or
    - iv. ARDC reasonably determines that there is an operational need for the content to be removed.

Agree

Disagree

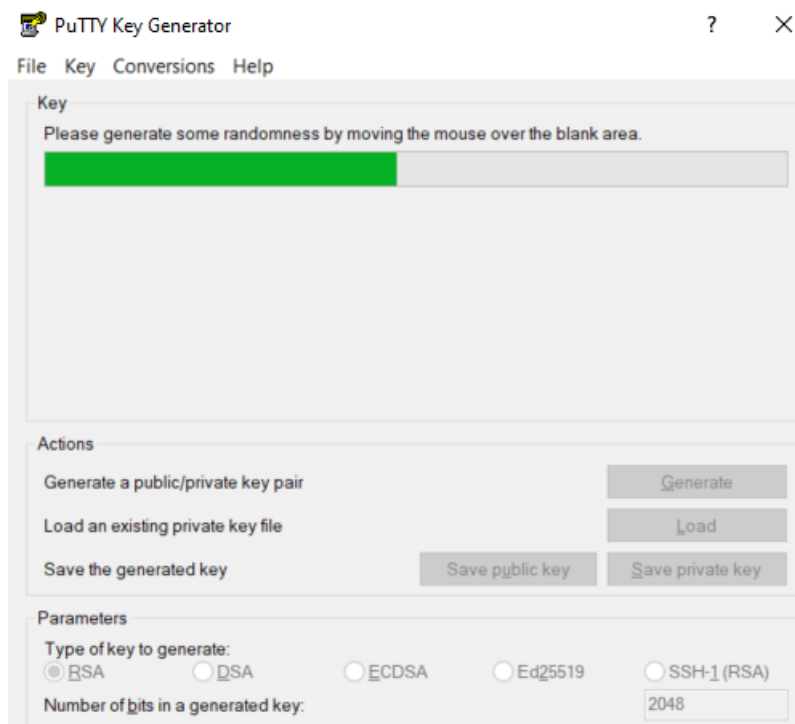
6. After that, you will see the Nectar Dashboard page



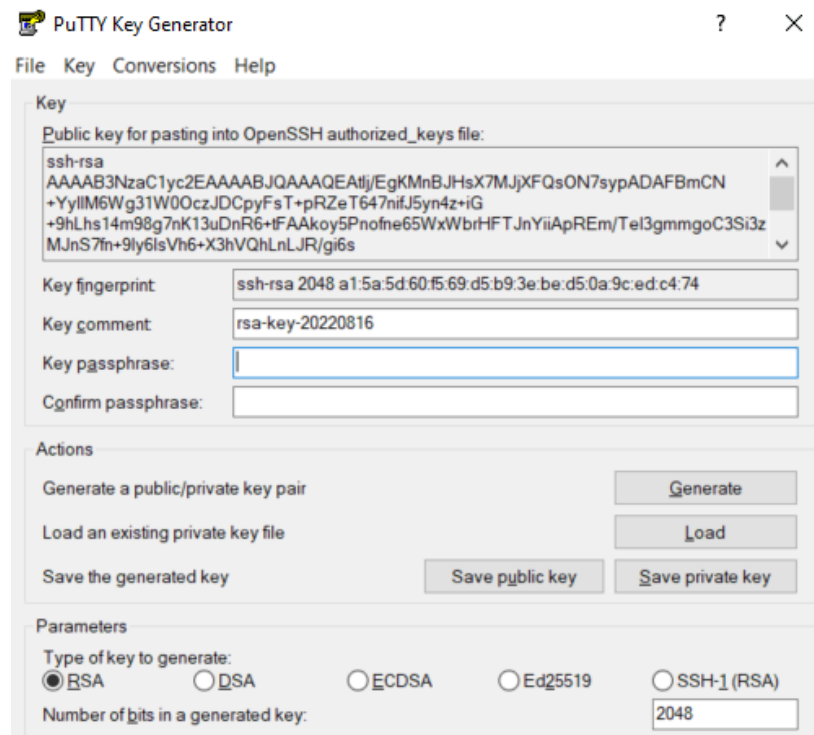
## 1.2 Setup SSH Keys

### 1.2.1 Windows Users - PuTTY

1. Launch PuttyGen.exe from the Start search field or using Win + R
2. Click on Generate button and wait until the process has been finished



3. If the process has been finished, it will be looked like the image below. Copy the rsa key into your clipboard and paste it into the temporary file (to be pasted into the text field in the Nectar interface)



The screenshot shows the PuTTY Key Generator window. The 'Key' section displays the public key for pasting into the OpenSSH authorized\_keys file, the key fingerprint, the key comment, and the key passphrase. The 'Actions' section includes buttons for 'Generate', 'Load', 'Save public key', and 'Save private key'. The 'Parameters' section shows the type of key to generate (RSA) and the number of bits in a generated key (2048).

Public key for pasting into OpenSSH authorized\_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAQEAtlj/EgKMnBJHsX7MjXfQsON7syADAfBmCN
+YyIlM6Wg31W0OczJDCpyFsT+pRZeT647nifJ5yn4z+iG
+9hLhs14m98g7nK13uDnR6+FAAkoy5Pnofne65WxWbrHFTJnYiApREm/Tel3gmmgoC3Si3z
MJnS7fn+9ly6isVh6+X3hVQhLnLJR/gi6s
```

Key fingerprint: ssh-rsa 2048 a1:5a:5d:60:f5:69:d5:b9:3e:be:d5:0a:9c:ed:c4:74

Key comment: rsa-key-20220816

Key passphrase:

Confirm passphrase:

Actions

Generate a public/private key pair

Load an existing private key file

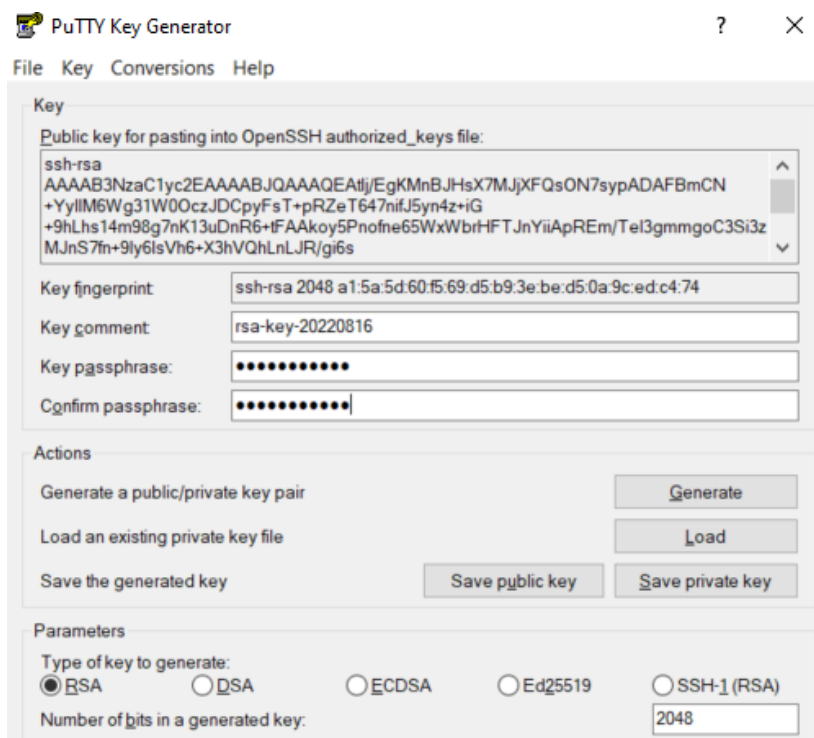
Save the generated key

Parameters

Type of key to generate: ☒ RSA ☐ DSA ☐ ECDSA ☐ Ed25519 ☐ SSH-1 (RSA)

Number of bits in a generated key: 2048

4. For a security reason, fill in the key passphrase before saving the public key and private key to the local drive.



The screenshot shows the PuTTY Key Generator window. The 'Key' section displays the public key for pasting into the OpenSSH authorized\_keys file, the key fingerprint, the key comment, and the key passphrase. The 'Actions' section includes buttons for 'Generate', 'Load', 'Save public key', and 'Save private key'. The 'Parameters' section shows the type of key to generate (RSA) and the number of bits in a generated key (2048).

Public key for pasting into OpenSSH authorized\_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAQEAtlj/EgKMnBJHsX7MjXfQsON7syADAfBmCN
+YyIlM6Wg31W0OczJDCpyFsT+pRZeT647nifJ5yn4z+iG
+9hLhs14m98g7nK13uDnR6+FAAkoy5Pnofne65WxWbrHFTJnYiApREm/Tel3gmmgoC3Si3z
MJnS7fn+9ly6isVh6+X3hVQhLnLJR/gi6s
```

Key fingerprint: ssh-rsa 2048 a1:5a:5d:60:f5:69:d5:b9:3e:be:d5:0a:9c:ed:c4:74

Key comment: rsa-key-20220816

Key passphrase: .....

Confirm passphrase: .....

Actions

Generate a public/private key pair

Load an existing private key file

Save the generated key

Parameters

Type of key to generate: ☒ RSA ☐ DSA ☐ ECDSA ☐ Ed25519 ☐ SSH-1 (RSA)

Number of bits in a generated key: 2048

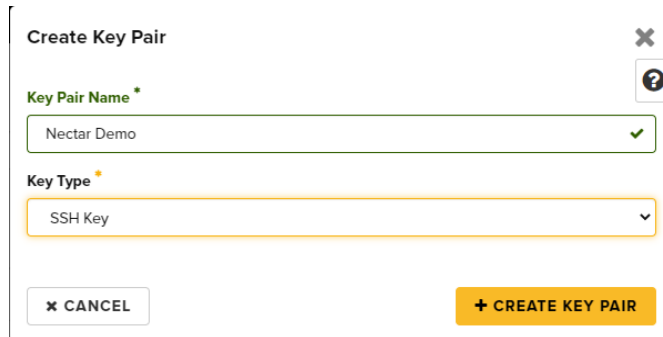
5. Save the public key and the private key to safe places on computer

 privkey	16/08/2022 17:18	PuTTY Private Key ...	2 KB
 pubkey.pub	16/08/2022 17:18	PUB File	1 KB

### 1.2.2 Using Nectar Public Key (Alternative)

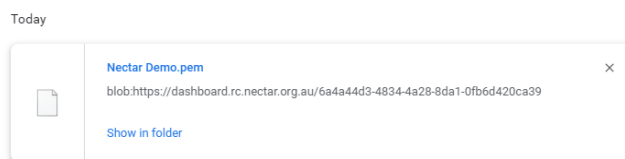
1. Creating PEM file using Nectar:

- Go to Key Pairs tab
- Click on the +Create Key Pair button
- Input Key Pair Name and choose Key Type to SSH Key
- Click +Create Key Pair button in the popup box



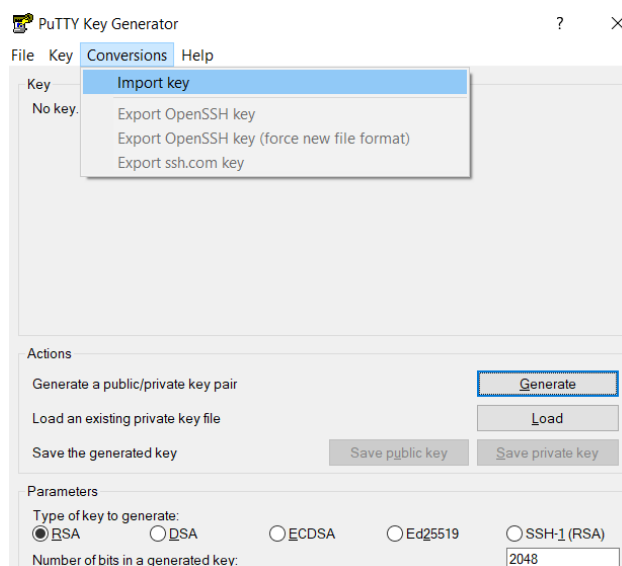
The 'Create Key Pair' dialog box is shown. It has a title bar with a close button (X) and a help button (?). The 'Key Pair Name' field contains 'Nectar Demo' with a green checkmark. The 'Key Type' dropdown menu is set to 'SSH Key'. At the bottom, there are two buttons: 'CANCEL' and '+ CREATE KEY PAIR'.

2. It will automatically download the .pem file like in the image below, save the .pem file to a safe place

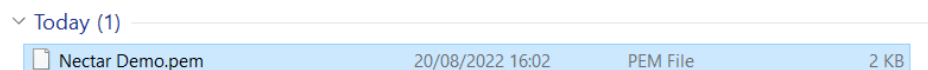


3. Open PuttyGen.exe

4. From the menu select Conversions → Import key



5. Choose the .pem file that has been downloaded before



6. Repeat point 1.2.1 step 3, 4, and 5

## 1.3 Creating Virtual Machine

1. Go back to the NectarCloud Dashboard. On the left side panel click “Compute” and then click “images”
2. Select NeCTAR Ubuntu 18.04 LTS (Bionic) amd64 by clicking on the box (There is Ubuntu version 22.04 but I am using Ubuntu version 18.04 LTS (Bionic) amd64, it is up to your choice) and click Launch on the right side of the image you were selected

<input type="checkbox"/>	> NeCTAR R-Studio (Ubuntu 18.04 LTS Bionic)	Image	Active	Public	No	QCOW2	2.89 GB	LAUNCH
<input type="checkbox"/>	> NeCTAR R-Studio (Ubuntu 20.04 LTS Focal)	Image	Active	Public	No	QCOW2	3.06 GB	LAUNCH
<input type="checkbox"/>	> NeCTAR Rocky Linux 8 x86_64	Image	Active	Public	No	QCOW2	911.94 MB	LAUNCH
<input type="checkbox"/>	> NeCTAR Rocky Linux 9 x86_64	Image	Active	Public	No	QCOW2	732.38 MB	LAUNCH
<input checked="" type="checkbox"/>	> NeCTAR Ubuntu 18.04 LTS (Bionic) amd64	Image	Active	Public	No	QCOW2	389.25 MB	LAUNCH
<input type="checkbox"/>	> NeCTAR Ubuntu 18.04 LTS (Bionic) amd64 (with Docker)	Image	Active	Public	No	QCOW2	451.66 MB	LAUNCH
<input type="checkbox"/>	> NeCTAR Ubuntu 20.04 LTS (Focal) amd64	Image	Active	Public	No	QCOW2	381.61 MB	LAUNCH
<input type="checkbox"/>	> NeCTAR Ubuntu 20.04 LTS (Focal) amd64 (NVIDIA vGPU)	Image	Active	Public	No	QCOW2	916.86 MB	LAUNCH
<input type="checkbox"/>	> NeCTAR Ubuntu 22.04 LTS (Jammy) amd64	Image	Active	Public	No	QCOW2	1.23 GB	LAUNCH

3. Give your machine an appropriate name and description. Leave the default availability zone and count

### Launch Instance

Details

Source

Flavor\*

Networks

Security Groups

Key Pair

Configuration

Server Groups

Metadata

Please provide the initial hostname for the instance, the availability zone where it will be deployed, and the instance count. Increase the Count to create multiple instances with the same settings.

Project Name

pt-68922

Instance Name\*

Nectar Demo

Description

Availability Zone

Any Availability Zone

Count\*

1

Total Instances (2 Max)

50%

0 Current Usage

1 Added

1 Remaining

CANCEL

BACK

NEXT

LAUNCH INSTANCE

4. In the Flavor\* tab, click on the up arrow (number 1 in the image) near **m3.small** (it already moved to allocation section) and it will move to the allocation section (number 2 in the image). A standard Nectar account can use up to 8GB of RAM and 2 virtual CPUs



## Launch Instance

[Details](#)[Source](#)[Flavor](#)[Networks](#)[Security Groups](#)[Key Pair](#)[Configuration](#)[Server Groups](#)[Metadata](#)

Flavors manage the sizing for the compute, memory and storage capacity of the instance.

### Allocated

Name	VCPUS	RAM	Root Disk	Ephemeral Disk	Public	SU/hour	
> m3.small	2	4 GB	30 GB	0 GB	Yes	0.057	2

### Available

Name	VCPUS	RAM	Root Disk	Ephemeral Disk	Public	SU/hour	
> t3.xsmall	1	1 GB	10 GB	0 GB	Yes	0.014	1
> t3.small	2	2 GB	10 GB	0 GB	Yes	0.029	↑
> m3.xsmall	1	2 GB	30 GB	0 GB	Yes	0.029	↑
> t3.medium	4	4 GB	10 GB	0 GB	Yes	0.057	↑
> m3.medium	4	8 GB	30 GB	0 GB	Yes	0.114	↑
> m3.large	8	16 GB	30 GB	0 GB	Yes	0.228	↑
> m3.xlarge	16	32 GB	30 GB	0 GB	Yes	0.457	↑
> m3.xlarge	32	64 GB	30 GB	0 GB	Yes	0.913	↑

[✕ CANCEL](#)[< BACK](#)[NEXT >](#)[LAUNCH INSTANCE](#)

- Click the Key Pair tab and click Import Key Pair (because we do not have allocated key)

## Launch Instance

[Details](#)[Source](#)[Flavor](#)[Networks](#)[Security Groups](#)[Key Pair](#)[Configuration](#)[Server Groups](#)[Metadata](#)

A key pair allows you to SSH into your newly created instance. You may select an existing key pair, import a key pair, or generate a new key pair.

[+ CREATE KEY PAIR](#)[IMPORT KEY PAIR](#)

### Allocated

Displaying 0 items

Name	Type	Fingerprint
Select a key pair from the available key pairs below.		

Displaying 0 items

### Available

Select one

Click here for filters or full text search.

Displaying 0 items

Name	Type	Fingerprint
No items to display.		

Displaying 0 items

☐ Set admin password

[✕ CANCEL](#)[< BACK](#)[NEXT >](#)[LAUNCH INSTANCE](#)

- Give your Key Pair name and choose SSH Keys for the Key Type. After that, copy your SSH RSA key that has been generated and has been kept in the temporary file to Public Key text area, then click on the import key pair button.

### Import Key Pair

Key Pairs are how you login to your instance after it is launched. Choose a key pair name you will recognize and paste your SSH public key into the space provided.

**Key Pair Name**

**Key Type**

SSH Key

**Load Public Key from a file**

[Choose File](#) No file chosen

**Public Key** (Modified) Content size: 397 bytes of 16.00 KB

ssh-rsa

rsa-key-20220816

[CANCEL](#) [IMPORT KEY PAIR](#)

- Nectar's firewall will by default block the ports we need to use our server. Click on the "Security Groups" tab. Click the up arrows next to ssh, http and icmp so that they move to the "Allocated" section. We need ssh allowed to login to our virtual machine and we need icmp if we want to ping it (ssh, http, and icmp in the image below has already been clicked from available section to allocated section). Lastly click on "Launch Instance".

### Launch Instance

[Details](#)  
[Source](#)  
[Flavor](#)  
[Networks](#)  
[Security Groups](#)  
[Key Pair](#)  
[Configuration](#)  
[Server Groups](#)  
[Metadata](#)

Select the security groups to launch the instance in.

**Allocated** 4

Displaying 4 items

Name	Description	
> default	Default security group	↓
> ssh	Allow SSH	↓
> http	Allow HTTP/S	↓
> icmp	Allow ICMP (eg. ping)	↓

Displaying 4 items

**Available** 0 Select one or more

Displaying 0 items

Name	Description
No items to display.	

Displaying 0 items

[CANCEL](#) [BACK](#) [NEXT](#) [LAUNCH INSTANCE](#)

- Click on the instance tab on the Nectar Dashboard to see the instance that has been created before.

The Nectar Dashboard shows the following instance details:

Instance Name	Image Name	IP Address	Flavour	Key Pair	Status	Availability Zone	Task	Power State	Age	Actions	
Nectar Demo	Nectar Ubuntu 18.04 LTS (bionic) amd64	qld 203.101.227.161 qld-data 10.255.138.151	m3.small	test	Active	a1	GRIScloud	None	Running	1 minute	CREATE SNAPSHOT

## 1.4 Ping and Login

### 1.4.1 Ping

- Open terminal (linux) or command prompt (windows). Use the IP address from the instance page, ping the instance to see if it was already there.

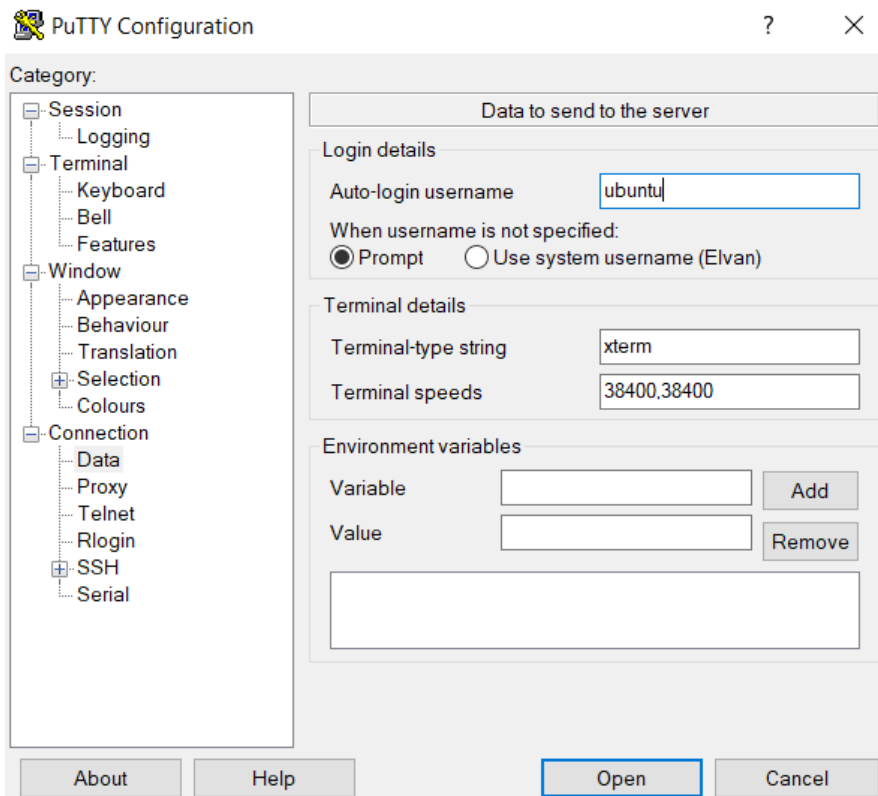
```
C:\WINDOWS\system32>ping 203.101.227.161

Pinging 203.101.227.161 with 32 bytes of data:
Reply from 203.101.227.161: bytes=32 time=42ms TTL=49
Reply from 203.101.227.161: bytes=32 time=42ms TTL=49
Reply from 203.101.227.161: bytes=32 time=43ms TTL=49
Reply from 203.101.227.161: bytes=32 time=42ms TTL=49

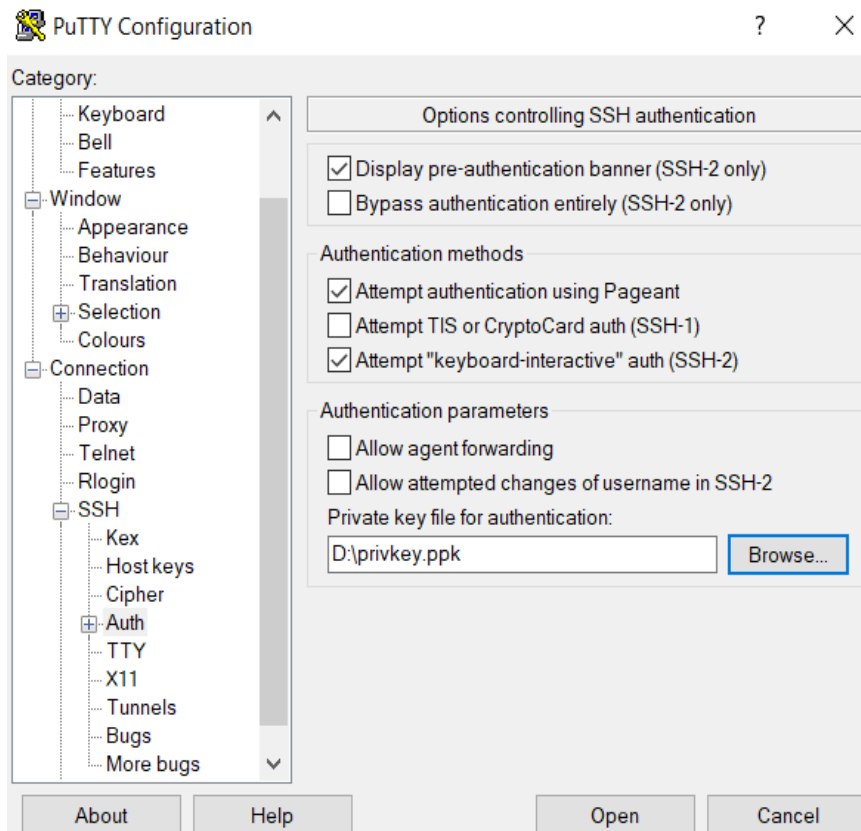
Ping statistics for 203.101.227.161:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 42ms, Maximum = 43ms, Average = 42ms
```

## 1.4.2 Login using Windows PuTTY

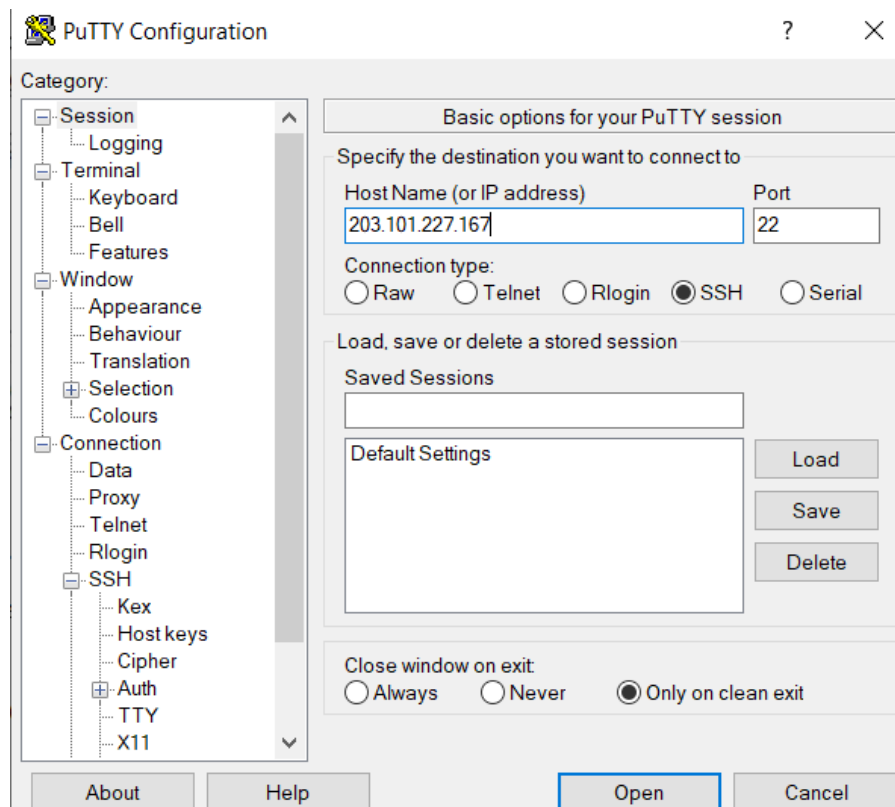
1. Open Putty, on the sidepane go to Connection → Data. In the auto-login username type ubuntu



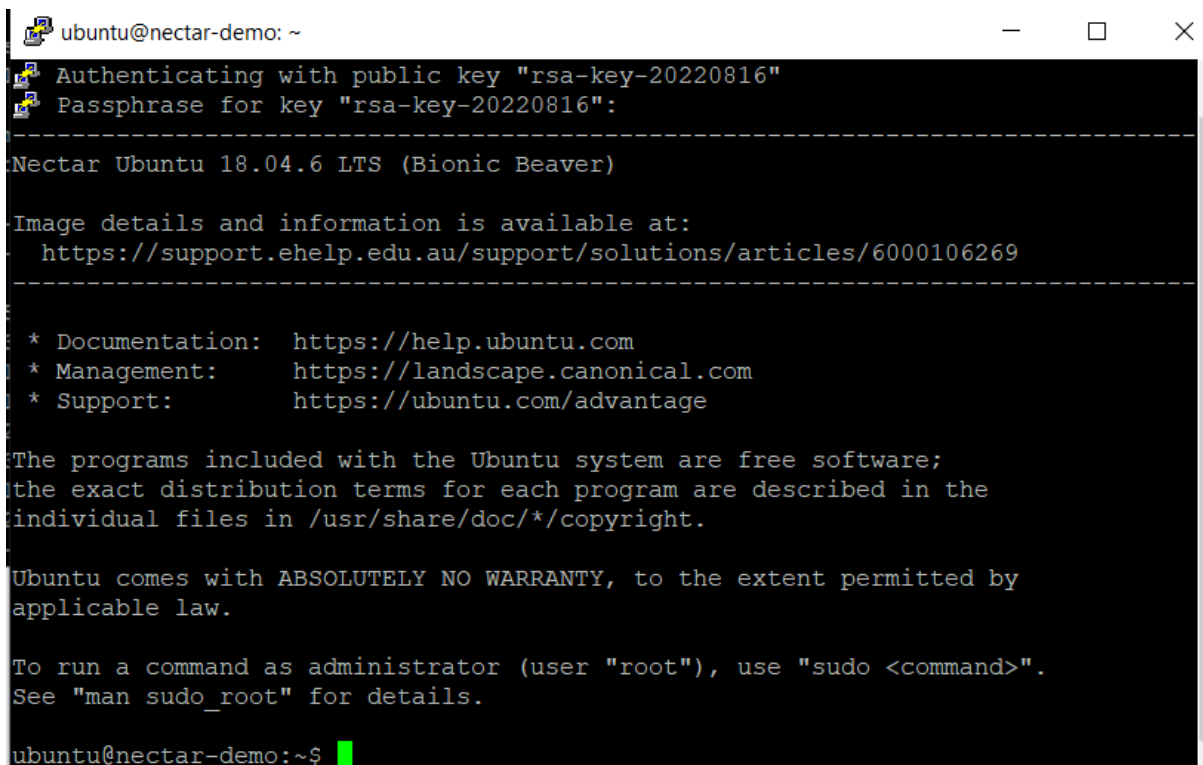
2. On the sidepane go to Connection → SSH → Auth. Click browse and select private key file (.ppk) that was generated earlier.



3. Click on the Session category and enter the IP address from the Nectar instances page, then click open button



4. Configuration window will be replaced by a console window that means you have logged into the Nectar instance



ubuntu@nectar-demo: ~

\* Support: <https://ubuntu.com/advantage>

The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/\*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo\_root" for details.

ubuntu@nectar-demo:~\$ ls

ubuntu@nectar-demo:~\$ ls -al

total 32

drwxr-xr-x	5	ubuntu	ubuntu	4096	Aug 16 07:42	.
drwxr-xr-x	3	root	root	4096	Aug 16 07:27	..
-rw-r--r--	1	ubuntu	ubuntu	220	Apr 4 2018	.bash_logout
-rw-r--r--	1	ubuntu	ubuntu	3771	Apr 4 2018	.bashrc
drwx-----	2	ubuntu	ubuntu	4096	Aug 16 07:42	.cache
drwx-----	3	ubuntu	ubuntu	4096	Aug 16 07:42	.gnupg
-rw-r--r--	1	ubuntu	ubuntu	807	Apr 4 2018	.profile
drwx-----	2	ubuntu	ubuntu	4096	Aug 16 07:27	.ssh

ubuntu@nectar-demo:~\$

## Task 2. Add Web server, DB server, and Website within an LXC Container

### 2.1 Adding LXC Web server

#### 2.1.1 Login to Nectar

#### 2.1.2 Update and upgrade packages

Using command “sudo apt update” and “sudo apt upgrade”

#### 2.1.3 LXC and LXD

1. Install lxc and lxd using “sudo apt install lxc lxd”
2. Add ubuntu user to the lxd group using “sudo usermod --append --groups lxd ubuntu”
3. Setup lxd with the command below and accept all default options using “sudo lxd init”

```
ubuntu@nectar-demo:~$ sudo lxd init
Would you like to use LXD clustering? (yes/no) [default=no]:
Do you want to configure a new storage pool? (yes/no) [default=yes]:
Name of the new storage pool [default=default]:
Would you like to connect to a MAAS server? (yes/no) [default=no]:
Would you like to create a new local network bridge? (yes/no) [default=yes]:
What should the new bridge be called? [default=lxdbr0]:
What IPv4 address should be used? (CIDR subnet notation, "auto" or "none") [default=auto]:
What IPv6 address should be used? (CIDR subnet notation, "auto" or "none") [default=auto]:
Would you like LXD to be available over the network? (yes/no) [default=no]:
Would you like stale cached images to be updated automatically? (yes/no) [default=yes]:
Would you like a YAML "lxd init" preseed to be printed? (yes/no) [default=no]:
```

4. List the lxc containers on the system

```
ubuntu@nectar-demo:~$ lxc list
To start your first container, try: lxc launch ubuntu:18.04

+-----+-----+-----+-----+-----+-----+
| NAME   | STATE | IPV4   | IPV6   | TYPE   | SNAPSHOTS |
+-----+-----+-----+-----+-----+-----+
```

5. For the assignment I am going to use Ubuntu because I have better understanding in the bash code than Alpine's code which is used in the lab

```
ubuntu@nectar-demo:/$ lxc image list images: | grep -i -E 'ubuntu.*amd64'
```

Image Name	Architecture	OS	Version	Size	Created	Updated	Public
ubuntu/16.04 (7 more)	amd64	Ubuntu	16.04	88.67MB	Aug 19, 2022	at 12:00am (UTC)	
ubuntu/16.04/cloud (3 more)	amd64	Ubuntu	16.04	103.78MB	Aug 19, 2022	at 12:00am (UTC)	
ubuntu/18.04 (7 more)	amd64	Ubuntu	18.04	106.62MB	Aug 19, 2022	at 12:00am (UTC)	
ubuntu/18.04/cloud (3 more)	amd64	Ubuntu	18.04	113.33MB	Aug 19, 2022	at 12:00am (UTC)	
ubuntu/focal (7 more)	amd64	Ubuntu	focal	111.21MB	Aug 19, 2022	at 12:00am (UTC)	
ubuntu/focal/cloud (3 more)	amd64	Ubuntu	focal	122.21MB	Aug 19, 2022	at 12:00am (UTC)	
ubuntu/jammy (7 more)	amd64	Ubuntu	jammy	113.25MB	Aug 19, 2022	at 12:00am (UTC)	
ubuntu/jammy/cloud (3 more)	amd64	Ubuntu	jammy	130.03MB	Aug 19, 2022	at 12:00am (UTC)	
ubuntu/kinetic (3 more)	amd64	Ubuntu	kinetic	112.49MB	Aug 19, 2022	at 12:00am (UTC)	
ubuntu/kinetic/cloud (1 more)	amd64	Ubuntu	kinetic	129.15MB	Aug 19, 2022	at 12:00am (UTC)	

6. Download and launch image (I am using ubuntu 18.04) using “lxc launch images ubuntu/18.04 web”

```
ubuntu@nectar-demo:/home$ lxc launch images:ubuntu/18.04 web
Creating web
Starting web
```

7. Check the container using command “lxc list”

```
ubuntu@nectar-demo:/$ lxc list
```

NAME	STATE	IPV4	IPV6	TYPE	SNAPSHOTS
web	RUNNING	10.116.245.126 (eth0)	fd42:a9fc:94d4:4ae1:216:3eff:fef4:a502 (eth0)	PERSISTENT	0

8. Log into the container using command “lxc exec web bash”. In my case, web is my container name.

```
ubuntu@nectar-demo:/$ lxc exec web bash
root@web:~#
```

9. Install Apache2 Web Server using command “sudo apt-get install apache2”

10. To start the Apache Web Server use command “sudo systemctl start apache2”

11. Bind a port to the host to redirect traffic to the container using iptables:

“sudo -E bash -c ‘iptables -t nat -I PREROUTING -i eth0 -p TCP -d \$PUBLIC\_IP --dport \$PORT -j DNAT --to-destination \$CONTAINER\_IP:\$PORT -m comment --comment “bind port to lxc ubuntu container”’

\$PORT = use 80 for web server

\$PUBLIC\_IP = your VM IP which is Nectar public IP

\$CONTAINER\_IP = your container IP from “lxc list”



## 2.2 Adding Database Server

### 2.2.1 Install MySQL server

Type command “sudo apt-get install mysql-server” and start using “sudo systemctl start mysql”

### 2.2.2 Open MySQL

Type command “sudo mysql”

### 2.2.3 Create root password

1. Type command “ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY root;” (the password is 'root')
2. Type “exit” to exit from mysql

### 2.2.4 Login again using root as user

```
root@web:~# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.7.39-0ubuntu0.18.04.2 (Ubuntu)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

### 2.2.5 Granting the root user privileges

Type command “GRANT PRIVILEGE ON database.table TO 'username'@'host';”

```
mysql> GRANT ALL PRIVILEGES ON *.* TO 'root'@'localhost';
```

### 2.2.6 Create database

Type command “CREATE DATABASE database\_name;”

```
mysql> CREATE DATABASE test;
Query OK, 1 row affected (0.00 sec)
```

### 2.2.7 Create table for the website

Type “CREATE TABLE table\_name (column\_name datatype, column\_name2 datatype, ...);”. ALTER TABLE in the image below is used for changing column name

```
mysql> CREATE TABLE Users( user_id INT NOT NULL AUTO_INCREMENT, Name varchar(255), PRIMARY KEY(user_id) );
Query OK, 0 rows affected (0.01 sec)

mysql> ALTER TABLE Users CHANGE Name name varchar(255);
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

## 2.3 Adding Simple Functional Website

### 2.3.1 Install PHP

1. Type “sudo apt-get install php”
2. Check it using “php --version”

### 2.3.2 Install php-mysqli to connect mysqli with php

Type “sudo apt-get install php-mysqli”

### 2.3.3 Create an index.php file in /var/www/html folder

```
<!DOCTYPE html>
<?php
try {
    if ( $_SERVER['REQUEST_METHOD'] == 'POST' ) {
        $servername = "localhost";
        $username = "root";
        $password = "root";
        $database = "test";
        $conn = mysqli_connect($servername, $username, $password,
$database);

        if (!$conn) {
            die("Connection failed: ". mysqli_connect_error());
        }

        $name = $_POST["fullname"];
        if($_POST["login"]) {
            $sql = "SELECT * FROM Users WHERE name='$name'";

            $result = $conn->query($sql);

            if ($result->num_rows > 0) {
                while($row = $result->fetch_assoc()) {
                    echo "welcome ".$row["name"]."<br>";
                }
            } else {
                echo "<script>alert('You are not
registered!')</script>";
            }
        } else {
            $sql = "INSERT INTO Users (user_id, name) VALUES
(null,'$name')";

            $result = $conn->query($sql);

            if($result === TRUE) {
                echo "<script>alert('Registered
successfully!')</script>";
            } else {
```

```

                                echo "Error: ". $sql . "<br>" .
mysqli_error($conn);
                                }
                                }
                                }
                                mysqli_close($conn);
} catch (Exception $e) {
    echo $e;
}
?>
<html lang="en">
  <head>
    <meta charset=UTF-8">
    <meta name="description" content="Testing Web Page in LXC">
    <meta name="keywords" content="Web, Test">
    <meta name="author" content="Elvan Alandi">
    <title>Testing Website</title>
  </head>
  <body>
    <h1>Login Using Name</h1>
    <form action="index.php" method="post">
      <label for="fullname">Full Name : </label>
      <input type="text" id="fullname" name="fullname"><br><br>
      <input type="submit" name="login" value="Login">
      <input type="submit" name="register" value="Register">
    </form>
  </body>
</html>

```

### 2.3.4 Test the web page

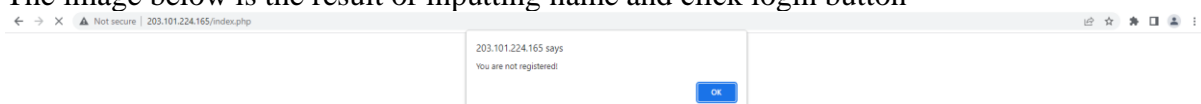


← → ↻ ⚠ Not secure | 203.101.224.165/index.php

## Login Using Name

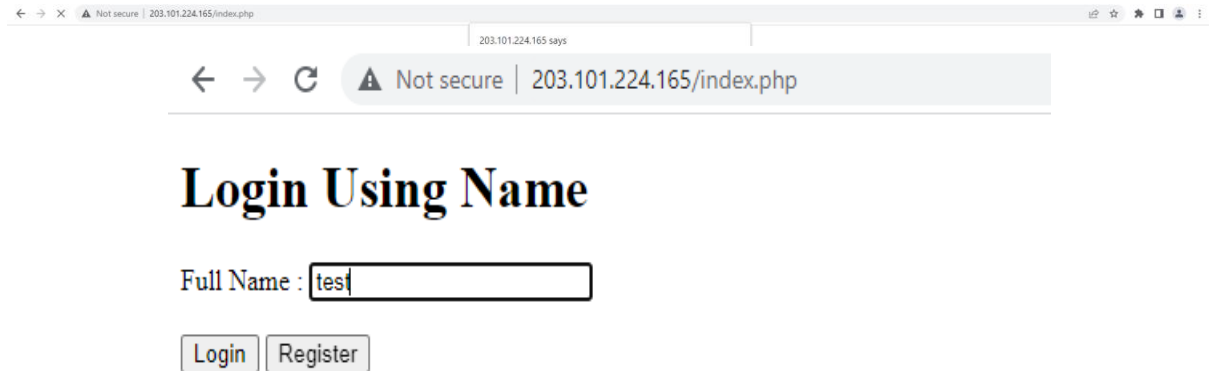
Full Name :

The image below is the result of inputting name and click login button



The image below is the result of inputting name and click register button

I input “test” in the full name input box



← → ↻ ⚠ Not secure | 203.101.224.165/index.php

## Login Using Name

Full Name :

This is the result of successful login and the ‘test’ name inserted into the database.



← → ↻ ⚠ Not secure | 203.101.224.165/index.php

welcome test

## Login Using Name

Full Name :

```
mysql> select * from Users;
+-----+-----+
| user_id | name |
+-----+-----+
|      1 | elvan |
|      2 | test |
+-----+-----+
2 rows in set (0.00 sec)
```

## References

- [1] "What is PEM Format?", *knowledge.digicert.com*. [Online]. Available: <https://knowledge.digicert.com/quovadis/ssl-certificates/ssl-general-topics/what-is-pem-format.html#:~:text=PEM%20or%20Privacy%20Enhanced%20Mail,very%20distinct%20headers%20and%20footers>. [Accessed: 17- Aug- 2022].
- [2] "What are PEM files?", *s3.amazonaws.com*. [Online]. Available: [https://s3.amazonaws.com/smhelpcenter/smhelp941/classic/Content/security/concepts/what\\_are\\_pem\\_files.htm](https://s3.amazonaws.com/smhelpcenter/smhelp941/classic/Content/security/concepts/what_are_pem_files.htm). [Accessed: 17- Aug- 2022].
- [3] "Certificate and key format for importing - AWS Certificate Manager", *docs.aws.amazon.com*. [Online]. Available: <https://docs.aws.amazon.com/acm/latest/userguide/import-certificate-format.html#:~:text=PEM%20stands%20for%20Privacy%20Enhanced,a%20PEM%E2%80%93formatted%20file%20is%20>. [Accessed: 17- Aug- 2022].
- [4] "Privacy-Enhanced Mail - Wikipedia", *en.wikipedia.org*. [Online]. Available: [https://en.wikipedia.org/wiki/Privacy-Enhanced\\_Mail](https://en.wikipedia.org/wiki/Privacy-Enhanced_Mail). [Accessed: 17- Aug- 2022].
- [5] N. Cheema, "The Difference Between .PEM And .PPK And How To Generate .PPK File For SSH From Putty", *c-sharpcorner.com*, 2020. [Online]. Available: <https://www.c-sharpcorner.com/article/difference-between-pem-and-ppk/>. [Accessed: 18- Aug- 2022].
- [6] "42.4. Pluggable Authentication Modules (PAM)", *web.mit.edu*. [Online]. Available: [http://web.mit.edu/rhel-doc/5/RHEL-5-manual/Deployment\\_Guide-en-US/ch-pam.html](http://web.mit.edu/rhel-doc/5/RHEL-5-manual/Deployment_Guide-en-US/ch-pam.html). [Accessed: 18- Aug- 2022].
- [7] "What is a Container? - Docker", *Docker*. [Online]. Available: <https://www.docker.com/resources/what-container/>. [Accessed: 18- Aug- 2022].
- [8] D. Both, "A Linux user's guide to Logical Volume Management", *opensource.com*, 2016. [Online]. Available: <https://opensource.com/business/16/9/linux-users-guide-lvm>. [Accessed: 20- Aug- 2022].