



 lab



lab title

**Introduction to AWS**

**V1.43**



Course title

**BackSpace Academy**  
**AWS Certified Cloud Practitioner**



# Table of Contents

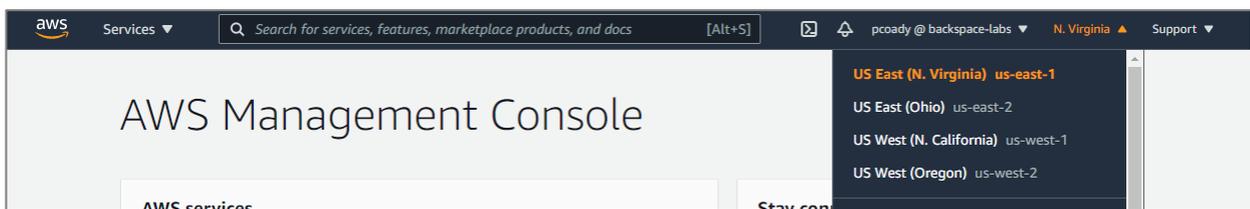
## Contents

<b>Table of Contents</b> .....	1
<b>About the Lab</b> .....	3
<b>Checking your AWS Usage and Monthly Bill</b> .....	4
<b>Creating an S3 Bucket and Uploading Files</b> .....	6
Uploading Files to your Bucket .....	7
Downloading files from your bucket .....	9
Troubleshooting.....	10
Clean Up.....	10
<b>Creating a SQL Database with RDS</b> .....	13
Creating a Security Group.....	13
Creating an RDS Database .....	16
Connecting to your RDS Instance .....	22
Troubleshooting Connection Issues .....	27
Clean Up.....	29
<b>Creating a Web Server with EC2</b> .....	32
Viewing your web server .....	39
Troubleshooting viewing your WordPress application .....	40
Finding the Username and Password for your WordPress application.....	42
Troubleshooting logging in to the WordPress application .....	44
Clean up .....	45
<b>Sending Emails with Amazon SES</b> .....	47
Requesting full access to SES.....	49
<b>Creating a Billing Alert with CloudWatch and SNS</b> .....	50
Enabling Billing Alerts .....	50
Creating a CloudWatch Alarm .....	51
<b>Creating an IAM User</b> .....	59
<b>Creating a Highly Available Architecture with Elastic Beanstalk</b> .....	62
Clean Up.....	65



## About the Lab

**Please note that not all AWS services are supported in all regions. Please use the US-East-1 (North Virginia) region for this lab.**



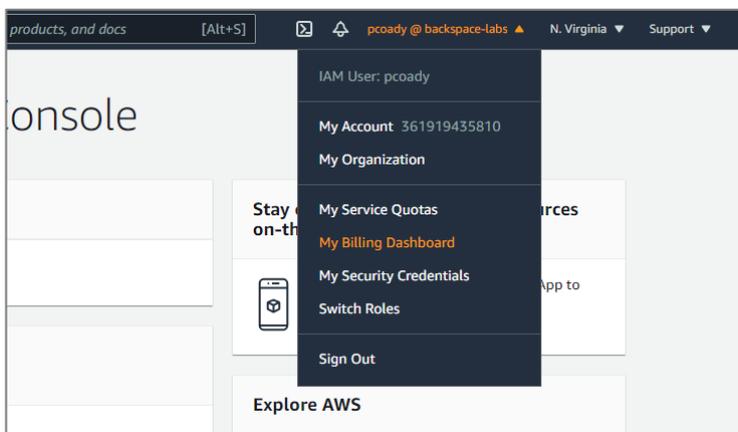
These lab notes are to support the hands on instructional videos of the Introduction to AWS section of the BackSpace AWS Cloud Practitioner Course.

**Please note that AWS services change on a weekly basis and it is extremely important you check the version number on this document to ensure you have the latest version with any updates or corrections.**

# ▶ **Checking** your AWS Usage and Monthly Bill

In this section we will learn how to use the **AWS Billing & Cost Management Dashboard** to keep track of costs.

From the AWS management console select "My Billing Dashboard" from the account drop down menu.



You will now see your total spend summary, spend by service and forecast spend (by clicking the Cost Explorer).

## Billing & Cost Management Dashboard



### Getting Started with AWS Billing & Cost Management

- Manage your costs and usage using [AWS Budgets](#)
  - Visualize your cost drivers and usage trends via [Cost Explorer](#)
  - Dive deeper into your costs using the [Cost and Usage Reports with Athena integration](#)
  - **Learn more:** Check out the [AWS What's New webpage](#)
- Do you have Reserved Instances (RIs)?**
- Access the [RI Utilization & Coverage reports](#)—and RI purchase recommendations—via [Cost Explorer](#).

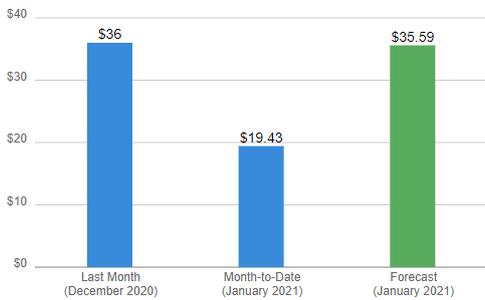
### Spend Summary

[Cost Explorer](#)

Welcome to the AWS Billing & Cost Management console. Your last month, month-to-date, and month-end forecasted costs appear below.

Current month-to-date balance for January 2021

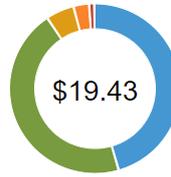
**\$19.43**



### Month-to-Date Spend by Service

[Bill Details](#)

The chart below shows the proportion of costs spent for each service you use.



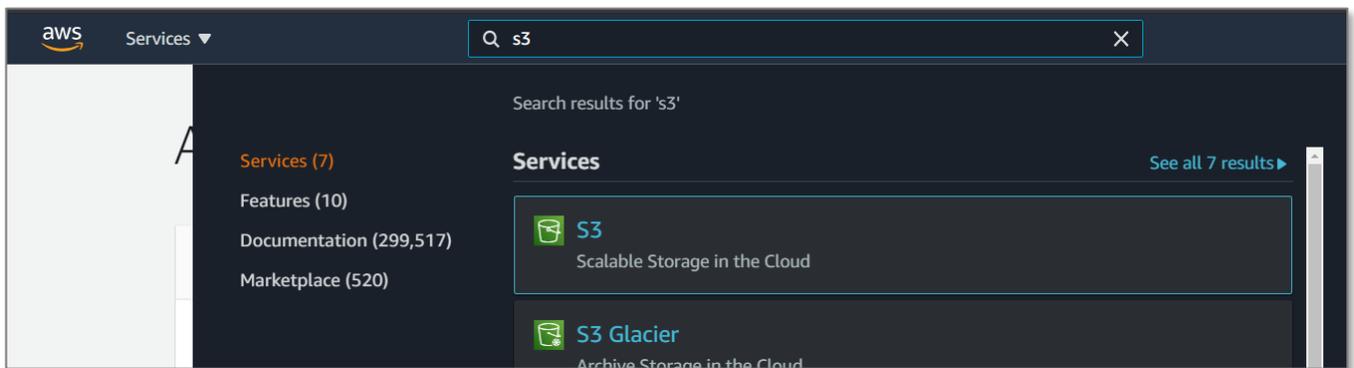
EC2	\$8.94
ELB	\$8.82
Route53	\$1.00
kms	\$0.53
Other Services	\$0.14
Tax	\$0.00
<b>Total</b>	<b>\$19.43</b>

# ▶ Creating an S3 Bucket and Uploading Files

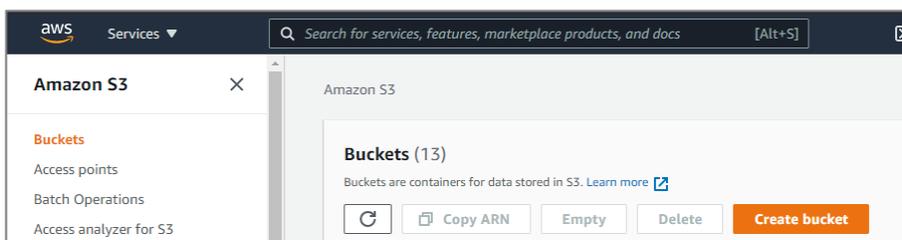
In this section we will create an S3 bucket, upload files to it and download files from it.

Click on the services menu and search S3.

Select S3



Click on Create Bucket



The create bucket dialog box will appear.

Enter a unique name for your bucket (it will need to be different from the one below)

Click 'Next'

Amazon S3 > Create bucket

## Create bucket

Buckets are containers for data stored in S3. [Learn more](#)

### General configuration

**Bucket name**

pcoady-backspace-intro-aws

Bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

**Region**

US East (N. Virginia) us-east-1

Copy settings from existing bucket - *optional*  
Only the bucket settings in the following configuration are copied.

**Choose bucket**

Leave other settings as is and click *Create bucket* (by default the bucket is private)

You will now see your bucket has been created.

Amazon S3

**Buckets (13)** [Refresh](#) [Copy ARN](#) [Empty](#) [Delete](#) [Create bucket](#)

Buckets are containers for data stored in S3. [Learn more](#)

Q pcoady X 1 match < 1 > ⚙

Name	Region	Access	Creation date
<a href="#">pcoady-backspace-intro-aws</a>	US East (N. Virginia) us-east-1	Bucket and objects not public	January 18, 2021, 02:12:33 (UTC+11:00)

## Uploading Files to your Bucket

Click on the link to the bucket

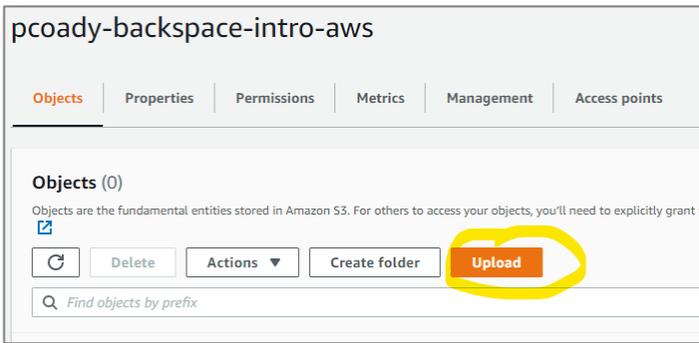
**Buckets (13)** [Refresh](#) [Copy ARN](#) [Empty](#) [Delete](#) [Create bucket](#)

Buckets are containers for data stored in S3. [Learn more](#)

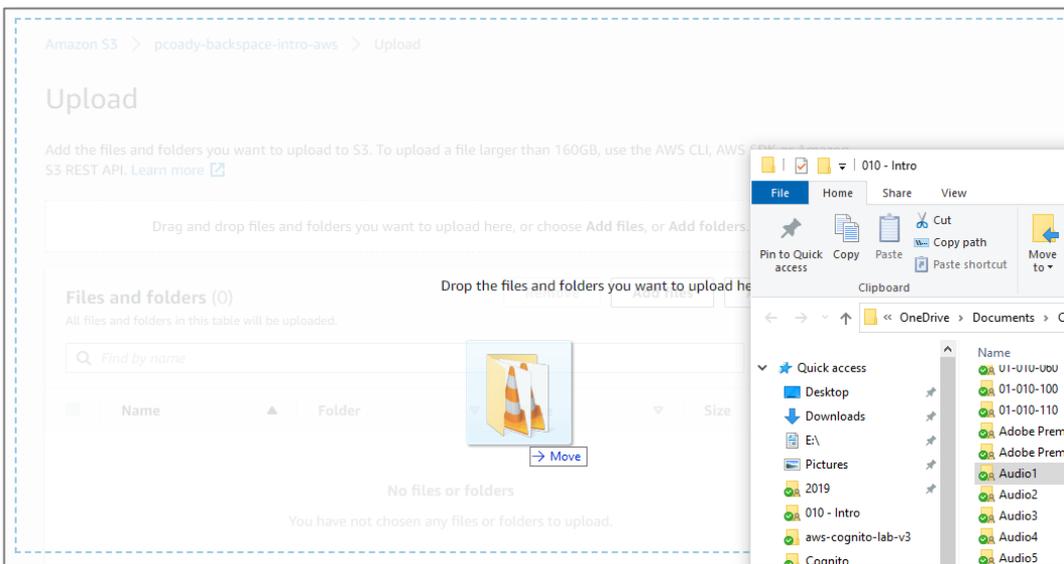
Q pcoady X 1 match < 1 > ⚙

Name	Region	Access	Creation date
<a href="#">pcoady-backspace-intro-aws</a>	US East (N. Virginia) us-east-1	Bucket and objects not public	January 18, 2021, 02:12:33 (UTC+11:00)

Click *Upload*

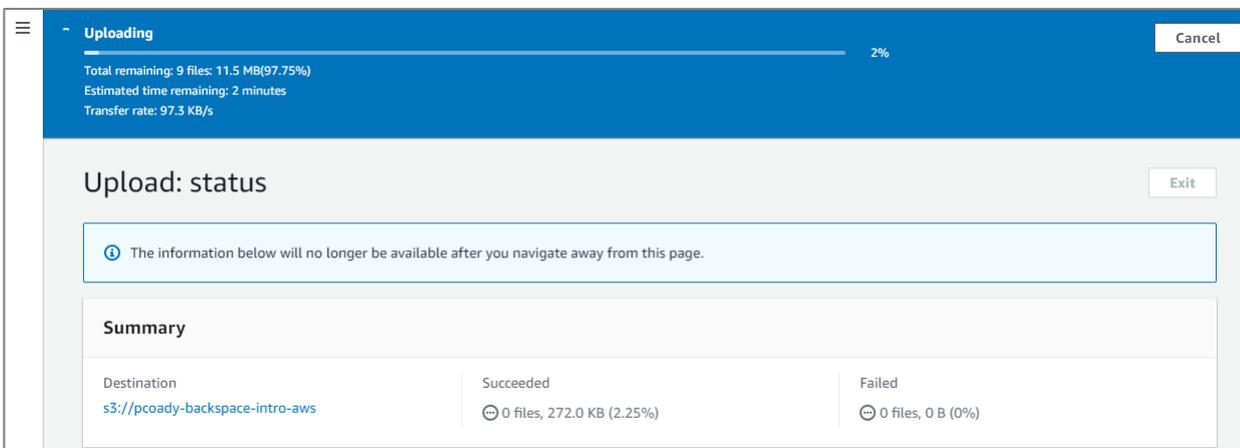


Drag a folder with files onto the form.



Scroll down and click *Upload*

Your files will begin uploading



Your upload will eventually complete.

The screenshot shows the 'Upload: status' page in the AWS S3 console. At the top, a green banner indicates 'Upload succeeded' with a checkmark icon and the text 'View details below.'. Below this, a blue box contains a warning: 'The information below will no longer be available after you navigate away from this page.' The main section is titled 'Upload: status' and contains a 'Summary' box with three columns: 'Destination' (s3://pcoady-backspace-intro-aws), 'Succeeded' (9 files, 11.8 MB (100.00%)), and 'Failed' (0 files, 0 B (0%)). Below the summary are two tabs: 'Files and folders' (selected) and 'Configuration'. The 'Files and folders' section shows 'Files and folders (9 Total, 11.8 MB)' with a search bar and a table of files.

Name	Folder	Type	Size	Status
aws1.mp3	Audio1/	audio/mpeg	2.8 MB	Succeeded

## Downloading files from your bucket

Click *Exit* to navigate back to the bucket details.

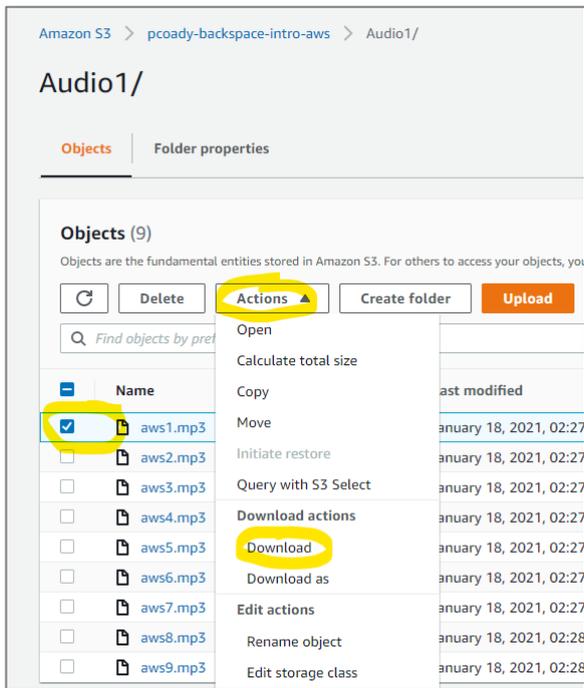
Click on the folder to view its contents

The screenshot shows the 'Objects' page in the AWS S3 console for the bucket 'pcoady-backspace-intro-aws'. The page has tabs for 'Objects', 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access points'. Below the tabs, there's a section for 'Objects (1)' with a description and a 'Learn more' link. There are buttons for 'Refresh', 'Delete', 'Actions', 'Create folder', and 'Upload'. A search bar is labeled 'Find objects by prefix'. Below the search bar is a table of objects.

<input type="checkbox"/>	Name	Type	Last modified	Size
<input type="checkbox"/>	Audio1/	Folder	-	

Select one of the files

Select *Actions* -> *Download*



## Troubleshooting

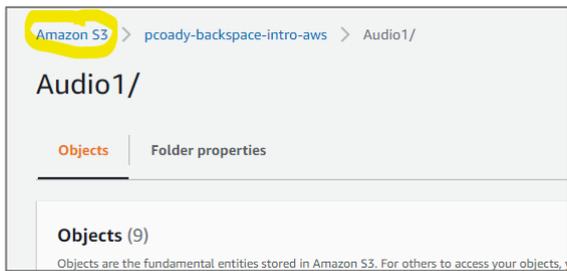
If you get the following screen it means you have clicked on the S3 URL and not the download link as detailed above. You cannot access files directly from a URL as they have private access.



## Clean Up

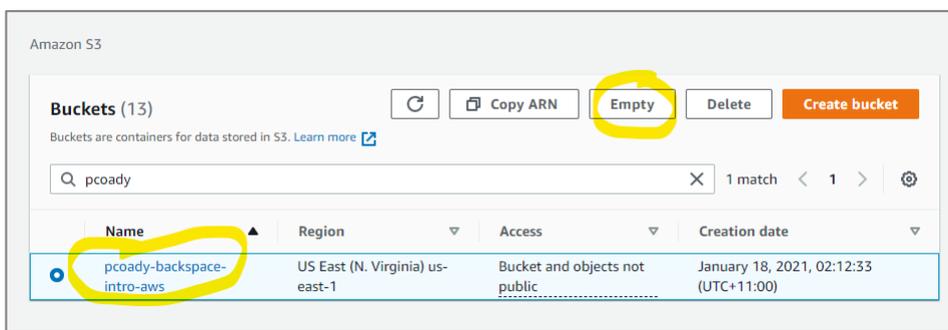
We will now delete the files and bucket so that you will not be billed by AWS.

Go back to the S3 dashboard.



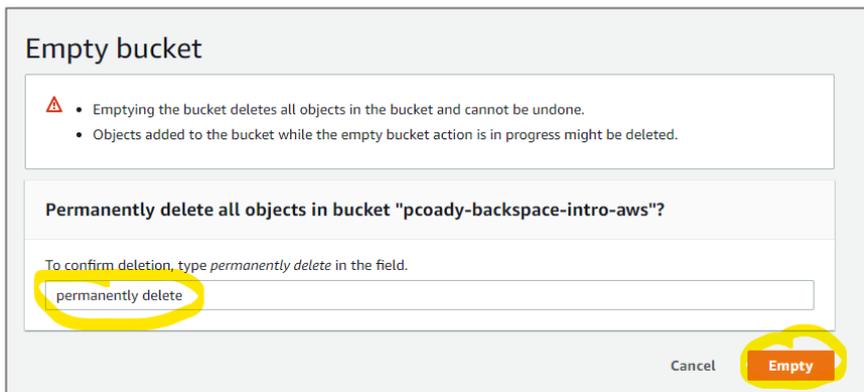
Select the bucket

Click *Empty*



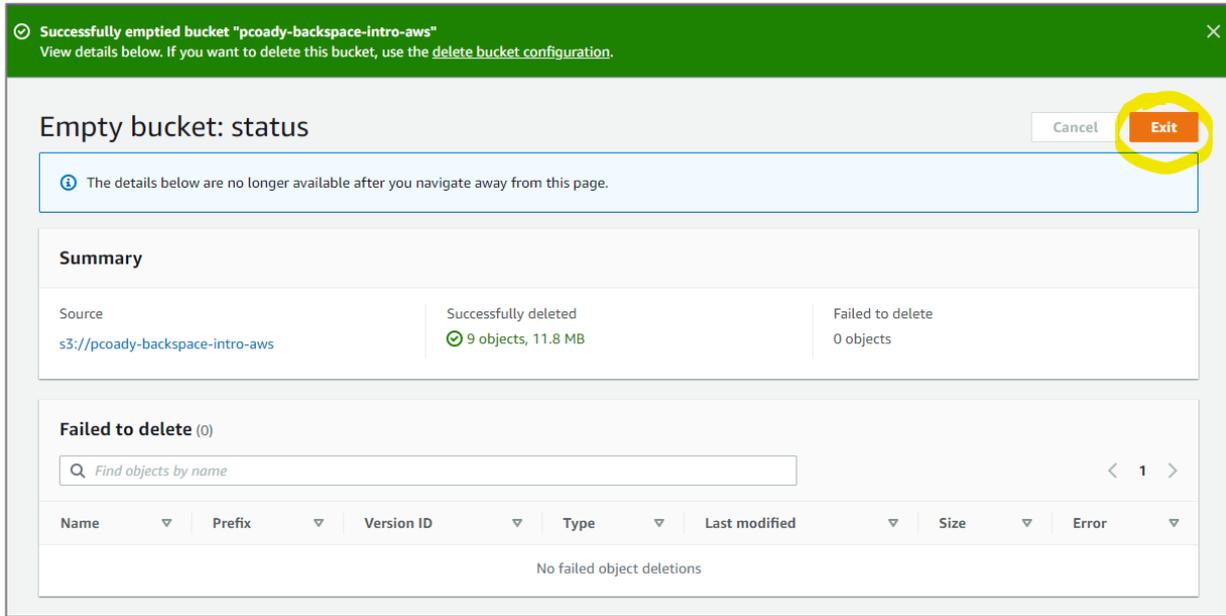
Enter *permanently delete*

Click *Empty*

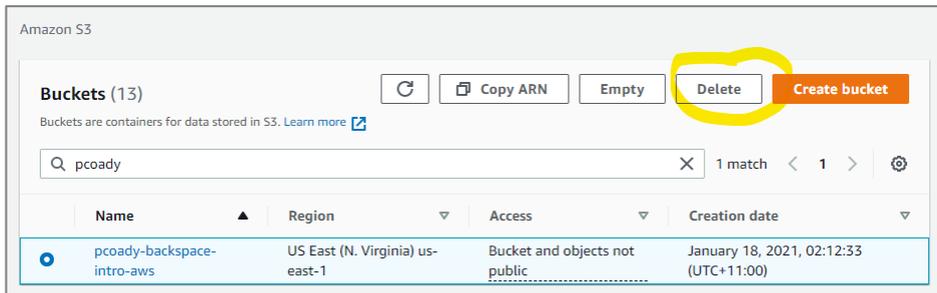


You will see a green success message

Click *Exit*

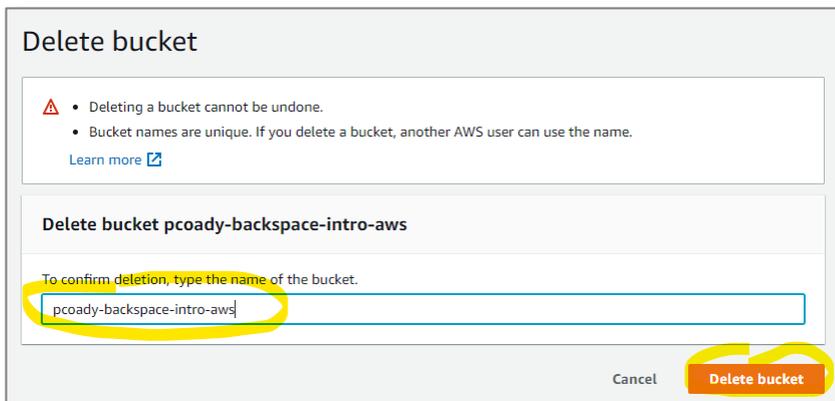


Click *Delete*



Confirm the name of the bucket to delete

Click *Delete bucket*



# ▶ Creating a SQL Database with RDS

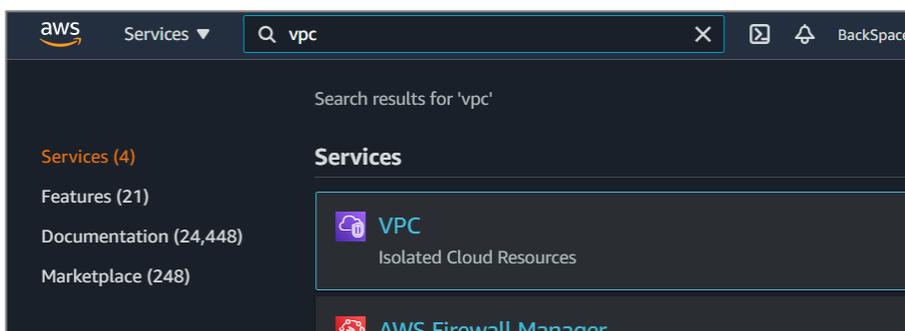
In this section, we will use the Relational Database Service to create a database. We will also connect into the database.

## Creating a Security Group

By default, inbound access from the Internet to our database instance is blocked. We will create a security group that defines an inbound rule that allows access from the Internet. We can then associate this security group to our database instance.

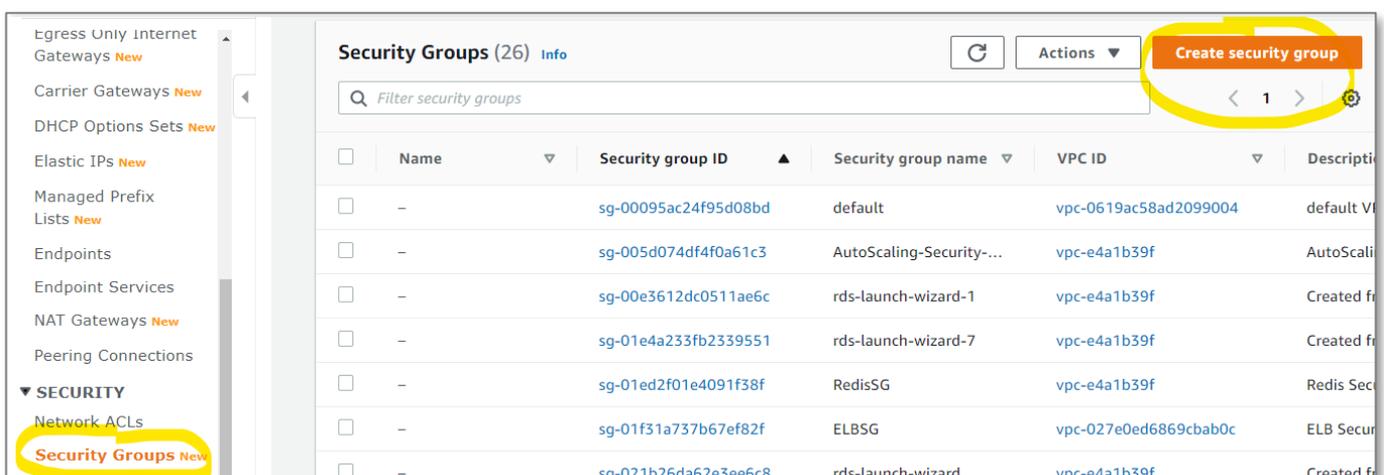
From the AWS console search *VPC*.

Select *VPC*



Scroll down and select *Security > Security Groups*

Click *Create security group*



Give it the name *backspace-rds-intro-lab*

Give it a description

**Create security group** [Info](#)

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To

**Basic details**

**Security group name** [Info](#)  
backspace-rds-intro-lab  
Name cannot be edited after creation.

**Description** [Info](#)  
Inbound internet access to MySQL RDS.

**VPC** [Info](#)  
vpc-e4a1b39f (Default VPC)

Click *Add rule* for Inbound rules

Select type *MySQL/Aurora*

Select source *Anywhere*

Click *Create security group*

### Inbound rules [Info](#)

Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>	Source <a href="#">Info</a>	Description - optional <a href="#">Info</a>
<input type="text" value="MYSQL/Aurora"/>	TCP	3306	<input type="text" value="Anywh..."/> <input type="text" value="0.0.0.0/0"/> <input type="text" value="::/0"/>	<input type="text"/>

### Outbound rules [Info](#)

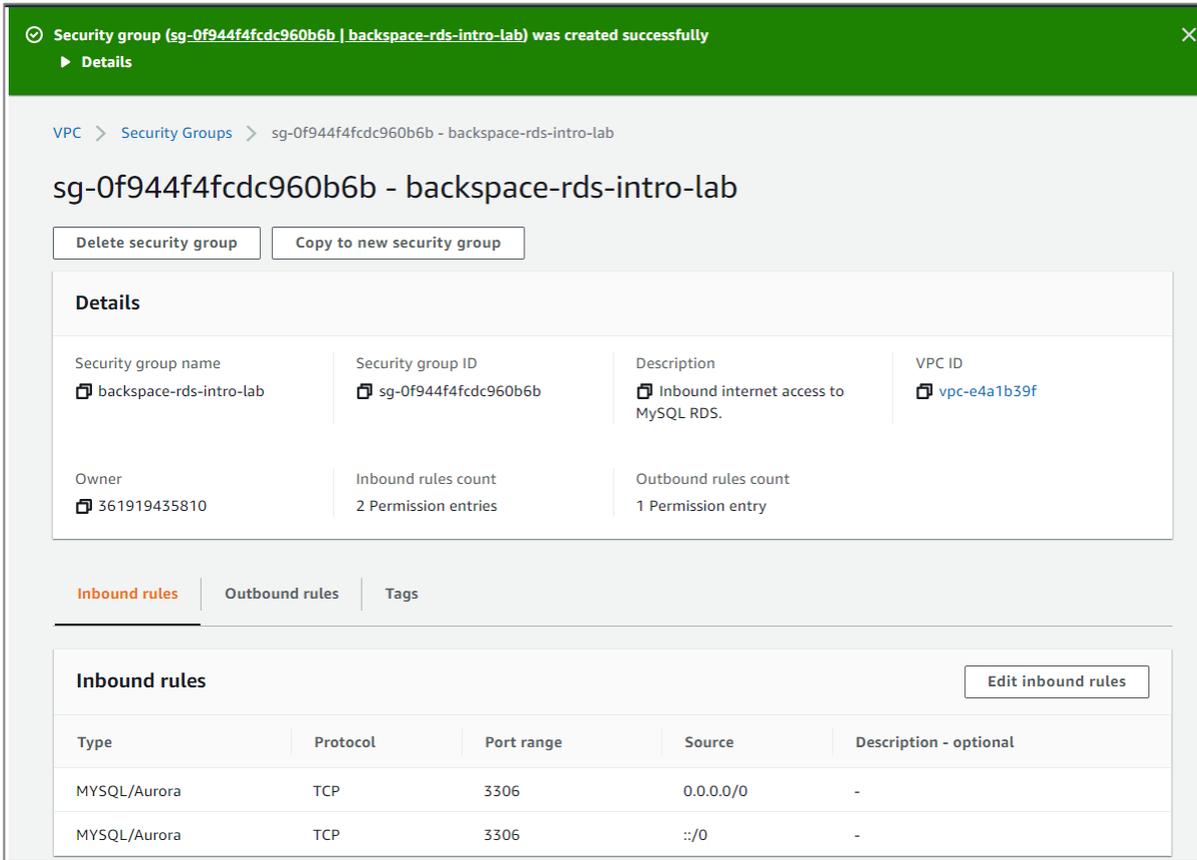
Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>	Destination <a href="#">Info</a>	Description - optional <a href="#">Info</a>
<input type="text" value="All traffic"/>	All	All	<input type="text" value="Custom"/> <input type="text" value="0.0.0.0/0"/>	<input type="text"/>

### Tags - optional

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

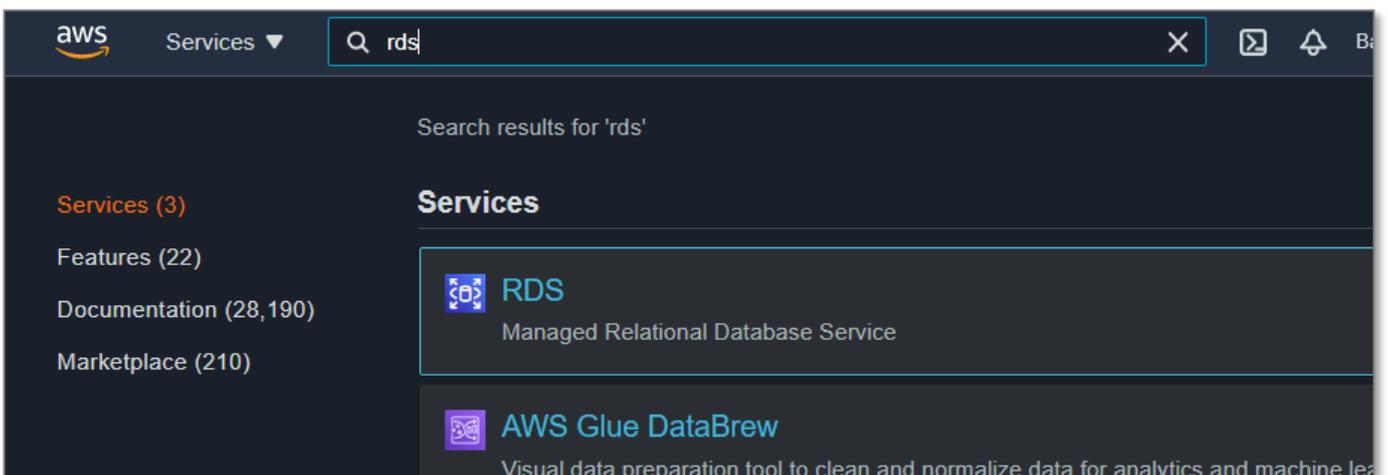
You can add up to 50 more tag



## Creating an RDS Database

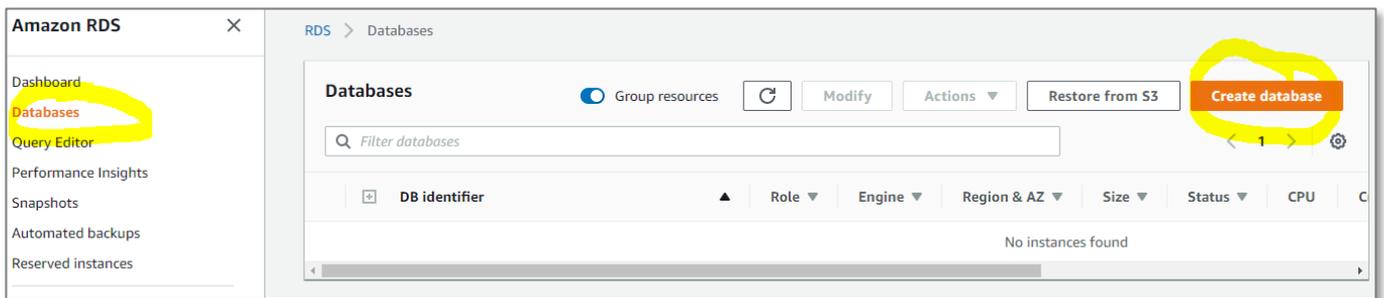
From the AWS console search *RDS*

Click on *RDS*

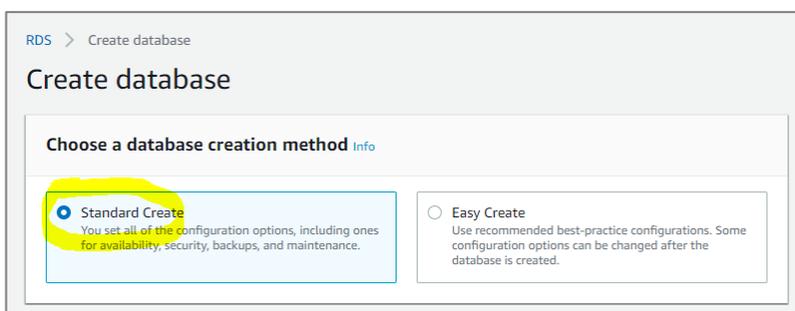


Select 'Databases'

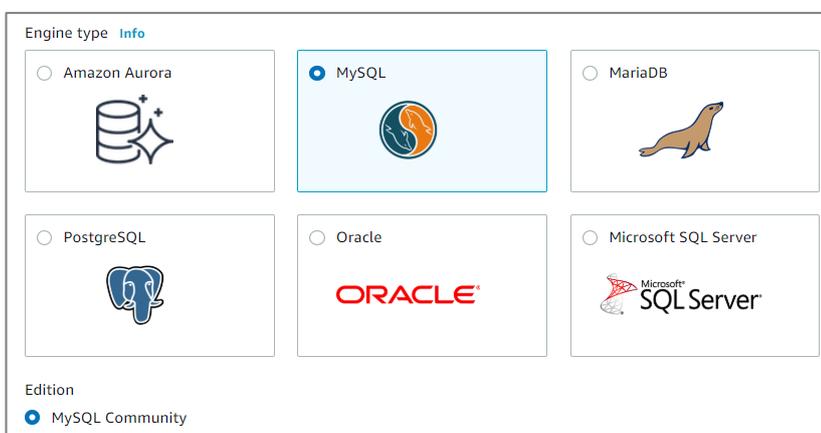
Select 'Create database'



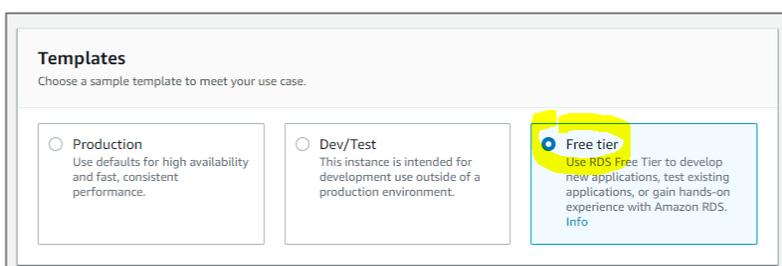
Select *Standard Create*



Select *MySQL*



Select *Free Tier*



In the *Settings* section give your instance a name/identifier.

Fill in a master username and password (remember this we will need it later)

**Settings**

**DB instance identifier** [Info](#)  
Type a name for your DB instance. The name must be unique cross all DB instances owned by your AWS account in the current AWS Region.

backspace-intro-aws

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ **Credentials Settings**

**Master username** [Info](#)  
Type a login ID for the master user of your DB instance.

admin

1 to 16 alphanumeric characters. First character must be a letter

**Auto generate a password**  
Amazon RDS can generate a password for you, or you can specify your own password

**Master password** [Info](#)  
\*\*\*\*\*

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), " (double quote) and @ (at sign).

**Confirm password** [Info](#)  
\*\*\*\*\*

In the *DB Instance size* section select db.t2.micro instance class

**DB instance size**

**DB instance class** [Info](#)  
Choose a DB instance class that meets your processing power and memory requirements. The DB instance class options below are limited to those supported by the engine you selected above.

- Standard classes (includes m classes)
- Memory Optimized classes (includes r and x classes)
- Burstable classes (includes t classes)**

db.t2.micro  
1 vCPUs 1 GiB RAM Not EBS Optimized

Include previous generation classes

Uncheck *Enable storage autoscaling*

**Storage**

**Storage type** [Info](#)  
General Purpose (SSD)

**Allocated storage**  
20 GiB  
(Minimum: 20 GiB, Maximum: 16384 GiB) Higher allocated storage may improve IOPS performance.

**Storage autoscaling** [Info](#)  
Provides dynamic scaling support for your database's storage based on your application's needs.

**Enable storage autoscaling**  
Enabling this feature will allow the storage to increase once the specified threshold is exceeded.

Scroll down to *Connectivity*

Expand *Additional connectivity configuration*

Select *yes for publicly accessible* (we will look at security later in the course)

Select *Choose existing* for VPC security group

Select the *backspace-rds-intro-lab* security group we created previously (click outside the list after selecting to close the list)

**Connectivity**

Virtual private cloud (VPC) [Info](#)  
VPC that defines the virtual networking environment for this DB instance.

Default VPC (vpc-e4a1b39f) ▼

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change the VPC selection.

▼ **Additional connectivity configuration**

Subnet group [Info](#)  
DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

default ▼

Public access [Info](#)

Yes  
Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

No  
RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

Existing VPC security groups

Choose VPC security groups ▲

Q |

- AutoScaling-Security-Group-1
- RedisSG
- WordPress Certified by Bitnami and Automattic-5-5-1-0 on Debian 10-AutogenByAWSMP-
- WebServerSG
- aws-cloud9-BackSpace-Labs-aa0e0177557d4b7da26fa3c1fe150530-InstanceSecurityGroup-1EQOMGDRQ5CW8
- LocalServerSG
- backspace-rds-intro-lab**
- default

Click outside the list to add the security group. You should then see the security group added.

Existing VPC security groups

Choose VPC security groups ▼

backspace-rds-intro-lab ✕ default ✕

Availability Zone [Info](#)

No preference ▼

Database port [Info](#)  
TCP/IP port that the database will use for application connections.

3306

Scroll down to *Database authentication*

Leave as *Password authentication*

**Database authentication**

Database authentication options [Info](#)

- Password authentication  
Authenticates using database passwords.
- Password and IAM database authentication  
Authenticates using the database password and user credentials through AWS IAM users and roles.
- Password and Kerberos authentication (not available for this version)  
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

Scroll down and expand *Additional configuration*

Enter a database name.

Uncheck *Enable automatic backups*

Leave all other options default.

**Additional configuration**  
Database options, backup disabled, backtrack disabled, Enhanced Monitoring disabled, maintenance, CloudWatch Logs, delete protection disabled

**Database options**

**Initial database name** [Info](#)  
test  
If you do not specify a database name, Amazon RDS does not create a database.

**DB parameter group** [Info](#)  
default.mysql5.7

**Option group** [Info](#)  
default.mysql-5-7

**Backup**  
Creates a point in time snapshot of your database

**Enable automatic backups**  
Enabling backups will automatically create backups of your database during a certain time window.

**Monitoring**

**Enable Enhanced monitoring**  
Enabling Enhanced monitoring metrics are useful when you want to see how different processes or threads use the CPU

Uncheck *Enable deletion protection* (we want to delete it easily when finished)

Click *Create database*

**Deletion protection**

**Enable deletion protection**  
Protects the database from being deleted accidentally. While this option is enabled, you can't delete the database.

**Estimated monthly costs**

The Amazon RDS Free Tier is available to you for 12 months. Each calendar month, the free tier will allow you to use the Amazon RDS resources listed below for free:

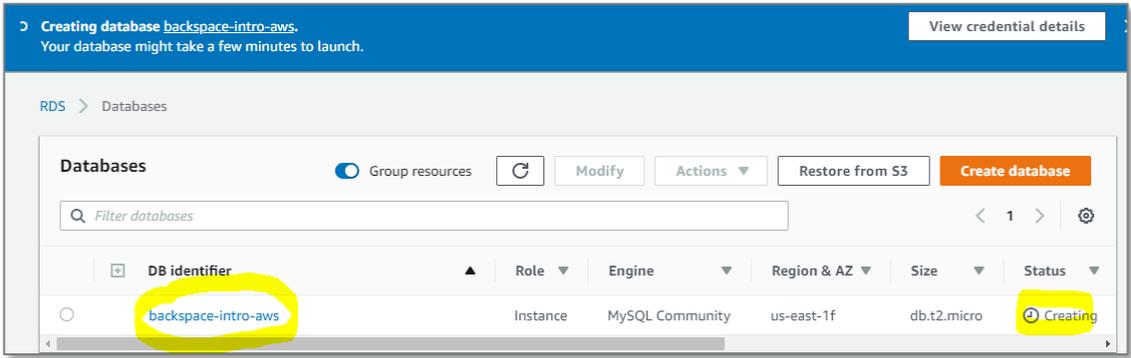
- 750 hrs of Amazon RDS in a Single-AZ db.t2.micro Instance.
- 20 GB of General Purpose Storage (SSD).
- 20 GB for automated backup storage and any user-initiated DB Snapshots.

[Learn more about AWS Free Tier.](#)

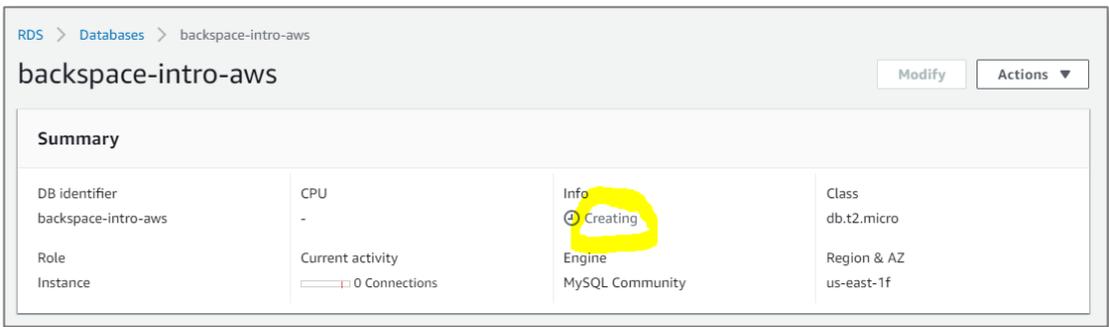
When your free usage expires or if your application use exceeds the free usage tiers, you simply pay standard, pay-as-you-go service rates as described in the [Amazon RDS Pricing page.](#)

Cancel **Create database**

Click on the database details link



Your instance will show status 'creating'.



## Connecting to your RDS Instance

To connect to your MySQL Database you will need to download and install the MySQL Workbench.

Instructions for Windows:

<https://dev.mysql.com/doc/workbench/en/wb-installing-windows.html>

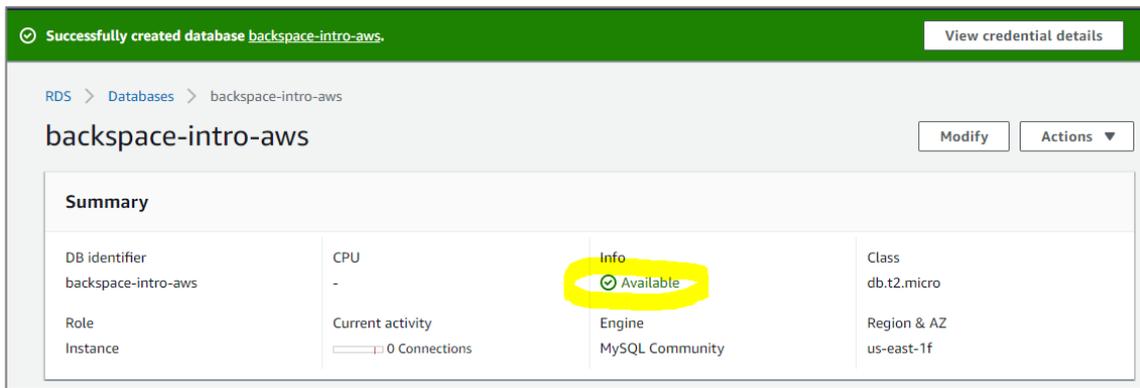
Instructions for Mac:

<https://dev.mysql.com/doc/workbench/en/wb-installing-mac.html>

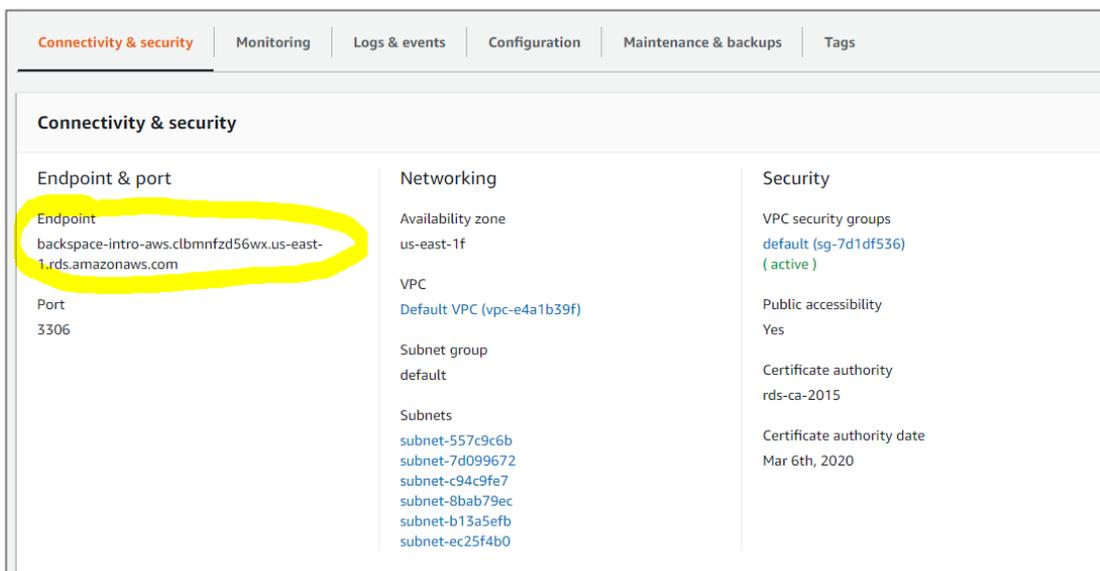
Instructions for Linux:

<https://dev.mysql.com/doc/workbench/en/wb-installing-linux.html>

Wait for your instance status to be 'available'



Scroll down and copy the database server endpoint



Open the MySQL Workbench application click to add a new connection



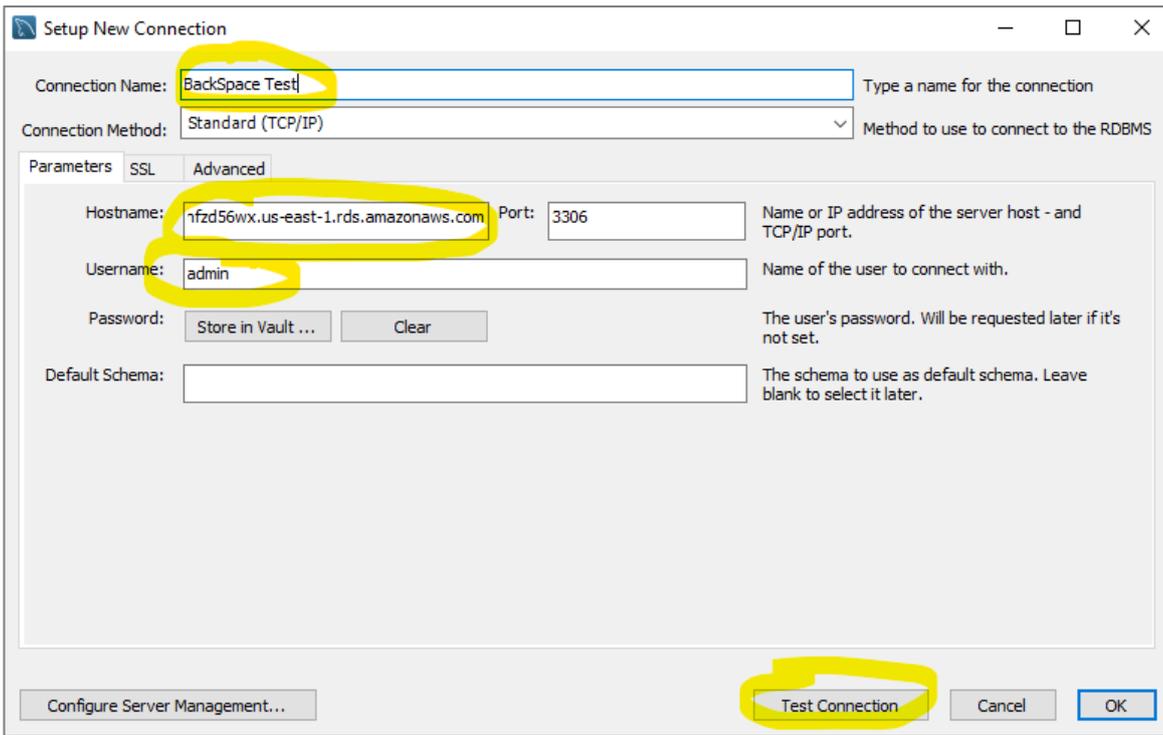
Give the connection a name.

The Hostname will be the RDS server endpoint.

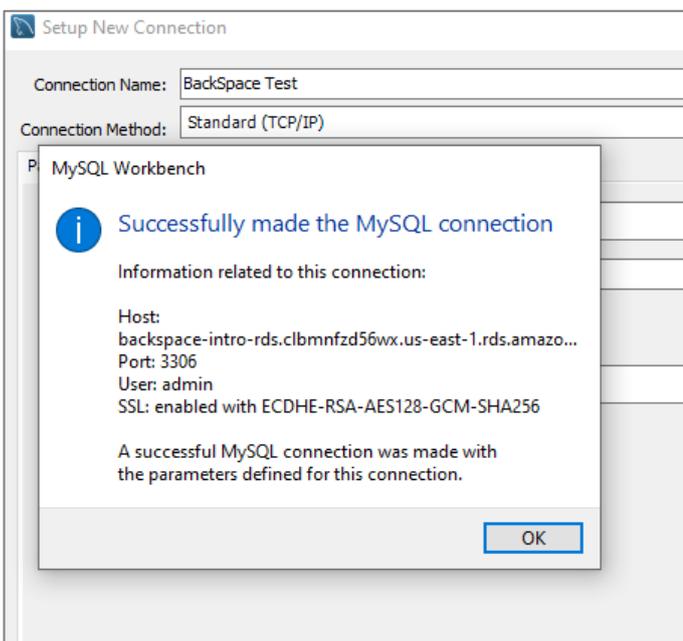
The port will be 3306.

The Username will be the master username we created in RDS (i.e. admin)

Click *Test Connection*

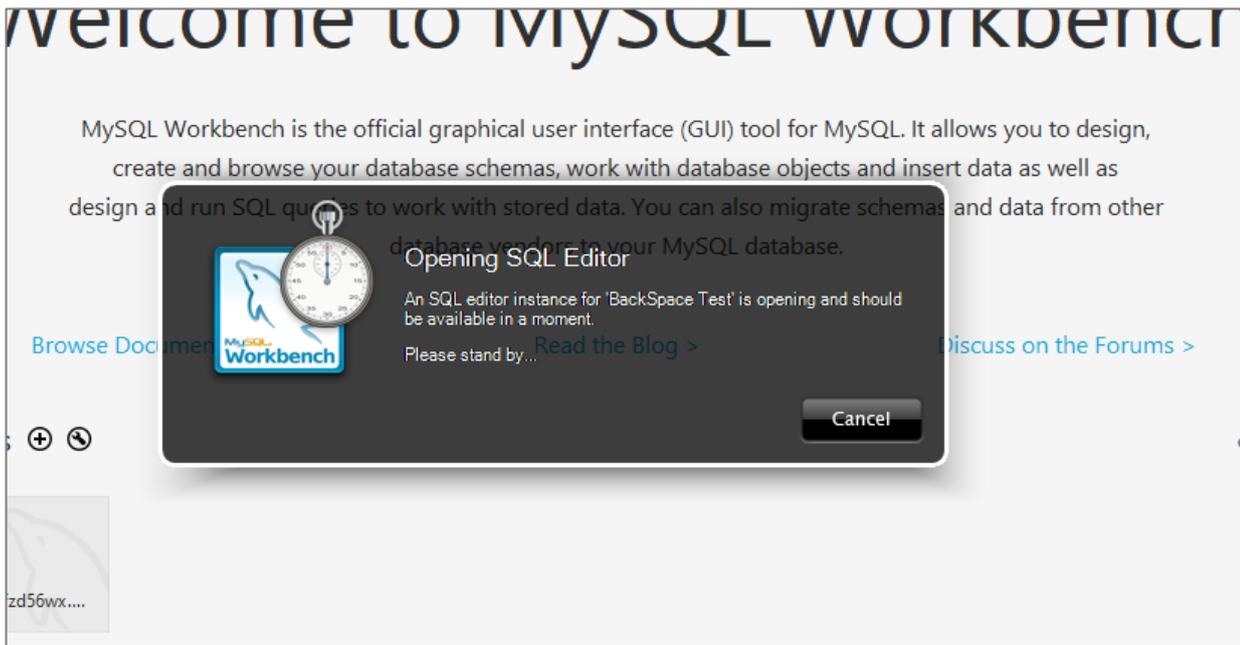


Enter the password you created in RDS for your master username



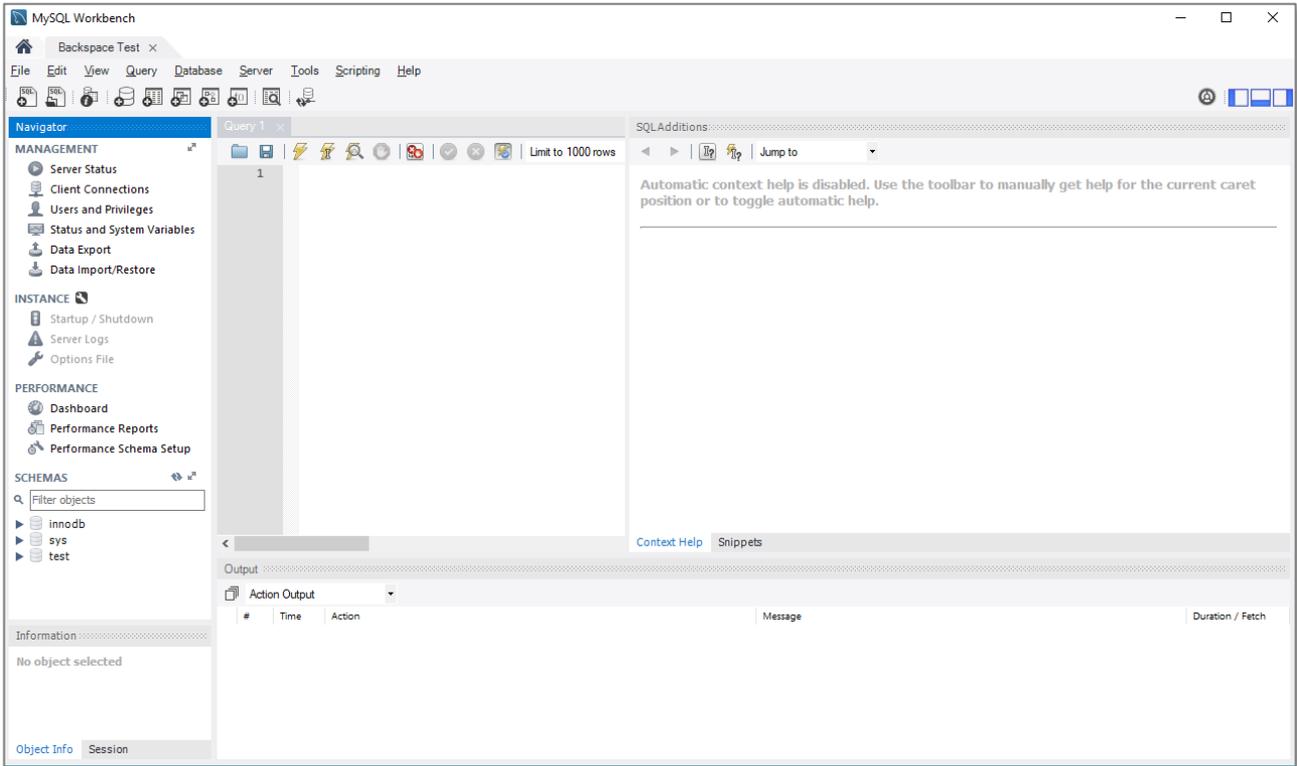
Click *OK*

Click on the Connection

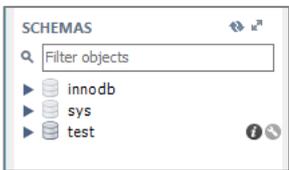


You will soon be connected to your database server

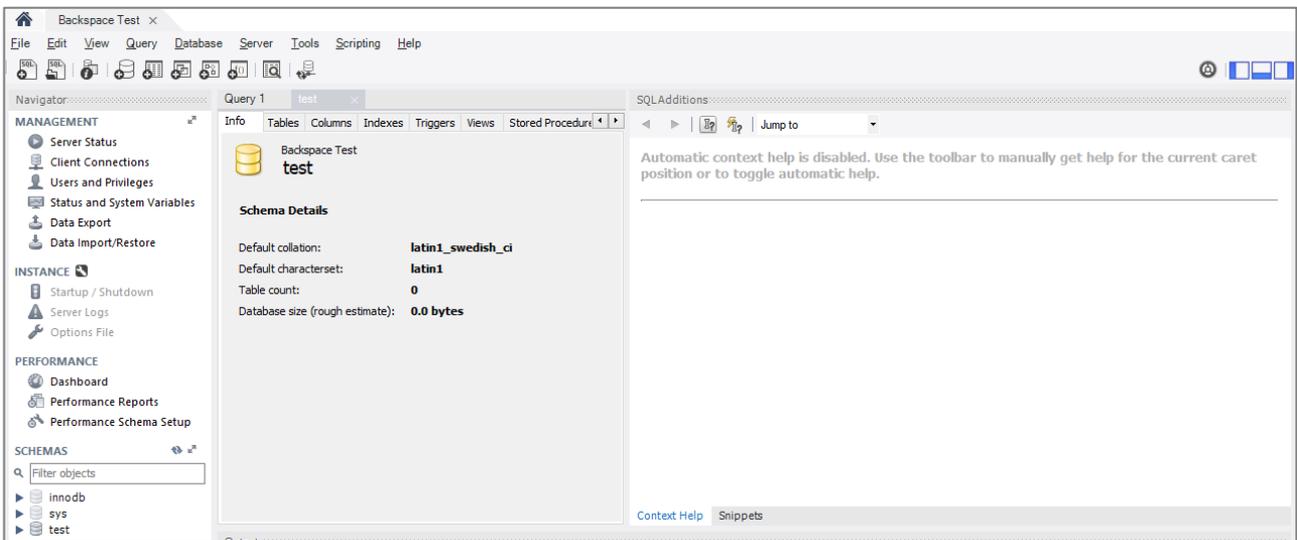
If you cannot connect then please see the "Troubleshooting Connection Issues" below.



Hover over the 'test' database under 'SCHEMAS' and click the information icon to get information about the database that was created by us in RDS.



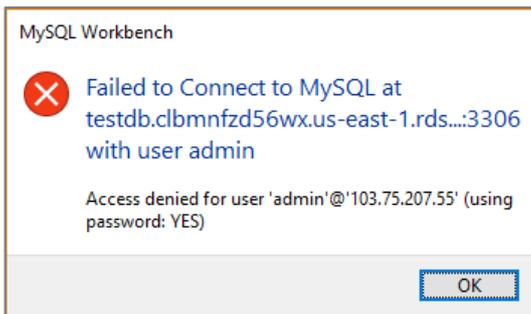
You then get an information screen for the database.



## Troubleshooting Connection Issues

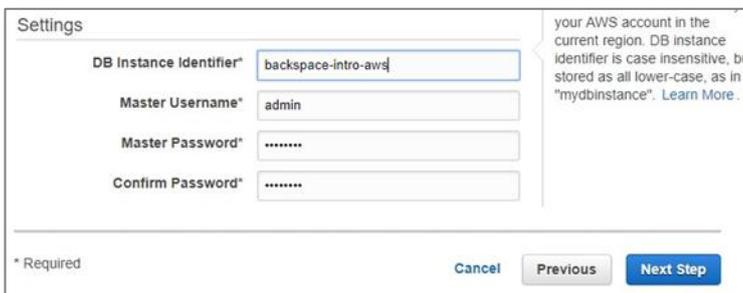
If you are getting connection errors then check the following:

### Wrong Username / Password

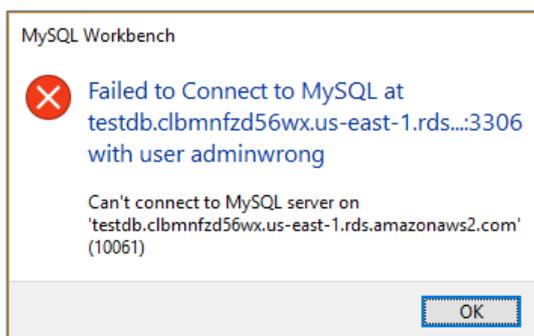


Make sure you use the correct username and password.

The username and password must be the one created when the RDS instance was created.



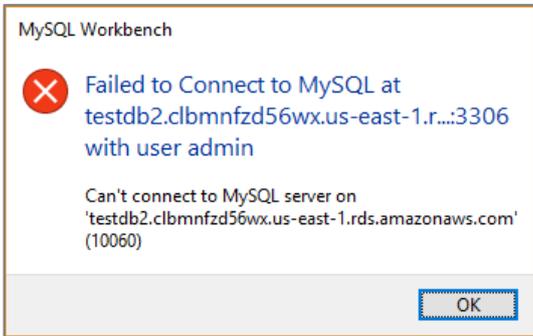
### Bad Connection String



This error means nothing exists at the endpoint. Check the connection endpoint and port are correct.

The hostname will be the RDS Instance Connection Endpoint without :3306 on the end.

## No Connection



This error means your server exists but you are unable to connect to it. This can be caused by:

- You have not selected 'public' when creating instance and the security group inbound rules will be incorrect. This will block traffic to your instance. See *Security Group Inbound Rules* below.
- You have a dynamic IP address or multiple IP addresses passing through a load balancer. See *Security Group Inbound Rules* below.
- Firewall at your end is blocking access to port 3306. See *Client-side Firewall* below.

## Security Group Inbound Rules

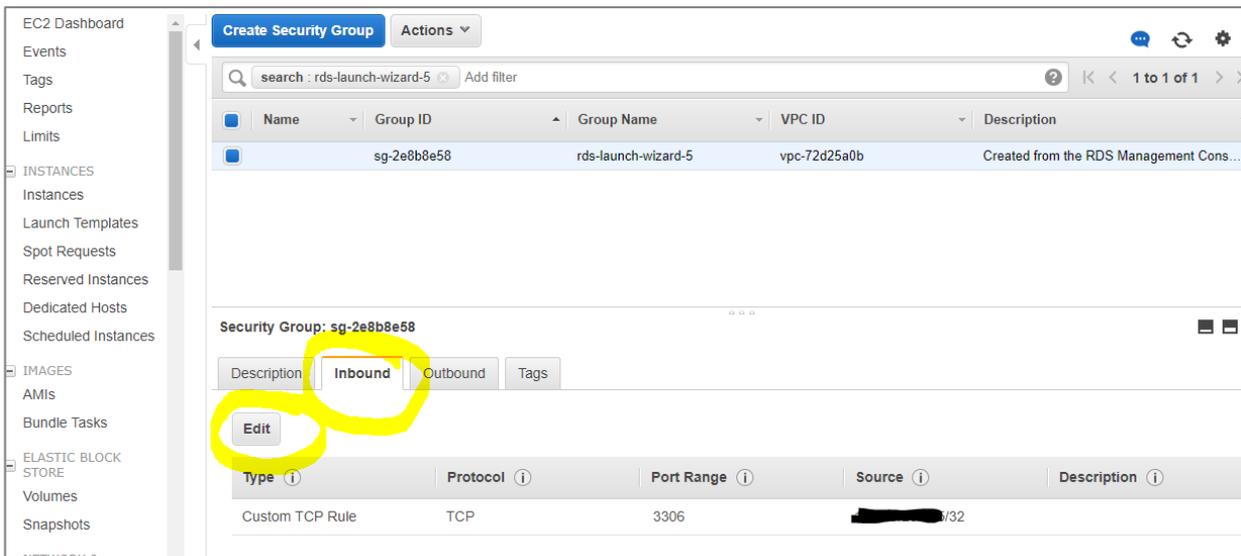
If you did not **select yes for publicly accessible** as detailed, your security group will block remote access. The security group may have an inbound rule for your IP address. If you are using a dynamic IP address or you are connecting from different networks then this will need to be changed to "anywhere" for the lab.

Click the security group

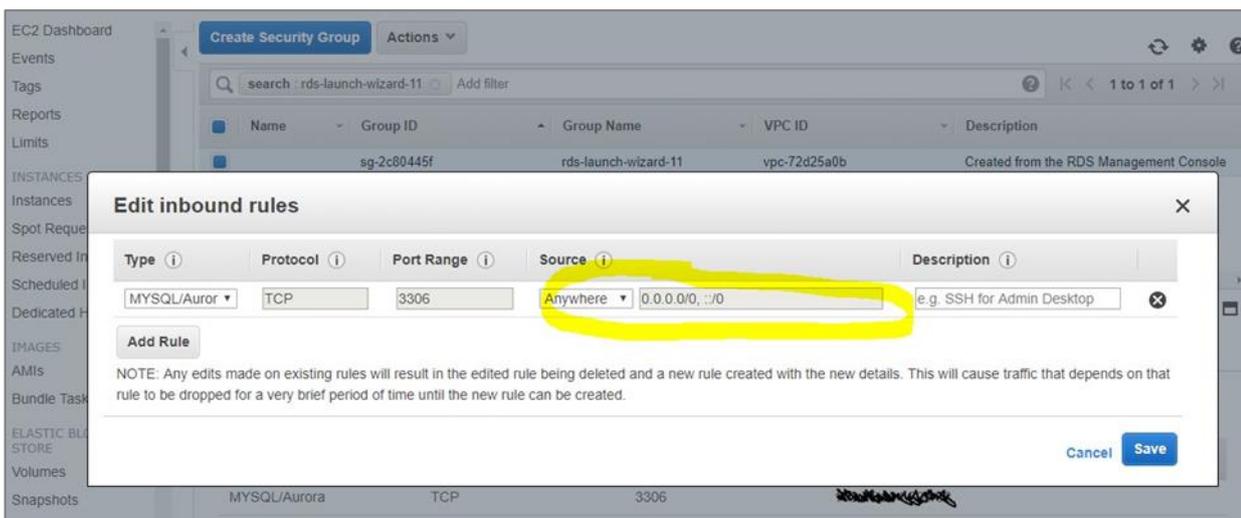


You will be taken to the EC2 console

Select the “Inbound” tab  
Click “Edit”



Change inbound rule to “Anywhere” 0.0.0.0/0, :::0



### Client-side Firewall

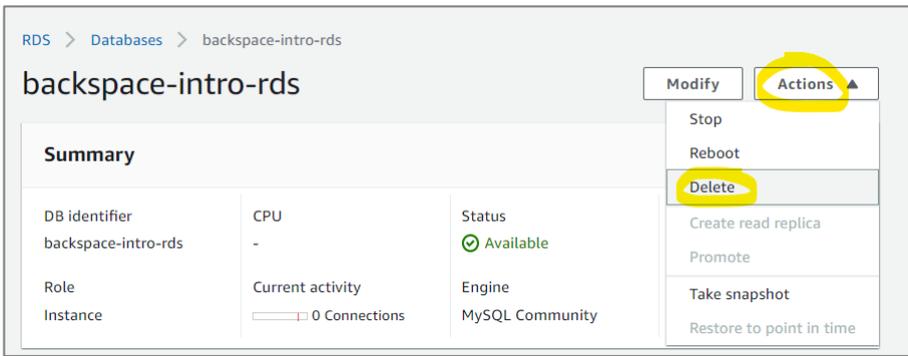
If you are still having problems connecting, a firewall at your end may be preventing access on port 3306. This is common if you are connecting from your work environment as port 3306 traffic may be blocked.

### Clean Up

To avoid incurring charges from AWS we will terminate the instance.

Go back to the RDS console.

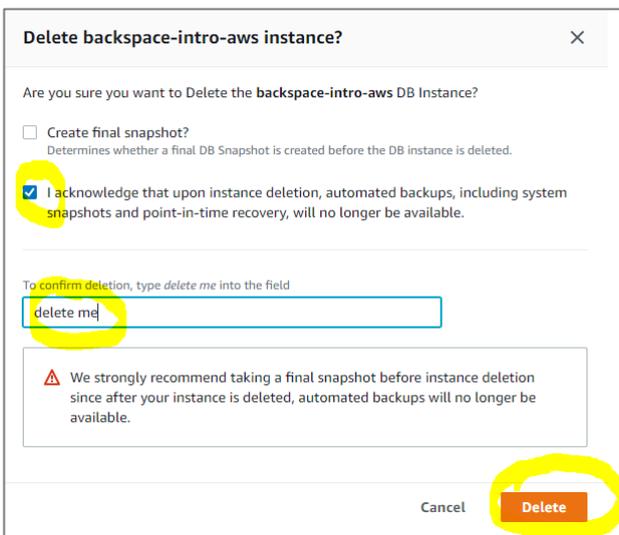
Click *Actions* -> *Delete* to terminate the instance



Select 'No' for 'Create final snapshot'

Check 'I acknowledge that upon instance deletion, automated backups, including system snapshots and point-in-time recovery, will no longer be available.'

Click 'Delete'



Click on the VPC security group we created previously

RDS > Databases > backspace-intro-rds

## backspace-intro-rds

Modify Actions

### Summary

DB identifier backspace-intro-rds	CPU 2.71%	Status Deleting	Class db.t2.micro
Role Instance	Current activity 0 Connections	Engine MySQL Community	Region & AZ us-east-1c

Connectivity & security | Monitoring | Logs & events | Configuration

Maintenance & backups | Tags

### Connectivity & security

Endpoint & port Endpoint backspace-intro-rds.clbmfzd56wx.us-east-1.rds.amazonaws.com Port	Networking Availability zone us-east-1c VPC Default VPC (vpc-e4a1b39f)	Security VPC security groups backspace-rds-intro-lab (sg-09ab0f7266fac969b) (active) default (sg-7d1df536) (active)
--	--	--

Select Actions -> Delete security group

Security Groups (1/1) Info

Actions Create security group

Filter security groups

search: sg-09ab0f7266fac969b Clear filters

<input checked="" type="checkbox"/>	Name	Security group ID		ID
<input checked="" type="checkbox"/>	-	sg-09ab0f7266fac969b	backspace-rds-intro-lab	vpc-e4a1b39f

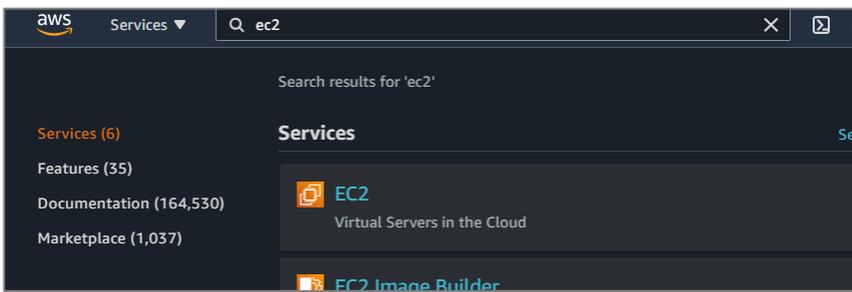
Actions menu items: Manage tags, Manage stale rules, Copy to new security group, Delete security group

# ▶ Creating a Web Server with EC2

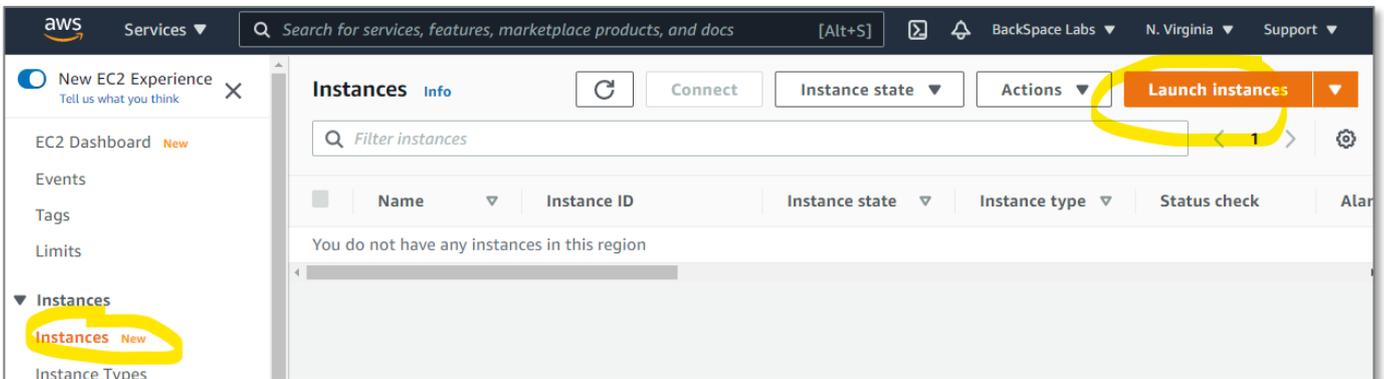
In this section, we will launch a publicly accessible WordPress application on Amazon EC2.

From the AWS console search *EC2*

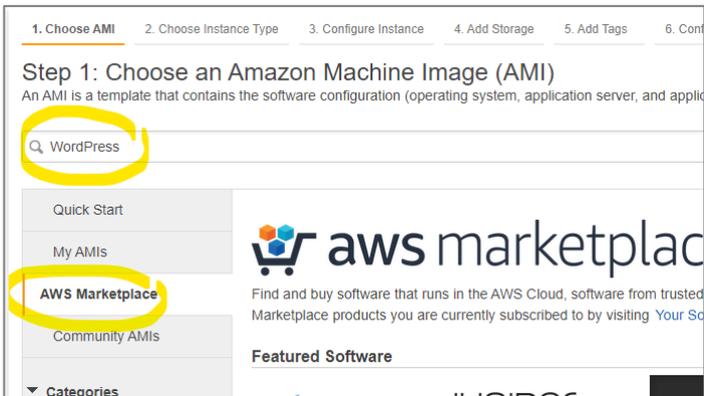
Click *EC2*



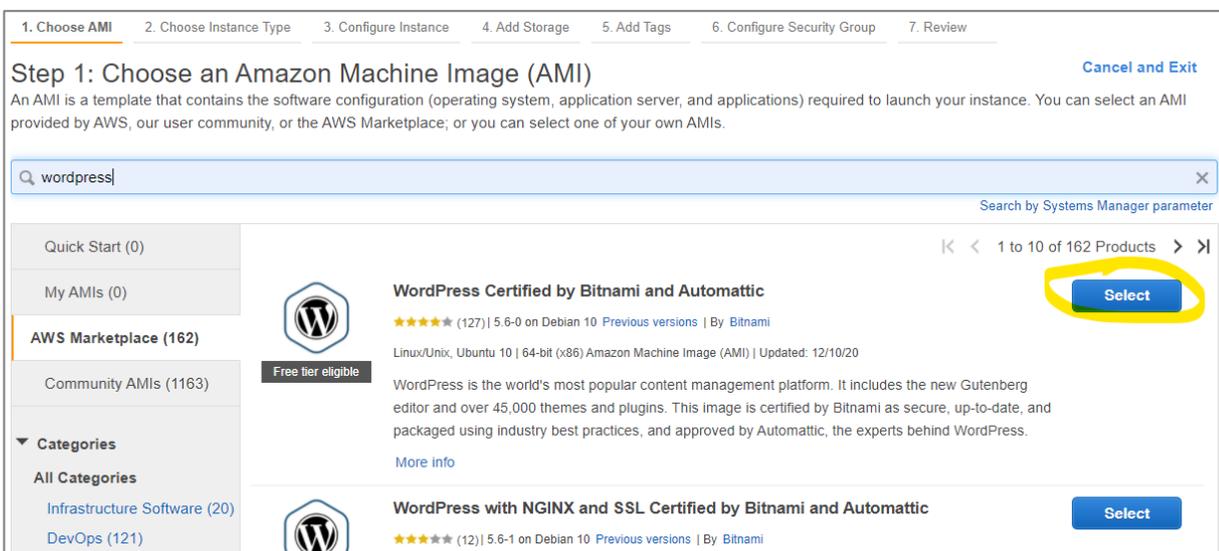
Select *Instances* -> *Launch Instances*



Select the *AWS Marketplace* and search for *WordPress*



Select the Bitnami AMI



Click *Continue* (This lab will be covered under the AWS free tier for accounts less than 12 months old)

### WordPress Certified by Bitnami



**WordPress Certified by Bitnami**

WordPress powers over 25% of all websites on the internet, making it the world's most popular blogging and content management platform. It is free and open source software developed entirely by its community, who have contributed over 45,000 themes, plugins, and widgets that enable an unlimited combination of features. Users can easily create and ...

[More info](#)  
[View Additional Details in AWS Marketplace](#)

**Product Details**

By Bitnami

Customer Rating ★★★★★ (117)

Latest Version 5.4.1-0-r01 on Debian 10

Base Operating System Linux/Unix, Debian 10

Delivery Method 64-bit (x86) Amazon Machine Image (AMI)

License Agreement [End User License Agreement](#)

On Marketplace Since 9/17/14

**Highlights**

- Jetpack plugin is included by default offering access to additional professional themes, performance improvements and marketing tools.

**Pricing Details**

**Hourly Fees**

Instance Type	Software	EC2	Total
t2.micro	\$0.00	\$0.012	\$0.012/hr
t2.small	\$0.00	\$0.023	\$0.023/hr
t2.medium	\$0.00	\$0.046	\$0.046/hr
t2.large	\$0.00	\$0.093	\$0.093/hr
t2.xlarge	\$0.00	\$0.186	\$0.186/hr
t2.2xlarge	\$0.00	\$0.371	\$0.371/hr
t3a.micro	\$0.00	\$0.009	\$0.009/hr
t3a.small	\$0.00	\$0.019	\$0.019/hr
t3a.medium	\$0.00	\$0.038	\$0.038/hr
t3a.large	\$0.00	\$0.075	\$0.075/hr
t3a.xlarge	\$0.00	\$0.15	\$0.15/hr
t3a.2xlarge	\$0.00	\$0.301	\$0.301/hr
t3.micro	\$0.00	\$0.01	\$0.01/hr
t3.small	\$0.00	\$0.021	\$0.021/hr
t3.medium	\$0.00	\$0.042	\$0.042/hr
t3.large	\$0.00	\$0.083	\$0.083/hr
t3.xlarge	\$0.00	\$0.166	\$0.166/hr
t3.2xlarge	\$0.00	\$0.333	\$0.333/hr

[Cancel](#) [Continue](#)

Choose the t2 micro instance.

Click Next: Configure Instance Details

1. Choose AMI   2. **Choose Instance Type**   3. Configure Instance   4. Add Storage   5. Add Tags   6. Configure Security Group   7. Review

### 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.

Filter by: All instance types Current generation Show/Hide Columns

**Currently selected:** t3a.small (Variable ECUs, 2 vCPUs, 2.2 GHz, AMD EPYC 7571, 2 GiB memory, EBS only)

**Note:** The vendor recommends using a **t3a.small** instance (or larger) for the best experience with this product.

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="radio"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="radio"/>	General purpose	t2.micro <span style="font-size: x-small; color: green;">Free tier eligible</span>	1	1	EBS only	-	Low to Moderate	Yes
<input type="radio"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes

[Cancel](#)   [Previous](#)   [Review and Launch](#)   [Next: Configure Instance Details](#)

Select enable for Auto-assign Public IP

Click Next: Add Storage

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

**Number of instances**  [Launch into Auto Scaling Group](#)

**Purchasing option**  Request Spot instances

**Network** vpc-e4a1b39f | Default VPC (default) [Create new VPC](#)

**Subnet** No preference (default subnet in any Availability Zone) [Create new subnet](#)

**Auto-assign Public IP** **Enable**

**Placement group**  Add instance to placement group

**Capacity Reservation** Open [Create new Capacity Reservation](#)

**IAM role** None [Create new IAM role](#)

**Shutdown behavior** Stop

**Stop - Hibernate behavior**  Enable hibernation as an additional stop behavior

**Enable termination protection**  Protect against accidental termination

**Monitoring**  Enable CloudWatch detailed monitoring

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

Click Next: Add Tags

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

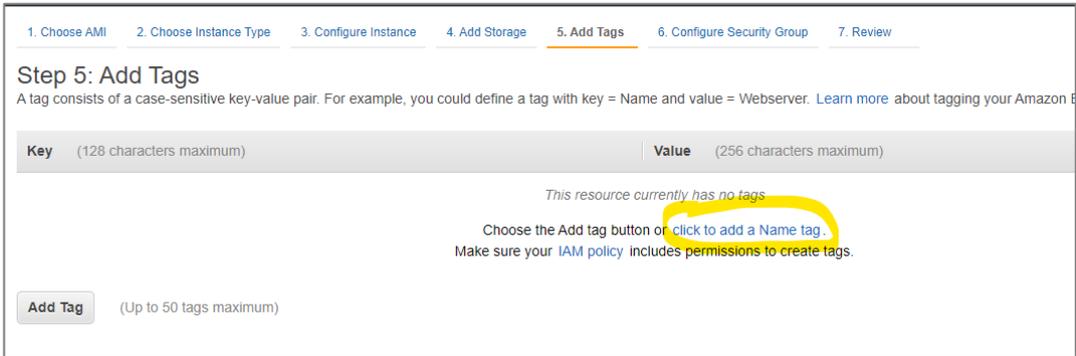
Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-01235d3cf67b2c8f8	10	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

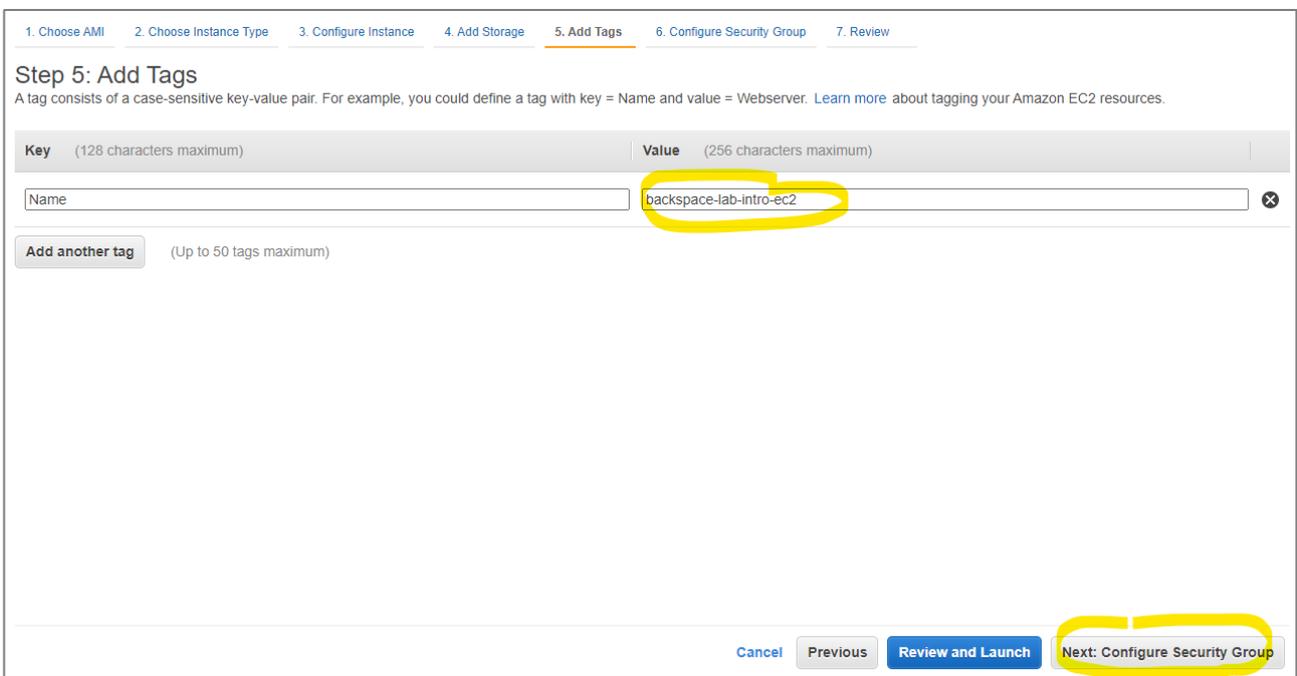
Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)

Click to add a Name tag



Give it a name and click Next: Configure Security Group



Click Review and Launch

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group:  Create a **new** security group  
 Select an **existing** security group

Security group name:

Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
HTTP	TCP	80	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
HTTPS	TCP	443	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

**Warning**  
 Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

## Click Launch

### 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.

**Warning**  
 Improve your instances' security. Your security group, WordPress Certified by Bitnami and Automattic-5-6-0 on Debian 10-AutogenByAWSMP-, is open to the world.  
 Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only. You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

**AMI Details** [Edit AMI](#)

 **WordPress Certified by Bitnami and Automattic**  
 This image may not be the latest version available and might include security vulnerabilities. Please check the latest, up-to-date, available version at <https://bitnami.com/stacks>.  
 Root Device Type: ebs Virtualization type: hvm

**Free tier eligible**

**Hourly Software Fees:** \$0.00 per hour on t2.micro instance. Additional taxes or fees may apply.  
 Software charges will begin once you launch this AMI and continue until you terminate the instance.

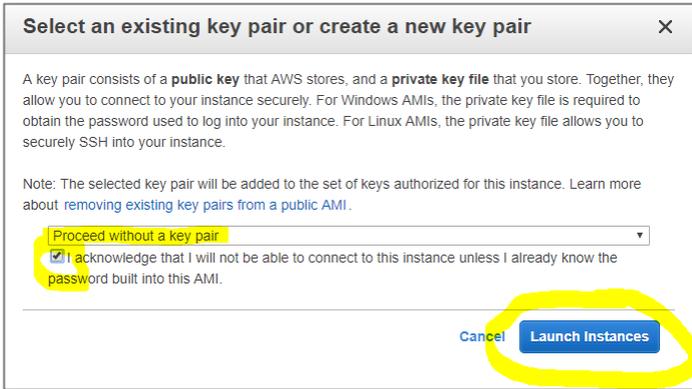
If you have an existing license entitlement to use this software, then you can launch this software without creating a new subscription. If you do not have an existing entitlement, then by launching this software, you will be subscribed to this software and agree that your use of this software is subject to the pricing terms and the seller's [End User License Agreement](#)

**Instance Type** [Edit instance type](#)

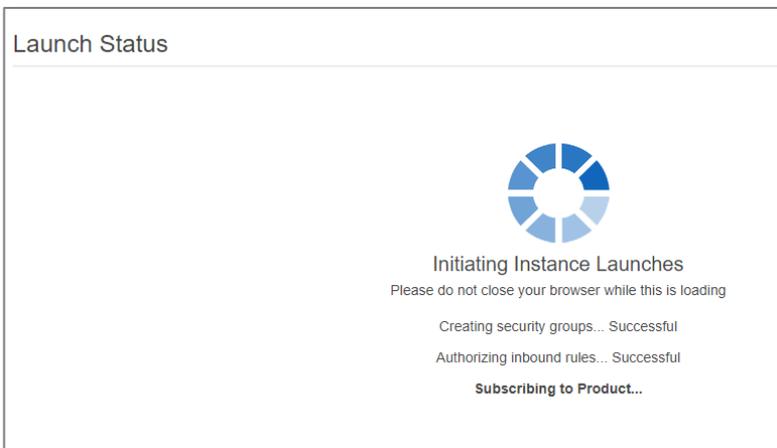
Select *Proceed without a key pair*

Select "I acknowledge that I will not be able to connect to this instance unless I already know the password built into this AMI."

Click *Launch Instances*



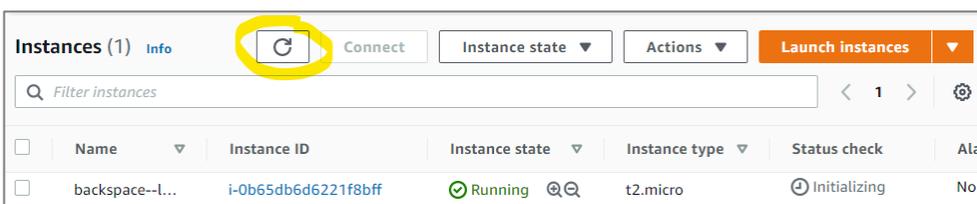
Wait for launch to initiate



When the launch process has started scroll to the bottom of the page and click "View Instances"

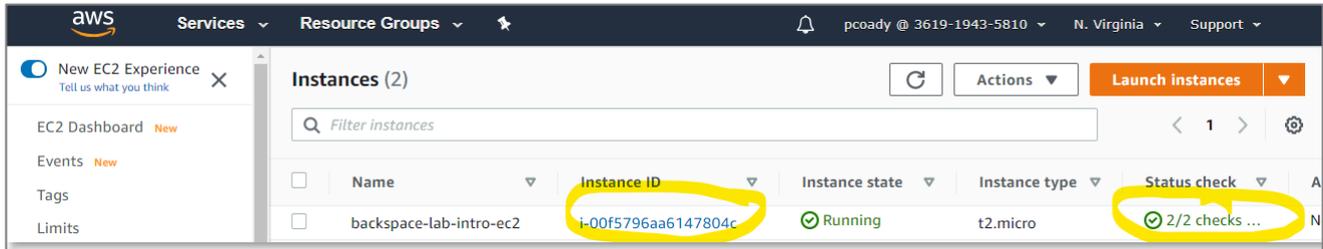


After a few minutes, the status of the instance will change to running and status checks will be completed (you will need to refresh the screen to see any changes).

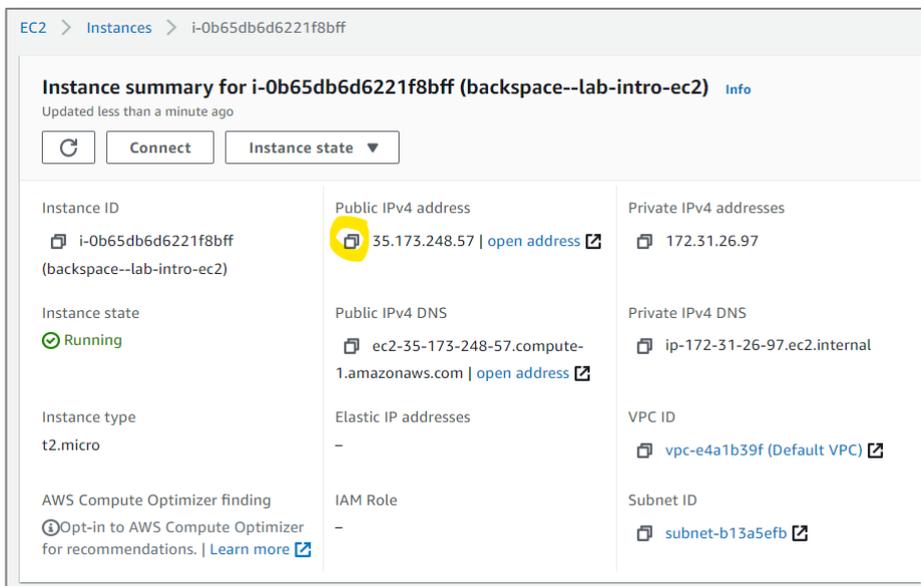


## Viewing your web server

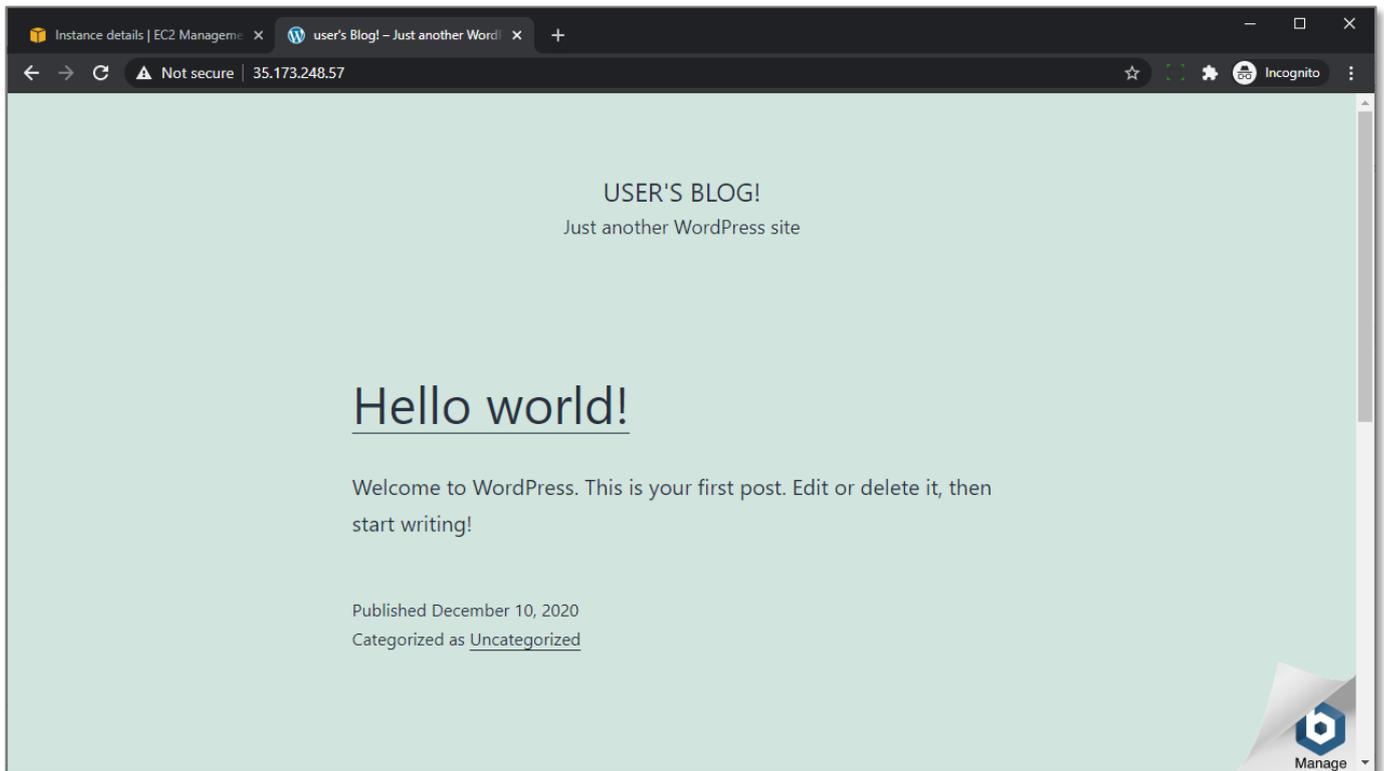
After the Status checks have completed click on the Instance ID to select the instance.



Copy the public IP address of your web server (don't click on *open address*).



Paste the IP address in your browser.



## Troubleshooting viewing your WordPress application

If you cannot view your website it probably hasn't finished the launch process completely.

If you navigate to your website and it displays a security message, you have tried to open with https not http.

If after quite some time you still can't view your website, it may be that your security group does not allow inbound requests on port 80 (http). The inbound rules should include:

```
80    tcp    0.0.0.0/0
```

Scroll down and click on the Security tab

Details | **Security** | Networking | Storage | Monitoring | Usage instructions | Tags

▼ Security details

IAM Role: -

Owner ID: 361919435810

Launch time: Tue Sep 01 2020 04:18:12 GMT+1000 (Australian Eastern Standard Time)

Security groups: sg-03dd052bbd08c91ff (WordPress Certified by Bitnami and Automatic-5-5-1 on Debian 10-AutogenByAWSMP-)

▼ Inbound rules

Filter rules

Port range	Protocol	Source	Security groups	WordPress
80	TCP	0.0.0.0/0	WordPress Certified by Bitnami and ...	true
22	TCP	0.0.0.0/0	WordPress Certified by Bitnami and ...	true
443	TCP	0.0.0.0/0	WordPress Certified by Bitnami and ...	true

If the rule is not present you will need to add it by clicking on the security group to open it:

Details | **Security** | Networking | Storage | Monitoring | Usage instructions | Tags

▼ Security details

IAM Role: -

Owner ID: 361919435810

Launch time: Tue Sep 01 2020 04:18:12 GMT+1000 (Australian Eastern Standard Time)

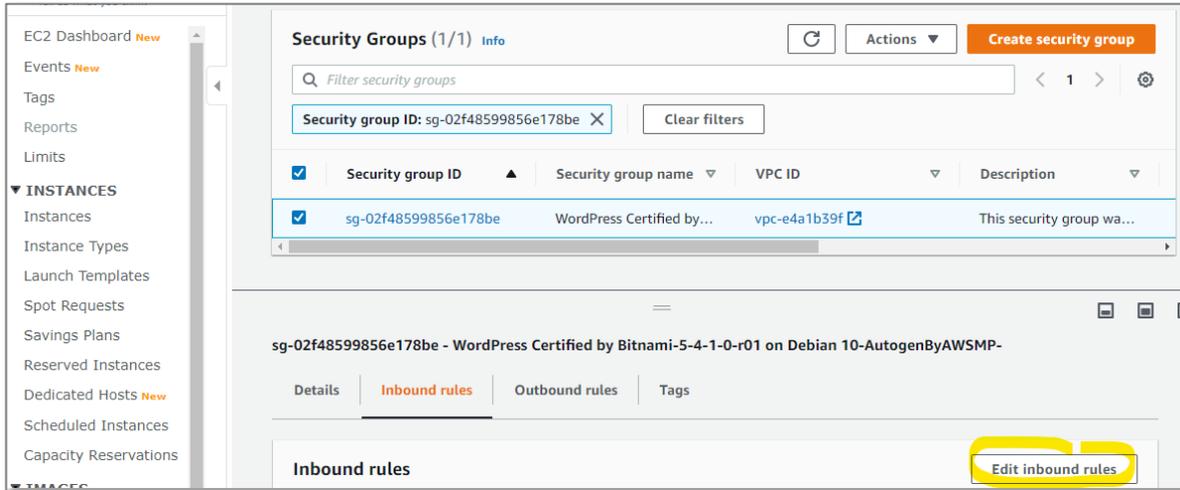
Security groups: [sg-03dd052bbd08c91ff \(WordPress Certified by Bitnami and Automatic-5-5-1 on Debian 10-AutogenByAWSMP-\)](#)

▼ Inbound rules

Filter rules

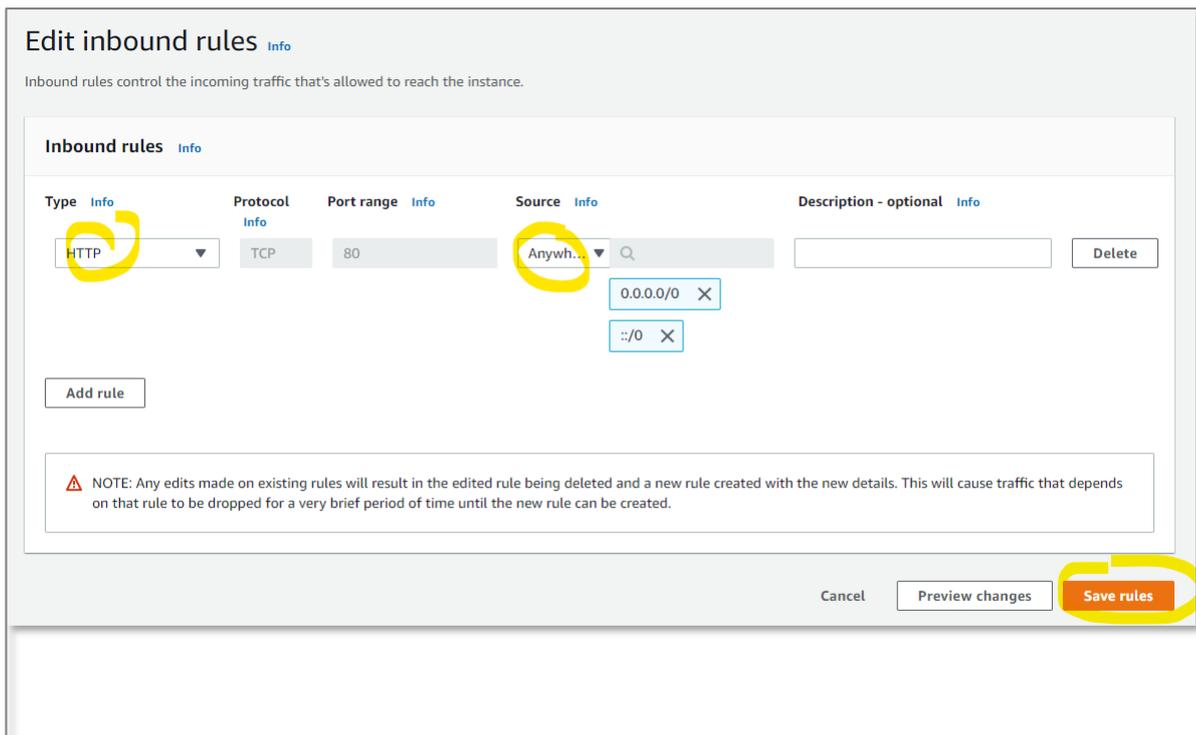
Click on the Inbound rules tab

Click on Edit



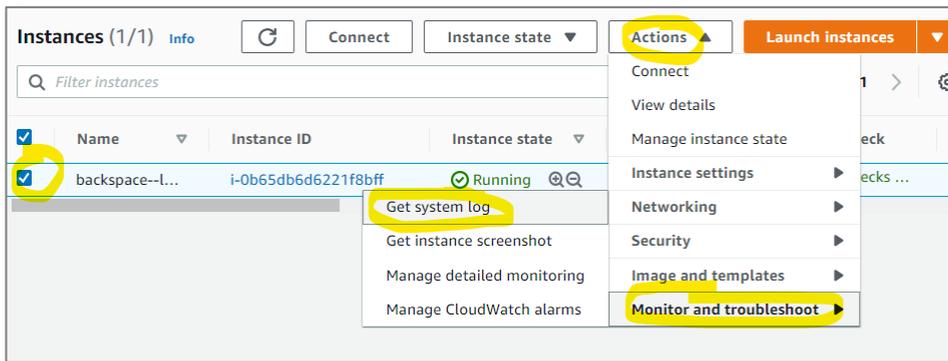
Add a rule for HTTP and Anywhere

Click Save rules

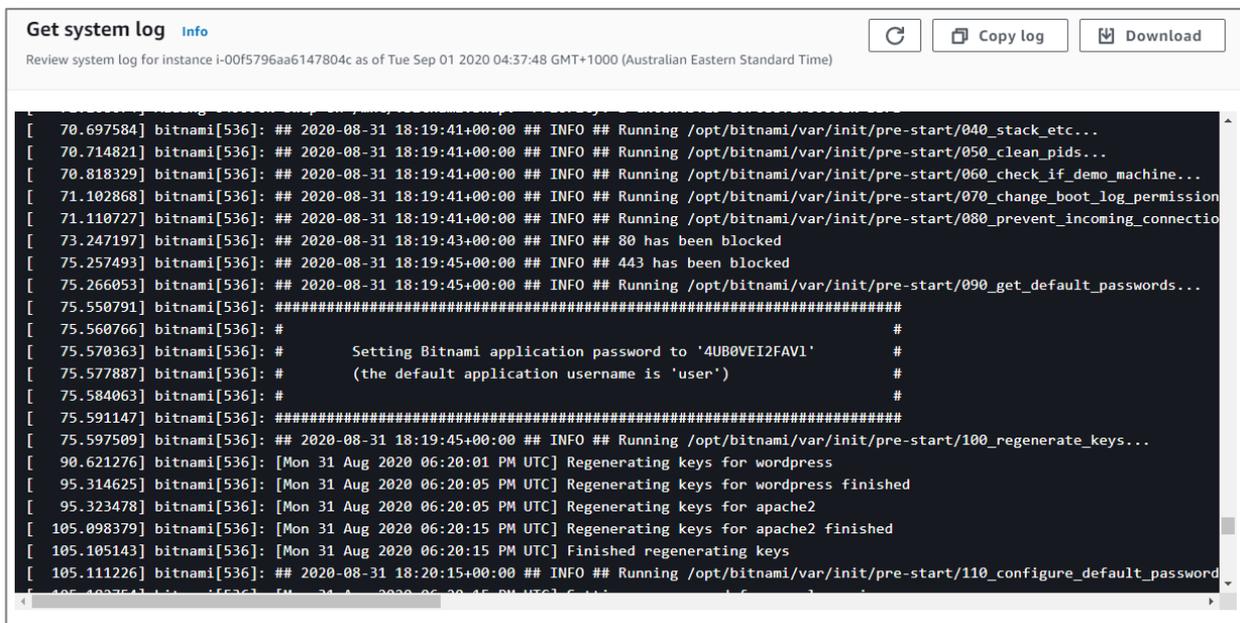


## Finding the Username and Password for your WordPress application

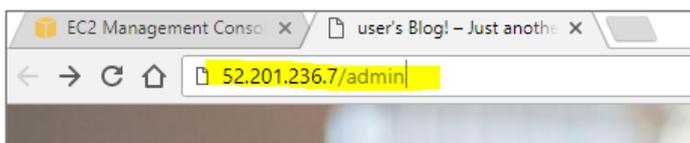
Go back to the EC2 console and select “Monitor and troubleshoot”, “Get System Log”. **Do not click on connect.**



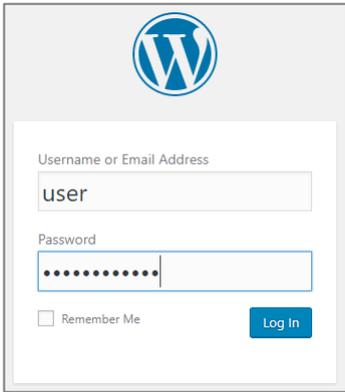
Scroll up until you find the log entry for the application username and password and copy it.



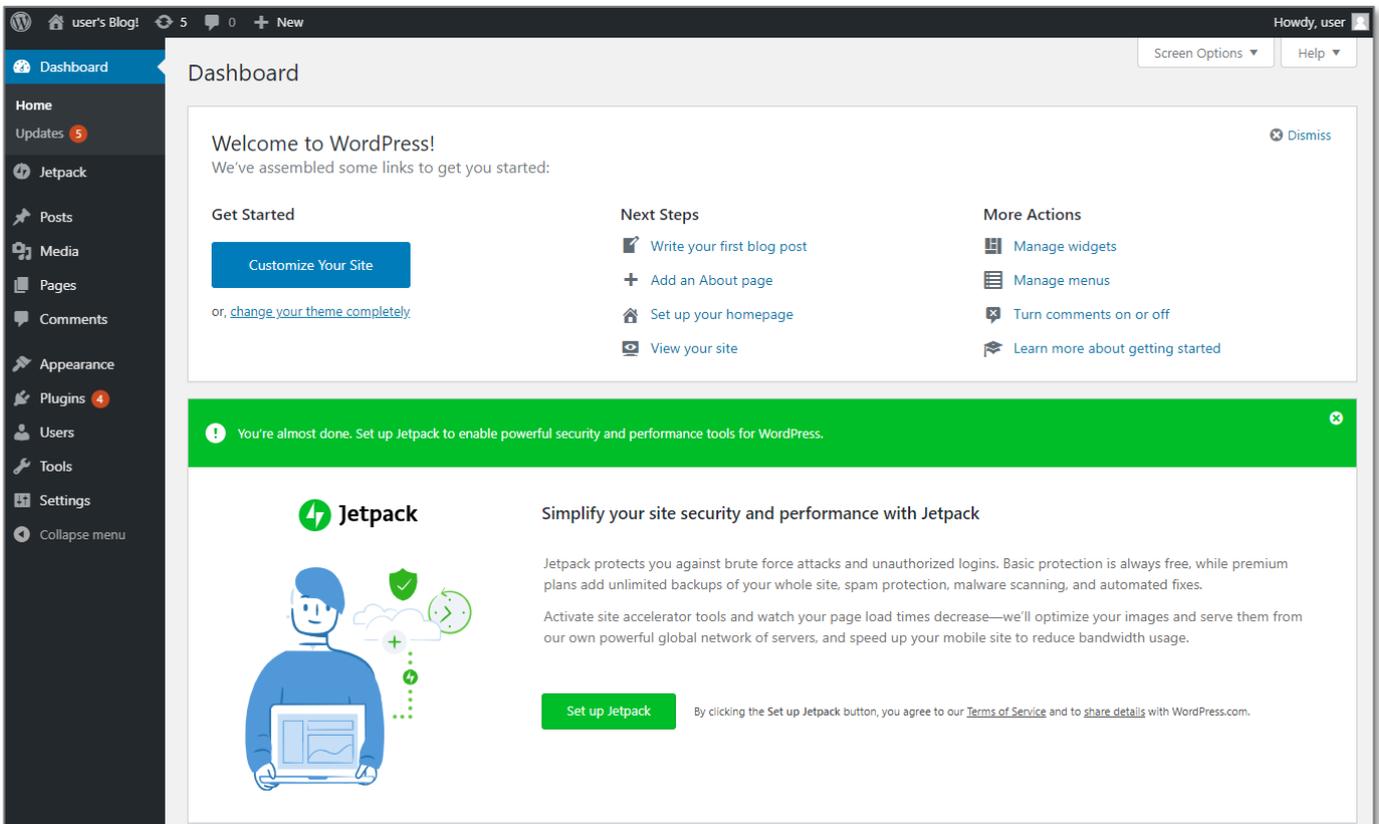
Go to the admin subdirectory of your website in your browser



Enter Username *user* and paste in the password

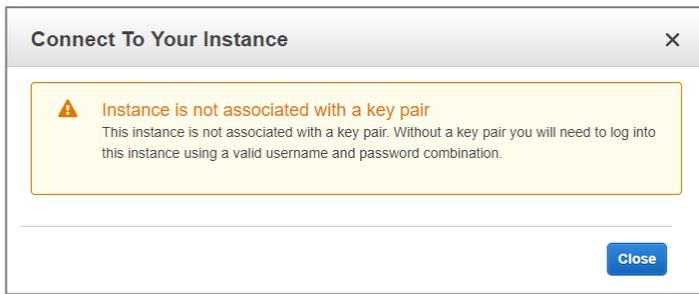


You will now be in the admin section of your WordPress application



## Troubleshooting logging in to the WordPress application

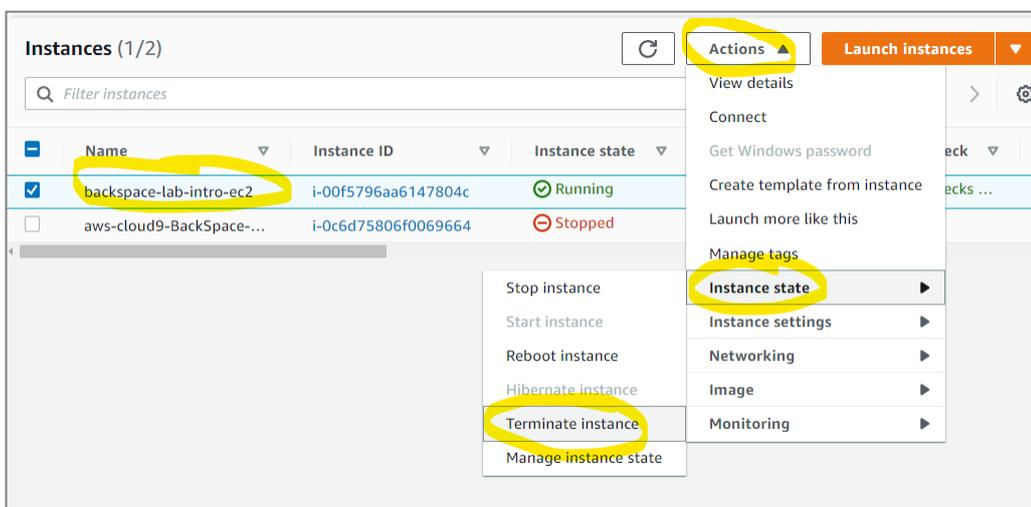
If you get the following message:



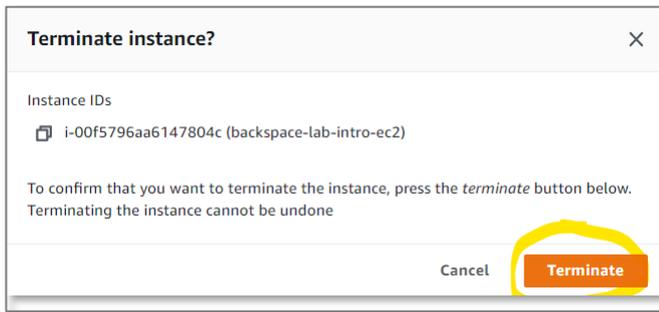
You have tried to connect to the Linux operating system by clicking on “Connect”. Do not click on connect, select “Actions – “Instance settings” - “Get System Log” as detailed previously.

## Clean up

Select *Actions* -> *Instance State* -> *Terminate*



Make sure you terminate the instance so that you are not billed for it anymore.

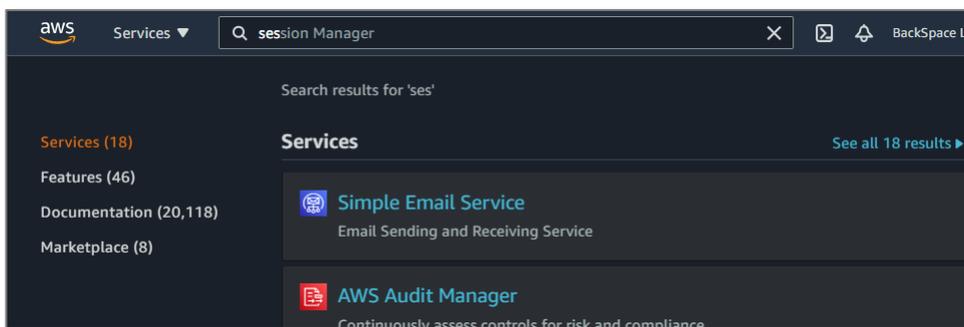


# ▶ Sending Emails with Amazon SES

In this section, we will use the Simple Email Service to send an email.

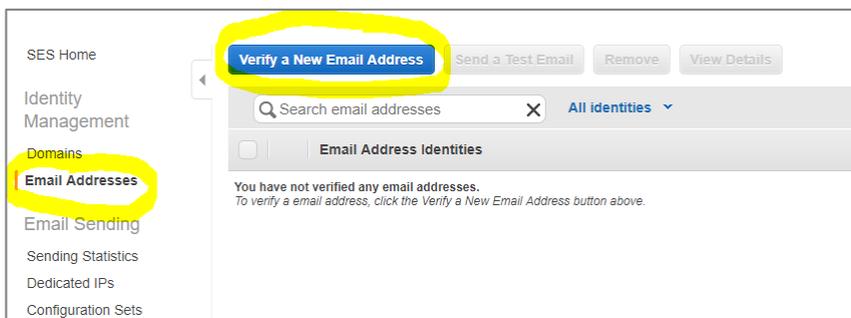
From the AWS console search *SES*

Click *Simple Email Service*

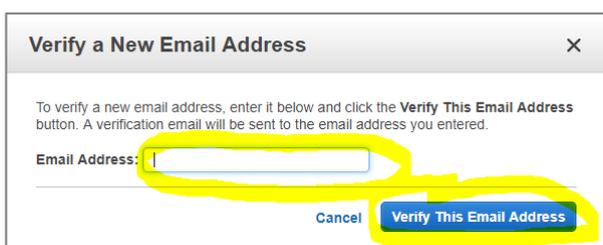


Click on 'Email addresses'

Click on 'Verify a New Email Address'



Enter your email address and click 'Verify this Email Address'



When you receive your verification email click on the supplied link.

You will then receive a success page

## Congratulations!

You have successfully verified an email address. You can now start sending email from this address.

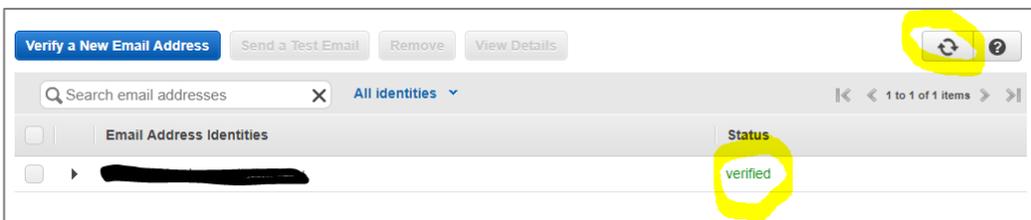
**For new Amazon SES users**—If you have not yet applied for a sending limit increase, then you are still in the [sandbox environment](#), and you can only send email to addresses that have been verified. To verify a new email address or domain, see the **Identity Management** section of the [Amazon SES console](#).

**For new Amazon Pinpoint users**—If you have not yet applied for a sending limit increase, then you are still in the [sandbox environment](#), and you can only send email to addresses that have been verified. To verify a new email address or domain, see the **Settings > Channels** page on the [Amazon Pinpoint console](#).

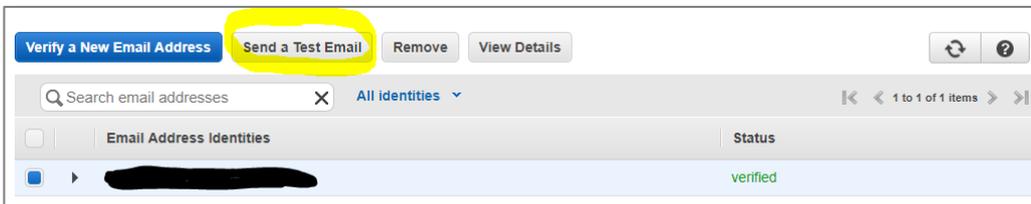
If you have already been approved for a sending limit increase, then you can start sending email to non-verified addresses.

Thank you for using Amazon Web Services!

Go back to the SES console page and refresh the information to see the email has been verified

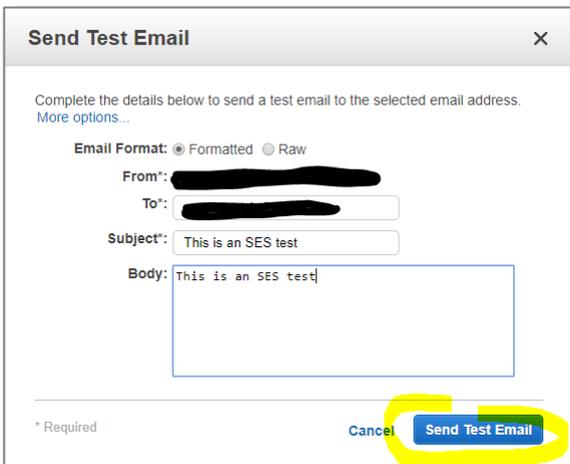


Click on the email address and select 'Send a test email'



Enter the same email address for from and to.

Fill out the email information and click 'Send test email'



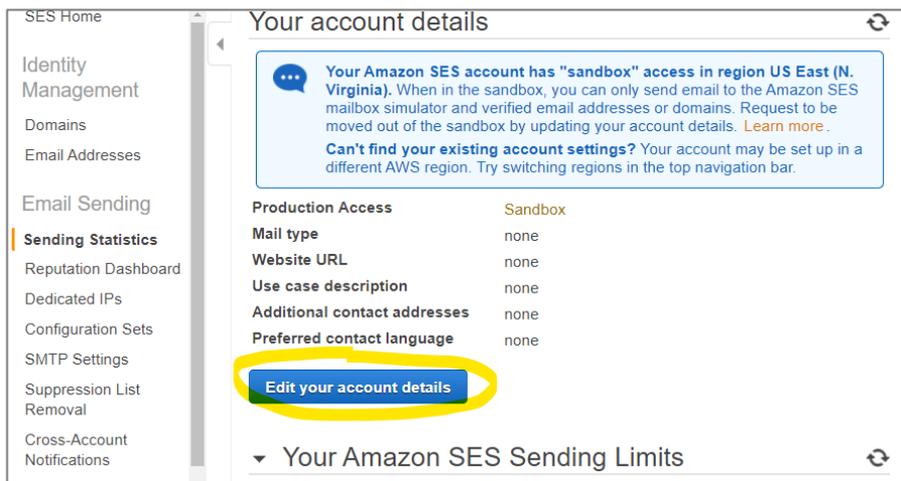
Check your email to see if it worked.

## Requesting full access to SES

New accounts only have sandbox access but this can be changed by applying to AWS.

Click on *Sending Statistics*

Click on *Edit your account details*



The screenshot shows the AWS SES console interface. On the left is a navigation menu with categories: Identity Management (Domains, Email Addresses), Email Sending (Sending Statistics, Reputation Dashboard, Dedicated IPs, Configuration Sets, SMTP Settings, Suppression List Removal, Cross-Account Notifications). The main content area is titled 'Your account details' and contains a notification box about sandbox access, a table of account settings, and a section for sending limits. The 'Edit your account details' button is highlighted with a yellow circle.

Production Access	Sandbox
Mail type	none
Website URL	none
Use case description	none
Additional contact addresses	none
Preferred contact language	none

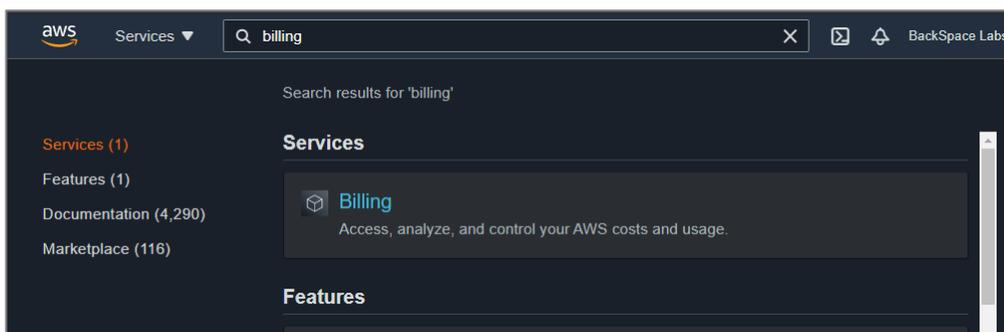
# ▶ **Creating** a Billing Alert with CloudWatch and SNS

In this section, we will create a CloudWatch billing alert that will send an email through the Simple Notification Service whenever our estimated monthly bill exceeds a certain level.

## Enabling Billing Alerts

From the AWS management console search *Billing*

Select *Billing*



Select *Billing Preferences*

Check *Receive Free Tier Usage Alerts*

Check *Receive Billing Alerts*

**IMPORTANT: Upcoming change to the Detailed Billing Report (DBR) and the Detailed Billing Report with Resources & Tags (DBR-RT)**  
On June 15, 2019, AWS will be changing the way unused reservation costs are presented in the legacy Detailed Billing Reports. If you currently use the DBR or DBR-RT portions of your reservation costs, then you should begin using the AWS Cost & Usage Reports. [Learn more](#)

## Preferences

- ▼ Billing Preferences
  - Receive PDF Invoice By Email**  
Turn on this feature to receive a PDF version of your invoice by email. Invoices are generally available within the first three days of the month.
  - Disable credit sharing**  
When credit sharing is disabled, credits will only be applied to the credit owner's account, and will not be shared across accounts in the same billing family. [Download credit preference history.](#)
  - ▶ **RI discount sharing** ⓘ
- ▼ Cost Management Preferences
  - Receive Free Tier Usage Alerts**  
Turn on this feature to receive email alerts when your AWS service usage is approaching, or has exceeded, the AWS Free Tier usage limits. If you wish to receive these email alerts, please specify the email address below.  
Email Address:
  - Receive Billing Alerts**  
Turn on this feature to monitor your AWS usage charges and recurring fees automatically, making it easier to track and manage your spending on AWS. You can set up alerts to receive email notifications when your charges reach a specified threshold. Once enabled, this preference cannot be disabled. [Manage Billing Alerts](#) or [try the new budgeting tool.](#)
- ▶ Detailed Billing Reports [Legacy]

[Save preferences](#)

Click *Save preferences*

## Creating a CloudWatch Alarm

From the AWS management console search *CloudWatch*

Select *CloudWatch*

Search results for 'CloudWatch'

Services (2)

Features (8)

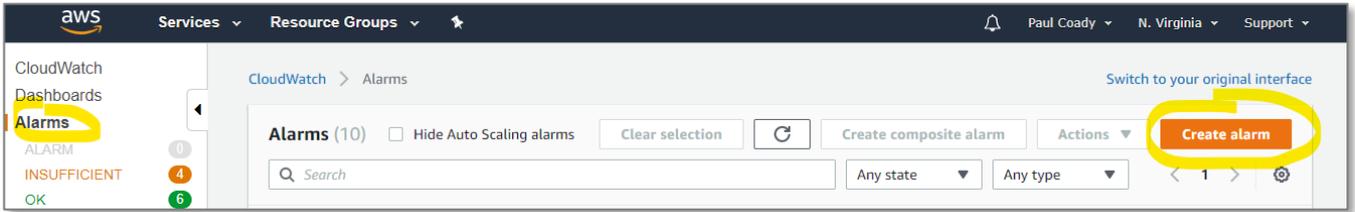
Documentation (51,182)

Marketplace (189)

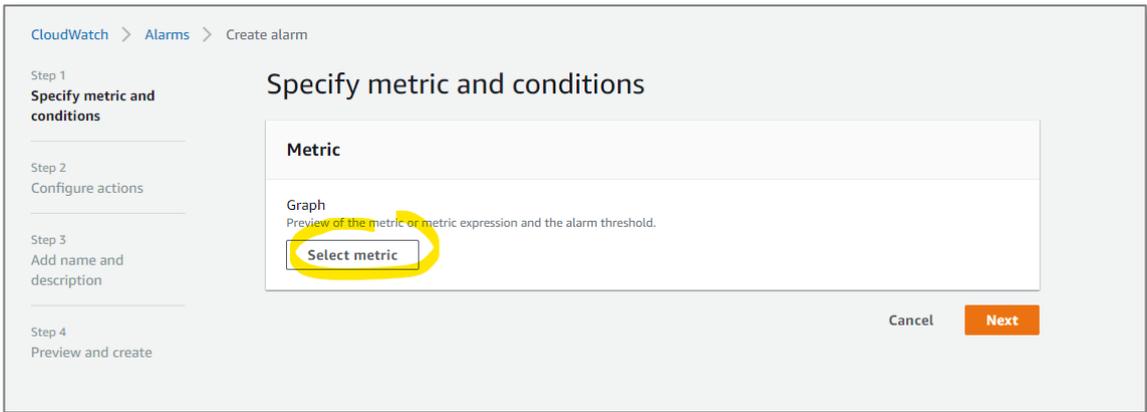
**CloudWatch**  
Monitor Resources and Applications

**Amazon EventBridge**

Click on 'Alarms', 'Create Alarm'

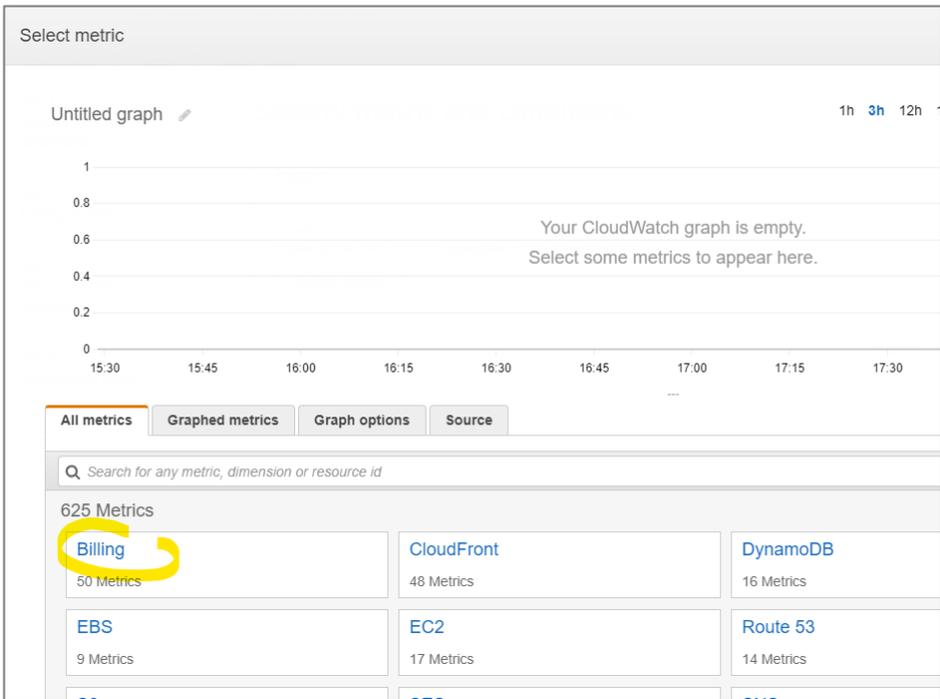


Click *Select metric*



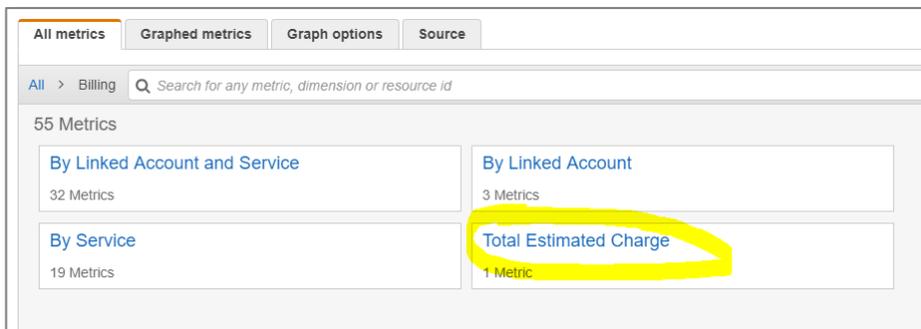
Search *Billing*

If you do not see any billing metrics it is most probably you are not operating in the US-East1 (N. Virginia) region. Please ensure you always operate from the US-East region during the course.



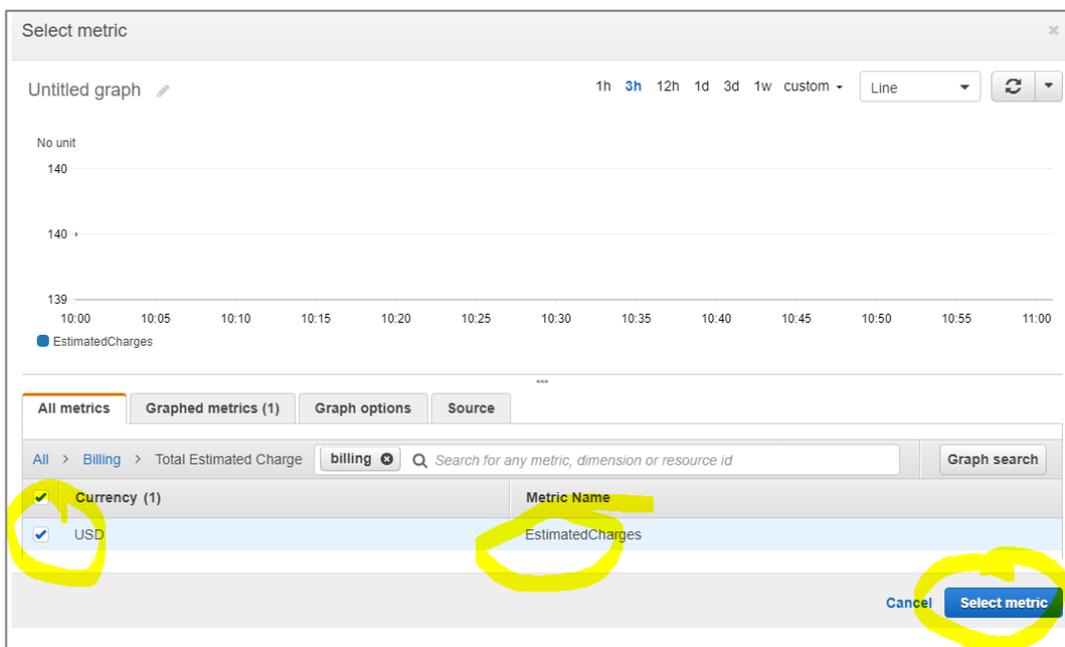
If you do not see any billing metrics it is most probably you are not operating in the US-East1 (N. Virginia) region. Please ensure you always operate from the US-East region during the course.

Select *Total Estimated Charge* from the billing metrics.



Select EstimatedCharges metric

Click *Select metric*



Set the alarm threshold to not exceed \$10

Click *Next*

CloudWatch > Alarms > Create alarm

Step 1  
**Specify metric and conditions**

Step 2  
Configure actions

Step 3  
Add name and description

Step 4  
Preview and create

### Specify metric and conditions

#### Metric

**Graph**  
This alarm will trigger when the blue line goes above the red line for 1 datapoints within 6 hours.

Namespace: AWS/Billing  
Metric name: EstimatedCharges  
Currency: USD  
Statistic: Maximum  
Period: 6 hours

#### Conditions

**Threshold type**

Static  
Use a value as a threshold

Anomaly detection  
Use a band as a threshold

Whenever EstimatedCharges is...

Define the alarm condition.

Greater > threshold

Greater/Equal >= threshold

Lower/Equal <= threshold

Lower < threshold

than... Define the threshold value.

10 USD

Must be a number

Additional configuration

Cancel **Next**

Select *in Alarm* for notification

Select *Create new topic*

Enter a unique topic name

Enter an email address to receive the alert

Click *Create Topic*

CloudWatch > Alarms > Create alarm

Step 1  
Specify metric and conditions

Step 2  
**Configure actions**

Step 3  
Add name and description

Step 4  
Preview and create

## Configure actions

### Notification

Alarm state trigger  
Define the alarm state that will trigger this action. Remove

**In alarm**  
The metric or expression is outside of the defined threshold.

**OK**  
The metric or expression is within the defined threshold.

**Insufficient data**  
The alarm has just started or not enough data is available.

Select an SNS topic  
Define the SNS (Simple Notification Service) topic that will receive the notification.

Select an existing SNS topic

**Create new topic**

Use topic ARN

Create a new topic...  
The topic name must be unique.

SNS topic names can contain only alphanumeric characters, hyphens (-) and underscores (\_).

Email endpoints that will receive the notification...  
Add a comma-separated list of email addresses. Each address will be added as a subscription to the topic above.

user1@example.com, user2@example.com

Create topic

Add notification

Click Next

Send a notification to...

Only email lists for this account are available

Email (endpoints)  
**aws@backspace.academy** - [View in SNS Console](#)

Add notification

### Auto Scaling action

Add Auto Scaling action

### EC2 action

This action is only available for EC2 Per-Instance Metrics

Add EC2 action

Cancel Previous **Next**

Give the alarm a unique name

Click Next

CloudWatch > Alarms > Create alarm

Step 1  
Specify metric and conditions

Step 2  
Configure actions

Step 3  
**Add a description**

Step 4  
Preview and create

### Add a description

**Name and description**

Define a unique name

Alarm name

Alarm description - optional  
Define a description for this alarm. Optionally you can also use markdown.

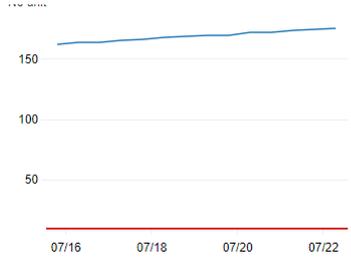
Alarm description

Up to 1024 characters (0/1024)

Cancel Previous **Next**

Click *Create alarm*

Step 4  
**Preview and create**



EstimatedCharges

Namespace  
AWS/Billing

Metric name  
EstimatedCharges

Currency  
USD

Statistic  
Maximum

Period  
6 hours

**Conditions**

Threshold type  
Static

Whenever **EstimatedCharges** is  
Greater (>)

than...  
10

► Additional configuration

Step 2: Configure actions Edit

**Actions**

Notification  
When in Alarm, send a notification  
to "pcoady-Account-Billing-Alert"

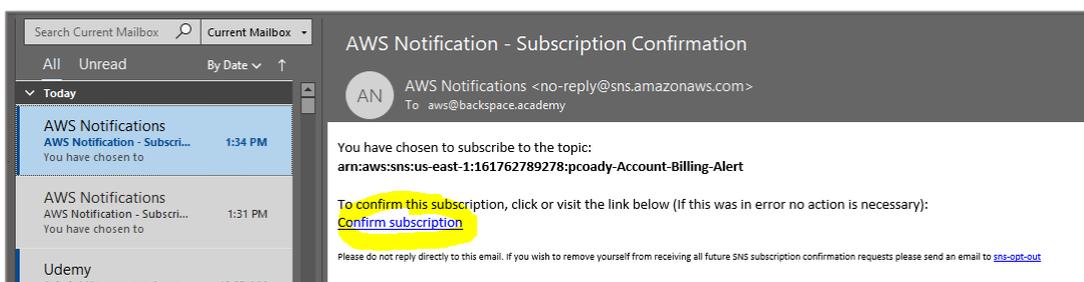
Step 3: Add a description Edit

**Name and description**

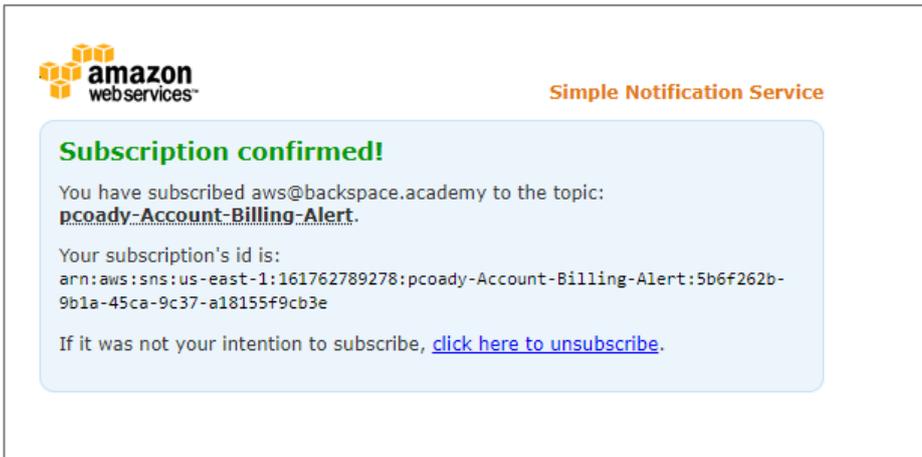
Name	Description
pcoady-Account-billing-alert	

Cancel Previous Create alarm

If you haven't already confirmed your email a confirmation email will be sent to you.



Click on the link in the email to confirm the SNS subscription



Go back to the CloudWatch console and refresh the screen.

The Alarm State will be INSUFFICIENT\_DATA until enough data has been collected by CloudWatch

Name	State	Conditions	Actions
test-ReadCapacityUnitsLimit-BasicAlarm	OK	ConsumedReadCapacityUnits >= 240 for 5 datapoints within 5 minutes	No r
test-name-index-WriteCapacityUnitsLimit-BasicAlarm	OK	ConsumedWriteCapacityUnits >= 240 for 60 datapoints within 1 hour	No r
test-name-index-ReadCapacityUnitsLimit-BasicAlarm	OK	ConsumedReadCapacityUnits >= 240 for 60 datapoints within 1 hour	No r
test-WriteCapacityUnitsLimit-BasicAlarm	OK	ConsumedWriteCapacityUnits >= 240 for 5 datapoints within 5 minutes	No r
pcoady-Account-billing-alert	Insufficient data	EstimatedCharges > 10 for 1 datapoints within 6 hours	Valid

After a few days you will have plenty of billing data to view

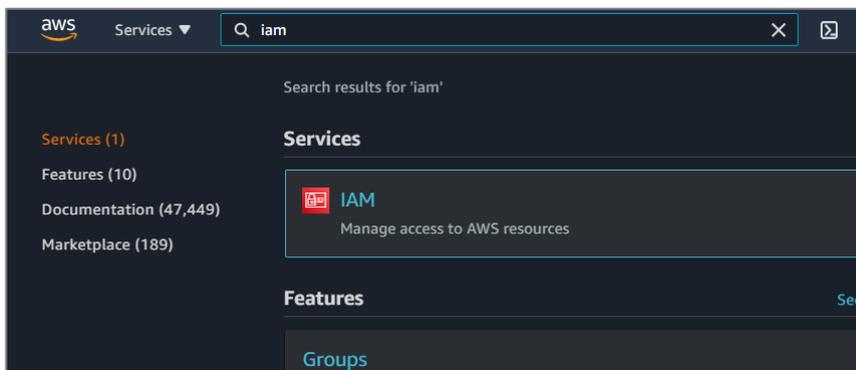
Name	State	Last state update	Conditions	Actions
pcoady-billing-alert	OK	2020-06-01 17:59:23	EstimatedCharges > 10 for 1 datapoints within 6 hours	-

# ▶ Creating an IAM User

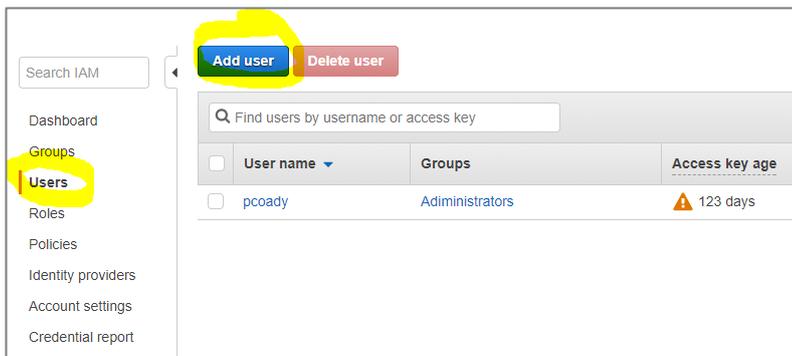
In this section, we will use the Identity and Access Management (IAM) service to create a user with console access and programmatic access.

From the AWS console click search IAM

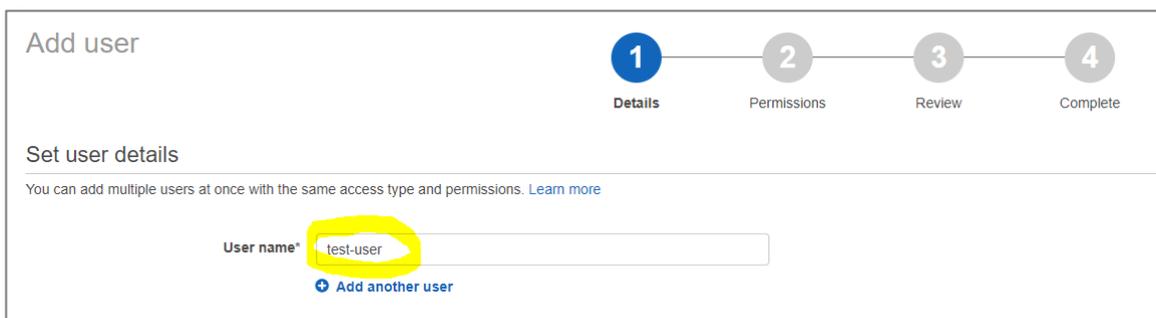
Select IAM



Select Users -> Add user



Give the user a name



Check Programmatic access

### Check AWS Management Console access

Click *Next: Permissions*

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

**Access type**

- Programmatic access**  
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.
- AWS Management Console access**  
Enables a **password** that allows users to sign-in to the AWS Management Console.

**Console password\***

- Autogenerated password
- Custom password

**Require password reset**  User must create a new password at next sign-in  
Users automatically get the `IAMUserChangePassword` policy to allow them to change their own password.

\* Required

Cancel **Next: Permissions**

We won't set any permissions for the user at this point.

Click *Next: Tags*

Add user

1 2 3 4

Set permissions for test-user

- Add user to group**
- Copy permissions from existing user
- Attach existing policies directly

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Add user to group

Create group Refresh

Cancel Previous **Next: Review**

We won't set any tags.

Click *Next Review*

Cancel Previous **Next: Review**

Click *Create user*

## Add user

1 2 3 4 5

### Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

**⚠ This user has no permissions**  
 You haven't given this user any permissions. This means that the user has no access to any AWS service or resource. Consider returning to the previous step and adding some type of permissions.

#### User details

<b>User name</b>	test-user
<b>AWS access type</b>	Programmatic access and AWS Management Console access
<b>Console password type</b>	Autogenerated
<b>Require password reset</b>	Yes

Cancel Previous Create user

Download the csv file containing the user credentials (access key and secret access key) to a safe location. You will need this for access using the Command Line Interface (CLI) later in the course.

## Add user

1 2 3 4 5

**✓ Success**  
 You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://backspace-labs.signin.aws.amazon.com/console>

Download .csv

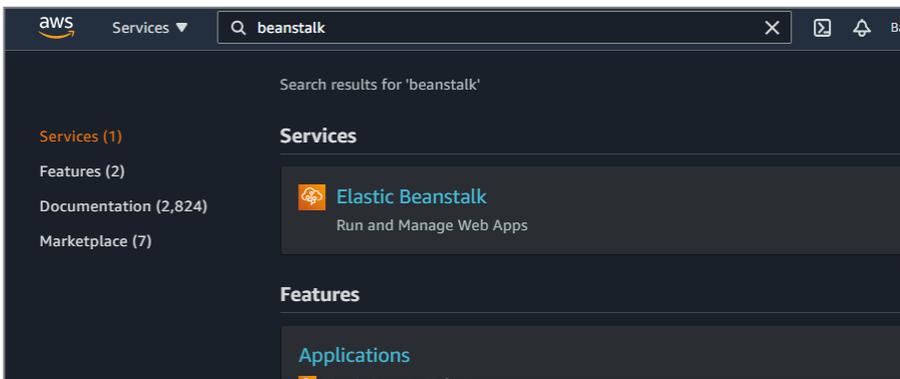
	User	Access key ID	Secret access key	Password	Email login instructions
▶	✓ test-user	AKIAVIRAUKQREM2SV2OF	***** Show	***** Show	Send email

Close

# ▶ **Creating** a Highly Available Architecture with Elastic Beanstalk

In this section, we will create a highly available and fault tolerant architecture using the AWS Elastic Beanstalk service.

From the Management Console search *Elastic Beanstalk*



Click *Create Application*



Give your application a name *Test Application*.

## Create a web app

Create a new application and environment with a sample application or your own code. By creating an environment, you allow AWS Elastic Beanstalk to manage AWS resources and permissions on your behalf. [Learn more](#)

### Application information

Application name

Test Application

Up to 100 Unicode characters, not including forward slash (/).

Scroll down to *Platform*

Select *Node.js* as the platform

**Platform**

Platform

Node.js

Platform branch

Node.js 12 running on 64bit Amazon Linux 2

Platform version

5.2.4 (Recommended)

Scroll down to *Application code*

Select *Sample application*

Click *Configure more options*

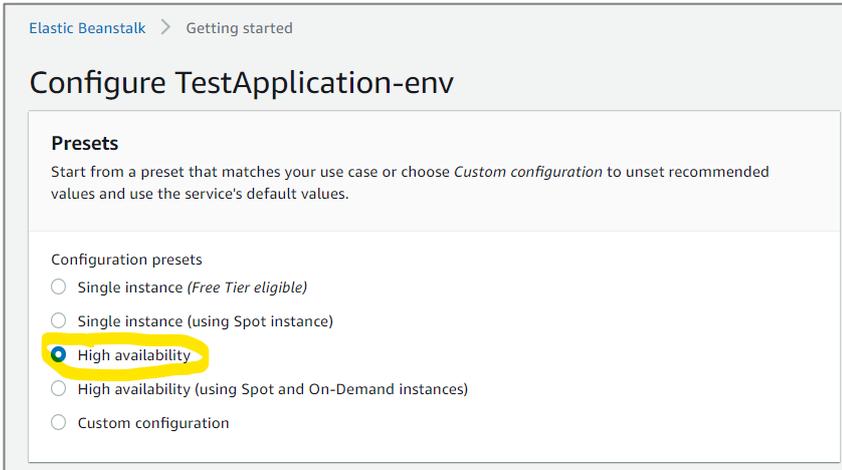
**Application code**

Sample application  
Get started right away with sample code.

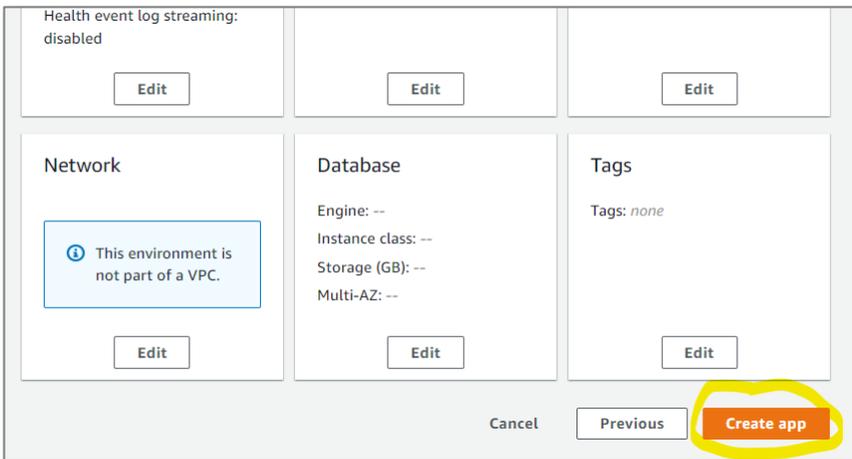
Upload your code  
Upload a source bundle from your computer or copy one from Amazon S3.

Cancel **Configure more options** Create application

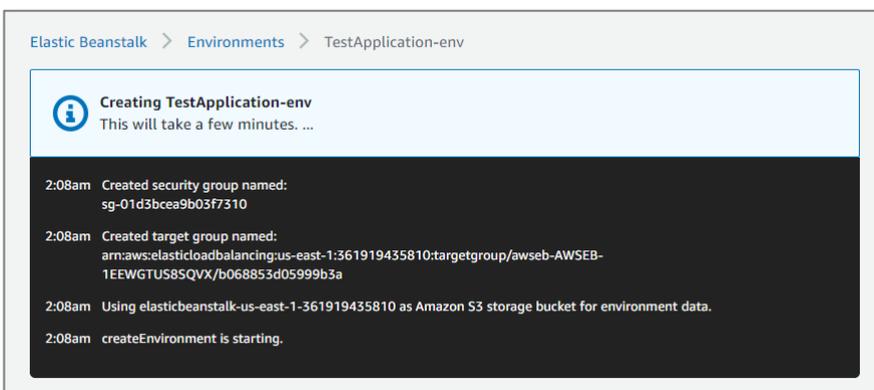
Select High availability



Scroll down and click *Create app*



Your application will now start being created.



After some time, your environment will be created.

Click on the website url

**TestApplication-env**

TestApplication-env.eba-m85h72sw.us-east-1.elasticbeanstalk.com (e-zm9zupp8kc)

Application name: **Test Application**

**Health**

Ok

Causes

**Running version**

Sample Application

Upload and deploy

**Platform**

nodejs

Node.js 12 running on 64bit  
Amazon Linux 2/5.2.4

Change

You will now see the Sample Application

**Congratulations**

Your first AWS Elastic Beanstalk Node.js application is now running on your own dedicated environment in the AWS Cloud

**What's Next?**

- [AWS Elastic Beanstalk overview](#)
- [AWS Elastic Beanstalk concepts](#)
- [Deploy an Express Application to AWS Elastic Beanstalk](#)
- [Deploy an Express Application with Amazon ElastiCache to AWS Elastic Beanstalk](#)
- [Deploy a Geddy Application with Amazon ElastiCache to AWS Elastic Beanstalk](#)
- [Customizing and Configuring a Node.js Container](#)
- [Working with Logs](#)

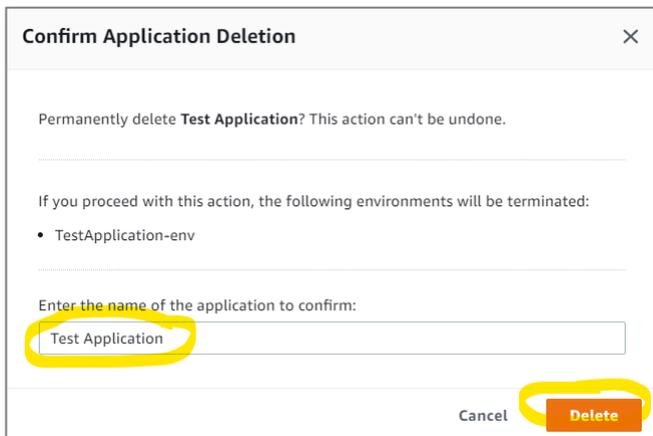
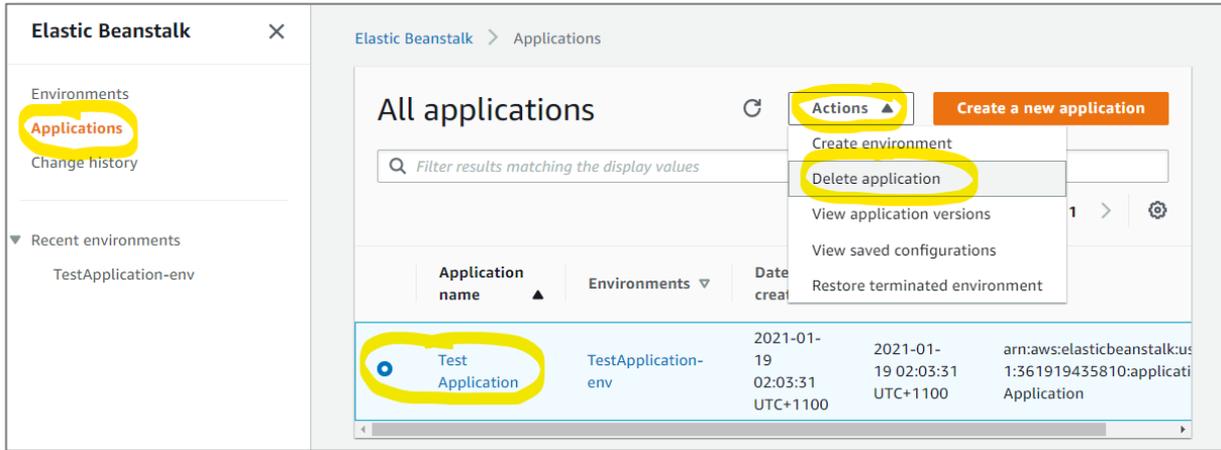
## Clean Up

We will now delete the environment so that you will not be billed by AWS.

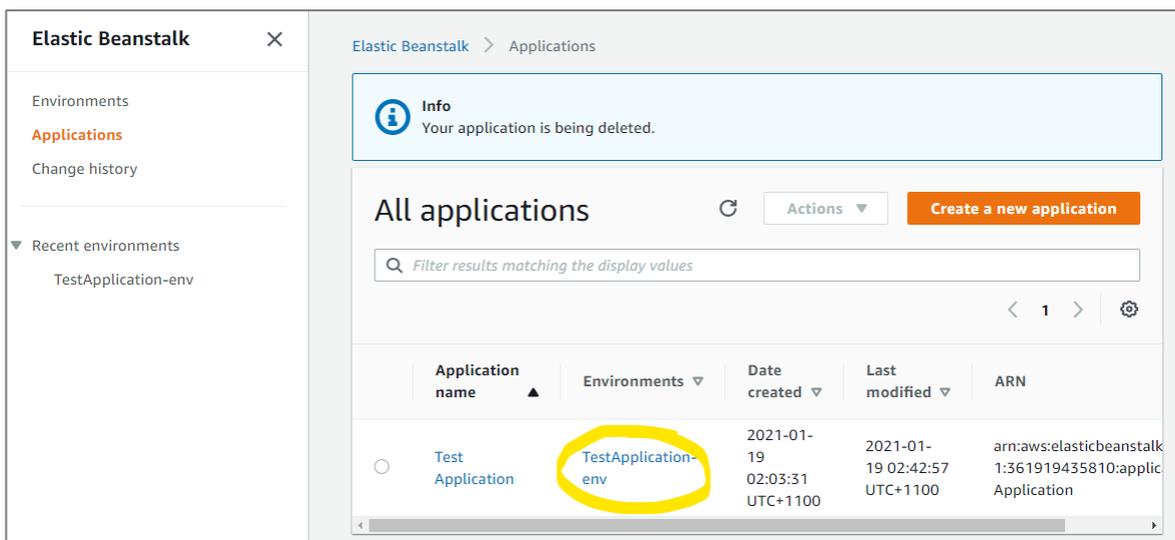
Click on *Applications*

Select the application

Select *Actions* -> *Delete application*



Click on the environment



You will now see your environment is being terminated.

Elastic Beanstalk is terminating your environment.  
[View Events](#)

### TestApplication-env

[TestApplication-env.eba-m85h72sw.us-east-1.elasticbeanstalk.com](#) (e-zm9zupp8kc)  
Application name: [Test Application](#)

[Refresh](#) [Actions](#)

Health	Running version	Platform
 Info <a href="#">Causes</a>	Sample Application <a href="#">Upload and deploy</a>	 Node.js 12 running on 64bit Amazon Linux 2/5.2.4 <a href="#">Change</a>