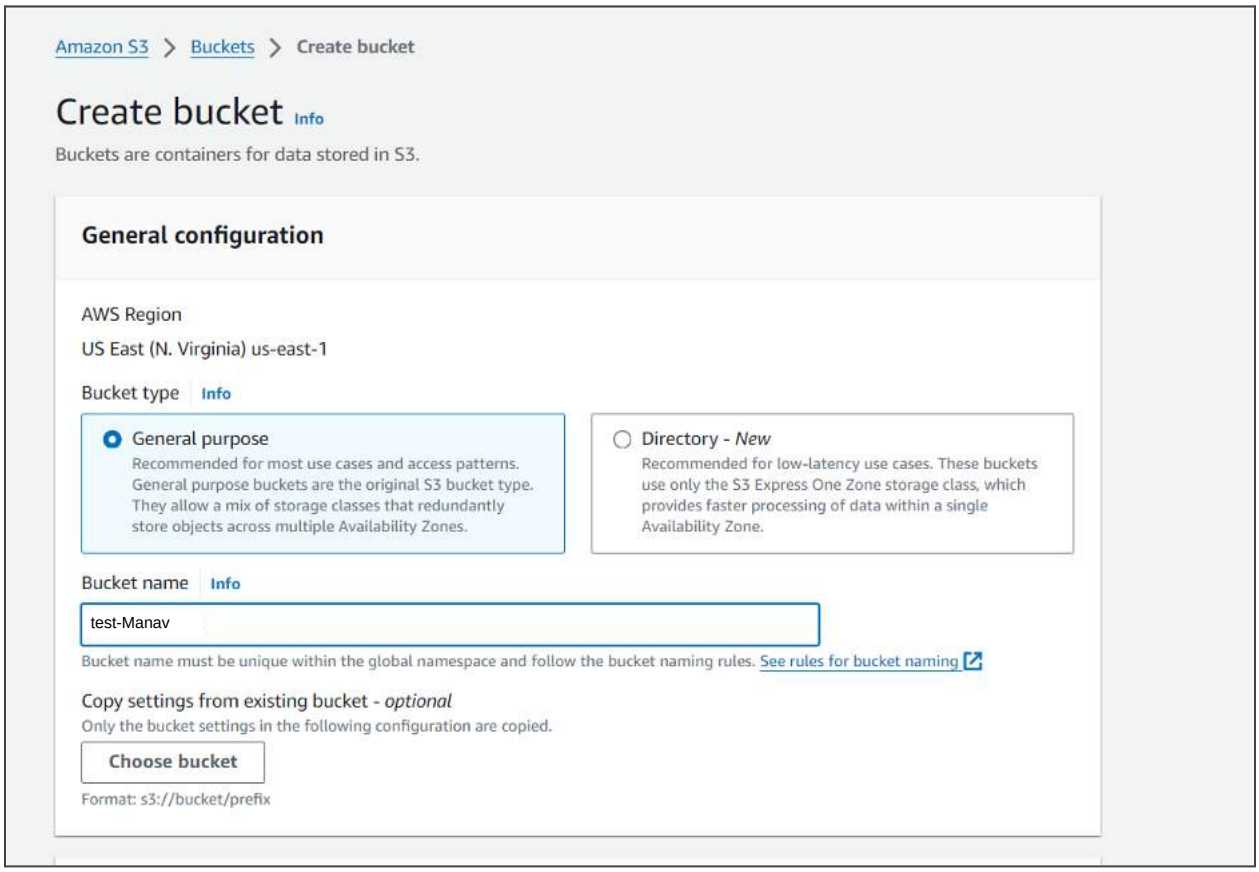
Experiment 2

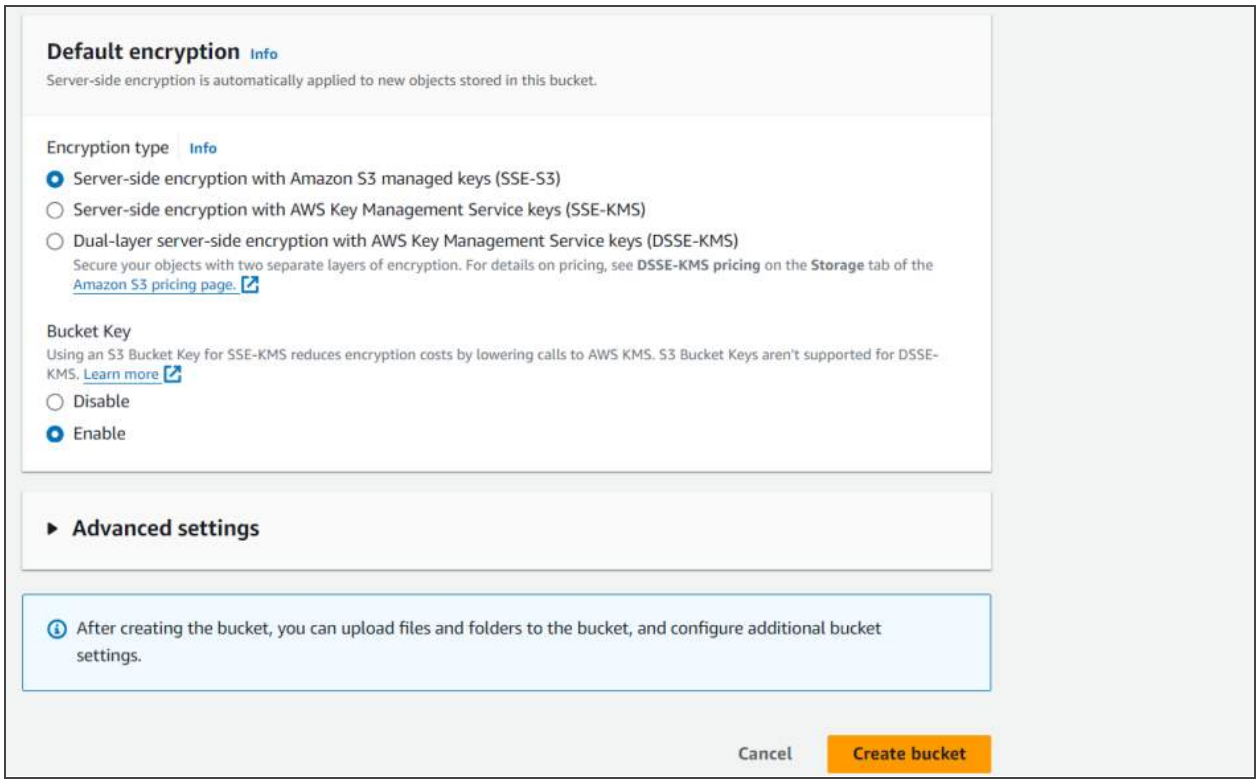
Aim: To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.

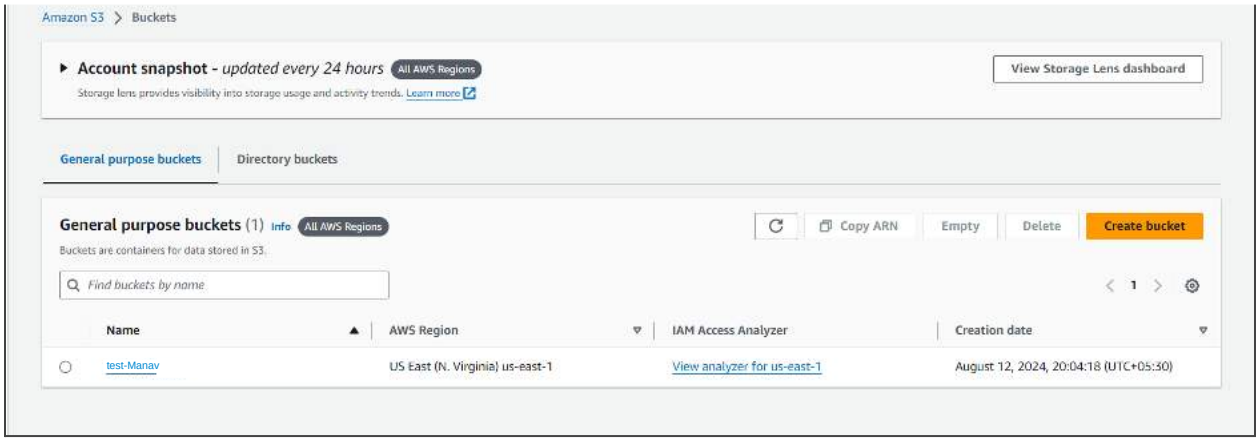
Contents:

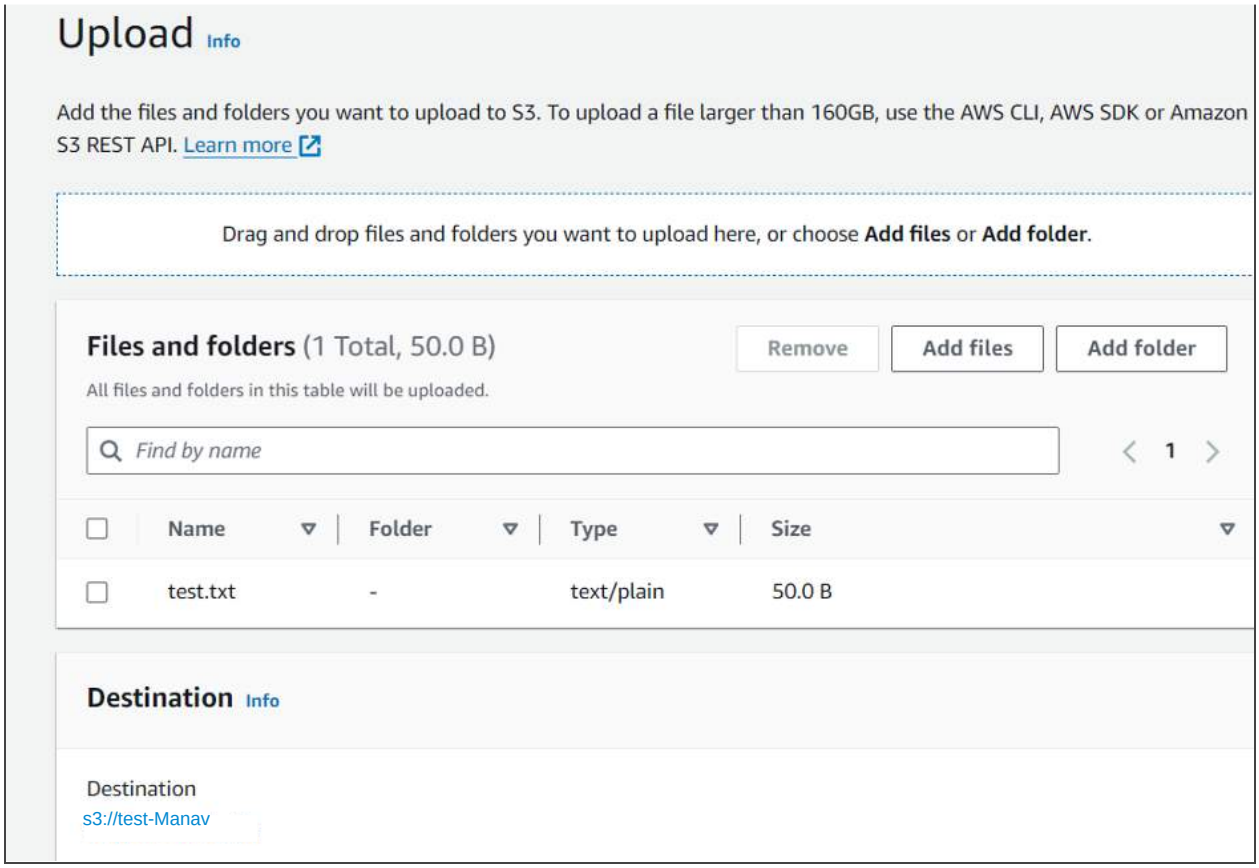
1. s3 bucket
2. ec2 instance
3. elastic beanstalk

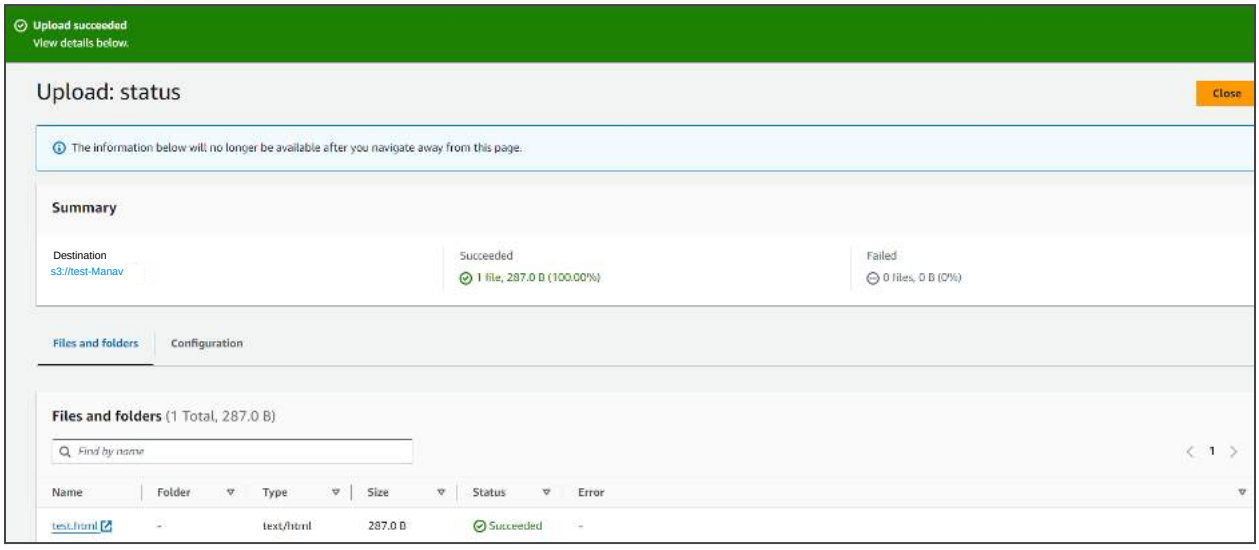
- s3 bucket

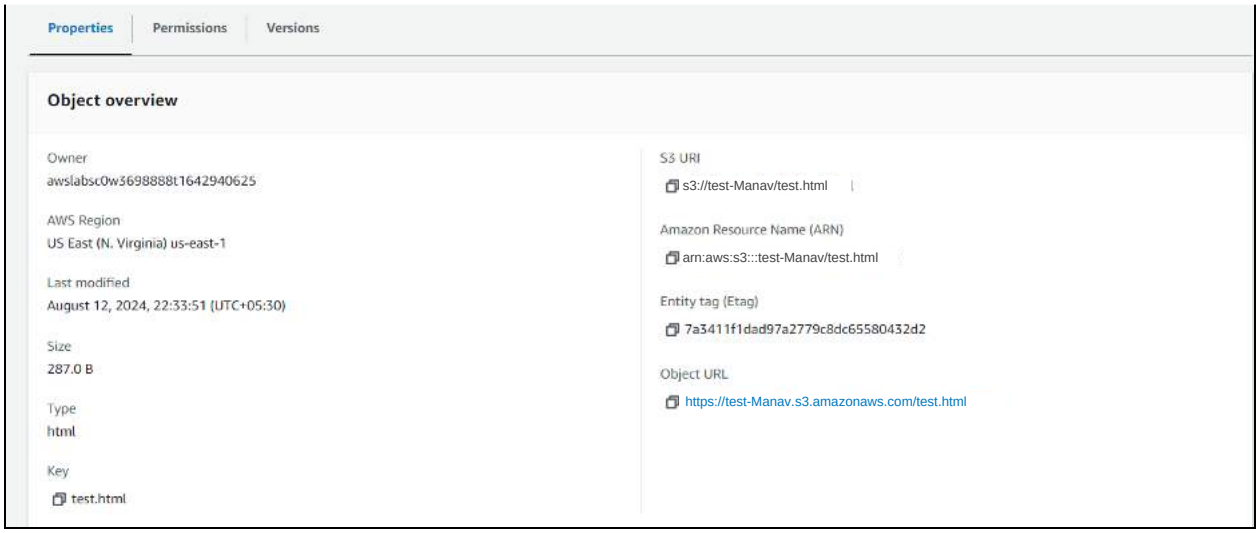


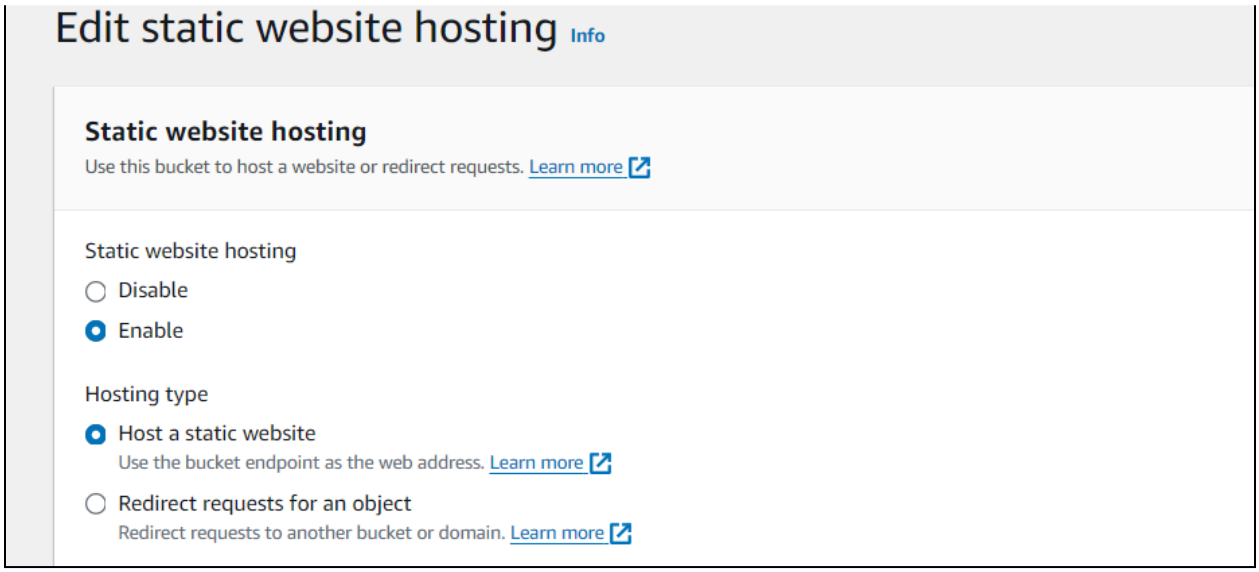

Default encryption Info
Server-side encryption is automatically applied to new objects stored in this bucket.
Encryption type Info
Server-side encryption with Amazon S3 managed keys (SSE-S3)
Server-side encryption with AWS Key Management Service keys (SSE-KMS)
Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
Secure your objects with two separate layers of encryption. For details on pricing, see DSSE-KMS pricing on the Storage tab of the Amazon S3 pricing page.
Bucket Key
Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. Learn more
Disable
Enable
Advanced settings
After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.
Cancel
Create bucket

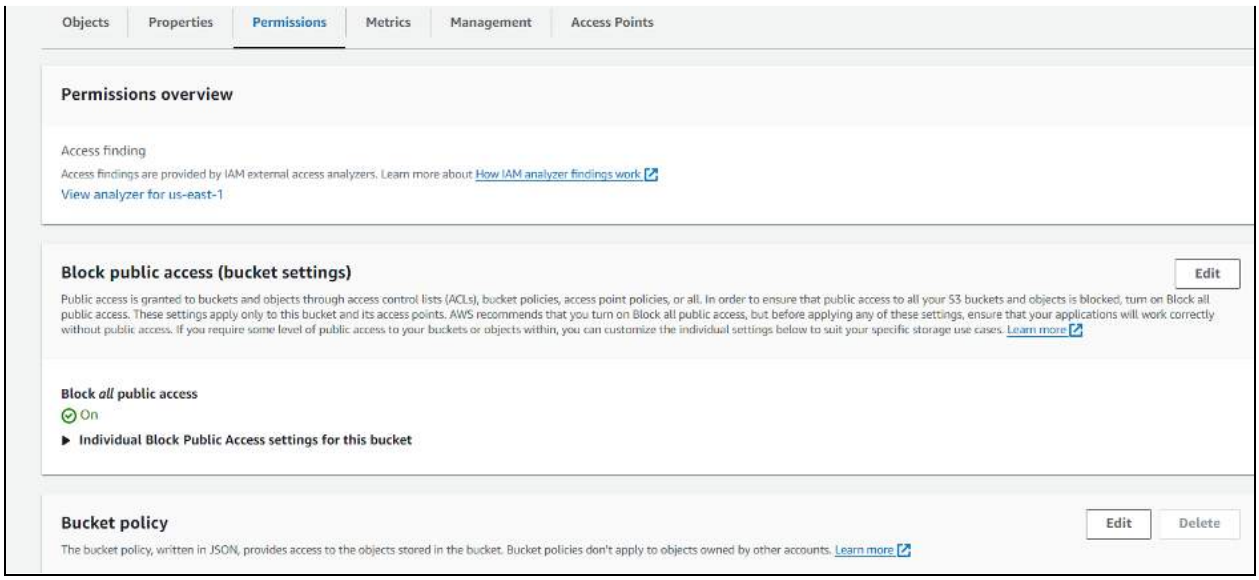

Amazon S3 > Buckets
Account snapshot - updated every 24 hours All AWS Regions
Storage lens provides visibility into storage usage and activity trends. Learn more
View Storage Lens dashboard
General purpose buckets
Directory buckets
General purpose buckets (1) Info All AWS Regions
Buckets are containers for data stored in S3.
Copy ARN
Empty
Delete
Create bucket
Find buckets by name
Name
AWS Region
IAM Access Analyzer
Creation date
test-Manav
US East (N. Virginia) us-east-1
View analyzer for us-east-1
August 12, 2024, 20:04:18 (UTC+05:30)


Upload Info
Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more
Drag and drop files and folders you want to upload here, or choose Add files or Add folder.
Files and folders (1 Total, 50.0 B)
Remove
Add files
Add folder
All files and folders in this table will be uploaded.
Find by name
1
Name
Folder
Type
Size
test.txt
-
text/plain
50.0 B
Destination Info
Destination
s3://test-Manav


Upload succeeded
View details below.
Upload: status
Close
The information below will no longer be available after you navigate away from this page.
Summary
Destination
s3://test-Manav
Succeeded
1 file, 287.0 B (100.00%)
Failed
0 files, 0 B (0%)
Files and folders
Configuration
Files and folders (1 Total, 287.0 B)
Find by name
1
Name
Folder
Type
Size
Status
Error
test.html
-
text/html
287.0 B
Succeeded
-


Properties
Permissions
Versions
Object overview
Owner
awslabsc0w3698888t1642940625
AWS Region
US East (N. Virginia) us-east-1
Last modified
August 12, 2024, 22:33:51 (UTC+05:30)
Size
287.0 B
Type
html
Key
test.html
S3 URI
s3://test-Manav/test.html
Amazon Resource Name (ARN)
arn:aws:s3:::test-Manav/test.html
Entity tag (Etag)
7a3411f1dad97a2779c8dc65580432d2
Object URL
https://test-Manav.s3.amazonaws.com/test.html

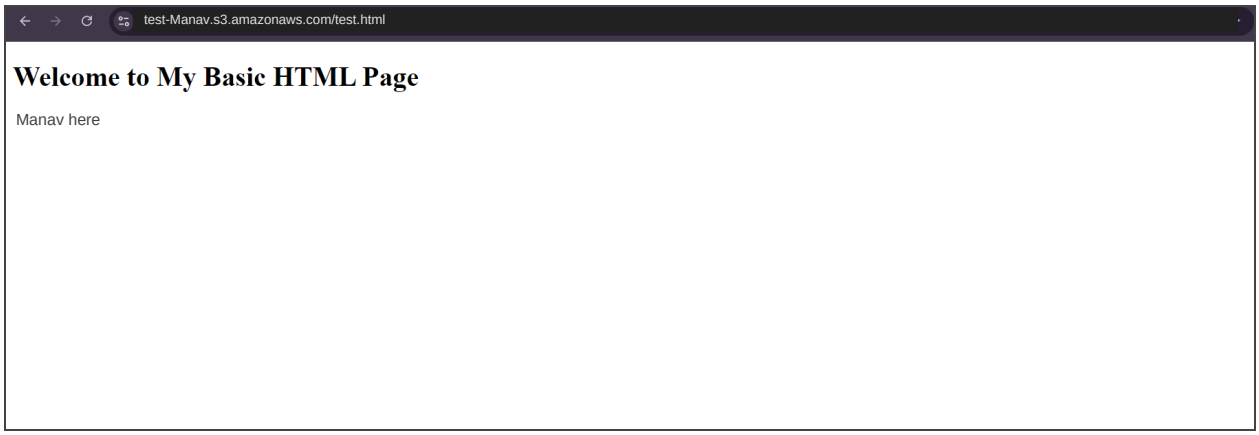

Edit static website hosting Info
Static website hosting
Use this bucket to host a website or redirect requests. Learn more
Static website hosting
Disable
Enable
Hosting type
Host a static website
Use the bucket endpoint as the web address. Learn more
Redirect requests for an object
Redirect requests to another bucket or domain. Learn more



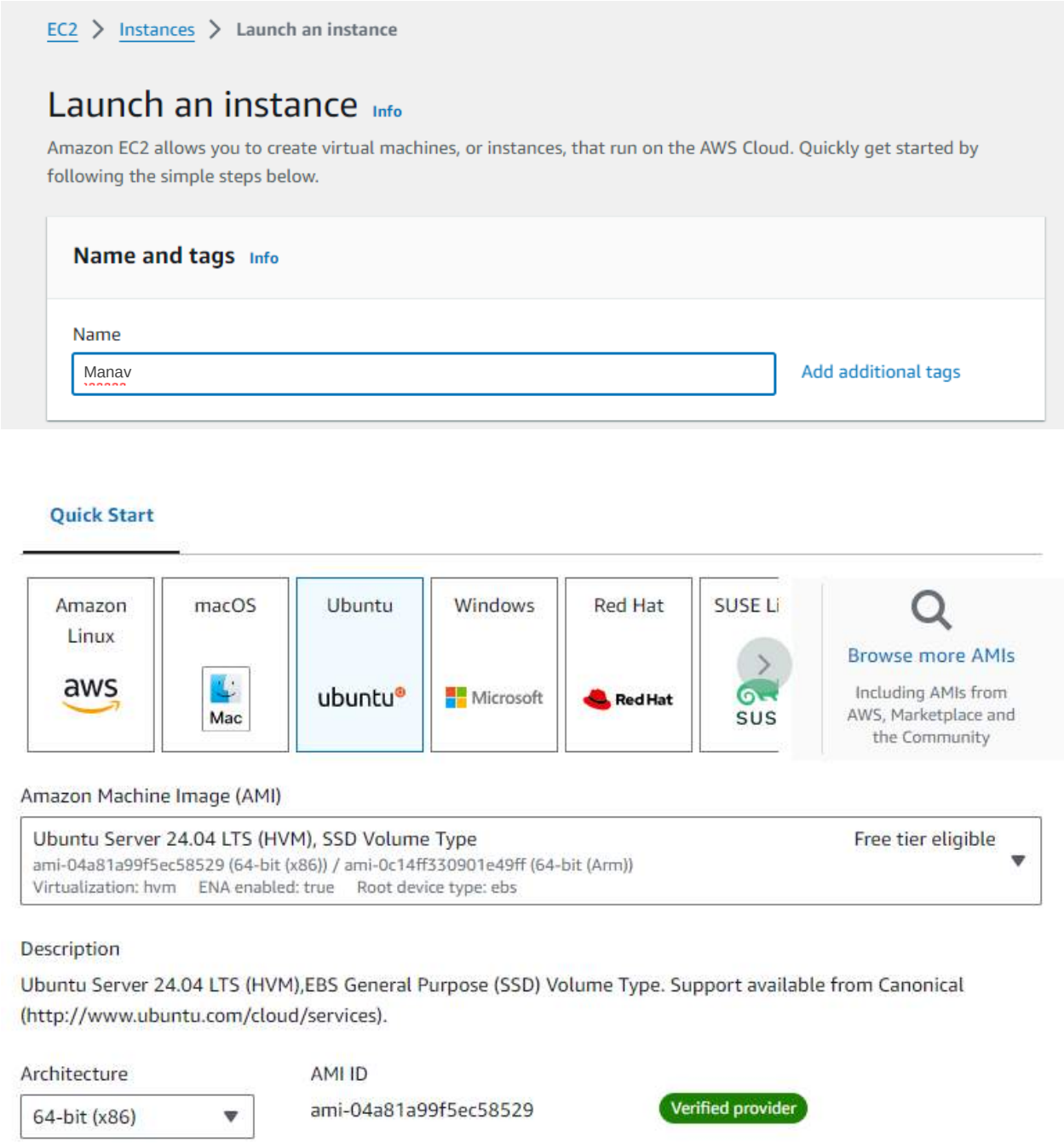
Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

```
{
  "Version": "2012-10-17",
 "Statement": [
   {
      "Sid": "PublicReadGetObject",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::test-Manav/*"
   }
 ]
}
```


test-Manav.s3.amazonaws.com/test.html
Welcome to My Basic HTML Page
Manav here

- Launching an EC2 instance



Instance type Info | Get advice

Instance type

t2.micro — Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

Configure storage Info

Advanced

1x 8 GiB gp3 Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

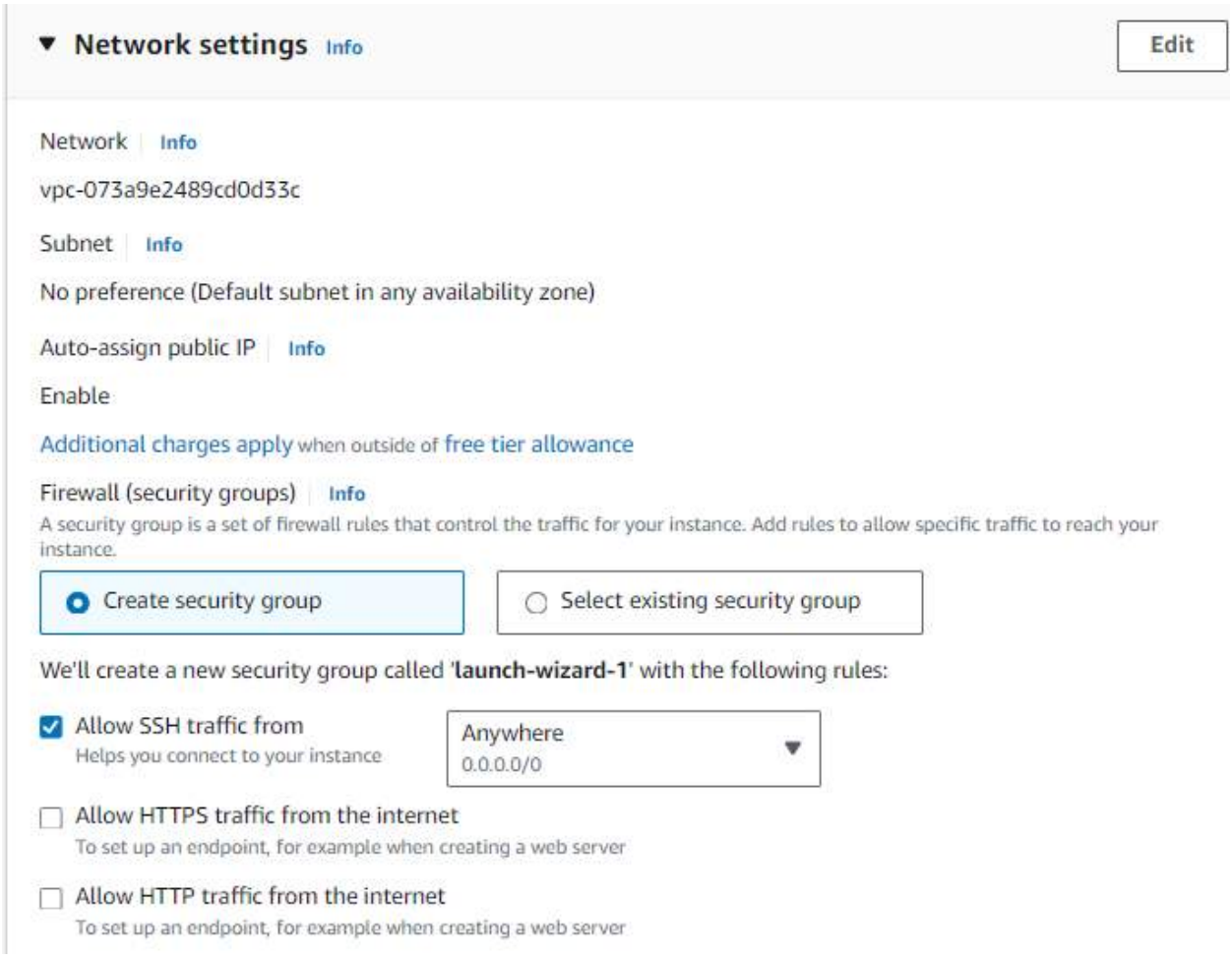
Add new volume

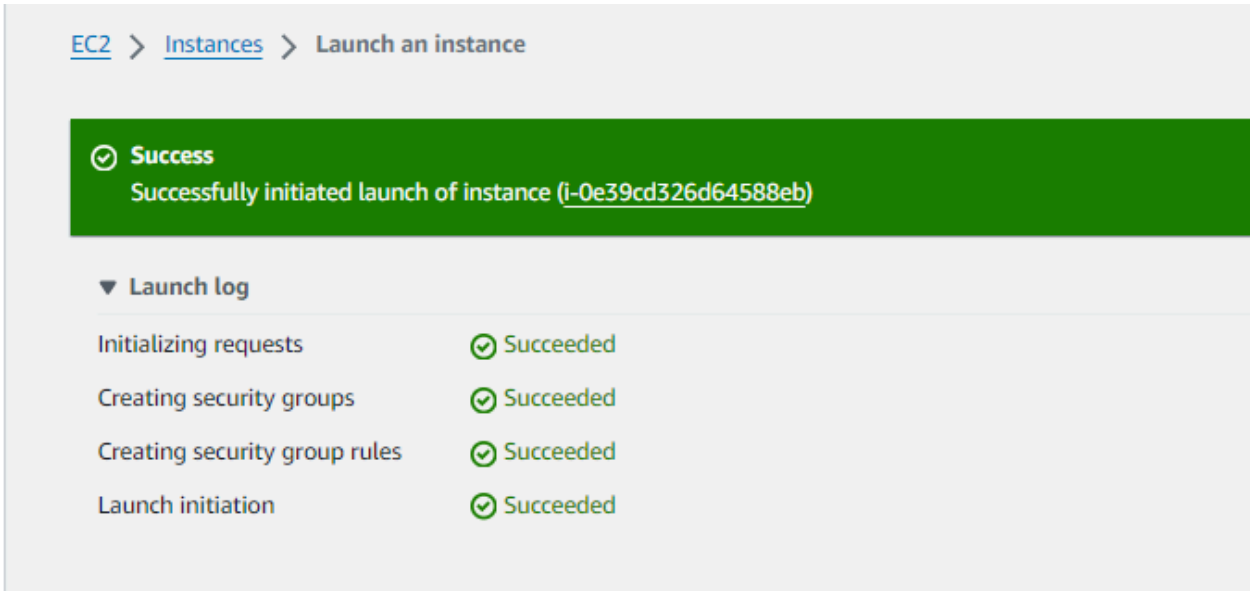
Click refresh to view backup information

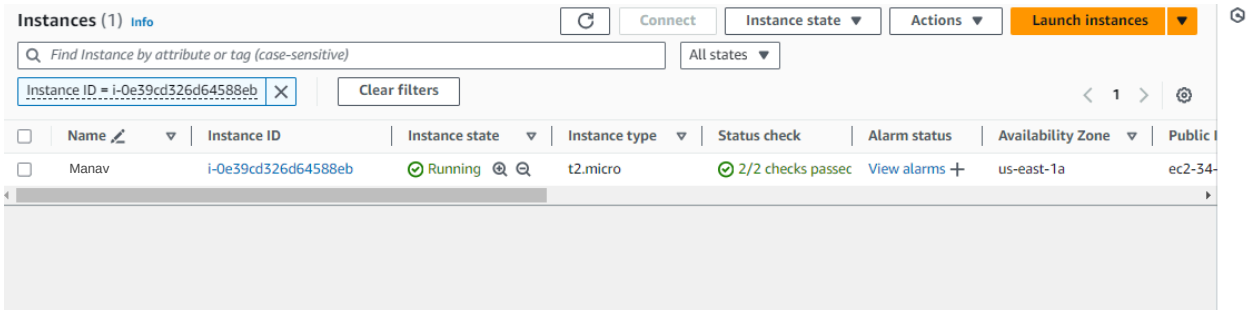
The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

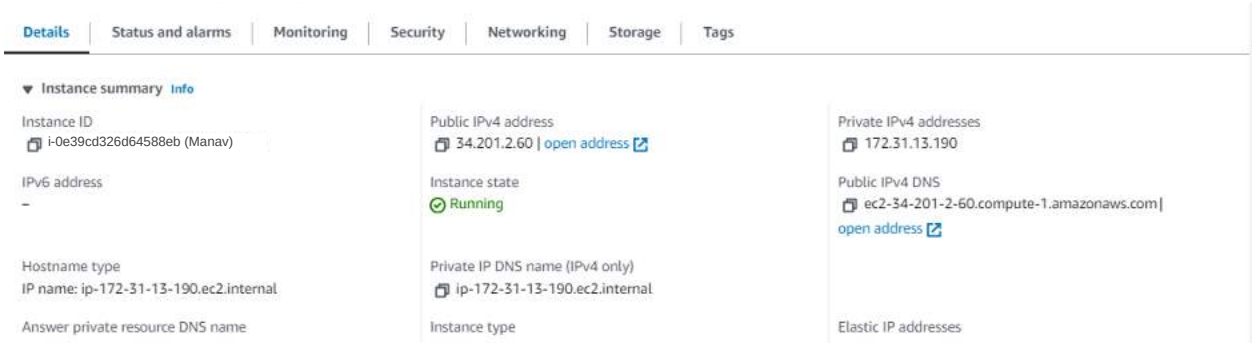
0 x File systems

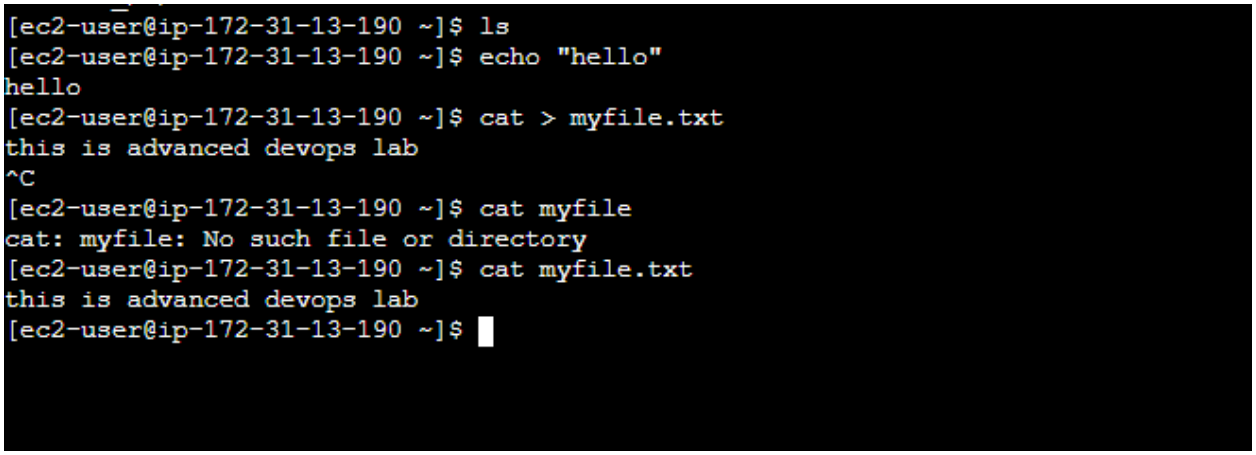
Edit


Network settings Info
Edit
Network Info
vpc-073a9e2489cd0d33c
Subnet Info
No preference (Default subnet in any availability zone)
Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance
Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
Create security group
Select existing security group
We'll create a new security group called 'launch-wizard-1' with the following rules:
Allow SSH traffic from
Helps you connect to your instance
Anywhere
0.0.0.0/0
Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server
Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server


EC2 > Instances > Launch an instance
Success
Successfully initiated launch of instance (i-0e39cd326d64588eb)
Launch log
Initializing requests
Succeeded
Creating security groups
Succeeded
Creating security group rules
Succeeded
Launch initiation
Succeeded


Instances (1) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Instance ID = i-0e39cd326d64588eb
Clear filters
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public I
Manav
i-0e39cd326d64588eb
Running
t2.micro
2/2 checks passec
View alarms
us-east-1a
ec2-34-

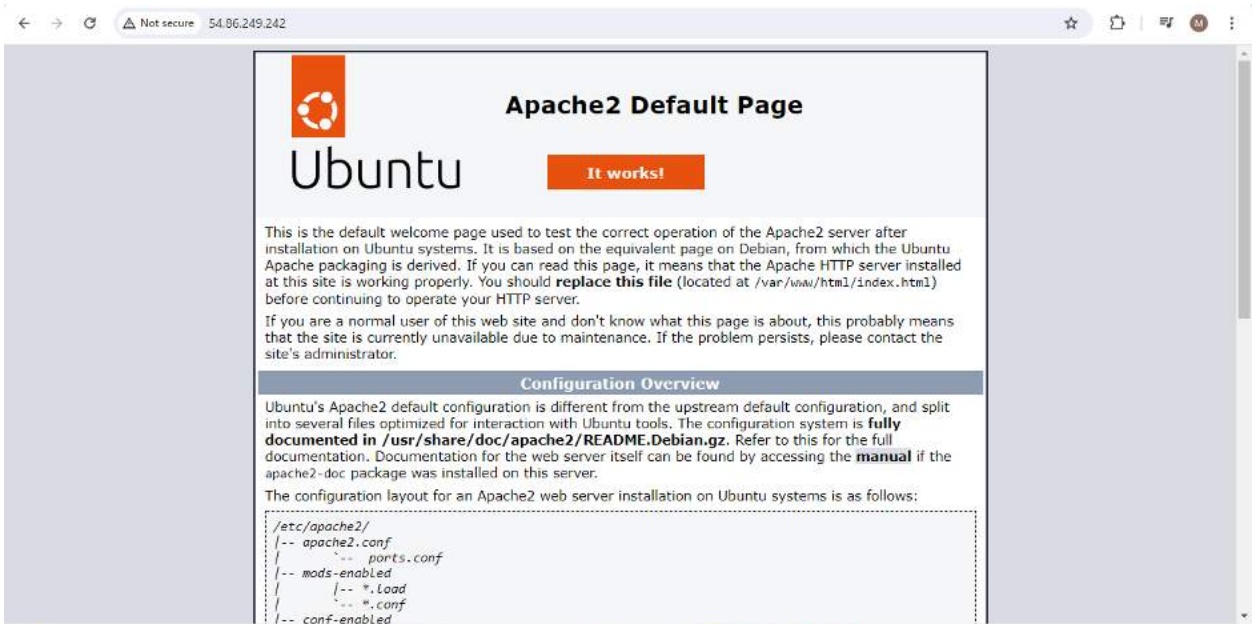

Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-0e39cd326d64588eb (Manav)
Public IPv4 address
34.201.2.60 | open address
Private IPv4 addresses
172.31.13.190
IPv6 address
-
Instance state
Running
Public IPv4 DNS
ec2-34-201-2-60.compute-1.amazonaws.com |
open address
Hostname type
IP name: ip-172-31-13-190.ec2.internal
Private IP DNS name (IPv4 only)
ip-172-31-13-190.ec2.internal
Answer private resource DNS name
Instance type
Elastic IP addresses


[ec2-user@ip-172-31-13-190 ~]$ ls
[ec2-user@ip-172-31-13-190 ~]$ echo "hello"
hello
[ec2-user@ip-172-31-13-190 ~]$ cat > myfile.txt
this is advanced devops lab
^C
[ec2-user@ip-172-31-13-190 ~]$ cat myfile
cat: myfile: No such file or directory
[ec2-user@ip-172-31-13-190 ~]$ cat myfile.txt
this is advanced devops lab
[ec2-user@ip-172-31-13-190 ~]$

```
root@ip-172-31-32-173:~# sudo su
root@ip-172-31-32-173:~# apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sql
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-brows
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1
0 upgraded, 10 newly installed, 0 to remove and 26 not upgraded.
Need to get 1680 kB/2083 kB of archives.
After this operation, 8094 kB of additional disk space will be used.
```

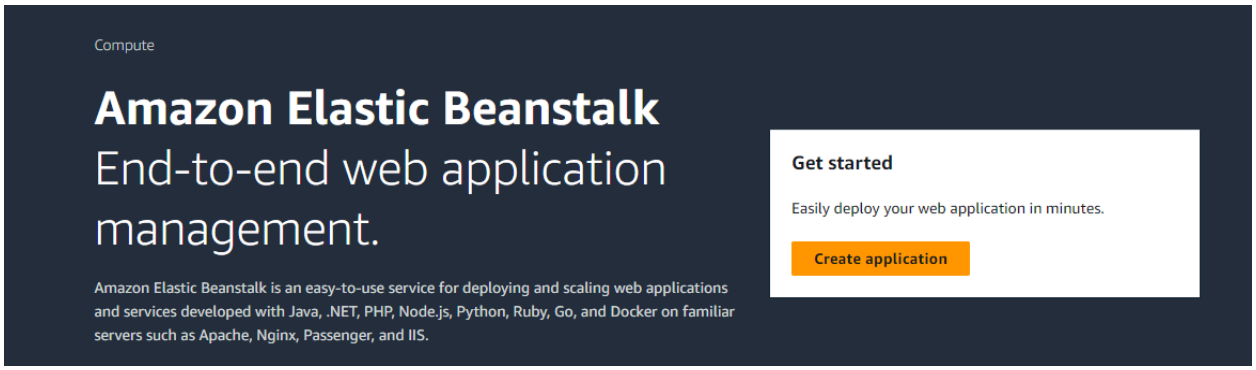
```
root@ip-172-31-32-173:~# systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Tue 2024-07-30 08:58:11 UTC; 44s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2619 (apache2)
      Tasks: 55 (limit: 1130)
     Memory: 5.4M (peak: 5.5M)
        CPU: 40ms
     CGroup: /system.slice/apache2.service
             ├─2619 /usr/sbin/apache2 -k start
             ├─2621 /usr/sbin/apache2 -k start
             └─2623 /usr/sbin/apache2 -k start

Jul 30 08:58:11 ip-172-31-32-173 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jul 30 08:58:11 ip-172-31-32-173 systemd[1]: Started apache2.service - The Apache HTTP Server.
```

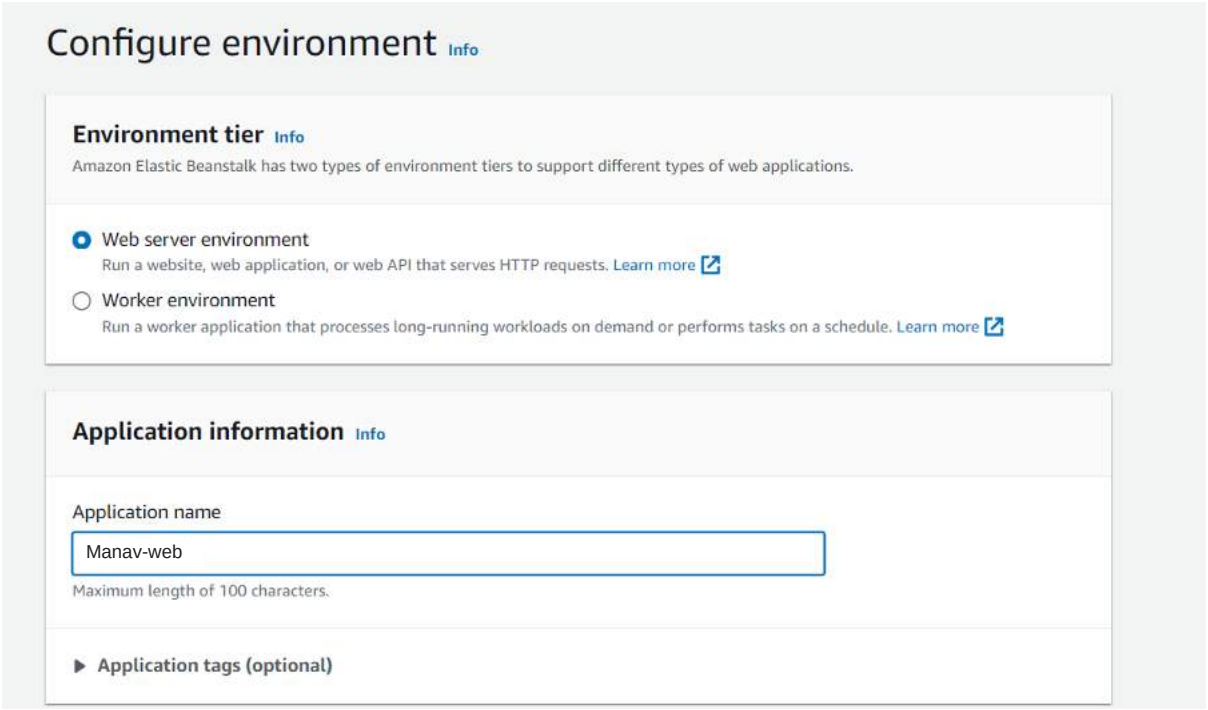
```
Jul 30 08:58:11 ip-172-31-32-173 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jul 30 08:58:11 ip-172-31-32-173 systemd[1]: Started apache2.service - The Apache HTTP Server.
root@ip-172-31-32-173:~# cd /var/www/html
```


Not secure
54.86.249.242
Apache2 Default Page
Ubuntu
It works!
This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should replace this file (located at /var/www/html/index.html) before continuing to operate your HTTP server.
If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.
Configuration Overview
Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is fully documented in /usr/share/doc/apache2/README.Debian.gz. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the manual if the apache2-doc package was installed on this server.
The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:
/etc/apache2/
|-- apache2.conf
| `-- ports.conf
|-- mods-enabled
| |-- *.load
| `-- *.conf
|-- conf-enabled


Not secure
54.86.249.242/index2.html
Hi..Manav here

Elastic beanstalk





Platform Info

Platform type

- (•) Managed platform
  Platforms published and maintained by Amazon Elastic Beanstalk. Learn more
- ( ) Custom platform
  Platforms created and owned by you. This option is unavailable if you have no platforms.

Platform

Node.js

Platform branch

Node.js 20 running on 64bit Amazon Linux 2023

Platform version

6.2.0 (Recommended)

Review Info

Step 1: Configure environment

Edit

Environment information

Environment tier Web server environment	Application name Manav-web
Environment name Manav-web-env	Application code Sample application
Platform arn:aws:elasticbeanstalk:eu-north-1::platform/PHP 8.3 running on 64bit Amazon Linux 2023/4.3.2	

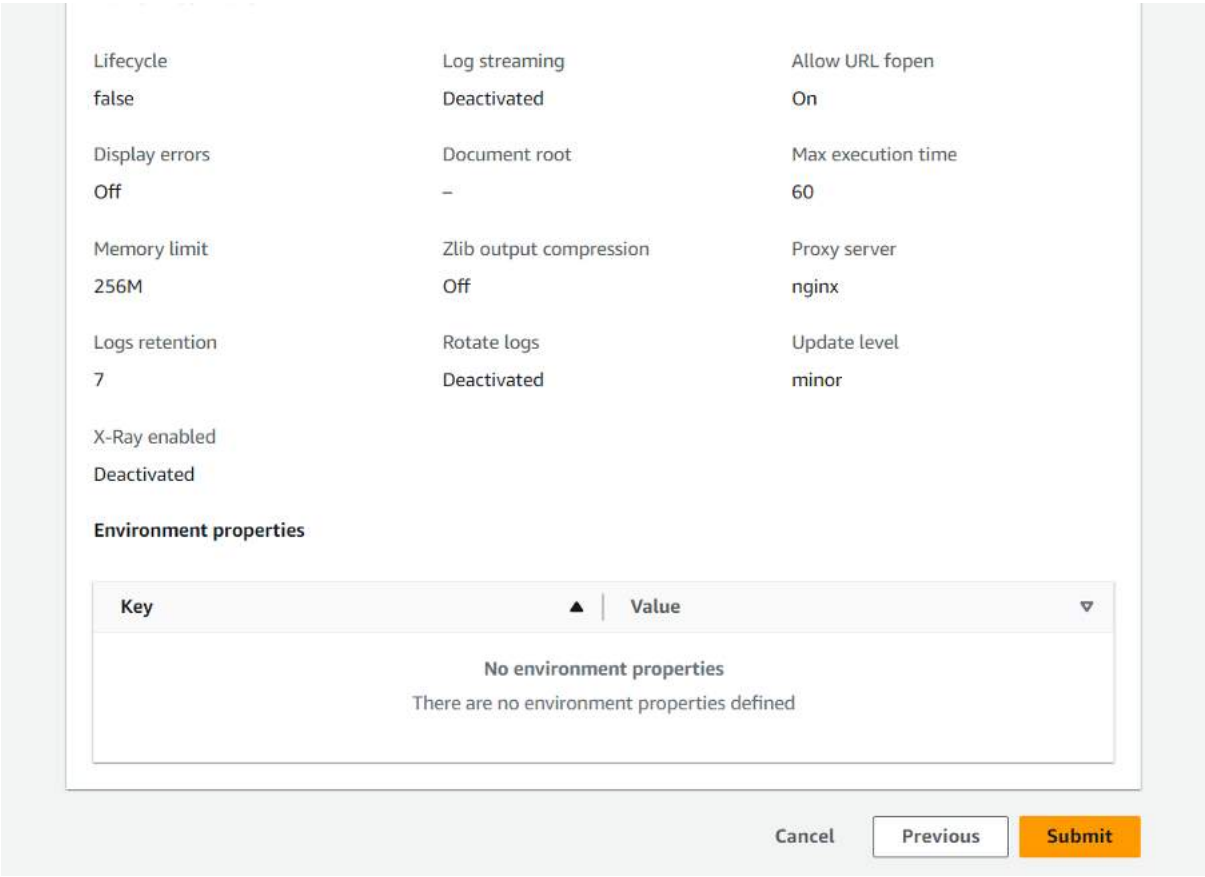
Step 2: Configure service access

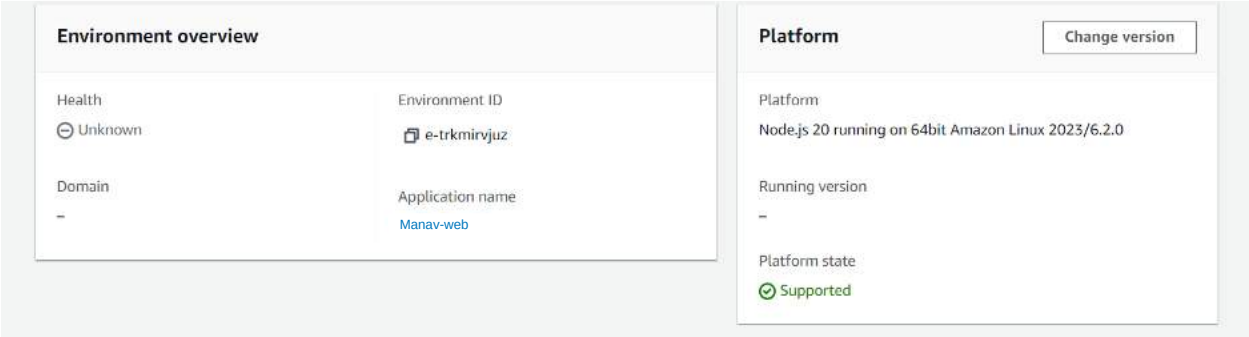
Edit

Service access Info

Configure the service role and EC2 instance profile that Elastic Beanstalk uses to manage your environment. Choose an EC2 key pair to securely log in to your EC2 instances.

Service role	EC2 instance profile


Lifecycle
false
Log streaming
Deactivated
Allow URL fopen
On
Display errors
Off
Document root
–
Max execution time
60
Memory limit
256M
Zlib output compression
Off
Proxy server
nginx
Logs retention
7
Rotate logs
Deactivated
Update level
minor
X-Ray enabled
Deactivated
Environment properties
Key
Value
No environment properties
There are no environment properties defined
Cancel
Previous
Submit


Environment overview
Health
Unknown
Environment ID
e-trkmirvjuz
Domain
–
Application name
Manav-web
Platform
Change version
Platform
Node.js 20 running on 64bit Amazon Linux 2023/6.2.0
Running version
–
Platform state
Supported

Congratulations

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

This environment is launched with Elastic Beanstalk Node.js Platform

What's Next?

- AWS Elastic Beanstalk overview
- AWS Elastic Beanstalk concepts
- Deploying an Express Application to AWS Elastic Beanstalk
- Deploying an Express application with clustering to Elastic Beanstalk
- Customizing and Configuring a Node.js Container
- Working with Logs