



lab



lab title

Introduction to AWS

V2.07



Course title

BackSpace Academy
AWS Certified Cloud Practitioner



▶ Table of Contents

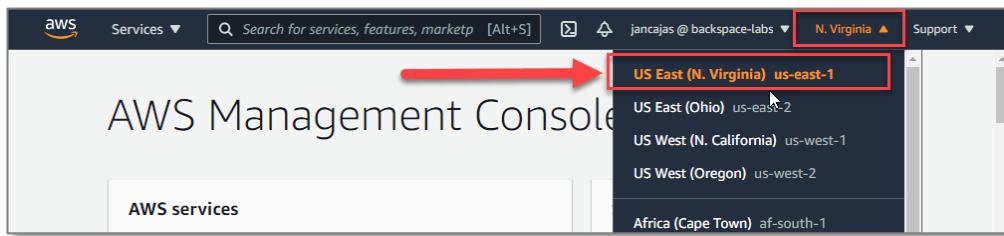
Contents

Table of Contents	1
About the Lab	3
Controlling Costs with AWS	4
Checking your AWS Usage and Monthly Bill	4
Using AWS Cost Explorer	5
View a Costs Report.....	6
Create an AWS Budget	7
Troubleshooting.....	13
Clean Up.....	14
Creating an S3 Bucket and Uploading Files.....	15
Create a Bucket.....	15
Uploading Files to your Bucket.....	16
Downloading files from your bucket	18
Troubleshooting.....	19
Clean Up.....	20
Creating a SQL Database with RDS.....	23
Creating a Security Group.....	23
Creating an RDS Database	26
Connecting to your RDS Instance	31
Troubleshooting Connection Issues	36
Clean Up.....	39
Creating a Web Server with EC2	41
Launch an EC2 Instance	41
Viewing your web server	47
Troubleshooting viewing your WordPress application	48
Finding the Username and Password for your WordPress application.....	51
Troubleshooting logging in to the WordPress application	53
Clean up	54
Monitoring User Activity with AWS CloudTrail.....	55
Clean Up.....	60

Sending Emails with Amazon SES	62
Requesting full access to SES	67
Creating an IAM User.....	68
Clean Up.....	70
Creating a Highly Available Architecture with Elastic Beanstalk	71
Clean Up.....	74

► 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.

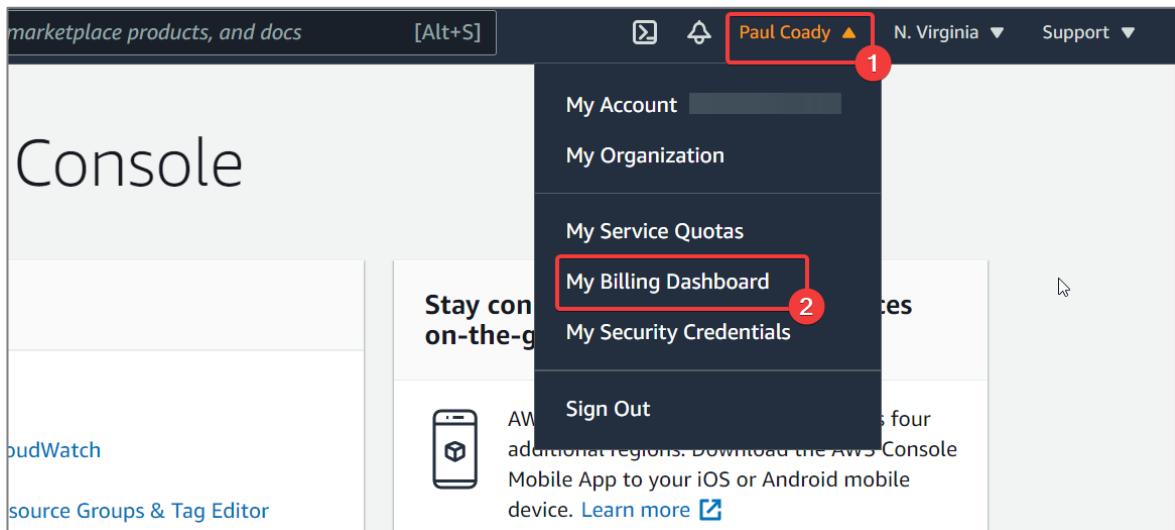
Please note that AWS services change on a weekly basis and it is extremely important you check the version number on this document to ensure you have the latest version with any updates or corrections.

▶ Controlling Costs with AWS

In this section we will learn how to use the AWS Billing & Cost Management Dashboard to keep track of costs. We will then create a Budget using AWS Budgets that will alert us when costs are exceeded.

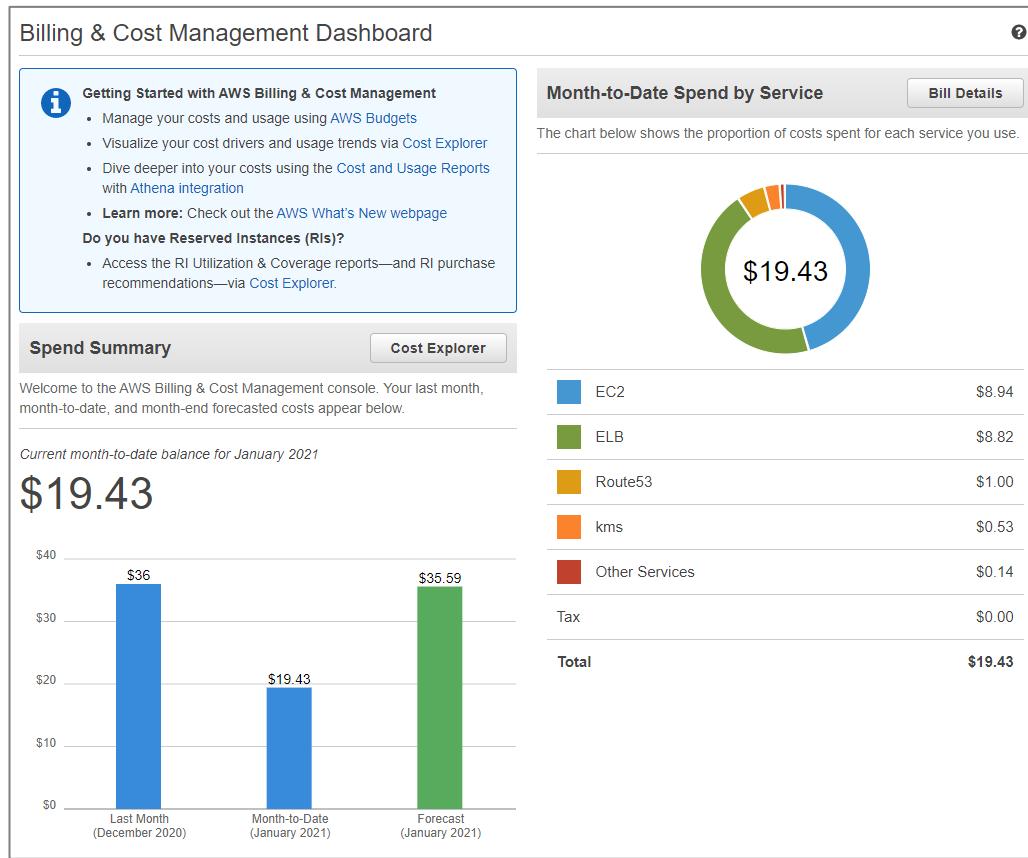
Checking your AWS Usage and Monthly Bill

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.

Yours will of course differ from the example below.



Using AWS Cost Explorer

Click [Cost Explorer](#)

aws Services ▾ Search for services, features, marketplace products, and docs

Home

Billing

Bills

Payments

Credits

Purchase orders

Cost & Usage Reports

Cost Categories

Cost allocation tags

Cost Management

Cost Explorer

Budgets

Billing & Cost Management Dashboard

i 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)?

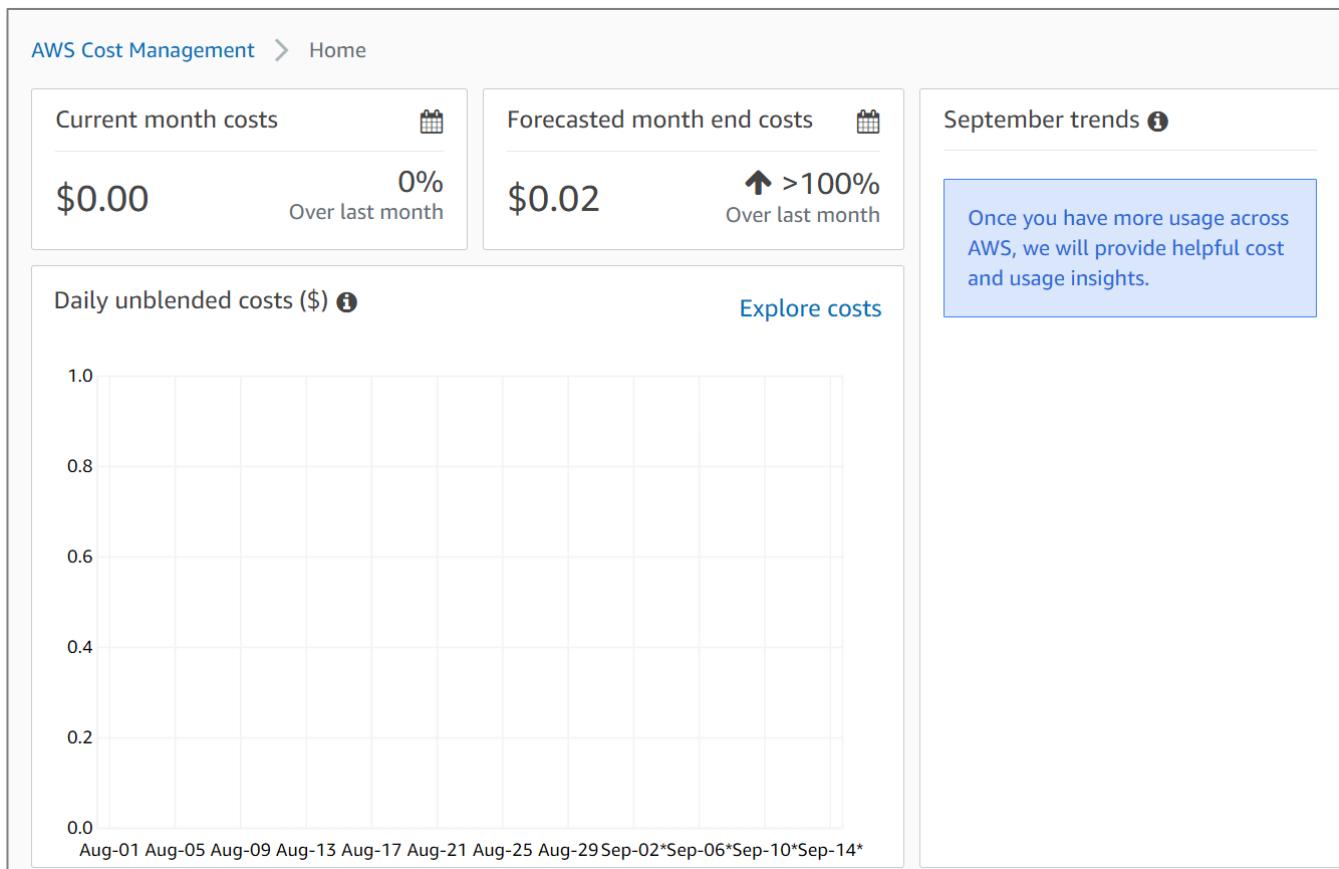
- Access the RI Utilization & Coverage reports—and RI purchase recommendations—via [Cost Explorer](#).

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.

You will now see a breakdown of costs and forecast costs.

Yours will of course differ from the example below.



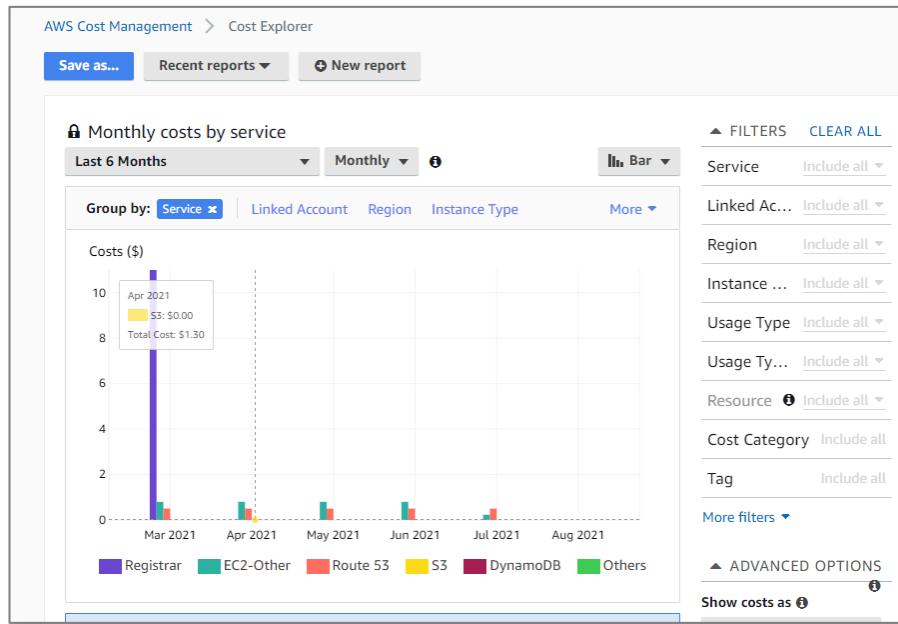
View a Costs Report

Select Reports

Select Monthly costs by service

The screenshot shows the "Reports" section of the AWS Cost Management interface. On the left sidebar, under the "Reports" menu item (marked with a red circle 1), there are options for Budgets, Cost Anomaly Detection, Rightsizing recommendations, Savings Plans, Overview, and Inventory. The main area shows a table of "All reports (9)". One report, "Monthly costs by service", is highlighted with a red box and a red circle 2. The table columns include Report name, Type, and Time range. Other reports listed include "Monthly costs by linked account" and "Monthly EC2 running hours costs and usage".

Yours will of course differ from the example below.

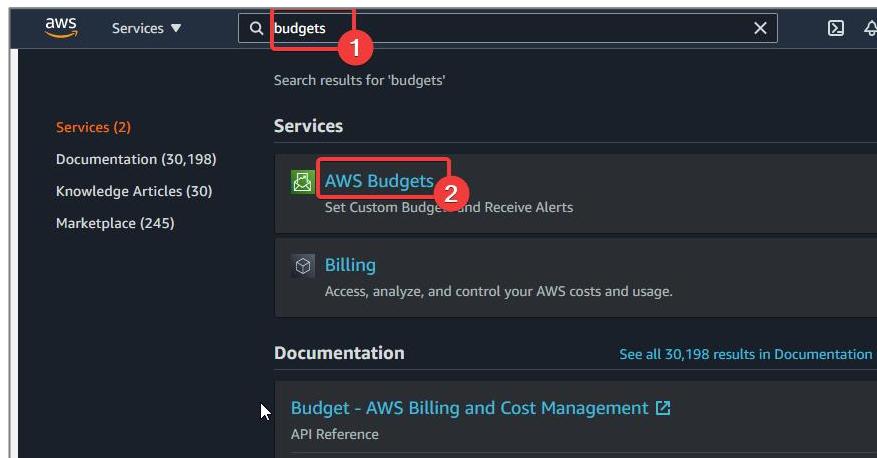


Create an AWS Budget

Make sure you are in US-East (N. Virginia) region.

From the AWS console search *Budgets*

Select *AWS Budgets*



Click on *Create a Budget*

AWS Billing

AWS Budgets

Set custom budgets that alert you when you exceed your budgeted thresholds

AWS Budgets is your hub for creating, tracking, and inspecting your budgets.

Create a budget

Pricing (US)

Select Cost budget then click Next

Billing Console > Budgets > Create budget

Step 1 Choose budget type

Step 2 Set your budget

Step 3 Configure alerts

Step 4 - Optional Attach actions

Step 5 Review

Choose budget type

Budget types

- Cost budget - Recommended**
Monitor your costs against a specified dollar amount and receive alerts when your user-defined thresholds are met. Using cost budgets, the budgeted amount you set represents your expected cloud spend. For example, you can set a cost budget for a business unit and then add additional parameters such as the associated member accounts.
- Usage budget**
Monitor your usage of one or more specified usage types or usage type groups and receive alerts when your user-defined thresholds are met. Using usage budgets, the budgeted amount represents your expected usage. For example, you can use a usage budget to monitor the usage of certain services such as Amazon EC2 and Amazon S3.
- Savings Plans budget**
Track the utilization or coverage associated with your Savings Plans and receive alerts when your percentage drops below a threshold you define. Setting a coverage target lets you see how much of your instance usage is covered by Savings Plans, while setting a utilization target lets you see if your Savings Plans are unused or underutilized.
- Reservation budget**
Track the utilization or coverage associated with your reservations and receive alerts when your percentage drops below a threshold you define. Setting a coverage target lets you see how much of your instance usage is covered by reservations, while setting a utilization target lets you see if your reservations are unused or underutilized. Reservation alerts are supported for Amazon EC2, Amazon RDS, Amazon Redshift, Amazon ElastiCache, and Amazon Elasticsearch reservations.

Cancel **Next**

For Period select *Monthly* then *Recurring budget* for the Budget effective date

Billing Console > Budgets > Create budget

Step 1 Choose budget type

Step 2 Set your budget

Step 3 Configure alerts

Step 4 - Optional Attach actions

Step 5 Review

Set your budget Info

How to set up your budget

Step 1: Set budget amount
Select the period and whether you would like to have a fixed budget or to specify a budget plan, then enter your budget amount.

Step 2: Scope your budget - optional
Add dimensions of data to narrow on a set of cost information. For example, you could select a number of AWS services to track as part of this budget.

Step 3: Enter in remaining budget details
Define the budget name.

Set budget amount

Period
Daily budgets do not support enabling forecasted alerts, daily budget planning, or attaching actions.

Monthly 1

Budget effective date

Recurring budget
Recurring budgets renew on the first day of every monthly billing period. 2

Expiring budget
Expiring monthly budgets stop renewing at the end of the selected expiration month.

Start month

Sep 2021

Select *Fixed* and enter your desired budget \$10.00

Choose how to budget

Fixed
Create a budget that tracks against a single monthly budgeted amount. 1

Monthly budget planning
Specify your budgeted amount for each budget period.

Enter your budgeted amount (\$)
Last month's cost: \$1.73

10.00 2

Name the budget then click *Next*

Details

Budget name
Provide a descriptive name for this budget.

Never exceed 10 dollars 1

Names must be between 1-100 characters.

Cancel Previous Next 2

Click *Add an alert threshold*

Budget amount

Your budgeted amount: **\$10.00**
To change your budgeted amount, go back to step 2.

No alert thresholds created. **Add an alert threshold**

Cancel Previous Next

Set a number for the threshold (80) then enter email information.

Leave *Amazon SNS Alerts* and *Amazon Chatbot Alerts* we are not using them.

▼ Alert #1 Remove

Set alert threshold

Threshold When should this alert be triggered? Trigger How should this alert be triggered?

80 % of budgeted amount Actual

Summary: When your actual cost is greater than 80.00% (\$8.00) of your **budgeted amount** (\$10.00), the alert threshold will be exceeded.

Notification preferences - *Optional*
Select one or more notification preferences to receive alerts.

Email recipients
Specify the email recipients you want to notify when the threshold has exceeded.
sample.email@email.com

Maximum number of email recipients is 10.

► Amazon SNS Alerts [Info](#)
► Amazon Chatbot Alerts

Click *Add alert threshold* to create another alert

▼ Alert #1 Remove

Set alert threshold

Threshold
 When should this alert be triggered?
 % of budgeted amount ▼

Trigger
 How should this alert be triggered?
 ▼

Notification preferences - Optional
Select one or more notification preferences to receive alerts.

Email recipients
Specify the email recipients you want to notify when the threshold has exceeded.

Maximum number of email recipients is 10.

► Amazon SNS Alerts [Info](#)

► Amazon Chatbot Alerts

+ Add alert threshold

Create another alert threshold for Forecasted Trigger.
 Set a number for the treshold (80) then select Forecasted.
 Then enter email information.

▼ Alert #2 Remove

Set alert threshold

Threshold
 When should this alert be triggered?
 % of budgeted amount ▼

Trigger
 How should this alert be triggered?
 ▼

Summary: When your forecasted cost is greater than 80.00% (\$8.00) of your **budgeted amount** (\$10.00), the alert threshold will be exceeded.

Notification preferences - Optional
Select one or more notification preferences to receive alerts.

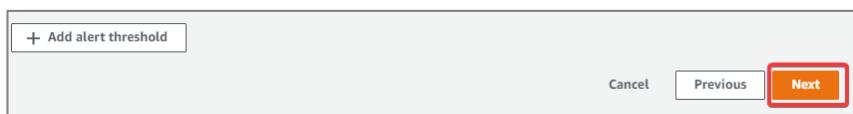
Email recipients
Specify the email recipients you want to notify when the threshold has exceeded.

Maximum number of email recipients is 10.

► Amazon SNS Alerts [Info](#)

► Amazon Chatbot Alerts

Click *Next*



Click **Next**

A screenshot of the AWS CloudWatch Metrics Insights alert configuration wizard. The second step, 'Add actions', is shown. It displays two alerts: 'Alert #1' and 'Alert #2', both of which have 0 actions attached. Each alert section shows its threshold (80%), the metric it's measured against (Actual Costs or Forecasted Costs), and its target (Email recipients or Amazon SNS). Both sections contain an 'Add Action' button. The 'Next' button at the bottom right is highlighted with a red box.

Then Click Create budget

Step 2: Set up your budget

Budget details

Name Never exceed 10 dollars	Start date Sep 2021	Budget amount \$10.00
Period Monthly	End date -	

► Additional budget parameters

Step 3: Configure alerts

Alerts

Alert #1	Alert #2
Threshold 80% of budgeted amount	Threshold 80% of budgeted amount
Threshold measured against Actual costs	Threshold measured against Forecasted costs

Step 4: Attach actions - optional

Actions

You have no budgets actions

Cancel Previous **Create budget**

You will see now the created Budget

Your budget Never exceed 10 dollars has been created successfully.

Billing Console > Budgets > Overview

Overview Info

Budgets (1) Info

<input type="checkbox"/>	Name	Thresholds	Budget	Amount ...	Forecast...	Current vs. budgeted
<input type="checkbox"/>	Never exceed 10 dollars	OK	\$10.00	-	-	0.00

Troubleshooting

Make sure you leave *Amazon SNS Alerts* and *Amazon Chatbot Alerts*, we are not using them.

If you have followed the lab notes carefully and still cannot create your budget then there may be a problem with your account such credit card information. AWS Support is free for billing and account issues.

Go to <https://console.aws.amazon.com/support>

Click *Create case*

AWS Support Center

How can we help?

Search AWS Support resources

Open support cases

Subject	Case ID	Created	Status
WorkDocs crashes windows file explorer	9175399951	1 day ago	Customer action completed

[View all cases](#) [Create case](#)

Select Account and billing support

AWS Support > Your support cases > Create case

Create case Info

Account and billing support Assistance with account and billing-related inquiries

Service limit increase Requests to increase the service limit of your AWS resources

Technical support Service-related technical issues and third-party applications

Clean Up

Select the *Budgets* you created
Then Click *Actions* then *Delete*

Billing Console > Budgets > Overview

Overview Info

Budgets (1/1) Info

Name	Thresholds	Budget	Amount ...	Copy	Current vs. budgeted
<input checked="" type="checkbox"/> Never exceed 10 dollars	<input checked="" type="checkbox"/> OK	\$10.00	-	-	0.00

[Download CSV](#) [Actions](#) [Create budget](#)

[Edit](#) [Delete](#)

1 **Never exceed 10 dollars**

2 [Actions](#)

3 [Delete](#)

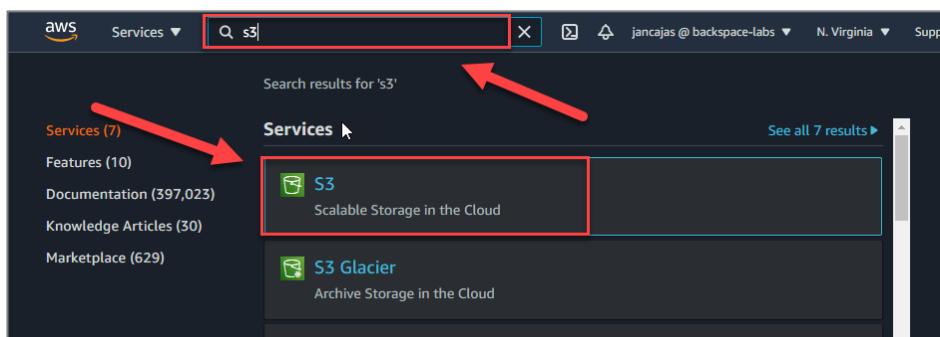
Creating an S3 Bucket and Uploading Files

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

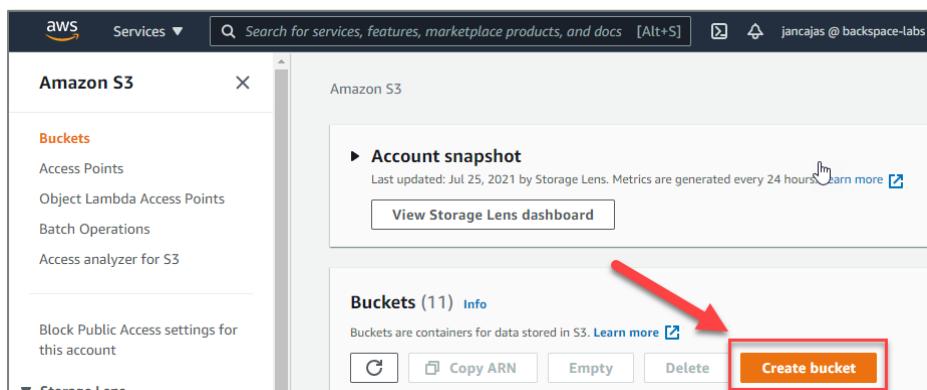
Create a Bucket

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 Info

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

General configuration	
Bucket name	<input type="text" value="jancajas-backspace-intro-aws"/> (Red arrow points here)
Bucket name must be unique and must not contain spaces or uppercase letters. See rules for bucket naming	
AWS Region	US East (N. Virginia) us-east-1
Copy settings from existing bucket - <i>optional</i> Only the bucket settings in the following configuration are copied. <input type="button" value="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.

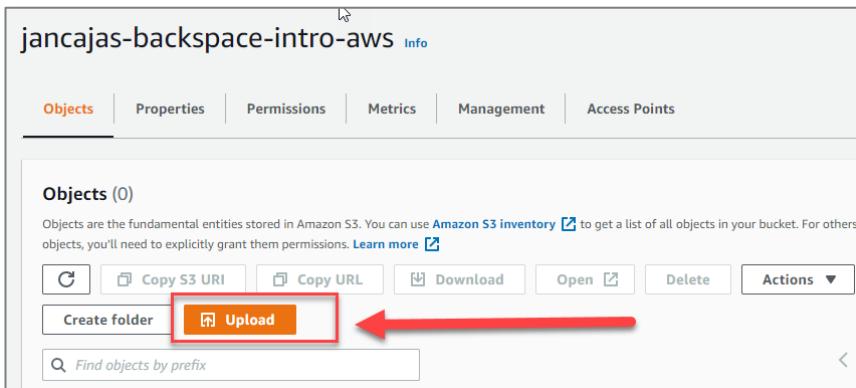
Buckets (12) <small>Info</small>				
Buckets are containers for data stored in S3. Learn more				
<input type="text"/> Find buckets by name				
Name	AWS Region	Access	Creation date	
jancajas-backspace-intro-aws	US East (N. Virginia) us-east-1	Bucket and objects not public	July 26, 2021, 11:01:34 (UTC+08:00)	<input type="button" value="Create bucket"/>

Uploading Files to your Bucket

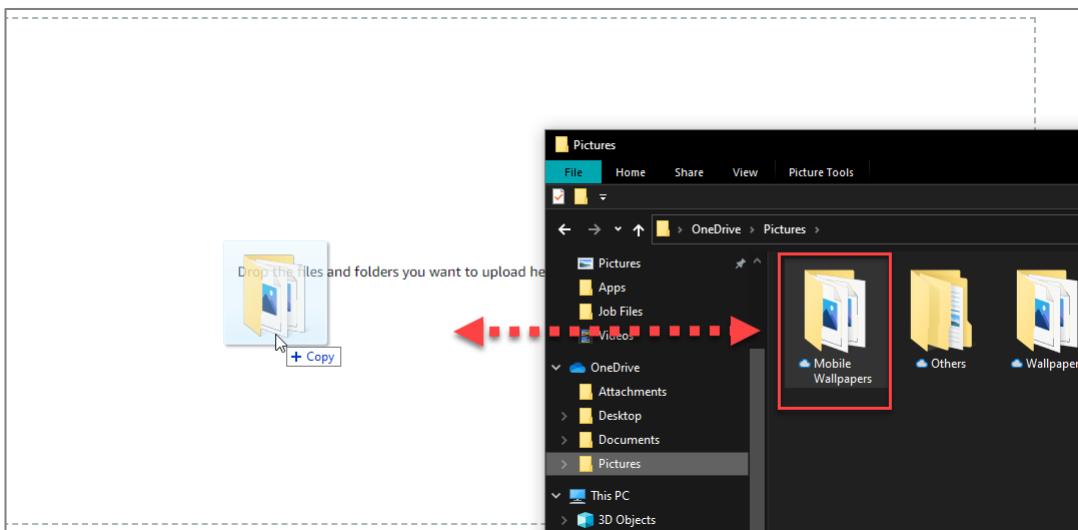
Click on the link to the bucket

Buckets (12) <small>Info</small>				
Buckets are containers for data stored in S3. Learn more				
<input type="text"/> Find buckets by name				
Name	AWS Region	Access	Creation date	
jancajas-backspace-intro-aws	US East (N. Virginia) us-east-1	Bucket and objects not public	July 26, 2021, 11:01:34 (UTC+08:00)	(Red arrow points here)

Click *Upload*



Drag a folder with files onto the form.



Scroll down and click *Upload*

Your files will begin *uploading*

The screenshot shows a progress bar at the top indicating an upload is in progress. The bar is mostly blue with a small white section on the right labeled '0%'. Below the bar, status information is displayed: 'Total remaining: 13 files: 7.1 MB(100.00%)', 'Estimated time remaining: calculating...', and 'Transfer rate: 0 B/s'. A 'Cancel' button is in the top right corner.

Upload: status

ⓘ The information below will no longer be available after you navigate away from this page.

Summary

Destination	Succeeded	Failed
s3://jancajas-backspace-intro-aws	0 files, 0 B (0%)	0 files, 0 B (0%)

Close

Your upload will eventually complete.

The screenshot shows a green header bar with the message 'Upload succeeded' and a link to 'View details below.' Below the bar, the 'Upload: status' summary is shown. The destination is s3://jancajas-backspace-intro-aws. The summary table shows 13 files succeeded (7.1 MB) and 0 failed. The 'Files and folders' tab is selected, showing a table of 13 total files (7.1 MB). One file, '113629_adapted_1080x1920.jpg', is listed with its folder path 'Mobile Wallpapers/' and type 'image/jpeg'. The size is 1.2 MB and the status is 'Succeeded'.

Upload succeeded
View details below.

Upload: status

ⓘ The information below will no longer be available after you navigate away from this page.

Summary

Destination	Succeeded	Failed
s3://jancajas-backspace-intro-aws	13 files, 7.1 MB (100.00%)	0 files, 0 B (0%)

Files and folders (13 Total, 7.1 MB)

Name	Folder	Type	Size	Status
113629_adapted_1080x1920.jpg	Mobile Wallpapers/	image/jpeg	1.2 MB	Succeeded

Downloading files from your bucket

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

Click on the *folder* to view its contents

jancajas-backspace-intro-aws [Info](#)

Objects | Properties | Permissions | Metrics | Management | Access Points

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

[C](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

Find objects by prefix

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	Mobile Wallpapers/	Folder	-	-	-

Select one of the files

Select -> Download

Amazon S3 > jancajas-backspace-intro-aws > Mobile Wallpapers/

Mobile Wallpapers/

Objects | Properties

Objects (13)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

[C](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#)

Find objects by prefix

<input type="checkbox"/>	Name	Type
<input checked="" type="checkbox"/>	113629_adapted_1080x1920.jpg	jpg
<input type="checkbox"/>	118464_adapted_1080x1920.jpg	jpg
<input type="checkbox"/>	118466_adapted_1080x1920.jpg	jpg
<input type="checkbox"/>	119613_adapted_1080x1920.jpg	jpg
<input type="checkbox"/>	63e05867cb8b8ac1f7dd9a707fcbd3ab.jpg	jpg
<input type="checkbox"/>	648ca287a66d3259a5151f45191a7a86.jpg	jpg
<input type="checkbox"/>	7TvcPvp_d.jpg	jpg
<input type="checkbox"/>	DNvId5PVAAA7BR.jpeg	jpeg

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.

```

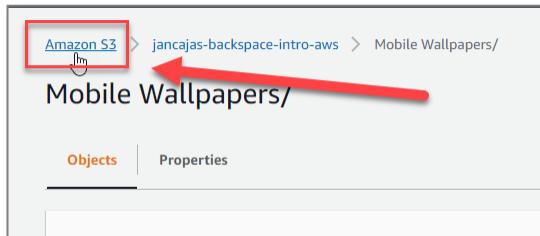
<Error>
  <Code>AccessDenied</Code>
  <Message>Access Denied</Message>
  <RequestId>D3D83296CB201602</RequestId>
  <HostId>
    GEzmWAsPegYl6V58dDQmJPoKPHjPpfmx6yhi6c9w6QmGpajaKgM1Z+aiwzxfIHH87rToHGvFRII=
  </HostId>
</Error>

```

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.



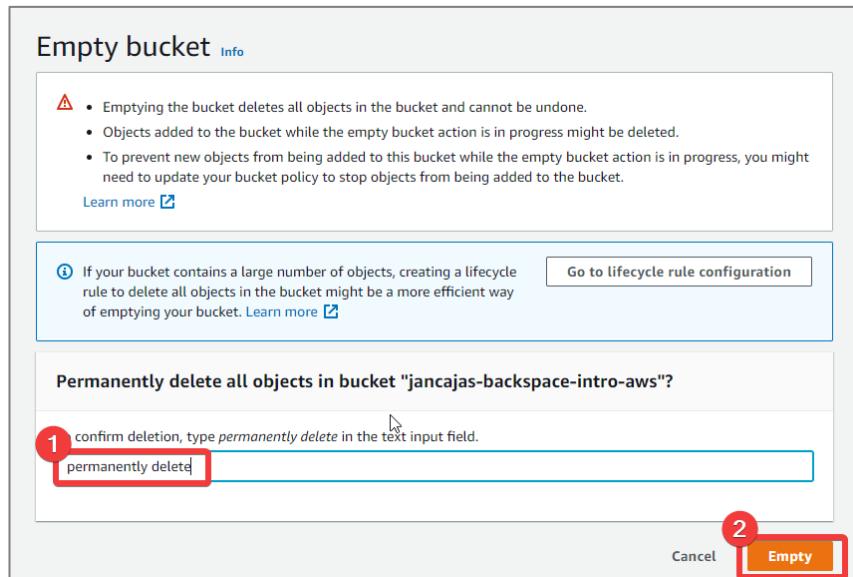
Select the bucket

Click *Empty*

Name	AWS Region	Access	Creation date
jancajas-backspace-intro-aws	US East (N. Virginia) us-east-1	Bucket and objects not public	July 26, 2021, 11:01:34 (UTC+08:00)

Enter *permanently delete*

Click *Empty*



You will see a green success message

Click *Exit*

✓ Successfully emptied bucket "jancajas-backspace-intro-aws"
 View details below. If you want to delete this bucket, use the [delete bucket configuration](#).

Empty bucket: status

The details below are no longer available after you navigate away from this page.

Summary		
Source s3://jancajas-backspace-intro-aws	Successfully deleted 13 objects, 7.1 MB	Failed to delete 0 objects

Failed to delete (0)

Name	Prefix	Version ID	Type	Last modified	Size	Error
No failed object deletions						

Click *Delete*

A screenshot of the AWS S3 Buckets list page. At the top, there are buttons for 'Info', 'Copy ARN', 'Empty', and 'Delete'. A red arrow points to the 'Delete' button. Below the buttons is a search bar with placeholder text 'Find buckets by name'. Underneath is a table with columns: Name, AWS Region, Access, and Creation date. A single row is selected, showing 'jancajas-backspace-intro-aws' in the Name column, 'US East (N. Virginia) us-east-1' in AWS Region, 'Bucket and objects not public' in Access, and 'July 26, 2021, 11:01:34 (UTC+08:00)' in Creation date.

Confirm the name of the bucket to delete

Click *Delete bucket*

A screenshot of the 'Delete bucket' confirmation dialog. It starts with a warning message: '⚠ Deleting a bucket cannot be undone.' followed by 'Bucket names are unique. If you delete a bucket, another AWS user can use the name.' with a 'Learn more' link. Below is a question 'Delete bucket "jancajas-backspace-intro-aws"?'. Step 1 is numbered '1' and highlights the text input field containing 'jancajas-backspace-intro-aws'. Step 2 is numbered '2' and highlights the 'Delete bucket' button.

Creating a SQL Database with RDS

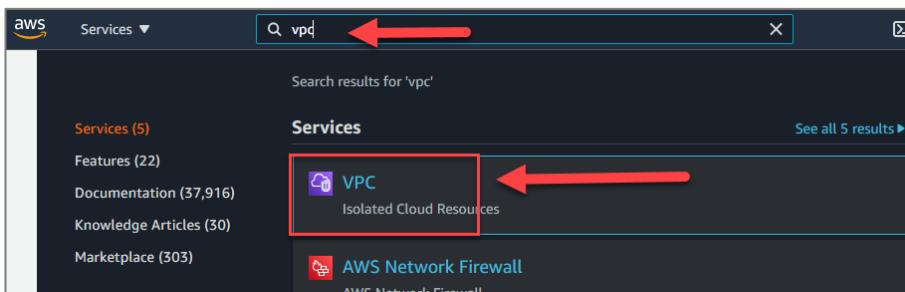
In this section, we will use the Relational Database Service to create a database. We will also connect into the database.

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.

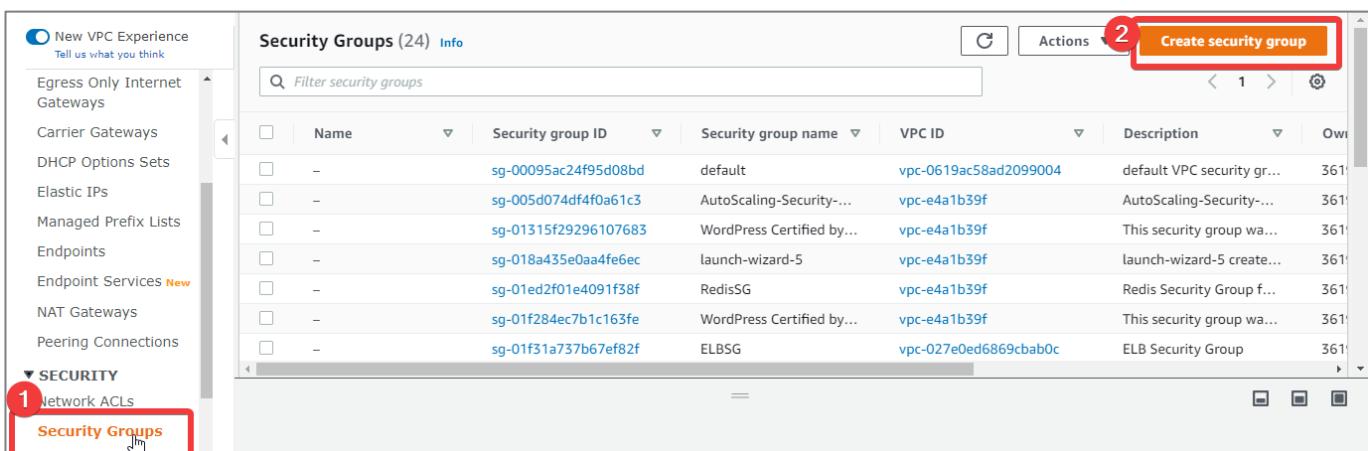
From the AWS console search VPC.

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 Name cannot be edited after creation.

Description Info

VPC Info

Click *Add rule* for *Inbound rules*

Select type *MySQL/Aurora*

Select source *Anywhere-IPv4 0.0.0.0/0* then click *add rule*

Inbound rules Info

Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>	Action
1 MySQL/Aurora	TCP	3306	2 Anywhere-IPv4 0.0.0.0/0		Delete
MySQL/Aurora	TCP	3306	2 Anywhere-IPv4 0.0.0.0/0		Delete
3 Add rule					

Select type *MySQL/Aurora*

Select source *Anywhere-IPv6 ::/0* then click *Create security group*

1

2

3

Security group (sg-04933fe776767520f | backspace-rds-intro-lab) was created successfully

Details			
Security group name backspace-rds-intro-lab	Security group ID sg-04933fe776767520f	Description Inbound internet access to MySQL RDS.	VPC ID vpc-e4a1b39f
Owner 361919435810	Inbound rules count 2 Permission entries	Outbound rules count 1 Permission entry	
Inbound rules	Outbound rules	Tags	Actions

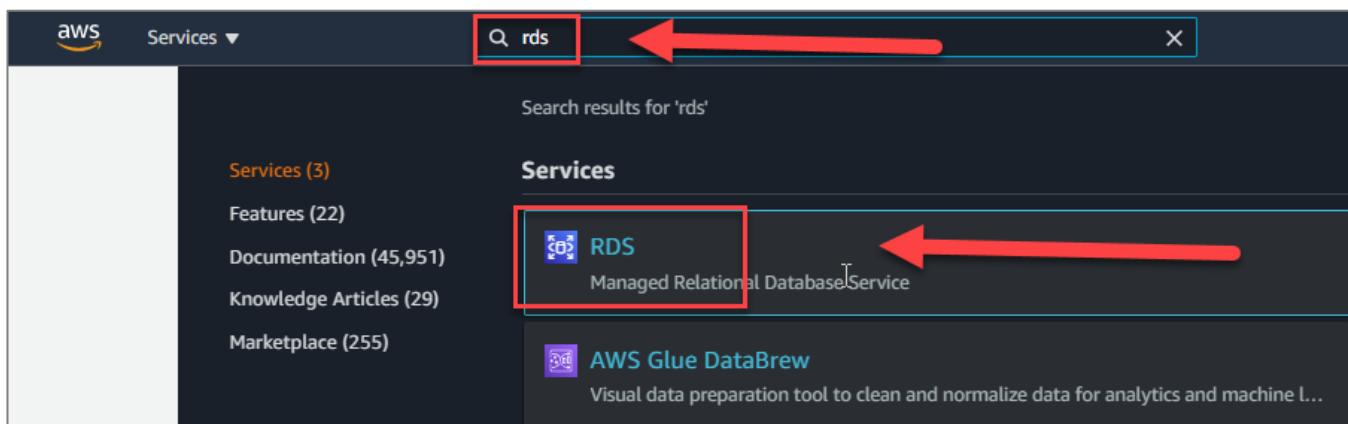
Inbound rules (2)

Name	Security group rule...	IP version	Type	Protocol	Port range
-	sgr-0ec81e86ee05e9026	IPv4	MySQL/Aurora	TCP	3306
-	sgr-0880949fff1171325	IPv6	MySQL/Aurora	TCP	3306

Creating an RDS Database

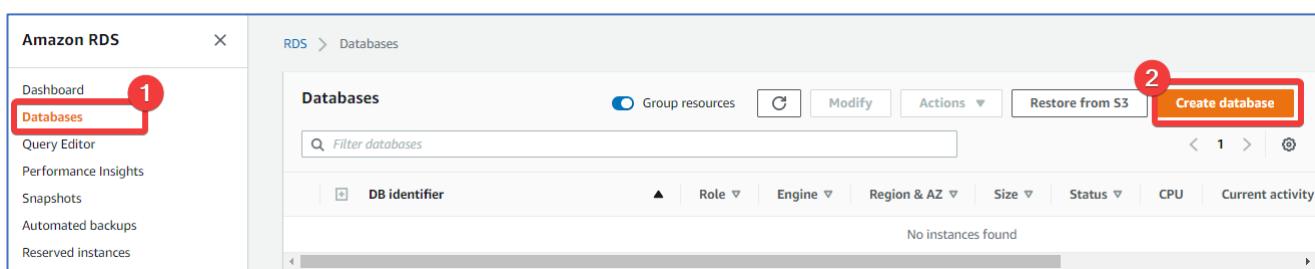
From the AWS console search *RDS*

Click on *RDS*

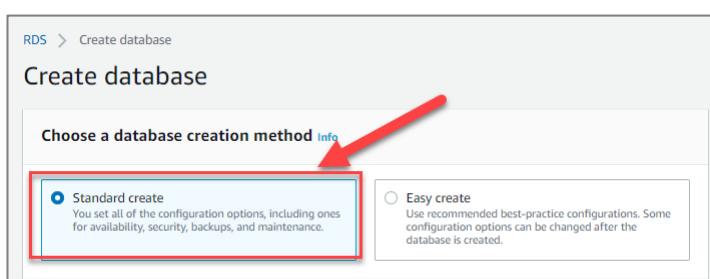


Select *Databases*

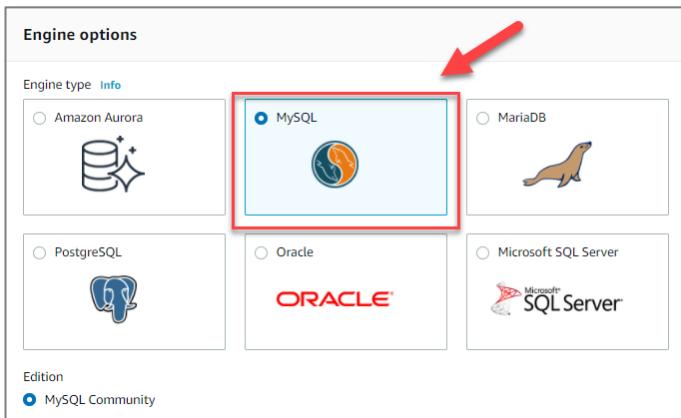
Select *Create database*



Select *Standard Create*



Select MySQL



Select Free Tier

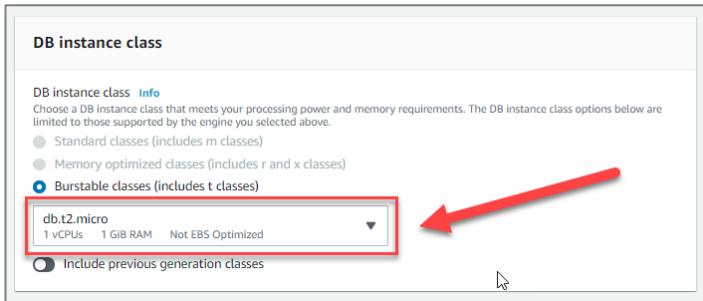


In the *Settings* section give your instance a name/identifier.

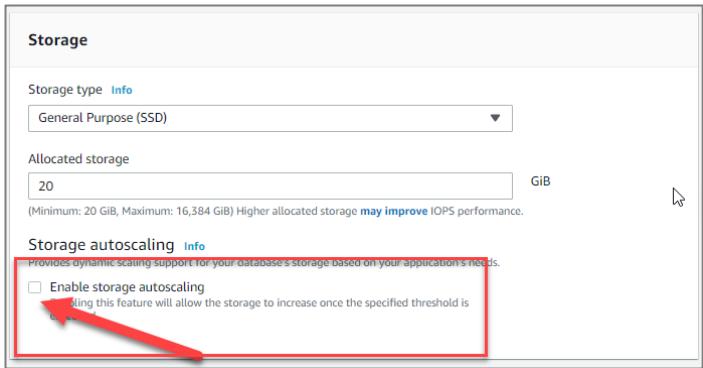
Fill in a master *username and password* (remember this we will need it later)

The screenshot shows the 'Settings' section of the AWS RDS setup wizard. The 'DB instance identifier' field contains 'backspace-intro-aws' (highlighted with a red box). In the 'Master username' field, 'admin' is typed (highlighted with a red box). The 'Master password' and 'Confirm password' fields both contain '*****' (highlighted with a red box). A red arrow points from the top right towards the DB instance identifier field.

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



Uncheck *Enable storage autoscaling*



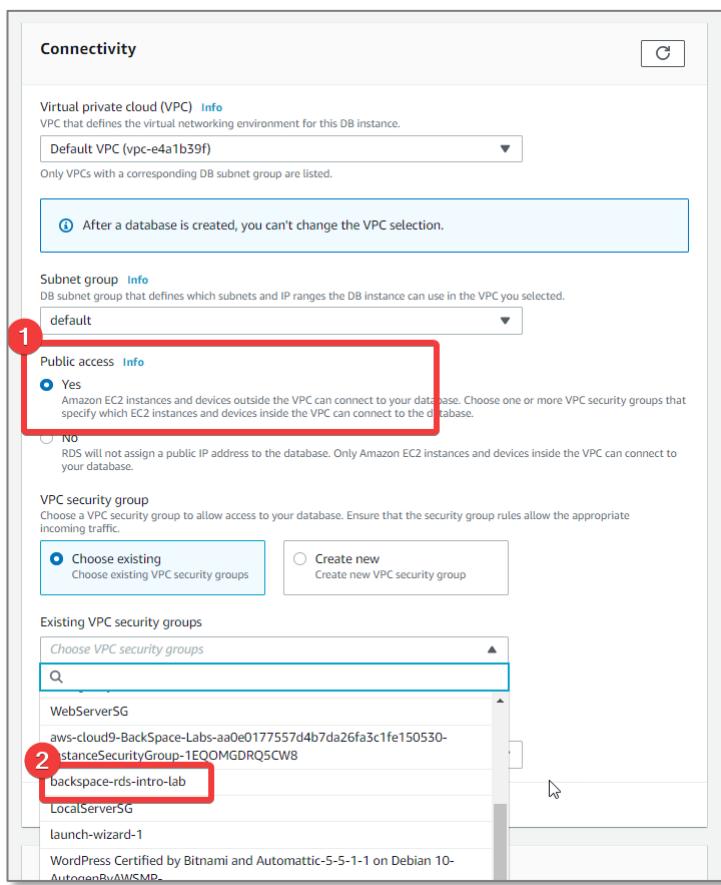
Scroll down to *Connectivity*

Expand *Additional connectivity configuration*

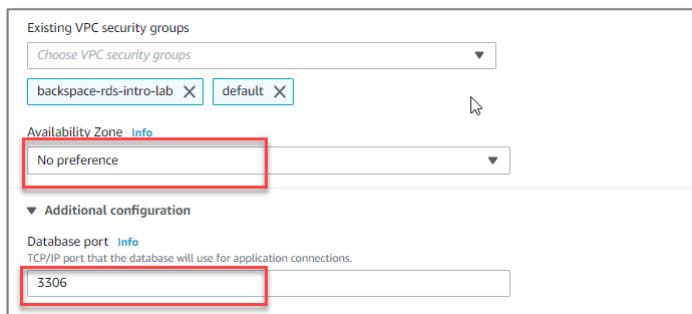
Select **yes** for *publicly accessible* (we will look at security later in the course)

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)

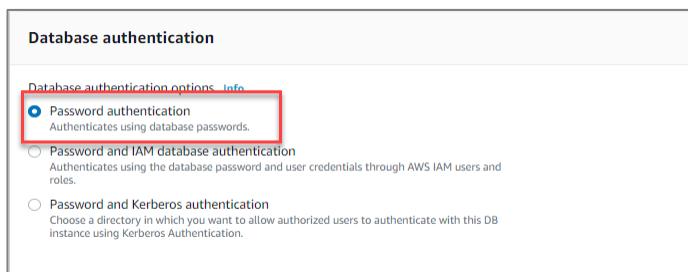


Click outside the list to add the security group. You should then see the security group added.



Scroll down to *Database authentication*

Leave as *Password authentication*

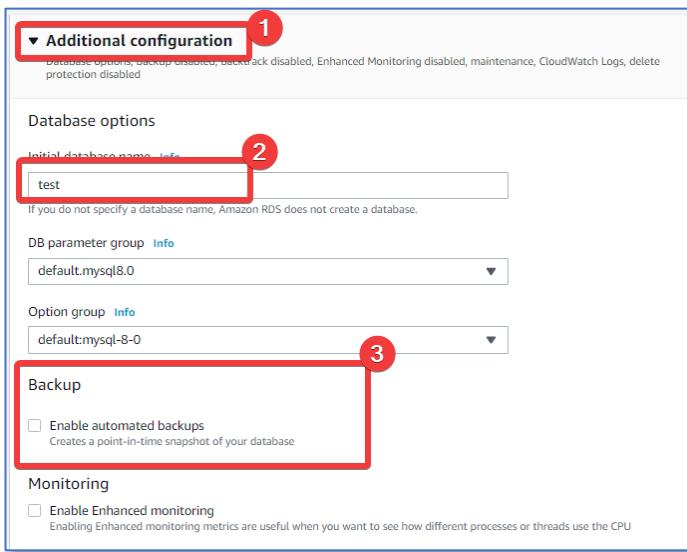


Scroll down and expand *Additional configuration*

Enter a database name.

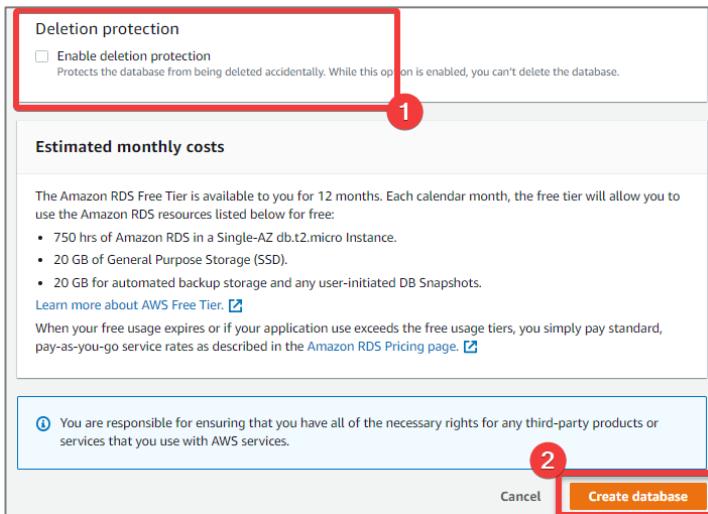
Uncheck *Enable automatic backups*

Leave all other options default.



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

Click *Create database*



Click on the database details link

The screenshot shows the AWS RDS 'Databases' page. At the top, a message says 'Creating database backspace-intro-aws' and 'Your database might take a few minutes to launch.' There is a 'View credential details' button. Below the message, the 'Databases' section has a 'Create database' button. A red arrow points to the 'DB identifier' column where 'backspace-intro-aws' is listed. Another red arrow points to the 'Status' column where 'Creating' is displayed.

Your instance will show status *Creating*

The screenshot shows the 'backspace-intro-aws' database details page. In the 'Summary' section, the 'Status' field is highlighted with a red box and a red arrow pointing to it, showing the value 'Creating'. Other fields include 'DB identifier: backspace-intro-aws', 'Role: Instance', 'CPU: -', 'Current activity', 'Engine: MySQL Community', 'Class: db.t2.micro', and 'Region & AZ: us-east-1f'.

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*

Successfully created database **backspace-intro-aws**

RDS > Databases > backspace-intro-aws

backspace-intro-aws

Summary

DB identifier backspace-intro-aws	CPU -	Status Available	Class db.t2.micro
Role Instance	Current activity	Engine MySQL Community	Region & AZ us-east-1f

Scroll down and copy the database server *endpoint*

Connectivity & security | Monitoring | Logs & events | Configuration | Maintenance & backups | Tags

Connectivity & security

Endpoint backspace-intro-aws.clbmnfzd56wx.us-east-1.rds.amazonaws.com	Networking	Security
Port 3306	Availability zone us-east-1f	VPC security groups backspace-rds-intro-lab (sg-04933fe776767520f) (active) default (sg-7d1df536) (active)
	VPC Default VPC (vpc-e4a1b39f)	Public accessibility Yes
	Subnet group default	Certificate authority rds-ca-2019
	Subnets subnet-ec2514b0 subnet-b13a5efb subnet-b8ab79ec subnet-c94c9fe7 subnet-7d099672 subnet-557c9c6b	Certificate authority date August 23, 2024, 01:08 (UTC±1:08)

Open the *MySQL Workbench* application click to add a new connection

MySQL Workbench

File Edit View Database Tools Scripting Help

Welcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

Browse Documentation > Read the Blog > Discuss on the Forums >

MySQL Connections + S

Filter connections

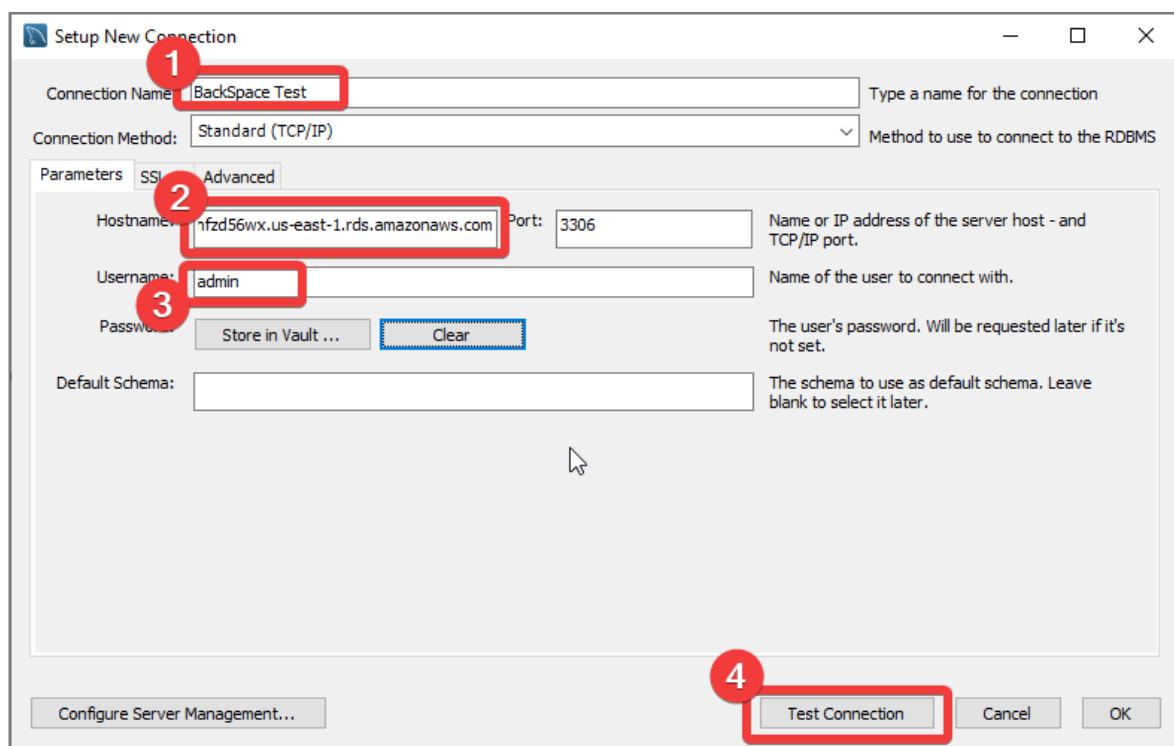
Give the connection a name.

The Hostname will be the RDS server endpoint.

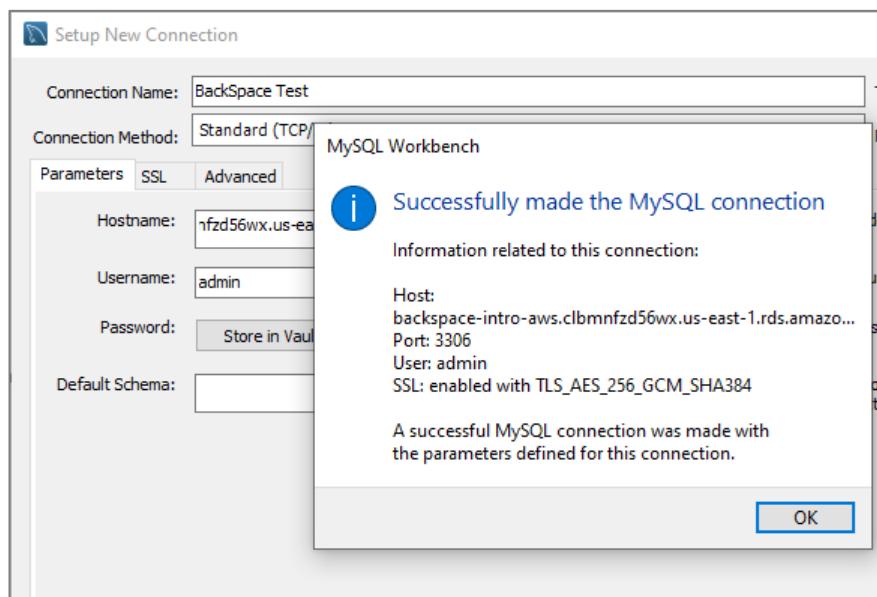
The port will be 3306.

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

Click *Test Connection*

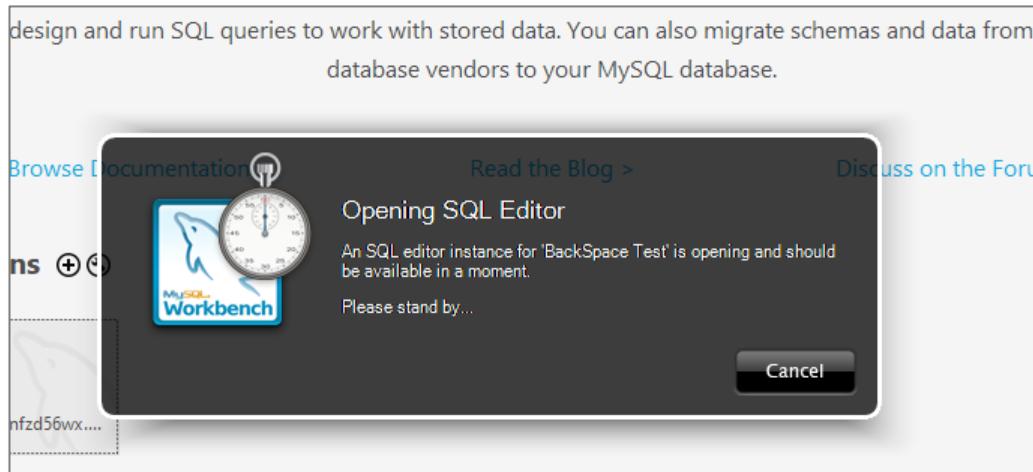
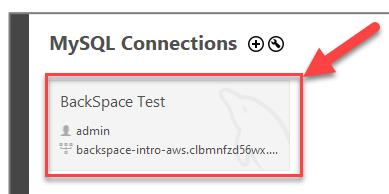


Enter the *password* you created in RDS for your master username



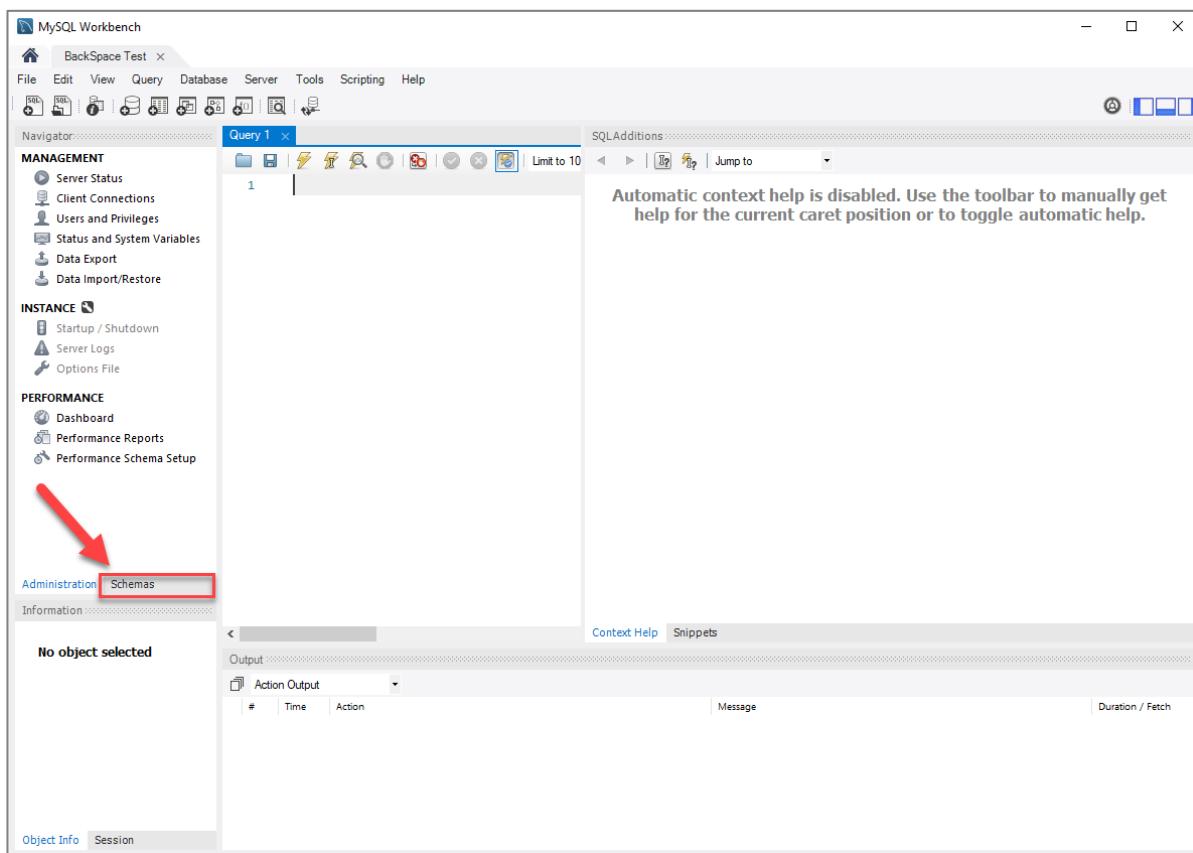
Click OK

Click on the Connection

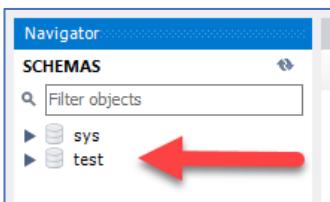


You will soon be connected to your database server

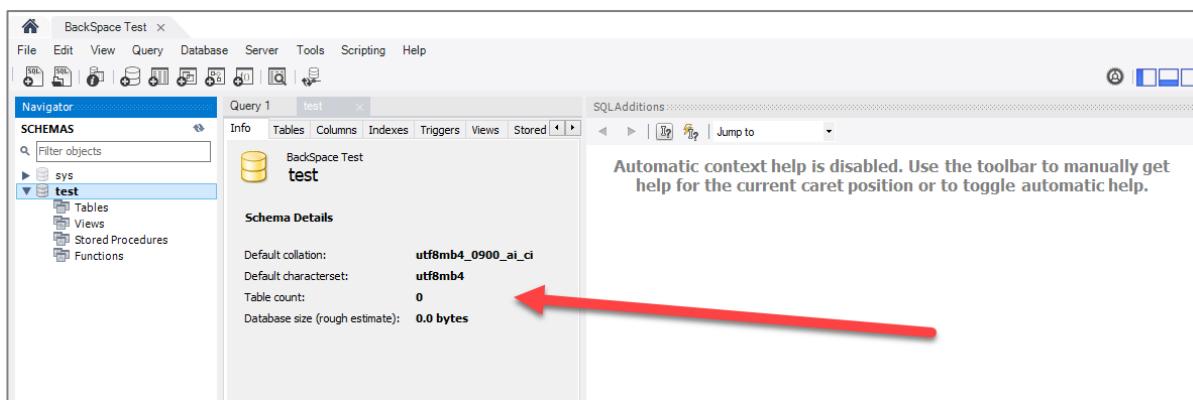
If you cannot connect then please see the "Troubleshooting Connection Issues" below.



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.



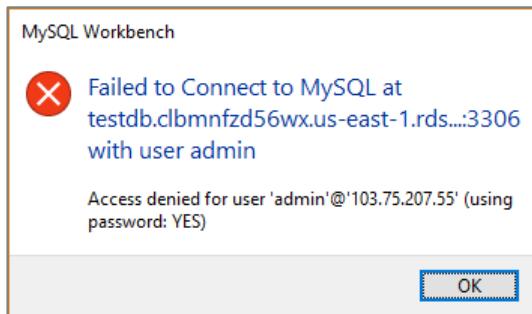
You then get an information screen for the database.



Troubleshooting Connection Issues

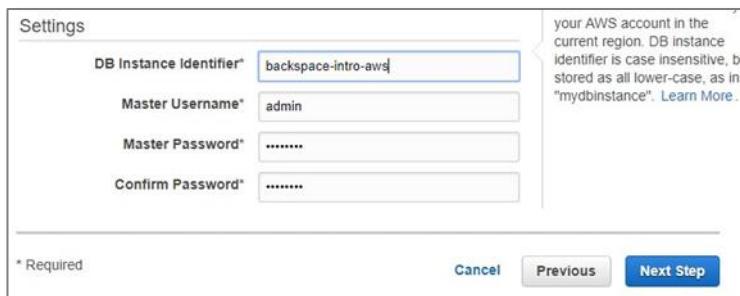
If you are getting connection errors then check the following:

Wrong Username / Password

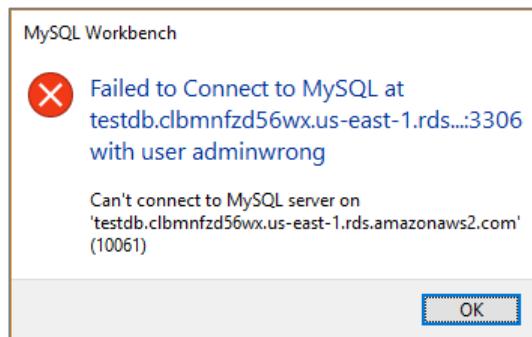


Make sure you use the correct username and password.

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



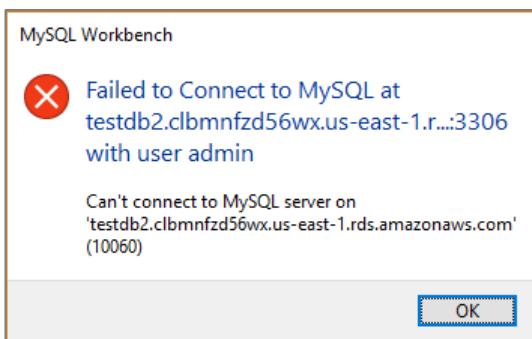
Bad Connection String



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**

Security Groups (1/4) Info						
	Name	Security group ID	Security group name	VPC ID	Description	Actions Create security group
<input type="checkbox"/>	–	sg-02dde92488961eb9a	WordPress Certified by...	vpc-800a67fd	This security group wa...	
<input type="checkbox"/>	aws-cloud9-Backsp...	sg-0a4a2df247d894188	aws-cloud9-Backspace...	vpc-800a67fd	Security group for AW...	
<input checked="" type="checkbox"/>	–	sg-0cf461def34692828	backspace-rds-intro-lab	vpc-800a67fd	Inbound internet acces...	
<input type="checkbox"/>	–	sg-7aaaba60	default	vpc-800a67fd	default VPC security gr...	

You will be taken to the EC2 console

Select the *Inbound rules* tab

Click *Edit inbound rules*

sg-0cf461def34692828 - backspace-rds-intro-lab

Details

Security group name backspace-rds-intro-lab	Security group ID sg-0cf461def34692828	Description Inbound internet access to MySQL RDS.	VPC ID vpc-800a67fd
Owner 991610390270	Inbound rules count 2 Permission entries	Outbound rules count 1 Permission entry	

Inbound rules (2)

Name	Security group rule...	IP version	Type	Protocol
-	sgr-00721e9daa350983c	IPv4	MySQL/Aurora	TCP
-	sgr-071999b4a0c451e9d	IPv6	MySQL/Aurora	TCP

Change inbound rule to *Anywhere IPv4 0.0.0.0/0*

Edit inbound rules

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-00721e9daa350983c	MySQL/Aurora	TCP	3306	Custom 0.0.0.0/0	
sgr-071999b4a0c451e9d	MySQL/Aurora	TCP	3306	Anywhere-IPv4 ::/0	

Add rule

Change inbound rule to *Anywhere IPv6 ::/0*, click *Save rules*

Edit inbound rules

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-00721e9daa350983c	MySQL/Aurora	TCP	3306	Custom 0.0.0.0/0	
sgr-071999b4a0c451e9d	MySQL/Aurora	TCP	3306	Anywhere-IPv6 ::/0	

Add rule

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

RDS > Databases > backspace-intro-aws

backspace-intro-aws

Summary			
DB identifier backspace-intro-aws	CPU <div style="width: 2.54%;">2.54%</div>	Status Available	Class db.t2.micro
Role Instance	Current activity <div style="width: 2%;">2 Connections</div>	Engine MySQL Community	Region & AZ us-east-1f

- Modify
- Actions ▾**
- Stop
- Reboot
- Delete**
- Create read replica
- Promote
- Take snapshot
- Restore to point in time

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*

Delete backspace-intro-aws instance?

Are you sure you want to Delete the backspace-intro-aws DB Instance?

Create final snapshot?
Determines whether a final DB Snapshot is created before the DB instance is deleted.

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

To confirm delete, type *delete me* into the field

delete me

⚠ We strongly recommend taking a final snapshot before instance deletion since after your instance is deleted, automated backups will no longer be available.

Cancel **Delete**

Click on the VPC security group we created previously

RDS > Databases > backspace-intro-aws

backspace-intro-aws

Summary

DB identifier backspace-intro-aws	CPU <div style="width: 2.54%;">2.54%</div>	Status ✖ Deleting	Class db.t2.micro
Role Instance	Current activity <div style="width: 2%;">2 Connections</div>	Engine MySQL Community	Region & AZ us-east-1f

Connectivity & security | Monitoring | Logs & events | Configuration | Maintenance & backups | Tags

Connectivity & security

Endpoint & port	Networking	Security
Endpoint backspace-intro-aws.clbmnfz56wx.us-east-1.rds.amazonaws.com	Availability zone us-east-1f	VPC security groups backspace-rds-intro-lab (sg-04933fe776767520f) (active) <default (sg-7d1df536)<br=""></default> (active)
Port	VPC Default VPC (vpc-e4a1b39f)	

Select Actions -> Delete security group

Security Groups (1/1) Info

Actions ▾ Create security group

- 1 Manage tags
- 2 Manage stale rules
- Copy to new security group
- Delete security groups**

<input checked="" type="checkbox"/>	Name	Security group ID	Security group name	VPC ID
<input checked="" type="checkbox"/>	-	sg-04933fe776767520f	backspace-rds-intro-lab	vpc-e4a1b39f

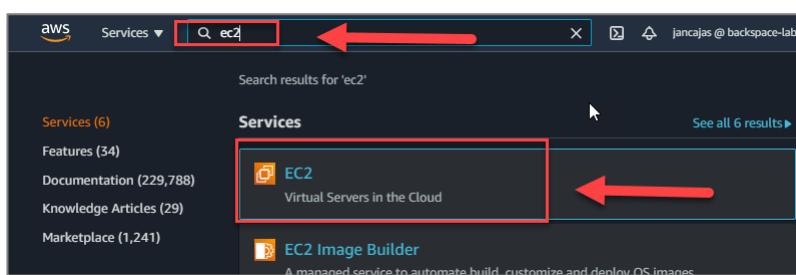
Creating a Web Server with EC2

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

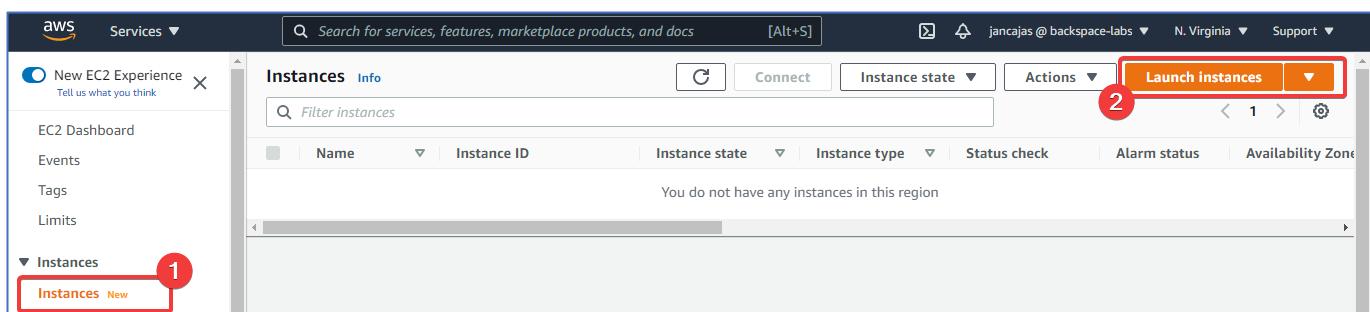
Launch an EC2 Instance

From the AWS console search *EC2*

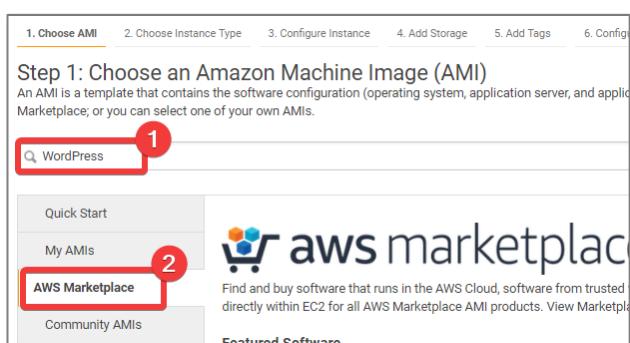
Click *EC2*



Select *Instances* -> *Launch Instances*



Select the AWS Marketplace and search for *WordPress*



Select the Bitnami AMI

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search: WordPress

Quick Start (0)
My AMIs (0)
AWS Marketplace (285)
Community AMIs (1408)
Categories
All Categories
Infrastructure Software (39)

WordPress Certified by Bitnami and Automattic
★★★★★ (128) | 5.7.2-40-r14 on Debian 10 | Previous versions | By Bitnami
Linux/Unix, Debian 10 (64-bit (x86) Amazon Machine Image (AMI)) | Updated: 7/16/21
WordPress is the world's most popular content management platform. It includes the new Gutenberg editor and over 45,000 themes and plugins. This image is certified by Bitnami as secure, up-to-date, and packaged using industry best practices, and approved by Automattic, the experts behind WordPress.
More info

WordPress with NGINX and SSL Certified by Bitnami and Automattic
★★★★★ (15) | 5.7.2-35-r15 on Debian 10 | Previous versions | By Bitnami

Cancel and Exit

Search by Systems Manager parameter

1 to 10 of 285 Products > >

Select

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

WordPress Certified by Bitnami and Automattic

Free tier eligible

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

Pricing Details

Hourly Fees			
Instance Type	Software	EC2	Total
t2.micro	\$0.00	\$0.012	\$0.023/hr
t2.small	\$0.00	\$0.023	\$0.046/hr
t2.medium	\$0.00	\$0.093	\$0.186/hr
t2.large	\$0.00	\$0.186	\$0.371/hr
t2.xlarge	\$0.00	\$0.371	\$0.742/hr
t2.2xlarge	\$0.00	\$0.742	\$1.484/hr
t3.micro	\$0.00	\$0.01	\$0.021/hr
t3.small	\$0.00	\$0.021	\$0.042/hr
t3.medium	\$0.00	\$0.083	\$0.166/hr
t3.large	\$0.00	\$0.166	\$0.333/hr
t3.xlarge	\$0.00	\$0.333	\$0.666/hr
t3.2xlarge	\$0.00	\$0.666	\$1.333/hr
t3a.micro	\$0.00	\$0.009	\$0.009/hr
t3a.small	\$0.00	\$0.019	\$0.038/hr
t3a.medium	\$0.00	\$0.038	\$0.075/hr
t3a.large	\$0.00	\$0.075	\$0.15/hr
t3a.xlarge	\$0.00	\$0.15	\$0.301/hr
t3a.2xlarge	\$0.00	\$0.301	\$0.602/hr
t3.2xlarge	\$0.00	\$0.602	\$1.204/hr

By Bitnami
Customer Rating ★★★★★ (128)
Latest Version 5.7.2-40-r14 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.

Continue

Choose the *t2 micro instance*.

Click Next: *Configure Instance Details*

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 families ▾ Current generation ▾ Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, ~ 1 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
0	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
1	t2	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
2	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
3	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
4	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
5	t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes
6	t2	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
7	t3	t3.nano	2	0.5	EBS only	Yes	Up to 1 gigabit	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

Select enable for *Auto-assign Public IP*

Click Next: *Add Storage*

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: 1 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

Domain join directory: No directory Create new directory

IAM role: None Create new IAM role

⚠ You do not have permissions to list instance profiles. Contact your administrator, or check your IAM permissions.

Shutdown behavior: Stop

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

Enable termination protection: Protect against accidental termination

Cancel Previous Review and Launch Next: Add Storage

Click Next: Add Tags

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-04571561c7770fafb	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

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. A copy of a tag can be applied to volumes, instances or both. Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	(128 characters maximum)	Value	(256 characters maximum)
This resource currently has no tags			

Choose the Add tag button or [click to add a Name tag](#). Make sure your [IAM policy](#) includes permissions to create tags.

Add Tag (Up to 50 tags maximum)

Give it a name and click Next: *Configure Security Group*

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	(128 characters maximum)	Value	(256 characters maximum)	Instances	Volumes	Network Interfaces
Name		backspace-lab-intro-ec2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Add another tag (Up to 50 tags maximum)

Cancel Previous Review and Launch Next: Configure Security Group

Click *Review and Launch*

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: This security group was generated by AWS Marketplace and is based on recom|

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

Add Rule

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.

Cancel Previous Review and Launch

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.

⚠ Improve your instances' security. Your security group, WordPress Certified by Bitnami and Automatic-5-7-2-40-r14 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 Automatic
 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

Launch



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

Select an existing key pair or create a new key pair

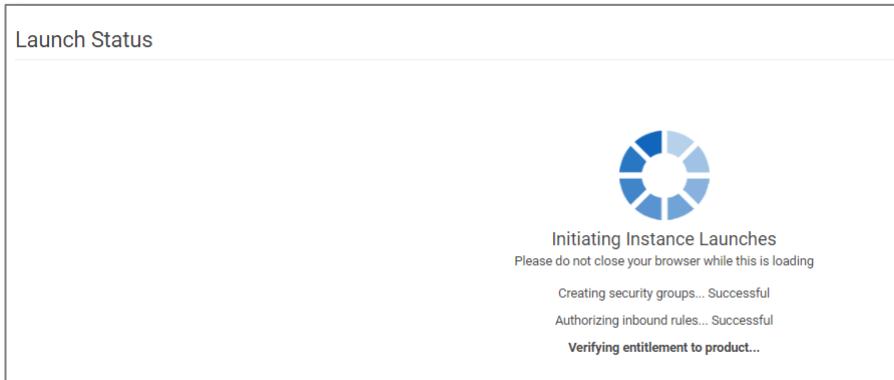
A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

1: The selected key pair will be added to the set of keys authorized for this instance. Learn more [about connecting to an instance from a public AMI.](#)

I acknowledge that without a key pair, I can connect to this instance only by using EC2 Instance Connect or the password built into the AMI. Note that EC2 Instance Connect is only supported on Amazon Linux 2 and Ubuntu. [Learn more.](#)

2 Launch Instances

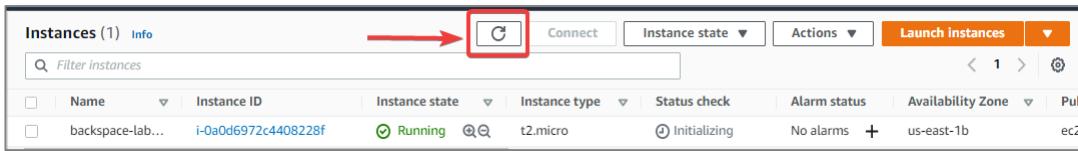
Wait for launch to initiate



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

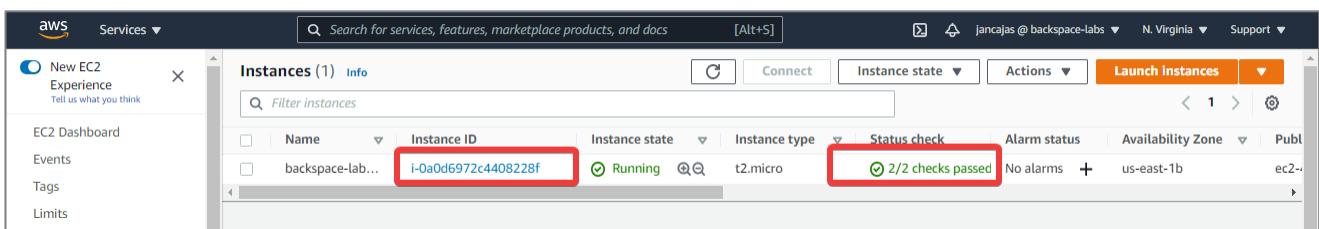


After a few minutes, the status of the instance will change to running and status checks will be completed (you will need to *refresh* the screen to see any changes).



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* this will open it in HTTPS not HTTP).

Instance summary for i-0a0d6972c4408228f (backspace-lab-intro-ec2) [Info](#)
Updated less than a minute ago

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0a0d6972c4408228f (backspace-lab-intro-ec2)	4.192.96.150 open address 🔗	172.31.15.216
Instance state	Public IPv4 DNS	Private IPv4 DNS
Running	ec2-44-192-96-150.compute-1.amazonaws.com open address 🔗	ip-172-31-15-216.ec2.internal
Instance type	Elastic IP addresses	VPC ID
t2.micro	-	vpc-e4a1b39f (Default VPC) 🔗
AWS Compute Optimizer finding	IAM Role	Subnet ID
Opt-in to AWS Compute Optimizer for recommendations. Learn more ↗	-	subnet-8bab79ec 🔗

Paste the IP address in your browser.

User's blog – Just another WordPress site

Not secure | 44.192.96.150

Hello world!

Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

Published July 26, 2021
Categorized as [Uncategorized](#)

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*

Port range	Protocol	Source	Security groups
80	TCP	0.0.0.0/0	WordPress Certified by Bitnami and Automatic-5-7-2-40-r1...
22	TCP	0.0.0.0/0	WordPress Certified by Bitnami and Automatic-5-7-2-40-r1...
443	TCP	0.0.0.0/0	WordPress Certified by Bitnami and Automatic-5-7-2-40-r1...

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

Click on the *Inbound rules* tab

Click on *Edit inbound rules*

The screenshot shows the AWS EC2 Dashboard with the 'Instances' section expanded. On the left, there's a sidebar with various options like New EC2 Experience, Events, Tags, Limits, Instances, and Images. The 'Instances' section is active, showing a list of security groups. One specific security group, 'sg-046e9504d2b132f9a', is selected and highlighted with a red box and a circled number 1. A modal window has opened for this security group, titled 'sg-046e9504d2b132f9a - WordPress Certified by Bitnami and Automattic-5-7-2-40-r14 on Debian 10-AutogenByAWSMP-'. Inside the modal, the 'Inbound rules' tab is selected and highlighted with a red box and a circled number 2. At the bottom right of the modal, the 'Edit inbound rules' button is also highlighted with a red box and a circled number 3.

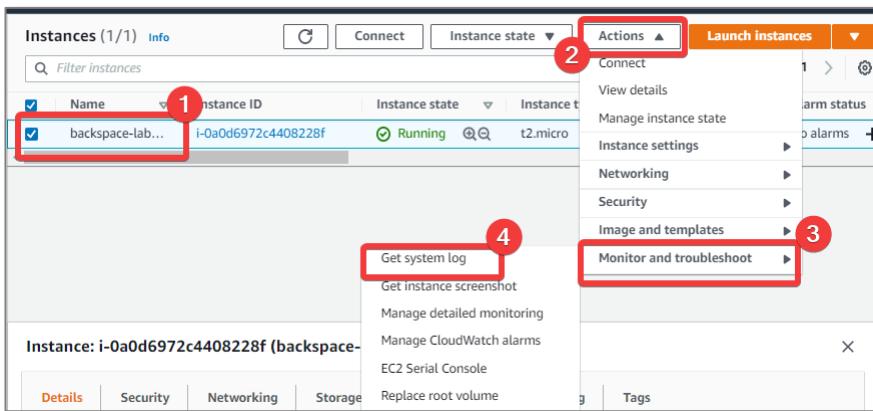
Add a rule for *HTTP* and *Anywhere IPv4 0.0.0.0/0*

Click *Save rules*

The screenshot shows the 'Edit inbound rules' dialog box. At the top, it says 'Edit inbound rules' and 'Inbound rules control the incoming traffic that's allowed to reach the instance.' Below this is a table for managing inbound rules. The first row shows a rule for 'sg-0392cfa23b35a8dd6' with 'Type' set to 'HTTP' (circled with number 1), 'Protocol' set to 'TCP', 'Port range' set to '80', and 'Source' set to 'Anywhere-IPv4' (circled with number 2). A dropdown menu for 'Source' is open, showing 'Custom' (selected), 'Anywhere-IPv4', 'Anywhere-IPv6', and 'My IP'. At the bottom right of the dialog, the 'Save rules' button is highlighted with a red box and a circled number 3.

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.

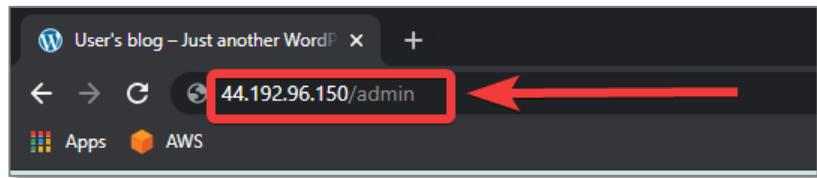
Get system log

```

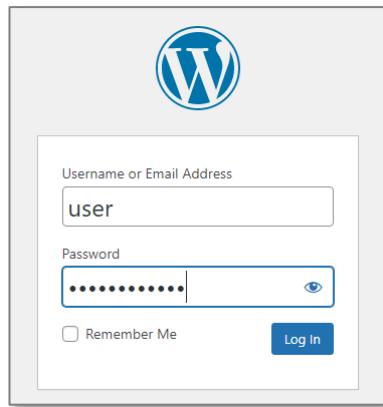
79.739455] bitnami[529]: Setting up swapspace version 1, size = 634.8 MiB (665595904 bytes)
[ 79.746662] bitnami[529]: no label, UUID=d1eca222-d8da-488e-8984-d4c049bb774a
[ 80.426699] Adding 64996k swap on /mnt/.bitnami.swap. Priority:-2 extents:15 across:2468620k SSFS
[ 79.781413] bitnami[529]: ## 2021-07-26 09:40:37+00:00 ## INFO ## Running /opt/bitnami/var/init/pre-start#040_check_if_demo_machine...
[ 80.948827] bitnami[529]: ## 2021-07-26 09:40:37+00:00 ## INFO ## Running /opt/bitnami/var/init/pre-start#050_change_boot_log_permissions...
[ 80.969760] bitnami[529]: ## 2021-07-26 09:40:38+00:00 ## INFO ## Running /opt/bitnami/var/init/pre-start#060_get_default_passwords...
[ 81.123939] bitnami[529]: ######
[ 81.136804] bitnami[529]: #
[ 81.147198] bitnami[529]: #      Setting Bitnami application password to 'HVSKSskYZ03b5' #
[ 81.159612] bitnami[529]: #      (the default application username is 'User' ) #
[ 81.176062] bitnami[529]: #
[ 81.178815] bitnami[529]: #####
[ 81.205644] bitnami[529]: ## 2021-07-26 09:40:38+00:00 ## INFO ## Running first-boot...
[ 85.431794] bitnami[529]: 2021-07-26T09:40:42.888Z - info: Saving configuration info to disk
[ 85.793516] bitnami[529]: 2021-07-26T09:40:43.251Z - info: Saving configuration info to disk
[ 85.803836] bitnami[529]: 2021-07-26T09:40:43.258Z - warn: No peerAddress provided. Skipping hosts file section
[ 86.746775] bitnami[529]: 2021-07-26T09:40:44.204Z - info: Data disk not present
[ 119.053550] bitnami[529]: 2021-07-26T09:41:16.510Z - info: Initializing module bnsupport
[ 119.069599] bitnami[529]: 2021-07-26T09:41:16.513Z - info: Initializing module gonit
[ 119.082786] bitnami[529]: 2021-07-26T09:41:16.513Z - info: Initializing module render-template
[ 119.096164] bitnami[529]: 2021-07-26T09:41:16.513Z - info: Initializing module php
[ 119.633416] bitnami[529]: 38;5;6mphp @|38;5;5m09:41:17.08 @|0m@|38;5;2mINFO @|0m ==> Configuring PHP options

```

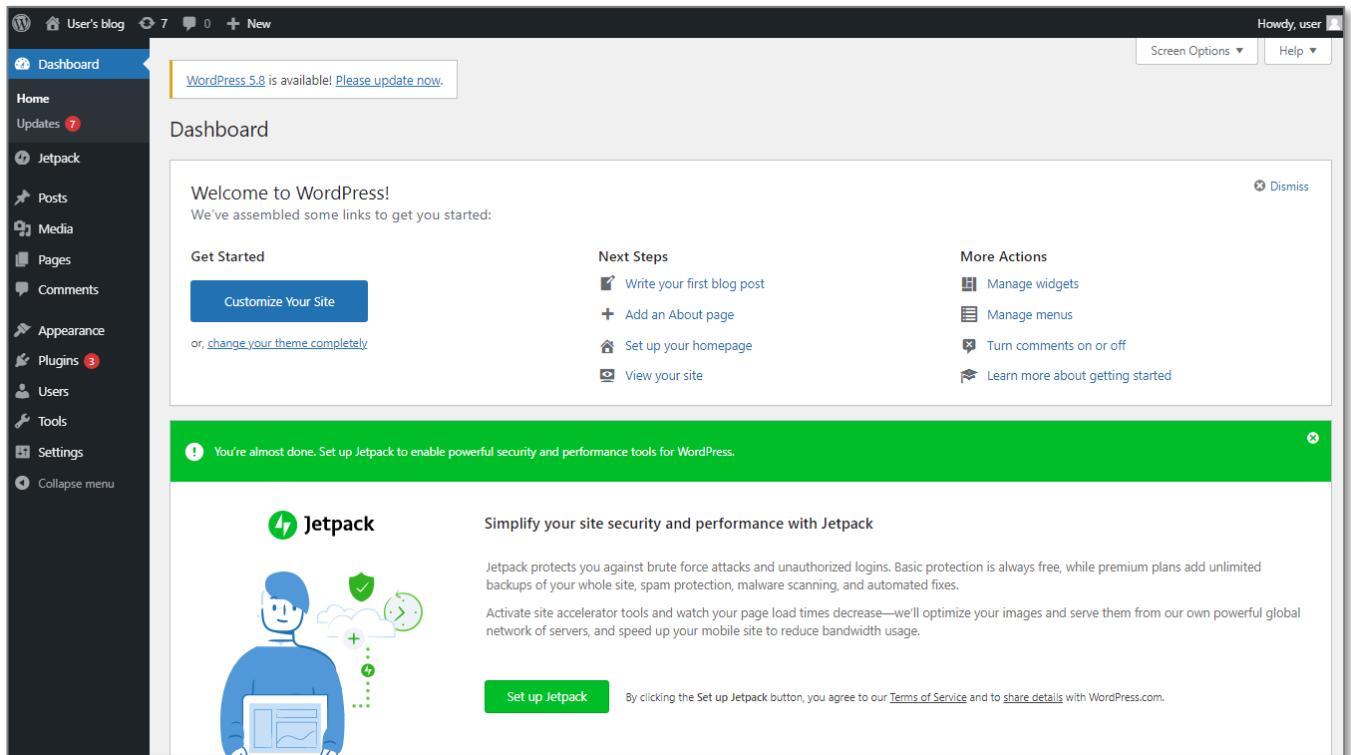
Go to the *admin* subdirectory of your website in your browser



Enter Username *user* and paste in the password

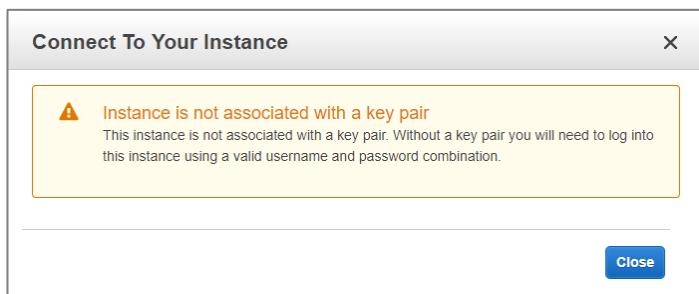


You will now be in the *admin* section of your WordPress application



Troubleshooting logging in to the WordPress application

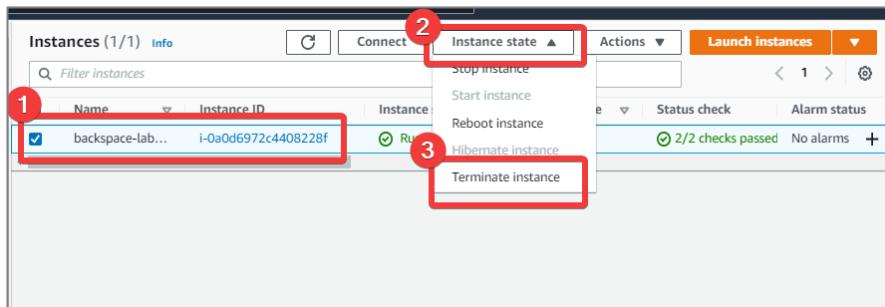
If you get the following message:



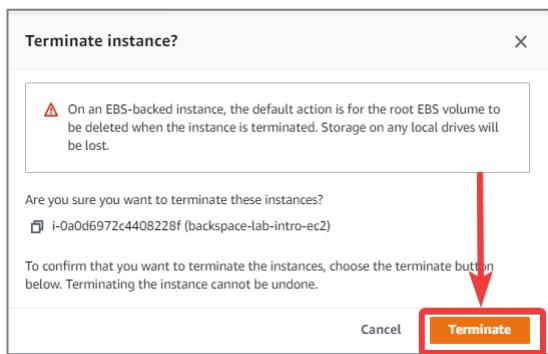
You have tried to connect to the Linux operating system by clicking on *Connect*. Do not click on connect, select *Actions – Instance settings - Get System Log* as detailed previously.

Clean up

Select *Actions -> Instance State -> Terminate*



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

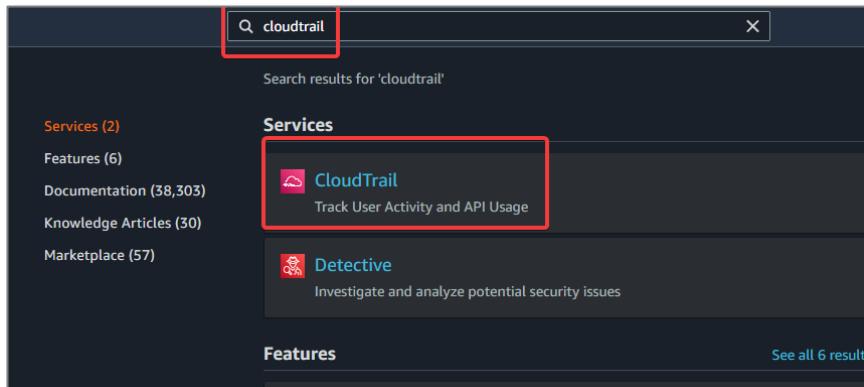


▶ Monitoring User Activity with AWS CloudTrail

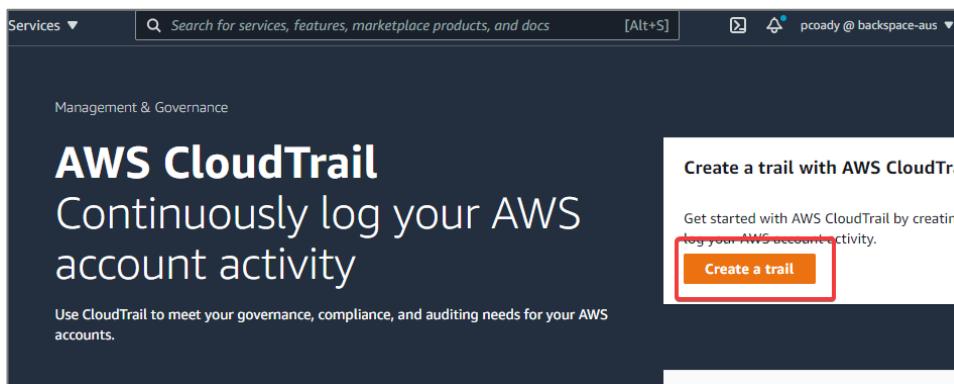
In this section, we will use the AWS CloudTrail service to create a trail and monitor user activity on our account.

From the AWS console search *CloudTrail*

Click *CloudTrail*



Click on *Create a trail*



Give the trail a name

Click on *Create trail*

CloudTrail > Quick trail create

Quick trail create

Trail details

Start logging management events by creating a trail with simplified settings. Logs are sent to an S3 bucket we create on your behalf. To choose a different bucket or additional events, go to the full [Create trail](#) workflow.

A trail created in the console is a multi-region trail. [Learn more](#)

Trail name
Enter a display name for your trail.
backspace-events

3-128 characters. Only letters, numbers, periods, underscores, and dashes are allowed.

Trail log bucket and folder

aws-cloudtrail-logs-161762789278-bf23a295

Logs will be stored in aws-cloudtrail-logs-161762789278-bf23a295/AWSLogs/161762789278

Though there is no cost to log these events, you incur charges for the S3 bucket that we create to store your logs.

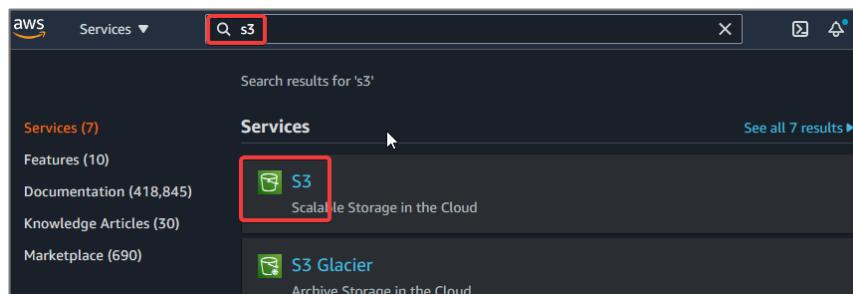
Create trail

Your trail is now active

CloudTrail > Trails

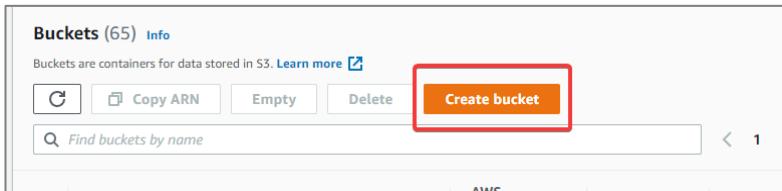
Name	Home region	Multi-region trail	Insights	Organization trail	S3 bucket	Log file prefix	CloudWatch Logs log group	Status
backspace-events	US East (N. Virginia)	Yes	Disabled	No	aws-cloudtrail-logs-161762789278-bf23a295			Logging

Go to the S3 console

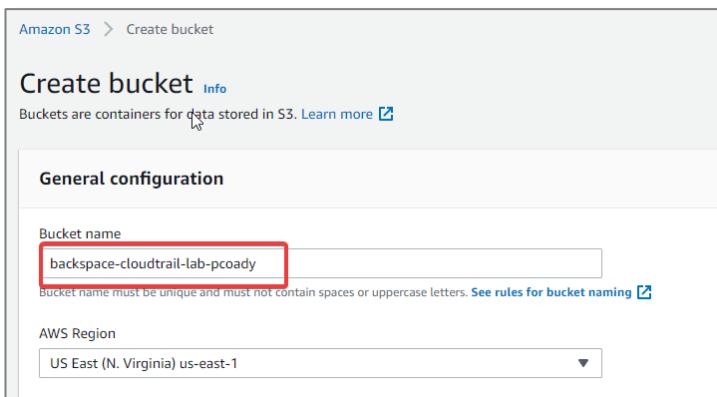


You will now see a bucket for storing the CloudTrail events has been created.

Click *Create bucket*

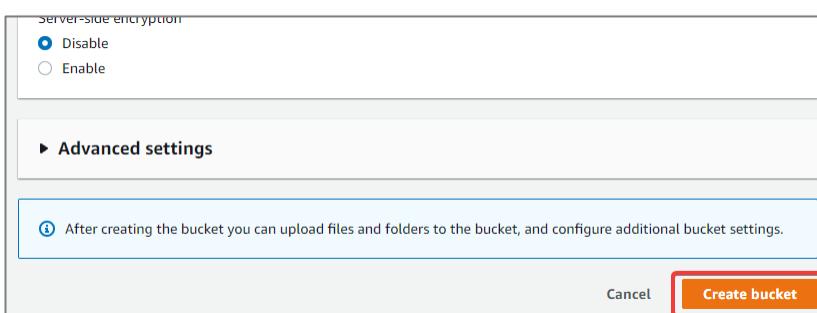


Give the bucket a unique name



Leave default settings

Click *Create bucket*



Now delete the bucket

Buckets (66) Info

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

[Copy ARN](#) [Empty](#) [Delete](#) [Create bucket](#)

Name	AWS Region
backspace-academy-media	US East (N. Virginia) us-east-1
backspace-academy-quicksight	US East (N. Virginia) us-east-1
aws-cloudtrail-lab-pcoady	US East (N. Virginia) us-east-1
	US East (N. Virginia) us-east-1

Now open the CloudTrail logs bucket

Buckets (65) Info

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

[Copy ARN](#) [Empty](#) [Delete](#) [Create bucket](#)

Name	AWS Region	Access	Creation date
aws-cloudtrail-logs-161762789278-bf23a295	US East (N. Virginia) us-east-1	Bucket and objects not public	October 12, 2021, 08:25:41 (UTC+11:00)
aws-sam-cli-managed-default-samclisourcebucket-y1a15f9203cy	US East (N. Virginia) us-east-1	Bucket and objects not public	June 29, 2020, 18:14:43 (UTC+10:00)

Navigate to find the logs in your region

Amazon S3 > aws-cloudtrail-logs-161762789278-bf23a295 [Info](#)

[Objects](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

[Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#)

[Create folder](#) [Upload](#)

[Find objects by prefix](#)

Name	Type	Last modified	Size	Storage class
AWSLogs/	Folder	-	-	-

Select the latest log file and click *Download*

Objects (5)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Actions ▾

2

Name	Type	Last modified	Size	Storage class
1_20211011T2130Z_z6TxnhT6gjwCgthX.json.gz	gz	2021-10-11T21:30:10 (UTC+11:00)	1.0 KB	Standard
161762789278_CloudTrail_us-east-1_20211011T2135Z_LGodrwTAd135HrQX.json.gz	gz	October 12, 2021, 08:34:07 (UTC+11:00)	21.7 KB	Standard
161762789278_CloudTrail_us-east-1_20211011T2140Z_Co9REMLhFu3Wrif.json.gz	gz	October 12, 2021, 08:39:48 (UTC+11:00)	6.1 KB	Standard
161762789278_CloudTrail_us-east-1_20211011T2140Z_eqK0ZOPjOK2j1K.json.gz	gz	October 12, 2021, 08:39:53 (UTC+11:00)	1.8 KB	Standard
161762789278_CloudTrail_us-east-1_20211011T2140Z_vcWQWln6c0dFyktr.json.gz	gz	October 12, 2021, 08:39:17 (UTC+11:00)	41.8 KB	Standard

Go to <https://jsoneditoronline.org>

Select *Code* on the left pane

Paste in the contents of the file

The screenshot shows the JSON Editor Online interface with two tabs open: "New document 1" and "New document 2".

New document 1: Contains a single JSON object with the following structure:

```
[{"Records": [{"eventVersion": "1.08", "userIdentity": {"type": "IAMUser", "principalId": "AIDAQMKUJR5QXMKET0Z2", "arn": "arn:aws:iam::161762789278:user/pcoady", "accountId": "161762789279", "accessKeyId": "ASIAISLKOQ1GPMQRK7YAK", "userName": "pcoady", "sessionContext": {"sessionIssuer": {}, "webIdFederationData": {}, "attributes": {"creationDate": "2021-10-11T18:52:28Z", "mfaAuthenticated": "true"}}, "eventTime": "2021-10-11T21:29:03Z"}]}
```

New document 2: Contains a single JSON object with the following structure:

```
{"Records": [{}]}
```

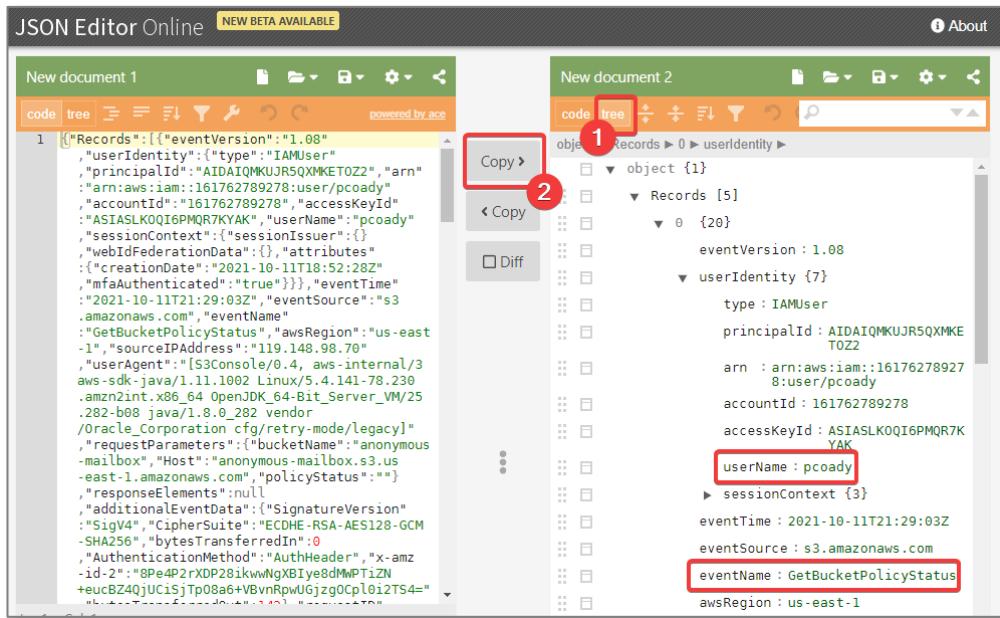
Both documents have standard file operations (New, Open, Save, etc.) and a toolbar labeled "powered by ace". The right panel shows a tree view of the selected node.

Select Tree on the right pane

Click *Copy*>

You will now be able to see the list of event records.

Expand one of the records to see the user details and the call made



Clean Up

Go back to the CloudTrail console

Delete the trail.

Name	Home region	Multi-region trail	Insights	Organization trail	S3 bucket	Log file prefix	CloudWatch Logs log group	Status
backspace-events	US East (N. Virginia)	Yes	Disabled	No	aws-cloudtrail-logs-161762789278-bf23a295		Logging	✓

Please note you will not be able to empty and delete the S3 bucket unless the CloudTrail trail has been deleted.

Go back to the S3 console

Empty the CloudTrail bucket

The screenshot shows the 'Amazon S3' service in the AWS console. The left sidebar has 'Buckets' selected (marked with a red circle labeled 1). The main area shows a table of buckets. The 'Empty' button in the top navigation bar is highlighted with a red box and a red circle labeled 3. A red circle labeled 2 highlights the blue circular icon next to the bucket name 'aws-cloudtrail-logs-161762789278-bf23a295'.

Name	AWS Region	Access	Created
aprc-s3-dev	Virginia) us-east-1	Objects can be public	19:59 (UTC+0)
audio-for-wordpress-703870264611a6e3623e0a87c5c90e8ae7d4d54c	US East (N. Virginia) us-east-1	Objects can be public	October 2021, (UTC+0)
aws-cloudtrail-logs-161762789278-bf23a295	US East (N. Virginia) us-east-1	Bucket and objects not public	October 2021, (UTC+0)

Delete the empty CloudTrail bucket

This screenshot is identical to the one above, but the 'Delete' button in the top navigation bar is highlighted with a red box and a red circle labeled 3. A red circle labeled 2 highlights the blue circular icon next to the bucket name 'aws-cloudtrail-logs-161762789278-bf23a295'.

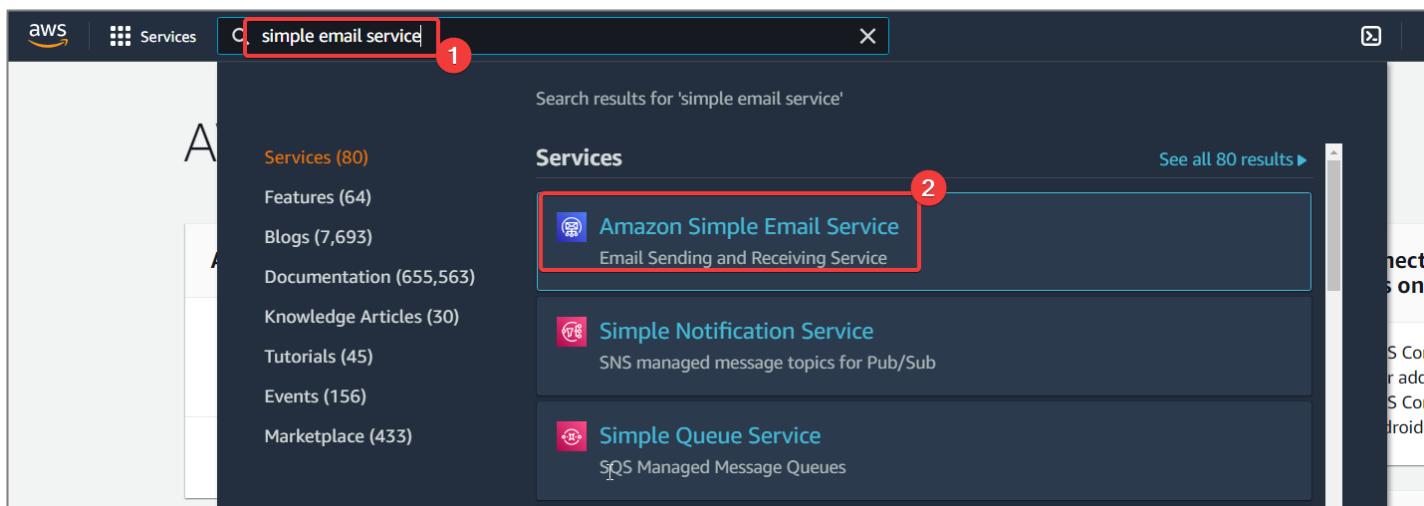
Name	AWS Region	Access	Created
audio-for-wordpress-703870264611a6e3623e0a87c5c90e8ae7d4d54c	US East (N. Virginia) us-east-1	Objects can be public	October 2021, (UTC+0)
aws-cloudtrail-logs-161762789278-bf23a295	US East (N. Virginia) us-east-1	Bucket and objects not public	October 2021, (UTC+0)

▶ 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 *Create Identity*



Select email address and type-in your Email Address

Identity details Info

Identity type

Domain
To verify ownership of a domain, you must have access to its DNS settings to add the necessary records.

Email address
To verify ownership of an email address, you must have access to its inbox to open the verification email.

2 Email address
@gmail.com

Email address can contain up to 320 characters, including plus signs (+), equals signs (=) and underscores (_).

Assign a default configuration set
Enabling this option ensures that the assigned configuration set is applied to messages sent from this identity by default whenever a configuration set isn't specified at the time of sending.

Click *Create Identity*

Tags - optional info
You can add one or more tags to help manage and organize your resources, including identities.

No tags associated with the resource.

Add new tag
You can add 50 more tags.

Cancel **Create identity**

You need to verify it from your entered email address

Amazon SES

Action required
To verify ownership of this identity, check your inbox for a verification request email and click the link provided.

Account dashboard Reputation metrics

Configuration

Verified identities

- Configuration sets
- Dedicated IPs
- Email templates
- Suppression list New
- Cross-account notifications New
- Email receiving

Use the classic console

Amazon SES > Configuration: Verified identities @gmail.com

@gmail.com

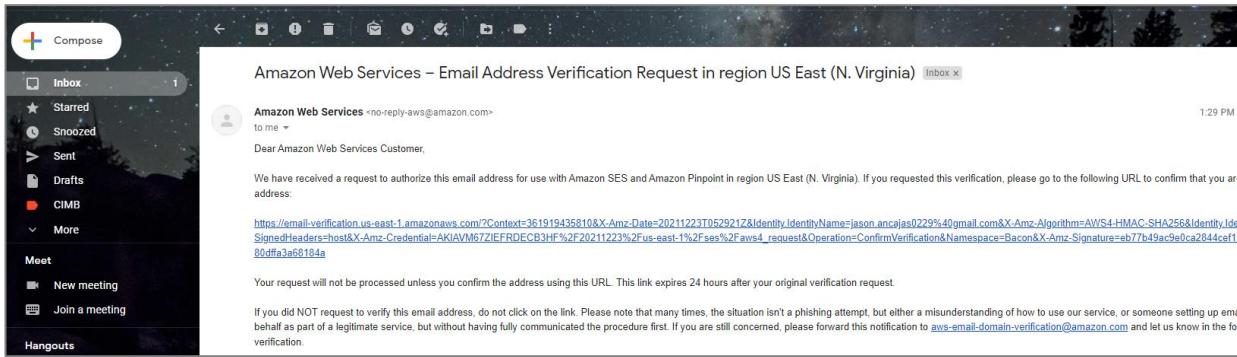
Delete Send test email

Legacy TXT records
Domain verification in Amazon SES is now based on *DomainKeys Identified Mail (DKIM)*, an email authentication standard that receiving mail servers use to validate an email's authenticity. Configuring DKIM in your domain's DNS settings confirms to SES that you're the identity owner, eliminating the need for TXT records. Domain identities that were verified using TXT records do not need to be reverified; however, we still recommend enabling DKIM signatures to enhance the deliverability of your mail with DKIM-compliant email providers. To access your legacy TXT records, download the record set as a .csv [\[CSV\]](#).

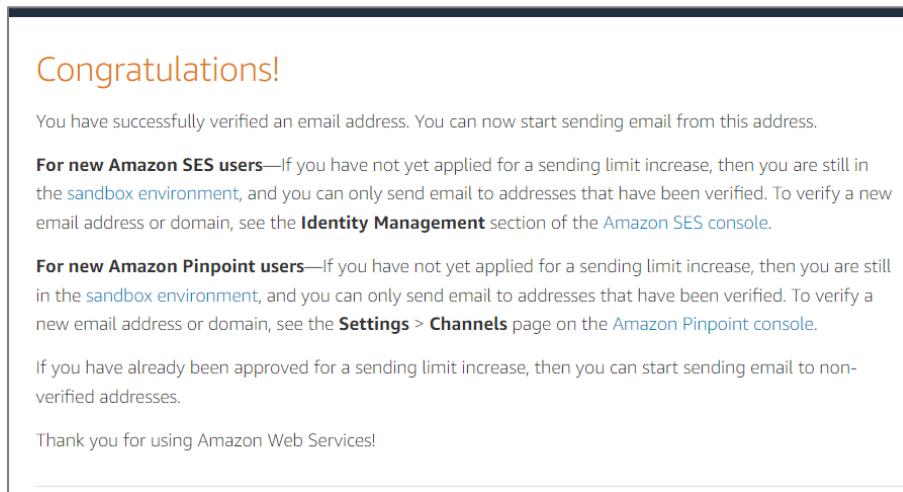
Summary for @gmail.com

Identity status	Amazon Resource Name (ARN)	AWS Region
<input checked="" type="radio"/> Unverified	arn:aws:ses:us-east-1:36191943	US East (N. Virginia)

Check your email



You will now receive a verification status.



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

Send a test email

The screenshot shows the Amazon SES Configuration: Verified identities page. On the left, there's a sidebar with 'Amazon SES' and a list of options like 'Account dashboard', 'Reputation metrics', 'Configuration', 'Verified identities' (which is selected and highlighted in blue), 'Configuration sets', 'Dedicated IPs', 'Email templates', 'Suppression list', 'Cross-account notifications', 'Email receiving', and 'Use the classic console'. The main content area shows a summary for the identity '@gmail.com'. It includes a section about Legacy TXT records, a summary table with 'Identity status' (Verified), 'Amazon Resource Name (ARN)' (arn:aws:ses:us-east-1:361919435810:identity/...@g mail.com), and 'AWS Region' (US East (N. Virginia)). At the top right, there are 'Delete' and 'Send test email' buttons, with 'Send test email' being highlighted with a red box.

Select *Formatted*

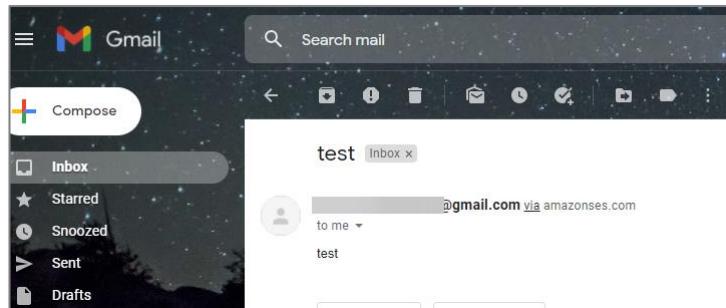
Select *Custom* for the Scenario

Enter the email address *recipient*

Click on *Send test email*

The screenshot shows the 'Send test email' dialog box. Step 1 highlights the 'Email format' section with the 'Formatted' option selected. Step 2 highlights the 'Scenario Info' dropdown set to 'Custom'. Step 3 highlights the 'Custom recipient' input field containing '@gmail.com'. Step 4 highlights the 'Subject' input field with 'test' typed in. Step 5 highlights the 'Body - optional' input field with 'test' typed in. Step 6 highlights the 'Send test email' button at the bottom right. Other visible fields include 'From-address' (@gmail.com), 'Configuration set - optional info' (Choose a configuration set dropdown), and 'Additional configurations - optional' (dropdown).

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 *Configuration*

Click on *Request Production access*

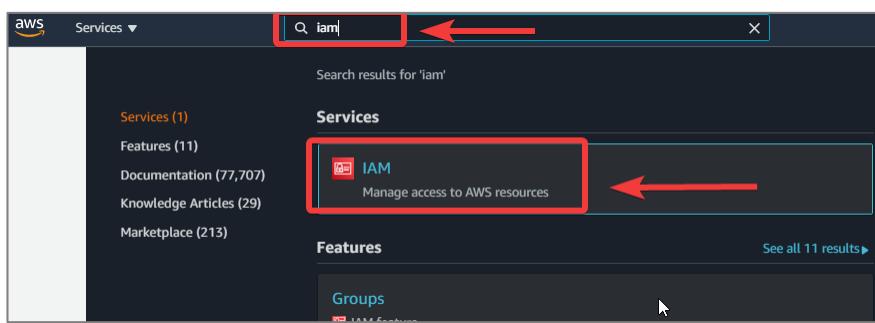
The screenshot shows the Amazon SES Account dashboard. On the left sidebar, under the 'Configuration' section, there is a red box with the number '1'. In the main content area, there is a warning message: 'Your Amazon SES account is in the sandbox in US East (N. Virginia). In a sandbox environment, you can use all of the features offered by Amazon SES; however, certain sending limits and restrictions apply. When you're ready to move out of the sandbox, submit a request for production access.' Below this message, there is a red box with the number '2' containing a button labeled 'Request production access'.

▶ Creating an IAM User

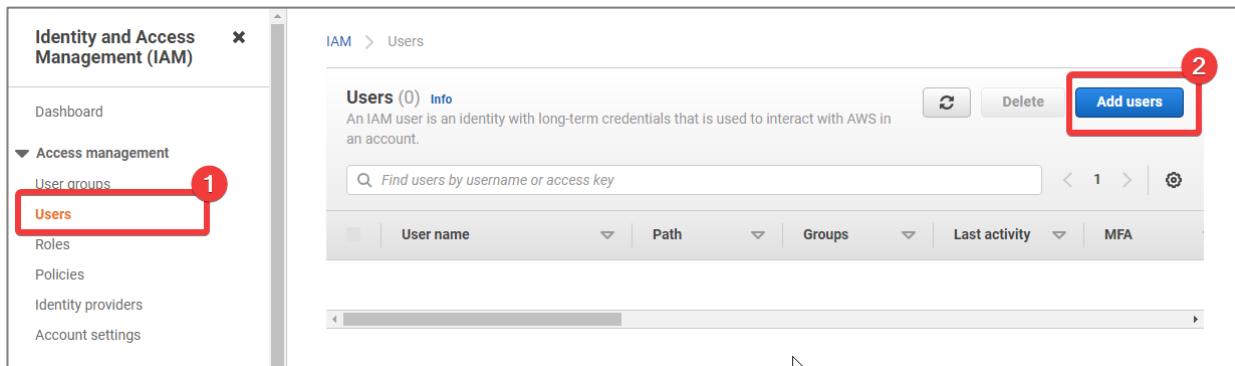
In this section, we will use the Identity and Access Management (IAM) service to create a user with console access and programmatic access.

From the AWS console click search IAM

Select IAM



Select Users -> Add user



Give the user a name

A screenshot of the 'Add user' wizard, step 1: Set user details. The form has a header 'Add user' with a progress bar showing step 1 of 5. The 'User name*' field is filled with 'test-user' and is highlighted with a red box and a red arrow. Below the field is a link '+ Add another user'.

Check Programmatic access

Check AWS Management Console access

Click Next: Permissions

Select AWS access type

Select how these users will access the AWS API.

Access type*

- 1 **Programmatic access**
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.
- 2 **AWS Management Console access**
Creates 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

We won't set any tags.

Add user

1 2 3 4 5

Set permissions

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

i Get started with groups
You haven't created any groups yet. Using groups is a best-practice way to manage users' permissions by job functions, AWS service access, or your custom permissions. Get started by creating a group. [Learn more](#)

Create group

Click Next Review

Cancel Previous **Next: Review**

Click *Create user*

Add user

Review

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

User details

User name	test-user
AWS access type	Programmatic access and AWS Management Console access
Console password type	Autogenerated
Require password reset	Yes
Permissions boundary	Permissions boundary is not set

Permissions summary

The user shown above will be added to the following groups.

Type	Name
Managed policy	IAMUserChangePassword

Cancel Previous **Create user**

You would normally download the csv file containing the user credentials (access key and secret access key) to a safe location.

Add user

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://991610390270.signin.aws.amazon.com/console>

Download .csv

	User	Access key ID	Secret access key	Password	Email login instructions
▶	test-user	AKIA6NYEVY37BLPSRDVT	***** Show	***** Show	Send email

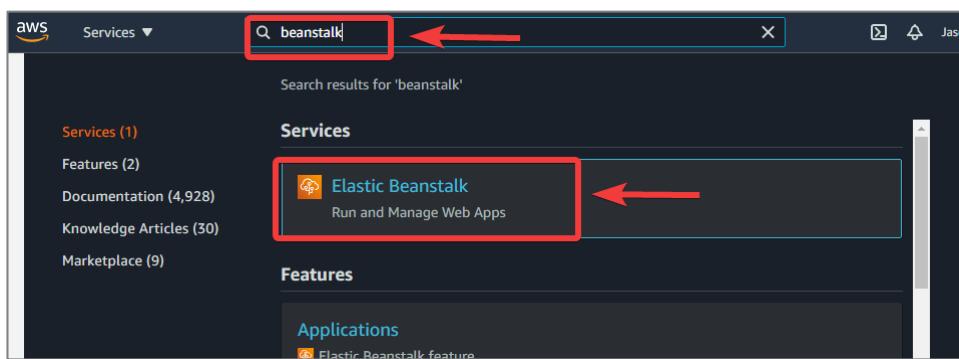
Clean Up

Go back to the *Users* dashboard and delete the user

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*.

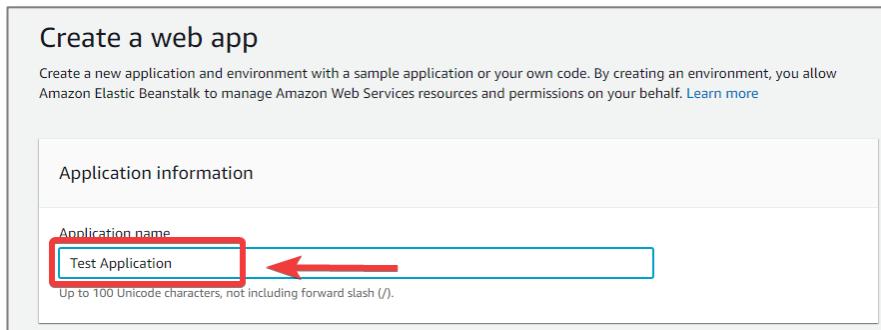
Create a web app

Create a new application and environment with a sample application or your own code. By creating an environment, you allow Amazon Elastic Beanstalk to manage Amazon Web Services 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 14 running on 64bit Amazon Linux 2

Platform version
5.4.3 (Recommended)



Scroll down to *Application code*

Select *Sample application*

Click *Configure more options*

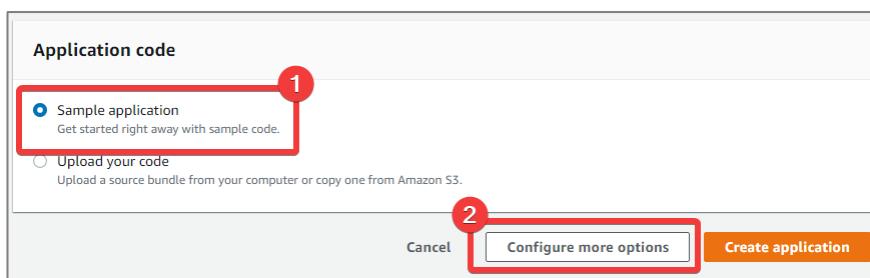
Application code

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  Create application



Select High availability

Elastic Beanstalk > Getting started

Configure Testapplication-env

Presets

Start from a preset that matches your use case or choose *Custom configuration* to unset recommended values and use the service's default values.

Configuration presets

- Single instance (*Free Tier eligible*)
- Single instance (using Spot instance)
- High availability ←
- High availability (using Spot and On-Demand instances)
- Custom configuration

Scroll down and click *Create app*

Monitoring

Health reporting system: Enhanced
Ignore application 4xx: disabled
Ignore load balancer 4xx: disabled
Health event log streaming: disabled

Managed Updates

Managed updates: enabled
Weekly update window: Wed:03:00 UTC

Notifications

Email address: --

Network

This environment is not part of a VPC.

Database

Engine: --
Instance class: --
Storage (GB): --
Multi-AZ: --

Tags

Tags: *none*

Cancel Previous **Create app** ↓

Your application will now start being created.
After some time, your environment will be created.

Elastic Beanstalk > Environments > Testapplication-env

Creating Testapplication-env
This will take a few minutes. ...

```

1:47pm Environment health has transitioned to Pending. Initialization in progress (running for 22 seconds). There are no instances.
1:47pm Created security group named:
sg-050e0d5ca70e4b3bc
1:47pm Created target group named:
arn:aws:elasticloadbalancing:us-east-1:991610390270:targetgroup/awseb-AWSEB-15H2H6C1JCINH/f45b9bc2dc20e36e
1:47pm Using elasticbeanstalk-us-east-1-991610390270 as Amazon S3 storage bucket for environment data.
1:47pm createEnvironment is starting.

```

Click on the website URL

The screenshot shows the AWS Elastic Beanstalk console for the 'Testapplication-env' environment. At the top, the application name is displayed. Below it, there are three main sections: 'Health' (green checkmark, 'Ok'), 'Running version' (Sample Application, 'Upload and deploy' button), and 'Platform' (Node.js 14 running on 64bit Amazon Linux 2/5.4.3, 'Change' button). The URL 'Testapplication-env.eba-k9xu2mb.us-east-1.elasticbeanstalk.com' is highlighted with a red box and has a red arrow pointing to it.

You will now see the Sample Application

The screenshot shows the 'Sample Application' page. On the left, a green panel displays 'Congratulations' and a message: 'Your first AWS Elastic Beanstalk Node.js application is now running on your own dedicated environment in the AWS Cloud'. Below this, it says 'This environment is launched with Elastic Beanstalk Node.js Platform'. On the right, a dark sidebar titled 'What's Next?' contains a list of links:

- [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

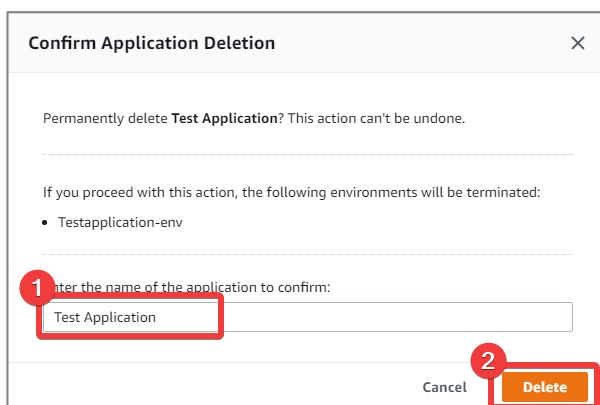
We will now delete the environment so that you will not be billed by AWS.

Click on *Applications*

Select the application

Select *Actions -> Delete application*

The screenshot shows the AWS Elastic Beanstalk Applications interface. On the left, a sidebar has 'Applications' selected (marked with a red box and number 1). The main area shows a table of applications. A specific row for 'Test Application' (marked with a red box and number 2) is selected. In the top right corner, a context menu is open over the row, with 'Delete application' highlighted (marked with a red box and number 4). The menu also includes options like 'Create environment', 'View application versions', 'View saved configurations', and 'Restore terminated environment'.



Click on the *Environments*

The screenshot shows the AWS Elastic Beanstalk Applications interface again. The sidebar has 'Environments' selected (marked with a red box and number 1). The main table shows the 'Test Application' row. An arrow points from the 'Environments' column header (marked with a red arrow) to the 'Testapplication-env' environment name in the row (marked with a red box and number 2). The row is highlighted with a red box.

You will now see your environment is being terminated.

The screenshot shows the AWS Elastic Beanstalk console for the environment 'Testapplication-env'. A prominent message at the top left states 'Elastic Beanstalk is terminating your environment.' with a 'View Events' link. The main content area displays the environment details:

- Health:** Pending (indicated by a circular arrow icon)
- Running version:** Sample Application
- Platform:** Node.js 14 running on 64bit Amazon Linux 2/5.4.3

Actions available include 'Upload and deploy' (button), 'Refresh' (button), and 'Actions ▾' (dropdown menu). A 'Causes' button is also present under the Health section.