



SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY

**Enterprise Standards and Best Practices for IT Infrastructure**

**4<sup>th</sup> Year 2<sup>nd</sup> Semester 2014**

Name: Chandranath R. D. S. P

SLIIT ID: IT 13 0629 34

Practical Session: WE Monday (June Intake)

Practical Number: Lab 1, 2 and 3

Date of Submission: 7/30/2016

Date of Evaluation : \_\_\_\_\_

Evaluators Signature : \_\_\_\_\_

## **Table of Content**

Purpose.....	01
What is AWS? .....	01
How to create a Windows Instance using AWS? .....	01
Working with Windows Virtual Instance.....	09
How to create a Linux Instance using AWS? .....	10
Working with Ubuntu Virtual Instance.....	14
How to create a DB Instance using AWS? .....	15

## Purpose

This report covers the process of how to create virtual operating system and database instances in AWS.

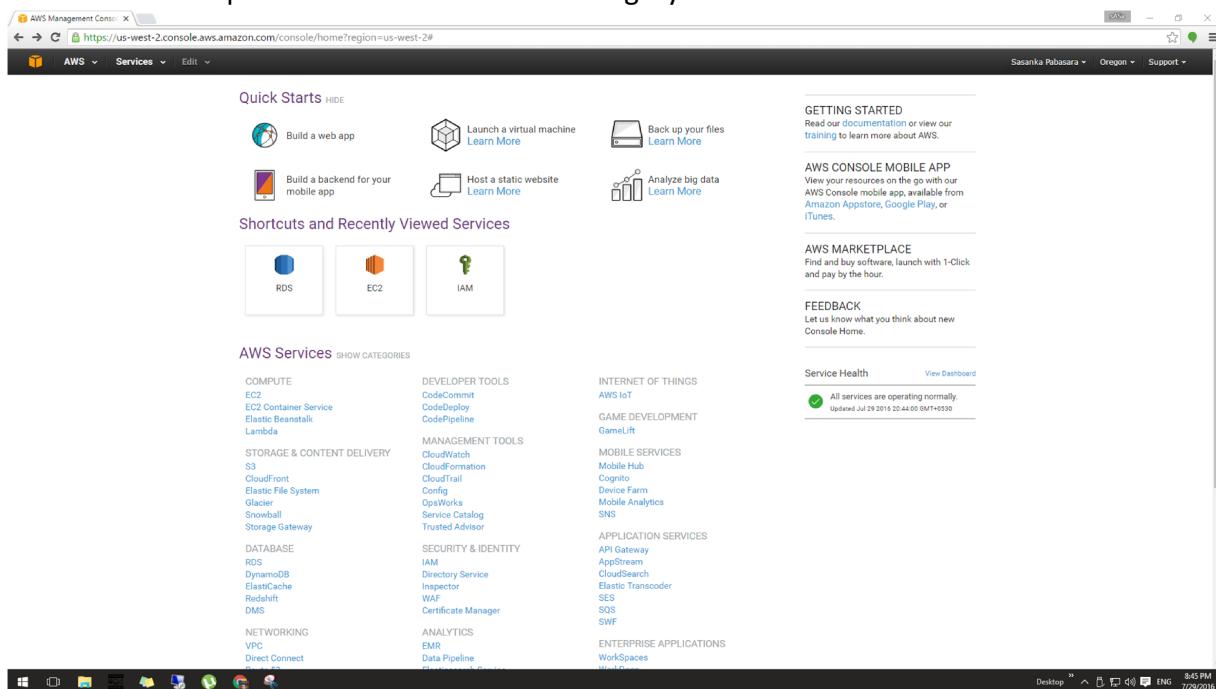
## What is AWS?

Amazon Web Services (AWS) helps us to work with virtual computing, using AWS console we can create and work with operating system instances and database instances in a virtual way. First we need to have an AWS account which is you have to provide your details alongside with your credit card number, after setting up your account you can start your projects in the virtual domain.

## How to create a Windows instance using AWS?

Open up your browser and go the AWS page then using your username and password login to the AWS server.

1. Select “EC2” component under “COMPUTE” category



Notes: all the components you previously worked with will be available under “Shortcuts and recently viewed Services”

## 2. Click “Launch Instance” to launch an operating system instance

## 3. Then you will be redirected into a new page which has a list of Amazon Machine Images (AMIs) then you need to select “Microsoft Windows Server 2012 R2 Base” which is a free AMI.

#### 4. After selecting you will be asked to choose the instance type

**Step 2: Choose an Instance Type**

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate
General purpose	<b>t2.micro</b> <small>[Free tier eligible]</small>	1	1	EBS only	-	Low to Moderate
General purpose	t2.small	1	2	EBS only	-	Low to Moderate
General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
General purpose	t2.large	2	8	EBS only	-	Low to Moderate
General purpose	m4.large	2	8	EBS only	Yes	Moderate
General purpose	m4.xlarge	4	16	EBS only	Yes	High
General purpose	m4.2xlarge	8	32	EBS only	Yes	High
General purpose	m4.4xlarge	16	64	EBS only	Yes	High
General purpose	m4.10xlarge	40	160	EBS only	Yes	10 Gigabit
General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate
General purpose	m3.large	2	7.5	1 x 32 (SSD)	-	Moderate
General purpose	m3.xlarge	4	15	2 x 40 (SSD)	Yes	High

**Next: Configure Instance Details**

#### 5. Then you can launch your instance by pressing “Launch” button

**Step 7: Review Instance Launch**

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

**AMI Details**

Microsoft Windows Server 2012 R2 Base - ami-26e72546  
Free tier eligible Microsoft Windows 2012 R2 Standard edition with 64-bit architecture. [English]  
Root Device Type: ebs Virtualization type: hvm

**Instance Type**

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

**Security Groups**

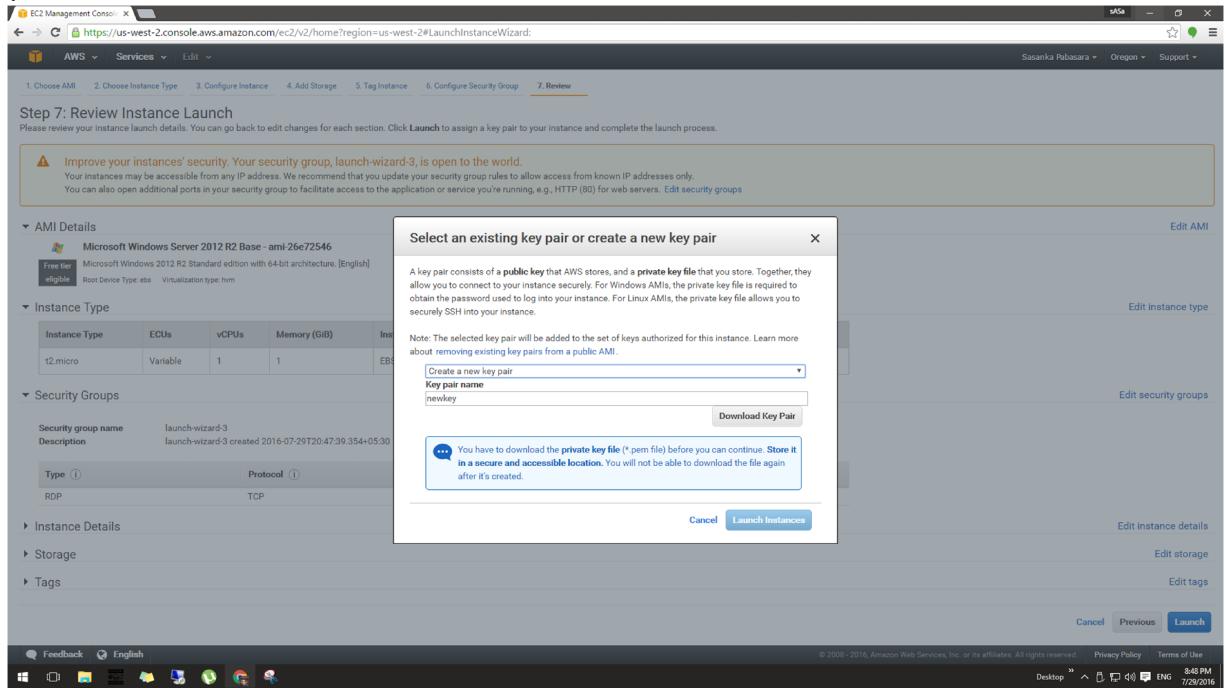
Security group name: launch-wizard-3  
Description: launch-wizard-3 created 2016-07-29T20:47:39.354+05:30

Type	Protocol	Port Range	Source
RDP	TCP	3389	0.0.0.0/0

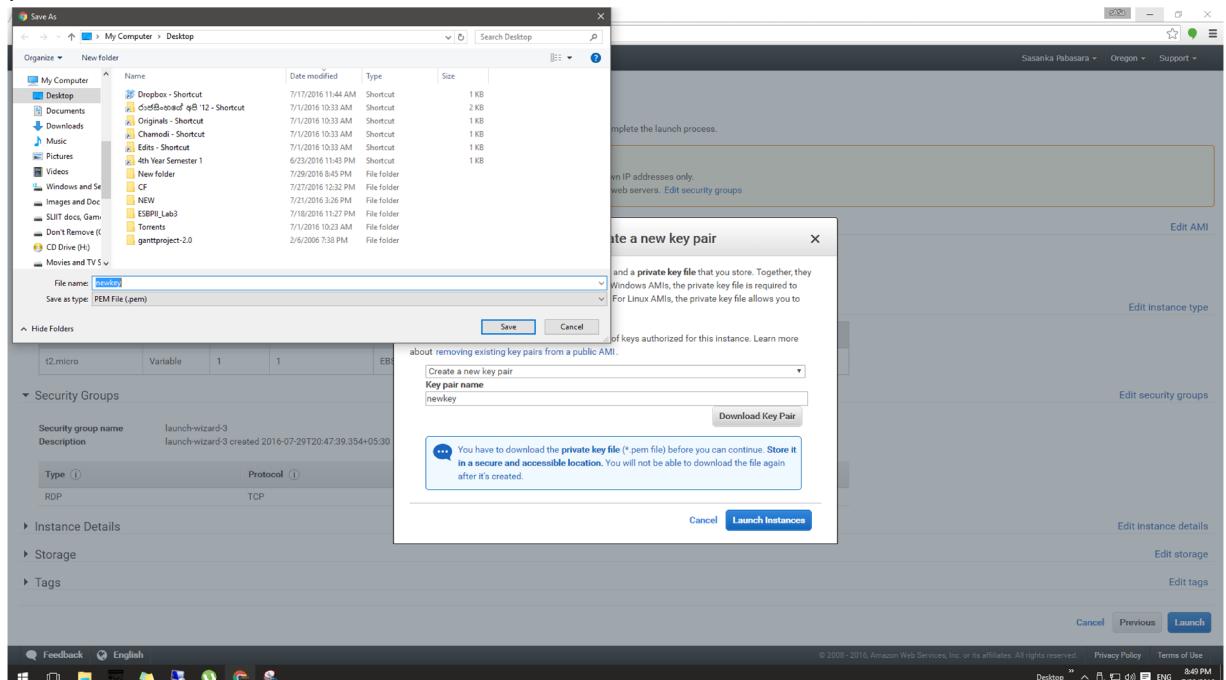
**Tags**

**Next: Launch**

6. Before launching you need to create and download a key pair, in the key pair you will get a private key and AWS stores a public key, together they allow you to connect to your instance.



Note: you can decide whether you want to create a new key pair or use an existing key pair



after downloading your key pair you can launch your instance.

7. As you can see there are 2 running instances and 1 pending instance, that pending one is the newly created instance, this may take a minite or two to convert the status pending to running.

The screenshot shows the AWS EC2 Management Console interface. On the left, a sidebar lists various services like Instances, Images, Auto Scaling, and Commands. The main area displays a table of instances with columns for Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS, Public IP, Key Name, Monitoring, and Launch. Three instances are listed:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS	Public IP	Key Name	Monitoring
	i-049da0e69dc69f12	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-52-42-153-75.us-west-2.compute.amazonaws.com	52.42.153.75	testkey	disabled
	i-0559ca268fb39701a	t2.micro	us-west-2a	pending	Initializing	None	ec2-52-36-250-77.us-west-2.compute.amazonaws.com	52.36.250.77	newkey	disabled
	i-0ed8c5cdbe7b3ccb	t2.micro	us-west-2b	running	2/2 checks ...	None	ec2-52-41-193-155.us-west-2.compute.amazonaws.com	52.41.193.155	testkey_linux	disabled

A modal window titled "Select an instance above" is open at the bottom, prompting the user to choose an instance to connect to.

8. When your instance is in running state you can right click on it and connet to your instance by providing your private key, the AWS system will decrypt your downloaded private key and will check with their public key for a match.

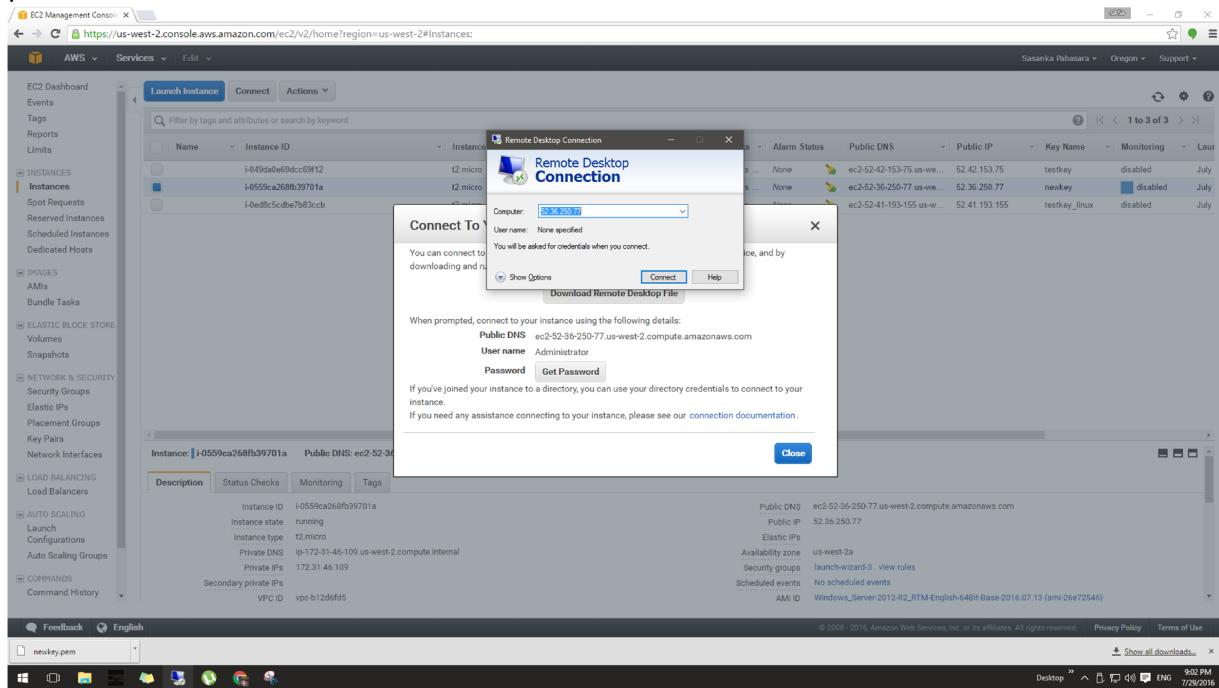
The screenshot shows the AWS EC2 Management Console with a modal dialog titled "Decrypt Password". It asks for the path to a private key file. A file explorer window is open on the desktop, showing a file named "newkey.pem" selected. The file details show it is a PEM file from 7/29/2016 8:49 PM. The "Decrypt Password" button is highlighted in blue.

The main EC2 console table shows the same three instances as before. The instance i-0559ca268fb39701a is now listed as "running".

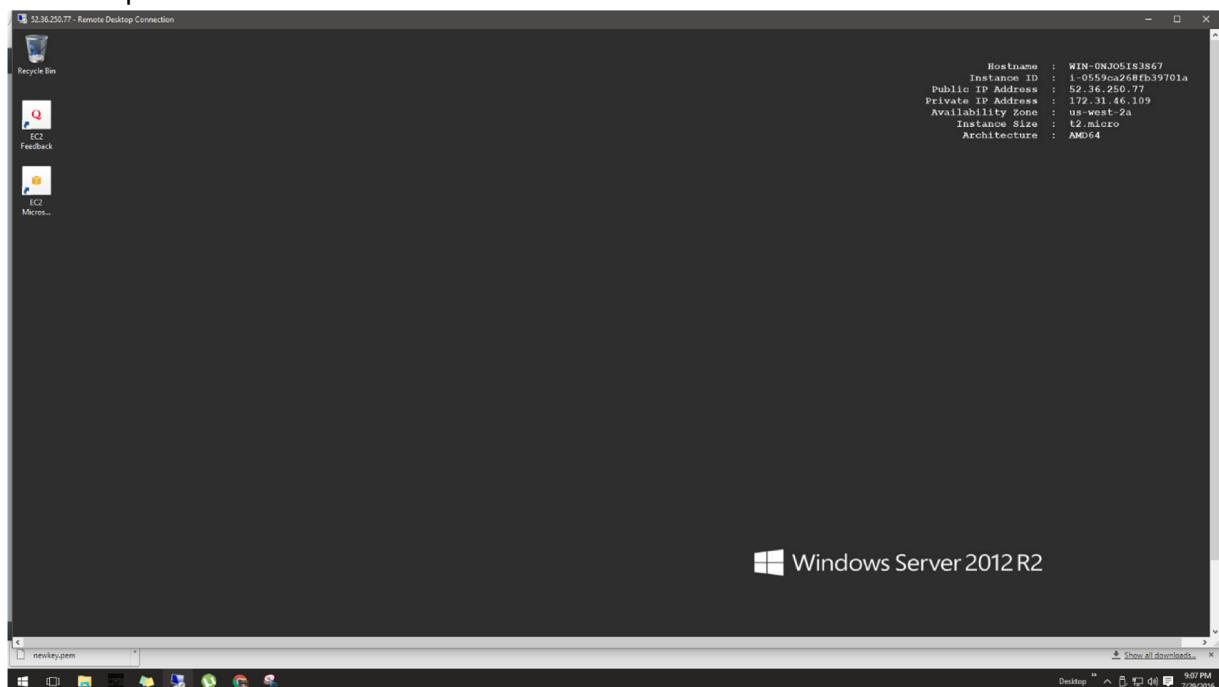
Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS	Public IP	Key Name	Monitoring
	i-049da0e69dc69f12	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-52-42-153-75.us-west-2.compute.amazonaws.com	52.42.153.75	testkey	disabled
	i-0559ca268fb39701a	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-52-36-250-77.us-west-2.compute.amazonaws.com	52.36.250.77	newkey	disabled
	i-0ed8c5cdbe7b3ccb	t2.micro	us-west-2b	running	2/2 checks ...	None	ec2-52-41-193-155.us-west-2.compute.amazonaws.com	52.41.193.155	testkey_linux	disabled

The desktop taskbar shows the file "newkey.pem" is currently being downloaded.

- After confirming your key pair successfully you can open up your Remote Desktop Connection application from your host computer and then use your IP address of the instance as the computer and the password which is decrypted in previous step as the password

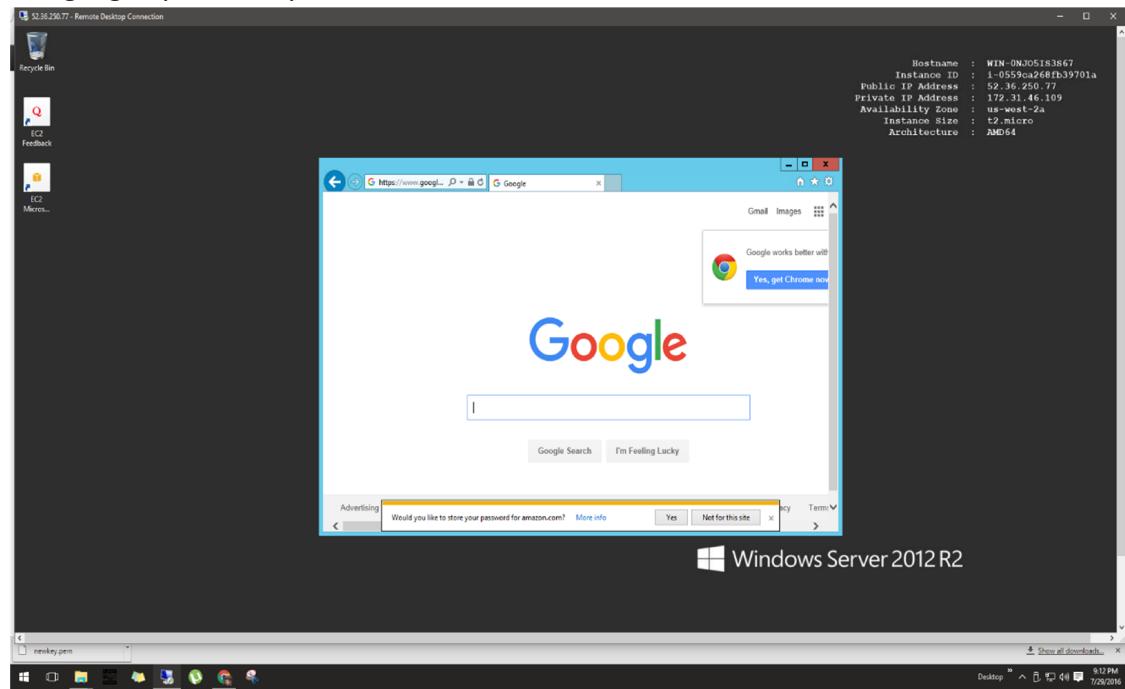


- Now you are in a virtual windows instance and you can do any work you do with your host computer in this virtual instance as well.



# Working with Windows Virtual Instance

- Googling in your newly launched windows instance ☺



## How to create a Linux instance using AWS?

1. Go back into step no 3 which is marked as "*How to create a windows instance using AWS?*" and select Amazon Linux AMI
2. Then after several steps same as before, you will get an Ubuntu instance in your instances tab

The screenshot shows the AWS EC2 Management Console. On the left, there's a sidebar with various navigation links: EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Spot Requests, Reserved Instances, Scheduled Instances, Dedicated Hosts, Images (AMIs), Bundle Tasks, Elastic Block Store (Volumes, Snapshots), Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces), Load Balancing (Load Balancers), Auto Scaling (Launch Configurations, Auto Scaling Groups), Commands (Command History, Documents, Managed Instances, Activations), and Feedback. The main area has tabs for Launch Instance, Connect, and Actions. A search bar at the top says "Filter by tags and attributes or search by keyword". Below it is a table of instances. One instance is selected: i-0ed8c5cdbe7fb3ccb. The table columns include Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS, Public IP, Key Name, Monitoring, and Launch Time. The selected instance's details are shown in a modal window below the table. The modal shows fields like Instance ID (i-0ed8c5cdbe7fb3ccb), Instance state (running), Instance type (t2.micro), Private DNS (ip-172-31-24-122.us-west-2.compute.internal), Private IPs (172.31.24.122), Secondary private IPs, VPC ID (vpc-b1206d5), Subnet ID (subnet-650b7801), Network interfaces (eth0), Source/dest. check (True), EBS-optimized (False), Root device type (ebs), Root device (/dev/xvda), and Block devices (/dev/xvda). To the right of the modal, there's a large amount of detailed information about the instance, including Public DNS (ec2-52-41-193-155.us-west-2.compute.amazonaws.com), Public IP (52.41.193.155), Elastic IPs, Availability zone (us-west-2b), Security groups (launch-wizard-2, view rules), Scheduled events (No scheduled events), AMI ID (amazon-ami-hvm-2016.03.3.x86\_64-gp2 (ami-7172b611)), Platform (-), IAM role (-), Key pair name (testkey\_linux), Owner (281527481726), Launch time (July 18, 2016 at 10:06:35 PM UTC+5:30 (277 hours)), Termination protection (False), Lifecycle (normal), Monitoring (basic), Alarm status (None), Kernel ID (-), RAM disk ID (-), Placement group (-), Virtualization (hvm), Reservation (F018c465501663f69d), and AMI launch index (0). At the bottom of the modal, there are buttons for Save, Cancel, and Delete. The footer of the console includes links for Feedback, English, Desktop, Privacy Policy, Terms of Use, and a timestamp (11:32 AM 7/30/2016).

3. To connect into Ubuntu instances you just can't right click and click connect just like we did in previous example.

Note: You will be needed two additional applications called putty and puttygen, where putty is going to help you to connect with your instance and puttygen will help you to download and decrypt your key pair.

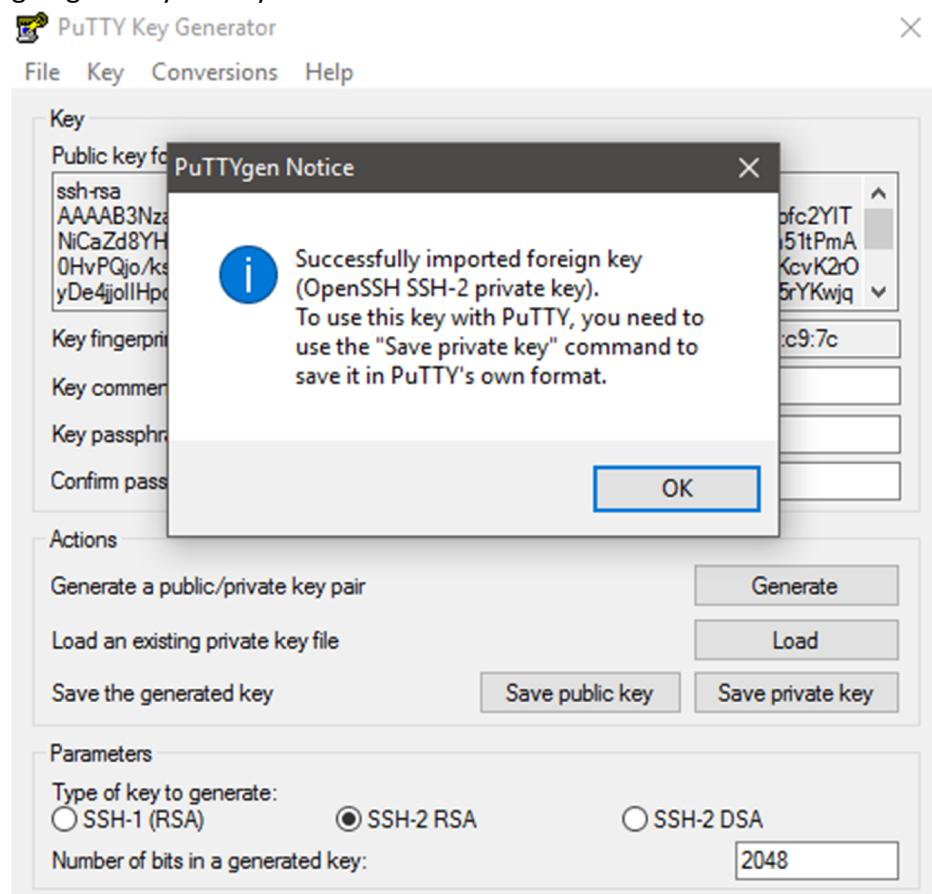
4. Putty is just like your Remote Desktop Connection application, you will need to provide IP address of your instance and the decrypted password.
  - a. IP address : Since this is an Ubuntu instance,

ubuntu@<Public DNS of your instance>

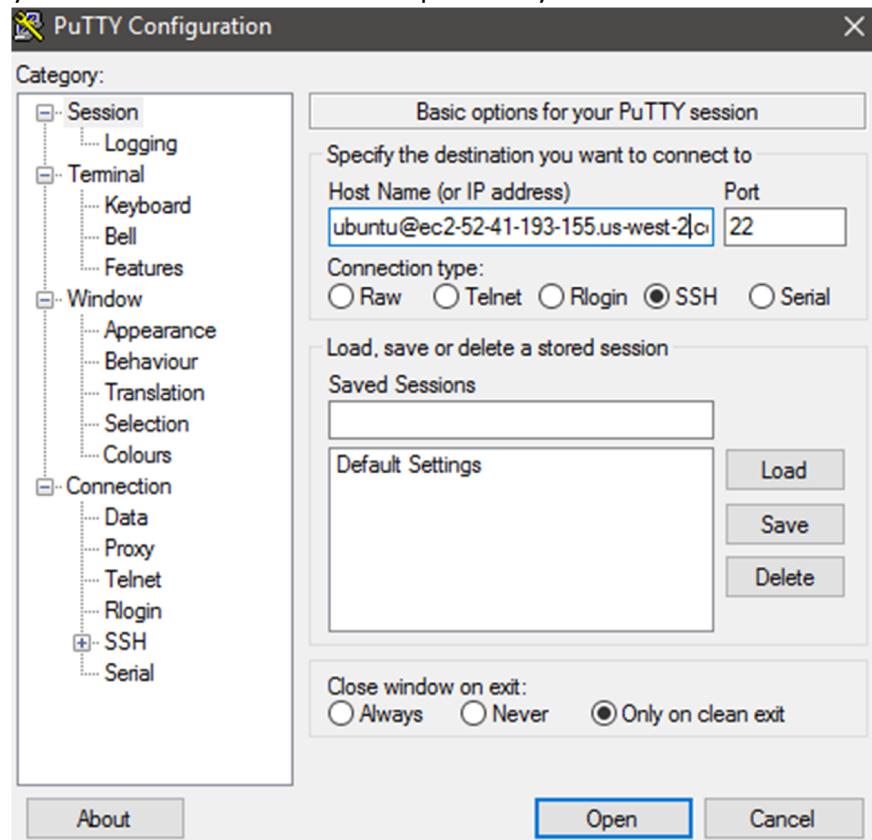
[ubuntu@ec2-52-41-193-155.us-west-2.compute.amazonaws.com](http://ubuntu@ec2-52-41-193-155.us-west-2.compute.amazonaws.com)

b. Password :

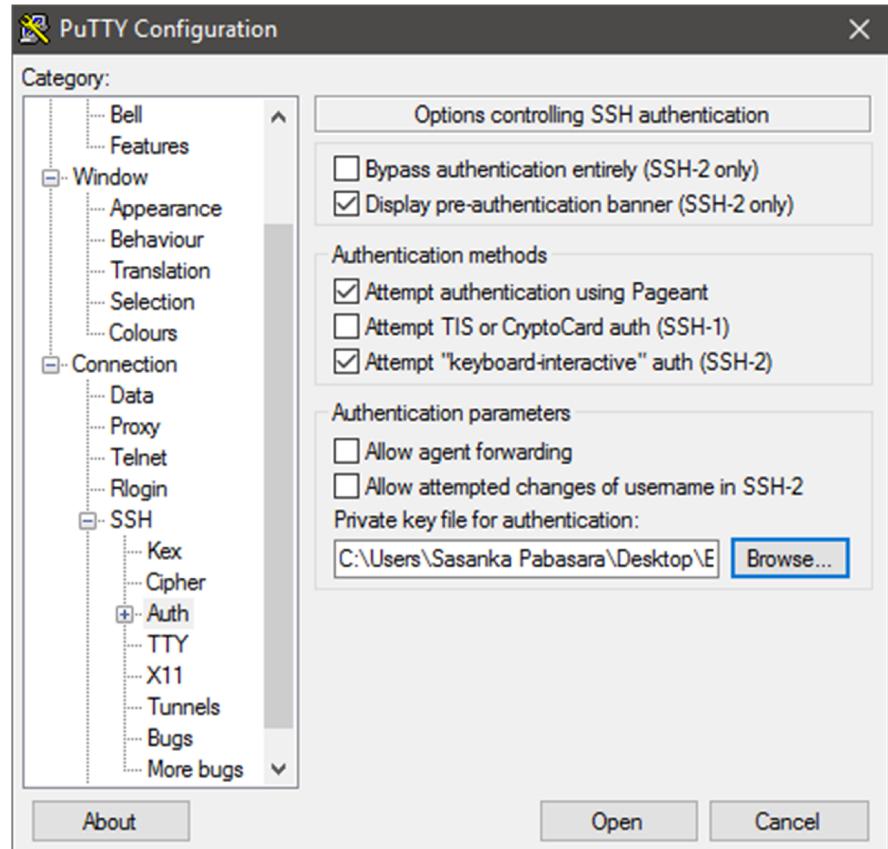
- i. Putty does not natively support the private key format (.pem) generated by Amazon EC2. Therefore puttygen, which can convert keys to the required Putty format (.ppk). You must convert your private key into this format (.ppk) before attempting to connect to your instance using putty. Click load and find your .pem file then load it into your puttygen application then click “Save Private Key” which is going to be your key.



- ii. Your private key is now in the correct format for use with putty. You can now connect to your instance using putty's SSH client. Provide your IP address as we discussed previously as the host name.



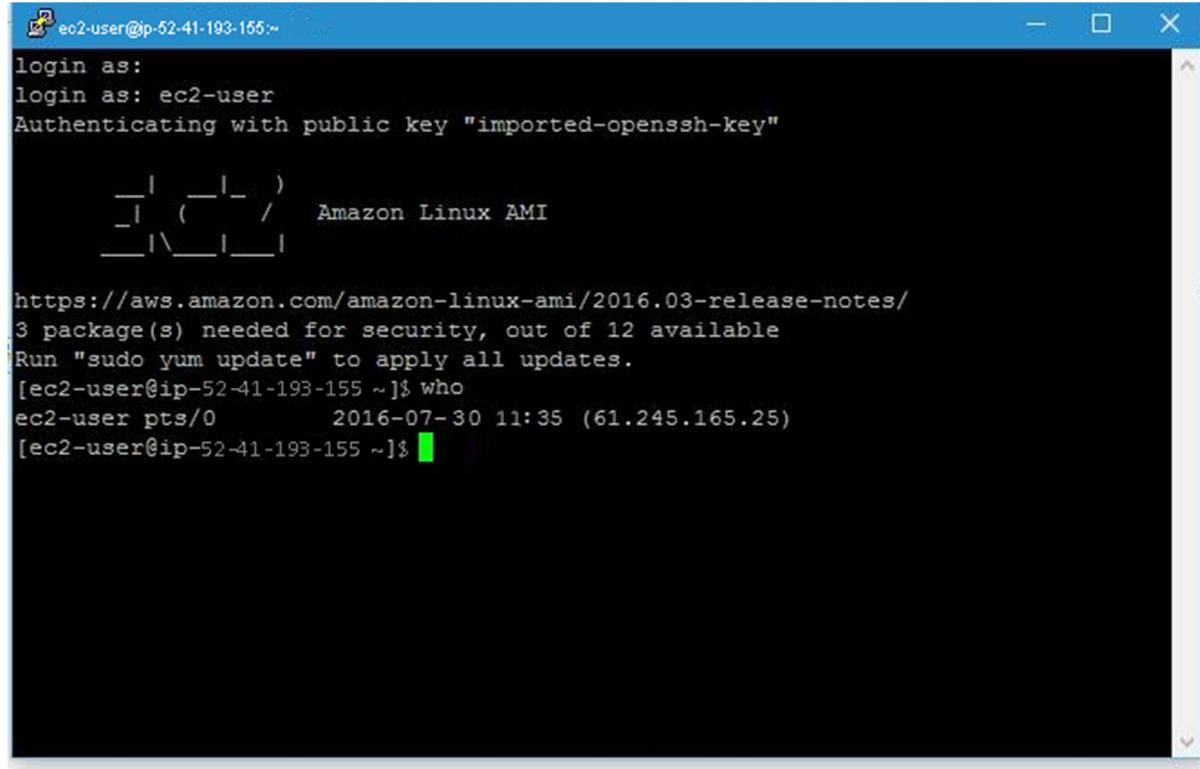
- iii. Go to Connection->SSH->Auth tab then browse your newly generated .ppk file and load it as the private key.



- iv. Click Open, now you will get an Ubuntu console instance

## Working with Ubuntu Virtual Instance

In here you can execute any Ubuntu command 😊

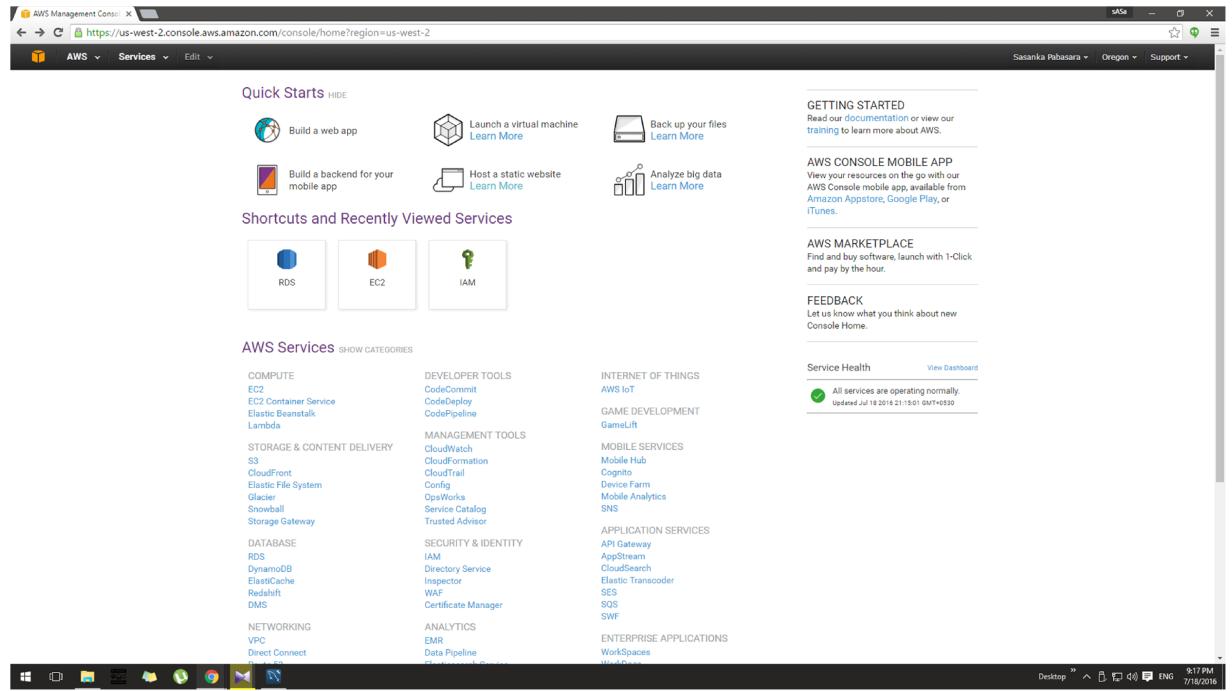


The screenshot shows a terminal window titled "ec2-user@ip-52-41-193-155:~". The window displays a login process for the "ec2-user" account, using a public key for authentication. It then shows the Amazon Linux AMI logo. Below this, it lists available security updates and the output of the "who" command, which shows the user is currently logged in at pts/0.

```
ec2-user@ip-52-41-193-155:~$ who
ec2-user pts/0    2016-07-30 11:35 (61.245.165.25)
```

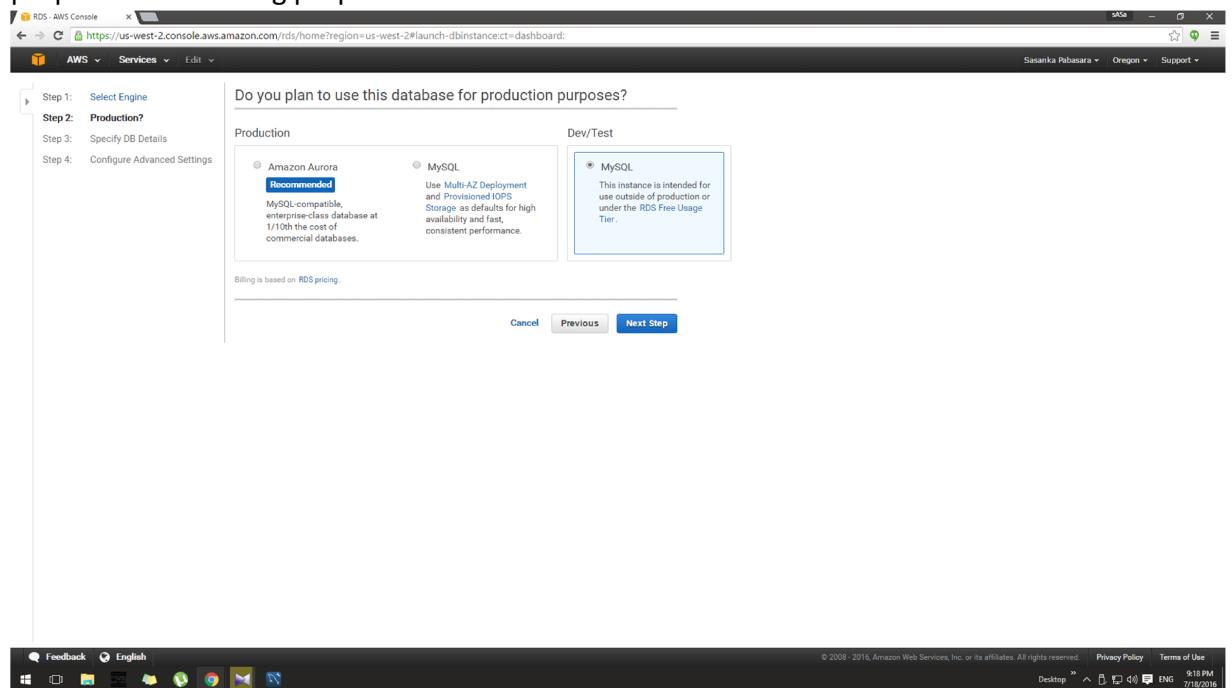
## How to create a DB Instance using AWS?

1. Select “RDS” component under “DATABASE” category



The screenshot shows the AWS Management Console Home page. At the top, there are "Quick Starts" like "Build a web app", "Launch a virtual machine", "Back up your files", "Build a backend for your mobile app", "Host a static website", and "Analyze big data". Below that is a section for "Shortcuts and Recently Viewed Services" with icons for RDS, EC2, and IAM. The main area is titled "AWS Services SHOW CATEGORIES" and lists various services under categories such as COMPUTE, STORAGE & CONTENT DELIVERY, DATABASE, NETWORKING, and more. The RDS icon is located under the DATABASE category. On the right side, there are sections for "GETTING STARTED", "AWS CONSOLE MOBILE APP", "AWS MARKETPLACE", and "FEEDBACK". A "Service Health" section indicates that all services are operating normally. The bottom of the screen shows the Windows taskbar.

2. Then you will get an interface, in there you can start the process to launch your DB instance
3. Then you need to decide whether you are creating this instance for production purpose or for testing purpose.



The screenshot shows the RDS AWS Console step 2: Production? interface. It asks "Do you plan to use this database for production purposes?". There are two options: "Production" and "Dev/Test". Under "Production", "Amazon Aurora" is selected and labeled as "Recommended". It is described as a MySQL-compatible, enterprise-class database at 1/10th the cost of commercial databases. Under "Dev/Test", "MySQL" is selected and described as being intended for use outside of production or under the RDS Free Usage Tier. At the bottom, it says "Billing is based on RDS pricing". There are "Cancel", "Previous", and "Next Step" buttons. The bottom of the screen shows the Windows taskbar.

#### 4. Then you need to fill the details as follows

The screenshot shows the 'Configure Advanced Settings' step of the AWS RDS instance creation wizard. The 'Instance Specifications' section is set for a MySQL 5.6.27 engine. The 'DB Instance Identifier' is 'ESBPII\_3'. The 'Master Username' is 'scbabara\_lab3' and the 'Master Password' is specified. A note at the bottom left says 'Estimate your monthly costs for the DB Instance using the RDS Instance Cost Calculator.' A warning message in a red box states: 'Provisioning less than 100 GB of General Purpose (SSD) storage for high throughput workloads could result in higher latencies upon exhaustion of the initial General Purpose (SSD) IO credit balance. Click here for more details.'

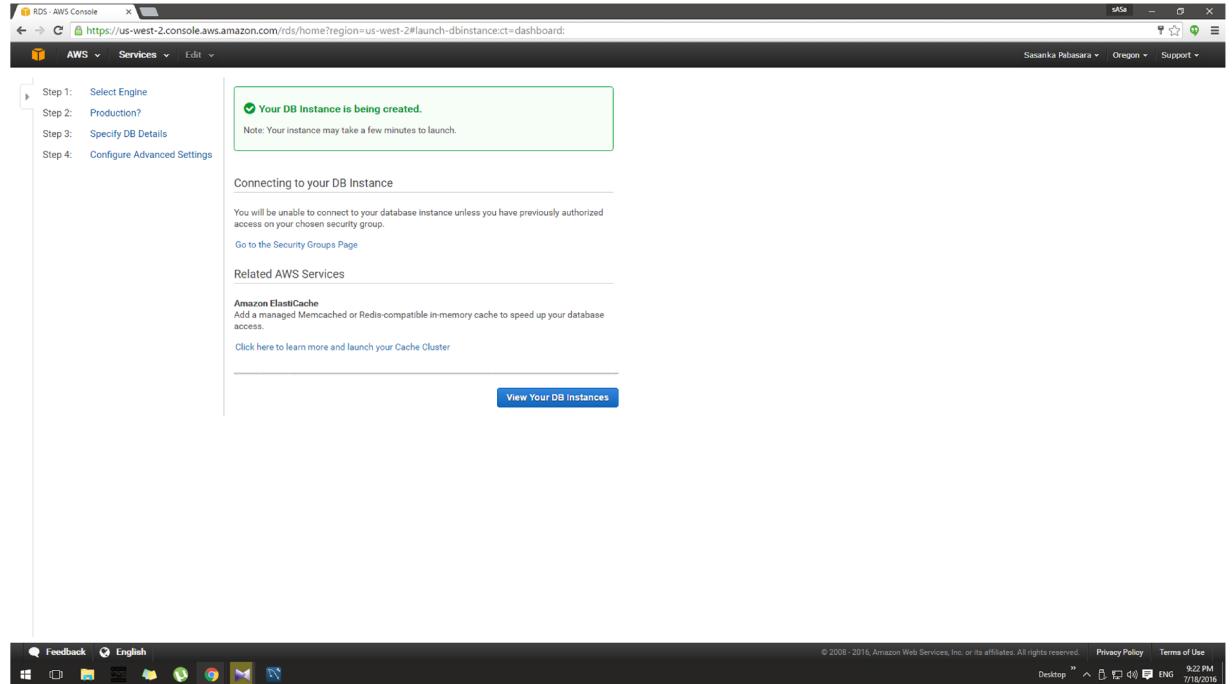
Note: You can decide the size of your storage for this db instance and you have to create a password for your instance.

#### 5. In the next step keep all the details as default

The screenshot shows the 'Configure Advanced Settings' step again, this time focusing on 'Network & Security' and 'Database Options'. Under 'Network & Security', it shows a VPC selected ('Default VPC (vpc-b126d65)'). Under 'Database Options', the 'Database Name' is left blank, 'Database Port' is 3306, 'DB Parameter Group' is 'default.mysql5.6', and 'Option Group' is 'default.mysql5.6'. The 'Monitoring' section has 'Enable Enhanced Monitoring' set to 'No'. The 'Maintenance' section has 'Auto Minor Version Upgrade' set to 'Yes' and 'Maintenance Window' set to 'No Preference'. At the bottom, there's a note about backup retention periods and a 'Launch DB Instance' button.

6. After completing these 5 steps you can finally launch your DB instance by pressing

#### View Your DB Instance



Then you will get a list of created db instances which are available for use marked as available in the status column, there are several status like creating -> backing up -> available. Every DB instance has an Endpoint address which is going to be the Hostname for the MySQL Workbench connection.

## 7. Copy down your endpoint address

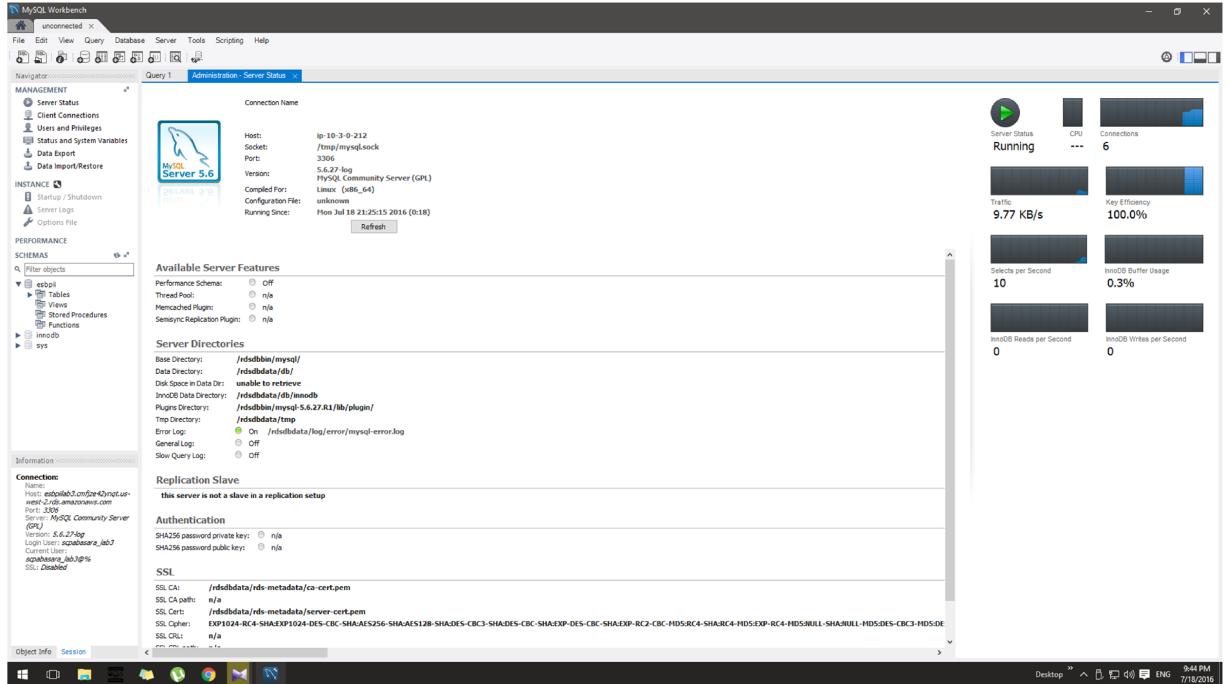
The screenshot shows the AWS RDS Dashboard. On the left, there's a sidebar with options like Instances, Clusters, Reserved Purchases, Snapshots, Security Groups, Parameter Groups, Option Groups, Subnet Groups, Events, Event Subscriptions, and Notifications. The main area is titled 'Launch DB Instance' and shows a table of DB instances. The first instance, 'esbpillab3', is highlighted. Its details are shown in a modal window: Engine (MySQL), Status (available), CPU (2.17%), Current Activity (None), Maintenance (None), Class (db.t2.small), VPC (vpc-b12d6fd5), Multi-AZ (No), Replication Role (None), and Encrypted (No). Below this, there are sections for 'Alarms and Recent Events' and 'Monitoring' (CPU, Memory, Storage, IOPS, Swap Usage) with their current values and thresholds.

## 8. Then open up your MySQL Workbench, and fill the details as bellow. You need to create a new connection.

The screenshot shows the MySQL Workbench interface. At the top, there's a toolbar with File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The main window has tabs for Navigator, Schemas, and Query. A 'Query 1' tab is active. In the center, a 'Connect to Database' dialog box is open. It has fields for 'Stored Connection' (dropdown), 'Connection Method' (Standard (TCP/IP)), 'Parameters' (SSL Advanced), 'Hostname' (yngt.us-west-2.rds.amazonaws.com), 'Port' (3306), 'Username' (sasanka\_lab3), 'Password' (Store in Vault...), and 'Default Schema' (Leave blank). There are 'OK' and 'Cancel' buttons at the bottom right of the dialog. The background shows the MySQL Workbench interface with various toolbars and panels.

Note: You need to store your password in the Workbench password store that is the same password you have entered earlier in step no 4. And the endpoint name should be copied into the hostname field without the port number and the colon.

9. Then you need to connect to your newly created schema using the SCHEMA category in your Navigator bar.



Note: You can see your DB instance connection details in the left side bottom part of your application, This application is running in your host computer but all the work we do with that instance is virtual so you can do anything to this instance using any host machine in the world without any conflicts.

#### Notes:

- And furthermore you can launch this DB instance in your previously created Windows Virtual Instance as well.
- And these 3 types of instances can be terminated, stopped, rebooted if you want.