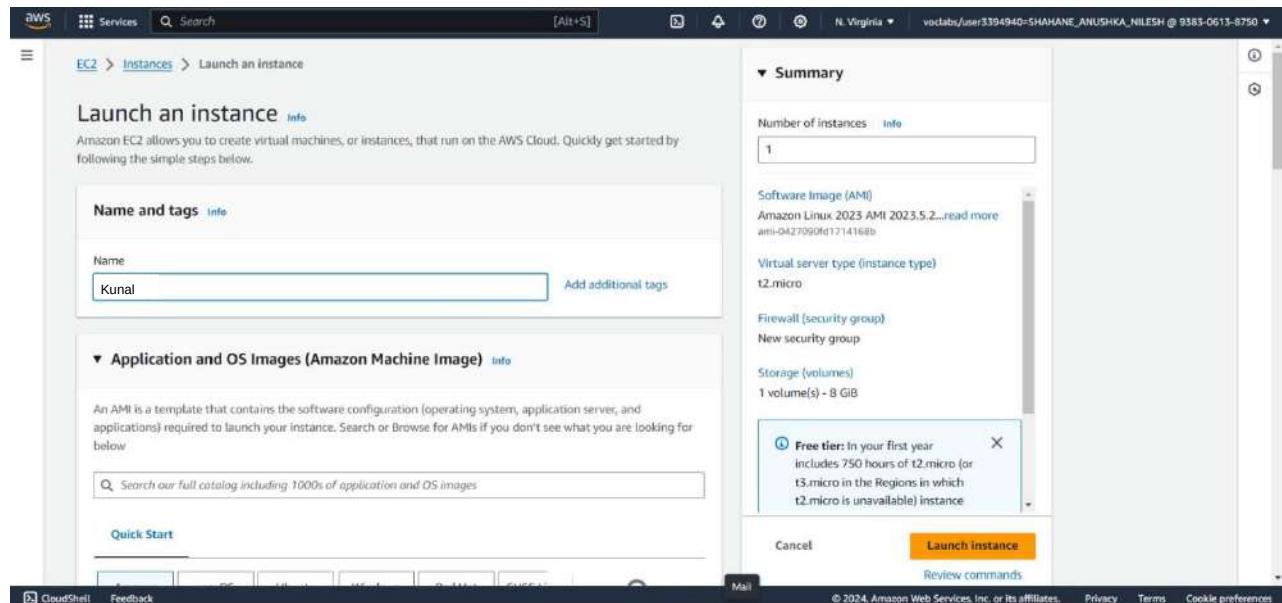


## ADVANCE DEVOPS EXPERIMENT 1



## Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags Info**

Name  
Kunal

Search our full catalog including 1000s of application and OS images

**Quick Start**

[Amazon Linux](#) [macOS](#) [Ubuntu](#) [Windows](#) [Red Hat](#) [SUSE Linux](#)

[Browse more AMIs](#)  
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type  
ami-04a81a99f5ec58529 (64-bit (x86)) / ami-0c14ff330901e49ff (64-bit (Arm))  
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible ▾

Description

Ubuntu Server 24.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Architecture      AMI ID  
64-bit (x86) ▾      ami-04a81a99f5ec58529      Verified provider

**▼ Configure storage** [Info](#) [Advanced](#)

1x  GiB  ▾ Root volume (Not encrypted)

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

[Add new volume](#)

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

ⓘ 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

# Kunal punjabi D15A 44

The screenshot shows the AWS EC2 Instances Launch an instance success page. At the top, there is a green success banner with the message "Successfully initiated launch of instance (i-0df3904aed5f9e9d9)". Below the banner, there is a "Launch log" button. The main content area is titled "Next Steps" and contains four cards:

- Create billing and free tier usage alerts**: To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds. Includes a "Create billing alerts" button.
- Connect to your instance**: Once your instance is running, log into it from your local computer. Includes a "Connect to instance" button and a "Learn more" link.
- Connect an RDS database**: Configure the connection between an EC2 instance and a database to allow traffic flow between them. Includes a "Connect an RDS database" button and a "Create a new RDS database" link.
- Create EBS snapshot policy**: Create a policy that automates the creation, retention, and deletion of EBS snapshots. Includes a "Create EBS snapshot policy" button.

At the bottom right, there is a navigation bar with links 1 through 6.

This screenshot is identical to the one above, but the "Launch log" section is expanded, showing the following log entries:

Event	Status
Initializing requests	Succeeded
Creating security groups	Succeeded
Creating security group rules	Succeeded
Launch initiation	Succeeded

The rest of the interface, including the "Next Steps" cards, is identical to the first screenshot.

# Kunal punjabi D15A 44

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with options like EC2 Global View, Events, Console-to-Code Preview, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, and Images. The main area is titled 'Instances (1) Info' and shows a table with one row. The row contains 'Name' (Kunal), 'Instance ID' (i-0df3904aed5f9e9d9), 'Instance state' (Running), 'Status check' (Initializing), 'Alarm status' (View alarms), 'Availability Zone' (us-east-1b), and 'Public IP' (ec2-54-197-204-120). Below the table, a message says 'Select an instance'.

The screenshot shows the 'Details' tab of the EC2 instance configuration page for instance i-0df3904aed5f9e9d9. The instance summary section includes fields for Instance ID (i-0df3904aed5f9e9d9), IPv6 address (-), Hostname type (IP name: ip-172-31-42-176.ec2.internal), Answer private resource DNS name (IPv4 (A)), and Auto-assigned IP address (-). The Public IPv4 address is 54.197.204.120, and the Private IP4 address is 172.31.42.176. The instance state is Running, and the instance type is t2.micro. The VPC ID is not explicitly shown here but is present in the full URL.

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

ubuntu@ip-172-31-42-176:~$ ls
ubuntu@ip-172-31-42-176:~$ echo "hello"
hello
ubuntu@ip-172-31-42-176:~$ cat > myfile.txt
This is Advance devops lab
^C
ubuntu@ip-172-31-42-176:~$ cat myfile
cat: myfile: No such file or directory
ubuntu@ip-172-31-42-176:~$ cat myfile.txt
This is Advance devops lab
ubuntu@ip-172-31-42-176:~$
```

## Hosting a static website using EC2 instance:

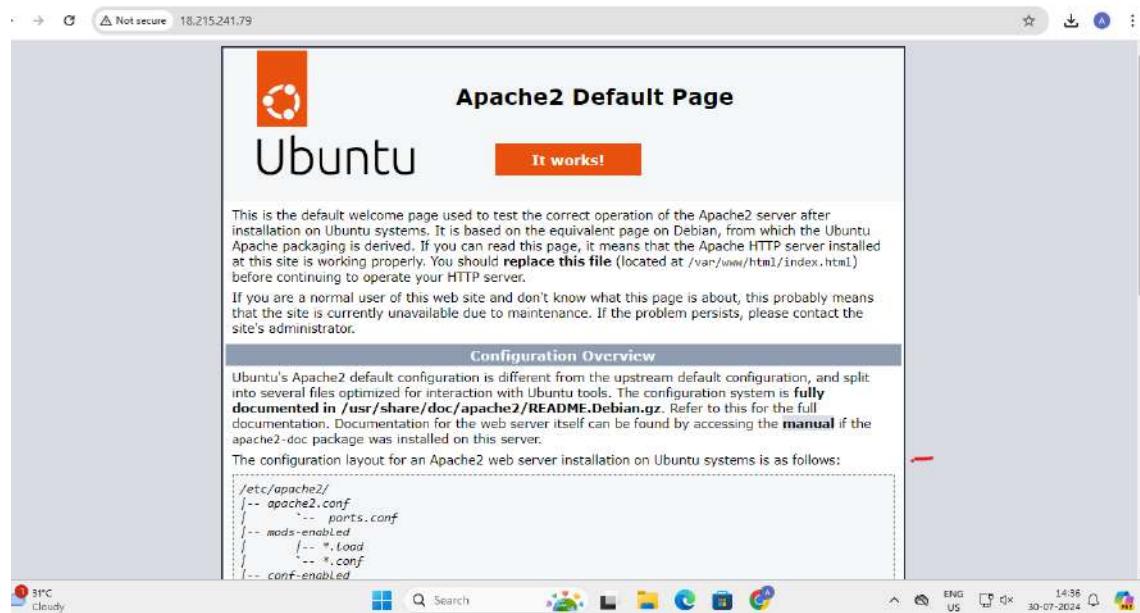
```
*** System restart required ***
Pending kernel upgrade!
Running kernel version:
  6.8.0-1009-aws
Diagnostics:
  The currently running kernel version is not the expected kernel version 6.8.0-1012-aws.
Last login: Tue Jul 30 08:37:47 2024 from 18.206.107.28
ubuntu@ip-172-31-41-78:~$ sudo su
root@ip-172-31-41-78:/home/ubuntu# apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.58-1ubuntu8.4).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ip-172-31-41-78:/home/ubuntu# systemctl
```

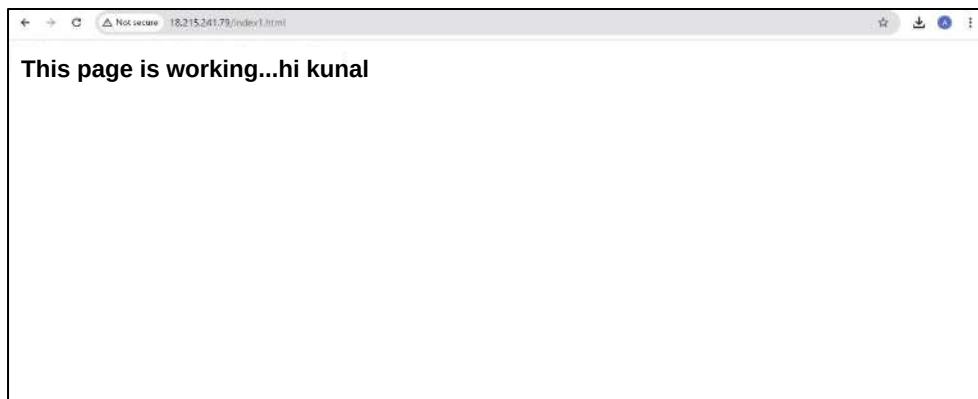
```
[ 12917 /usr/sbin/apache2 -k start
[ 12919 /usr/sbin/apache2 -k start
[ 12921 /usr/sbin/apache2 -k start

Jul 30 08:44:17 ip-172-31-41-78 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jul 30 08:44:17 ip-172-31-41-78 systemd[1]: Started apache2.service - The Apache HTTP Server.
root@ip-172-31-41-78:/home/ubuntu# cd/var/www/html/
bash: cd/var/www/html/: No such file or directory
root@ip-172-31-41-78:/home/ubuntu# cd /var/www/html/
root@ip-172-31-41-78:/var/www/html# /var/www/html#
bash: /var/www/html#: No such file or directory
root@ip-172-31-41-78:/var/www/html# ]
```

```
command 'systemctl' from deb systemctl (1.4.4181-1.1)
Try: apt install <deb name>
root@ip-172-31-41-78:/home/ubuntu# 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 08:44:17 UTC; 12min ago
     Docs: https://httpd.apache.org/docs/2.4/
      Main PID: 12917 (apache2)
        Tasks: 55 (limit: 1130)
       Memory: 5.3M (peak: 5.4M)
          CPU: 74ms
        CGroup: /system.slice/apache2.service
                  ├─12917 /usr/sbin/apache2 -k start
                  ├─12919 /usr/sbin/apache2 -k start
                  └─12921 /usr/sbin/apache2 -k start

Jul 30 08:44:17 ip-172-31-41-78 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jul 30 08:44:17 ip-172-31-41-78 systemd[1]: Started apache2.service - The Apache HTTP Server.
root@ip-172-31-41-78:/home/ubuntu#
```





## Hosting using S3 bucket :

A screenshot of the 'Create bucket' configuration page in the AWS Management Console. The top navigation bar shows 'Amazon S3 &gt; Buckets &gt; Create bucket'. The main section is titled 'Create bucket' with a 'Info' link. A sub-section titled 'General configuration' is shown. Under 'AWS Region', it is set to 'US East (N. Virginia) us-east-1'. Under 'Bucket type', 'General purpose' is selected. The 'Bucket name' field contains 'test-123-Kunal'. Below the bucket name, a note states: 'Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming.' Under 'Copy settings from existing bucket - optional', there is a 'Choose bucket' button and a note: 'Only the bucket settings in the following configuration are copied.' The 'Format' is listed as 's3://bucket/prefix'.

**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 <a href="#">All AWS Regions</a> Storage lens provides visibility into storage usage and activity trends. <a href="#">Learn more</a>									
<a href="#">General purpose buckets</a>	<a href="#">Directory buckets</a>								
<b>General purpose buckets (1)</b> <a href="#">Info</a> <a href="#">All AWS Regions</a>	<a href="#">View Storage Lens dashboard</a>								
Buckets are containers for data stored in S3. <input type="text" value="Find buckets by name"/>	<a href="#"></a> <a href="#"></a> <a href="#">Empty</a> <a href="#">Delete</a> <a href="#" style="background-color: orange; color: white; border: 1px solid orange; padding: 2px 10px;">Create bucket</a>								
<table><thead><tr><th>Name</th><th>AWS Region</th><th>IAM Access Analyzer</th><th>Creation date</th></tr></thead><tbody><tr><td><a href="#">test-123-Kunal</a></td><td>US East (N. Virginia) us-east-1</td><td><a href="#">View analyzer for us-east-1</a></td><td>August 11, 2024, 19:49:09 (UTC+05:30)</td></tr></tbody></table>	Name	AWS Region	IAM Access Analyzer	Creation date	<a href="#">test-123-Kunal</a>	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	August 11, 2024, 19:49:09 (UTC+05:30)	
Name	AWS Region	IAM Access Analyzer	Creation date						
<a href="#">test-123-Kunal</a>	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	August 11, 2024, 19:49:09 (UTC+05:30)						

# Kunal punjabi D15A 44

The screenshot shows the AWS Lambda Test Grid upload summary page. At the top, a green header bar indicates "Upload succeeded" with a link to "View details below". Below this, a message states "The information below will no longer be available after you navigate away from this page." The main section is titled "Summary" and displays the destination "s3://test-123-Kunal" with a status of "Succeeded" (1 file, 0 B (0%)). The "Failed" section shows 0 files (0 B (0%)). Below the summary, there are tabs for "Files and folders" and "Configuration", with "Files and folders" selected. The "Files and folders" section shows one item: "Test.txt" (text/plain, 0 B, Succeeded). A search bar and navigation icons are also present.

The screenshot shows the AWS S3 object details page for "Test.txt" located in the "test-123-anushka" bucket. The left sidebar includes links for Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Block Public Access settings for this account, Storage Lens (Dashboards, Storage Lens groups, AWS Organizations settings), Feature spotlight, and AWS Marketplace for S3. The main content area shows the object "Test.txt" with the "info" tab selected. The "Properties" tab is also visible. The "Object overview" section provides the following details:

Attribute	Value
Owner	awslambda0w4201293t1653663267
AWS Region	US East (N. Virginia) us-east-1
Last modified	August 11, 2024, 19:58:50 (UTC+05:30)
Size	-
Type	txt
Key	
S3 URI	<a href="https://s3://test-123-anushka/test.txt">s3://test-123-anushka/test.txt</a>
Amazon Resource Name (ARN)	<a href="#">arn:aws:s3:::test-123-anushka/test.txt</a>
Entity tag (Etag)	<a href="#">d41d8cd98f00b204e9800998ecf8427e</a>
Object URL	<a href="https://test-123-kunal.s3.amazonaws.com/test.txt">https://test-123-kunal.s3.amazonaws.com/test.txt</a>

Kunal punjabi D15A 44

Successfully edited bucket policy.

Amazon S3 > Buckets > test-123-anushka

test-123-Kunal [Info](#)

Objects Properties Permissions Metrics Management Access Points

**Permissions overview**

Access finding

Access Findings are provided by IAM external access analyzers. Learn more about [How IAM analyzer Findings work](#).  
[View analyzer for us-east-1](#)

**Block public access (bucket settings)**

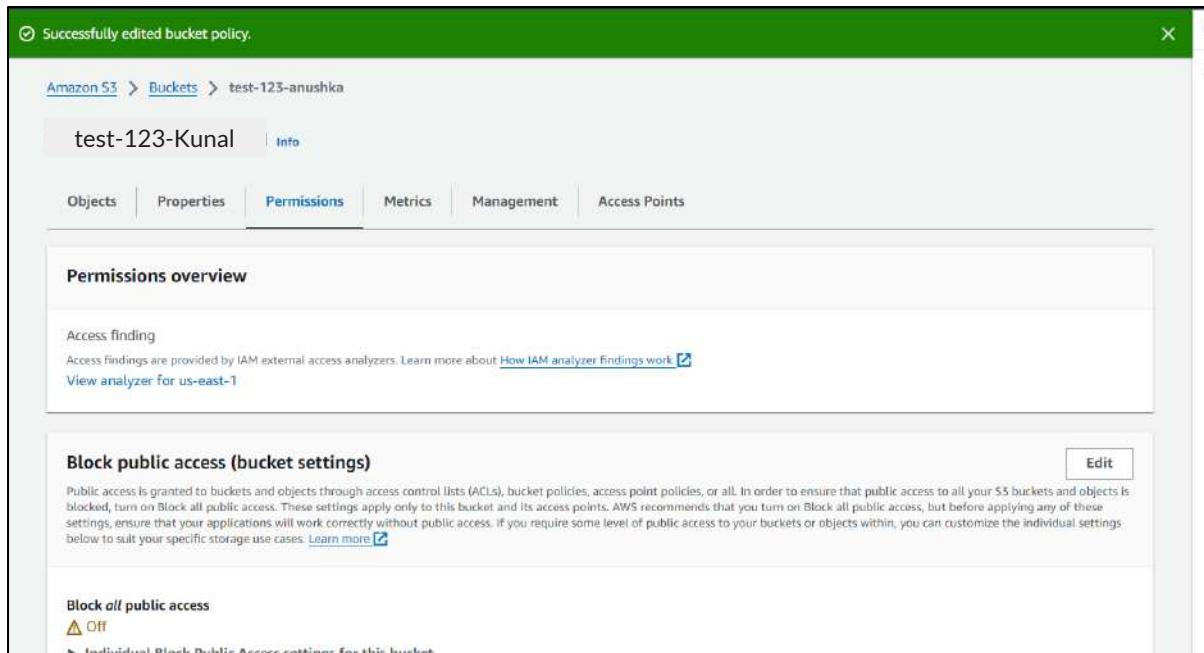
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#).

**Block all public access**

Off

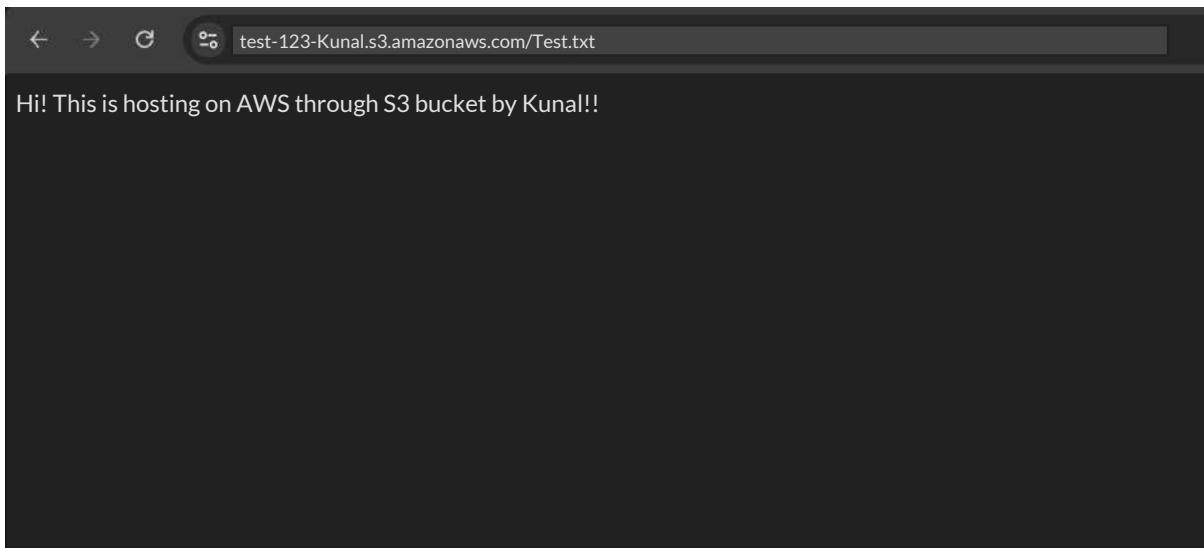
[Individual Block Public Access settings for this bucket](#)

[Edit](#)

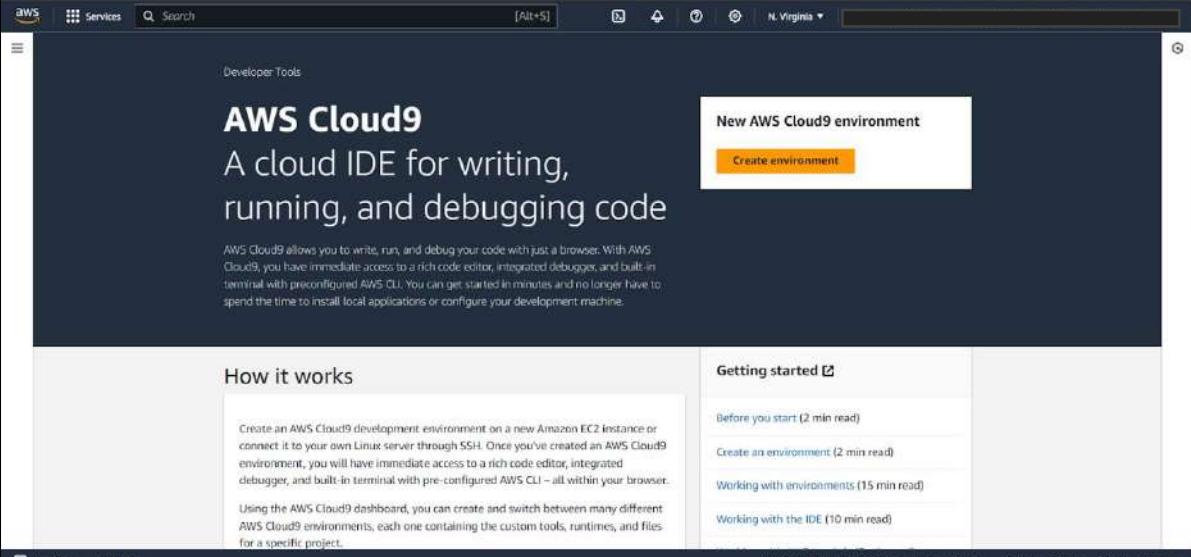


← → ⌂ test-123-Kunal.s3.amazonaws.com/Test.txt

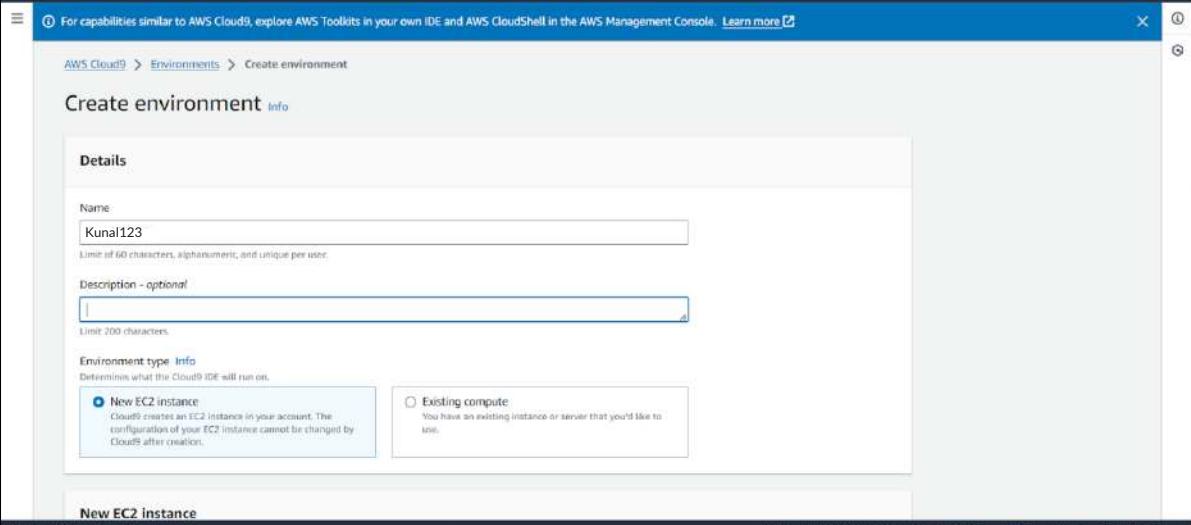
Hi! This is hosting on AWS through S3 bucket by Kunal!!



## Hosting using Cloud 9 :



The screenshot shows the AWS Cloud9 homepage. At the top right, there is a prominent yellow "Create environment" button. Below it, a section titled "New AWS Cloud9 environment" is visible. On the left, there's a "How it works" section with a detailed description of the service. On the right, there's a "Getting started" sidebar with links to various documentation pages. The bottom of the page includes standard AWS navigation links like CloudShell and Feedback.



The screenshot shows the "Create environment" dialog box. In the "Details" section, the "Name" field is filled with "Kunal123". Under "Environment type", the "New EC2 instance" option is selected, with a note explaining that Cloud9 creates a new EC2 instance in the user's account. The "Existing compute" option is also available but not selected. At the bottom of the dialog, there is a "New EC2 instance" button.

The screenshot shows the AWS Cloud9 interface. On the left, a sidebar lists 'My environments', 'Shared with me', 'All account environments', and 'Documentation'. The main area displays a progress bar for creating an environment, stating 'Creating Kunal 123 This can take several minutes. While you wait, see Best practices for using AWS Cloud9'. Below this, a message encourages exploring AWS Toolkits in the AWS Management Console. The central part of the screen shows the 'Environments (1)' section with a table. The table has columns: Name, Cloud9 IDE, Environment type, Connection, Permission, and Owner ARN. One environment is listed: 'Kunal123' (Status: Open, Type: EC2 instance, Connection: Secure Shell (SSH), Permission: Owner, ARN: arn:aws:sts:938306138750:assumed-role/vclabs/user\$394940:Punjabi\_Kunal\_Mahesh). At the bottom, there are links for CloudShell and Feedback, and a footer with copyright information.

The screenshot shows the AWS Cloud9 development environment. The top navigation bar includes 'Go', 'Run', 'Tools', 'Window', 'Support', 'Preview', and 'Run'. The main area displays the 'Welcome' page with the title 'AWS Cloud9' and the sub-headline 'Welcome to your development environment'. It explains that AWS Cloud9 allows writing, running, and debugging code in a browser. Below this is the 'Toolkit for AWS Cloud9' section, which provides a brief overview of the toolkit's features. A terminal window at the bottom shows a bash session with the command 'vclabs:~/environment \$'. A 'Getting started' sidebar on the right offers options like 'Create File', 'Upload Files...', and 'Clone from GitHub'.

A screenshot of a code editor interface. The main area shows the code for `Cloud9.html`:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Anushka Shahana's Website</title>
<style>
body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
    background-color: #f4f4f4;
}
.navbar {
    background-color: #003366;
    overflow: hidden;
}
.navbar a {
    float: left;
    display: block;
    color: #222222;
    text-align: center;
    padding: 14px 20px;
    text-decoration: none;
}
.navbar a:hover {
    background-color: #ddd;
    color: #000000;
}
.container {
    padding: 20px;
}
.header {
    text-align: center;
}
```

Below the code editor is a terminal window showing:

```
bash -[ip-172-31-72-68 ~] Immediate voulabs:~/environment $
```

A screenshot of a browser window and a code editor side-by-side.

The browser window displays a website with the title "Welcome to Kunal's Website". The navigation bar includes links for "Home", "About", and "Contact". The content area contains the following text:

Welcome to Kunal's Website

**About Me**

Hello! I'm Kunal Punjabi This is a simple example of my personal website hosted on AWS Cloud9

**Content Section**

I am student of VESIT DEPARTMENT:IT Class:D15A/44

The code editor shows the same `Cloud9.html` file as the previous screenshot, with the cursor positioned near the end of the `body` section.

## 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

The screenshot shows the 'Create bucket' configuration page in the AWS Management Console. The URL in the address bar is 'Amazon S3 > Buckets > Create bucket'. The main title is 'Create bucket' with an 'Info' link. A sub-instruction says 'Buckets are containers for data stored in S3.' Below this is a 'General configuration' section. Under 'AWS Region', it shows 'US East (N. Virginia) us-east-1'. Under 'Bucket type', there are two options: 'General purpose' (selected) and 'Directory - New'. The 'General purpose' option is described as recommended for most use cases and access patterns, noting that general purpose buckets are the original S3 bucket type and allow a mix of storage classes. The 'Directory - New' option is described as recommended for low-latency use cases, using only the S3 Express One Zone storage class. Below this is a 'Bucket name' field containing 'test-Kunal'. A note below the field states that the bucket name must be unique within the global namespace and follow bucket naming rules, with a link to 'rules for bucket naming'. There is also a 'Copy settings from existing bucket - optional' section with a 'Choose bucket' button and a note about only copying bucket settings. At the bottom, it says 'Format: s3://bucket/prefix'.

**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**

**Info** After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

[Cancel](#) [Create bucket](#)

Amazon S3 > Buckets											
▶ <b>Account snapshot - updated every 24 hours</b> <a href="#">All AWS Regions</a>	<a href="#">View Storage Lens dashboard</a>										
Buckets are containers for data stored in S3. <a href="#">Learn more</a>											
<a href="#">General purpose buckets</a> <a href="#">Directory buckets</a>											
<b>General purpose buckets (1)</b> <a href="#">Info</a> <a href="#">All AWS Regions</a>	<a href="#">Create bucket</a>										
Buckets are containers for data stored in S3. <a href="#">Learn more</a>											
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 5px;"><input type="text"/> Find buckets by name:</th> <th style="text-align: right; padding: 5px;">&lt; 1 &gt; <a href="#">@</a></th> </tr> <tr> <th style="text-align: left; padding: 5px;">Name</th> <th style="text-align: left; padding: 5px;">AWS Region</th> <th style="text-align: left; padding: 5px;">IAM Access Analyzer</th> <th style="text-align: left; padding: 5px;">Creation date</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;"><a href="#">test-Kunal</a></td> <td style="padding: 5px;">US East (N. Virginia) us-east-1</td> <td style="padding: 5px;"><a href="#">View analyzer for us-east-1</a></td> <td style="padding: 5px;">August 12, 2024, 20:04:18 (UTC+05:30)</td> </tr> </tbody> </table>		<input type="text"/> Find buckets by name:	< 1 > <a href="#">@</a>	Name	AWS Region	IAM Access Analyzer	Creation date	<a href="#">test-Kunal</a>	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	August 12, 2024, 20:04:18 (UTC+05:30)
<input type="text"/> Find buckets by name:	< 1 > <a href="#">@</a>										
Name	AWS Region	IAM Access Analyzer	Creation date								
<a href="#">test-Kunal</a>	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	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)

All files and folders in this table will be uploaded.

[Remove](#)

[Add files](#)

[Add folder](#)

[Find by name](#)

< 1 >

<input type="checkbox"/>	Name	Folder	Type	Size
<input type="checkbox"/>	test.txt	-	text/plain	50.0 B

### Destination Info

#### Destination

s3://test-Kunal

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-Kunal

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
<b>Object overview</b>		
Owner awslabsc0w3698888l1642940625	S3 URI <a href="s3://test-Kunal/test.html">s3://test-Kunal/test.html</a>	
AWS Region US East (N. Virginia) us-east-1	Amazon Resource Name (ARN) <a href="#">arn:aws:s3:::test-Kunal/test.html</a>	
Last modified August 12, 2024, 22:33:51 (UTC+05:30)	Entity tag (Etag) <a href="#">7a3411f1dad97a2779c8dc65580432d2</a>	
Size 287.0 B	Object URL <a href="https://test-Kunal.s3.amazonaws.com/test.html">https://test-Kunal.s3.amazonaws.com/test.html</a>	
Type html		
Key <a href="#">test.html</a>		

## 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](#) 

Objects Properties Permissions Metrics Management Access Points

### Permissions overview

Access finding

Access Findings are provided by IAM external access analyzers. Learn more about [How IAM analyzer findings work](#).

[View analyzer for us-east-1](#)

### Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

**Block all public access**

On

► Individual Block Public Access settings for this bucket

### 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](#)

Edit Delete

### 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-Kunal/*" } ] }
```



- Launching an EC2 instance

EC2 > Instances > Launch an instance

## Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags Info**

Name  Add additional tags

---

**Quick Start**

Amazon Linux  macOS  Ubuntu  Windows  Red Hat  SUSE Linux 

 [Browse more AMIs](#)  
Including AMIs from: AWS, Marketplace and the Community

**Amazon Machine Image (AMI)**

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type Free tier eligible ▾  
ami-04a81a99f5ec58529 (64-bit (x86)) / ami-0c14ff330901e49ff (64-bit (Arm))  
Virtualization: hvm ENA enabled: true Root device type: ebs

**Description**  
Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

**Architecture**  **AMI ID** ami-04a81a99f5ec58529 **Verified provider**

## ▼ 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) <a href="#">Info</a>		<a href="#">Connect</a>	Instance state ▾	Actions ▾	Launch instances	⋮
<input type="text"/> Find Instance by attribute or tag (case-sensitive)				All states ▾		⋮
<input type="checkbox"/>	Name ▾	Instance ID	Instance state	Instance type	Status check	Alarm status
<input type="checkbox"/>	Kunal	i-0e39cd326d64588eb	<span>Running</span> <a href="#">View details</a> <a href="#">Logs</a>	t2.micro	<span>2/2 checks passed</span> <a href="#">View alarms</a> <a href="#">+</a>	us-east-1a

[Details](#) | [Status and alarms](#) | [Monitoring](#) | [Security](#) | [Networking](#) | [Storage](#) | [Tags](#)

▼ Instance summary [Info](#)

Instance ID <a href="#">i-0e39cd326d64588eb (Kunal)</a>	Public IPv4 address <a href="#">34.201.2.60</a>   <a href="#">open address</a>	Private IPv4 addresses: <a href="#">172.31.13.190</a>
IPv6 address -	Instance state <span>Running</span>	Public IPv4 DNS <a href="#">ec2-34-201-2-60.compute-1.amazonaws.com</a>   <a href="#">open address</a>
Hostname type IP name: ip-172-31-13-190.ec2.internal	Private IP DNS name (IPv4 only) <a href="#">ip-172-31-13-190.ec2.internal</a>	Elastic IP addresses
Answer private resource DNS name	Instance type	

```
[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-browser
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
```

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**

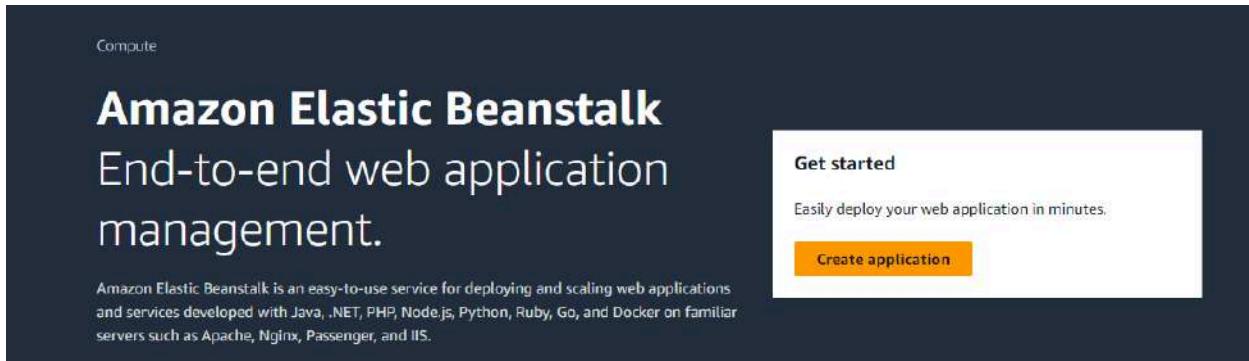
Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. 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:

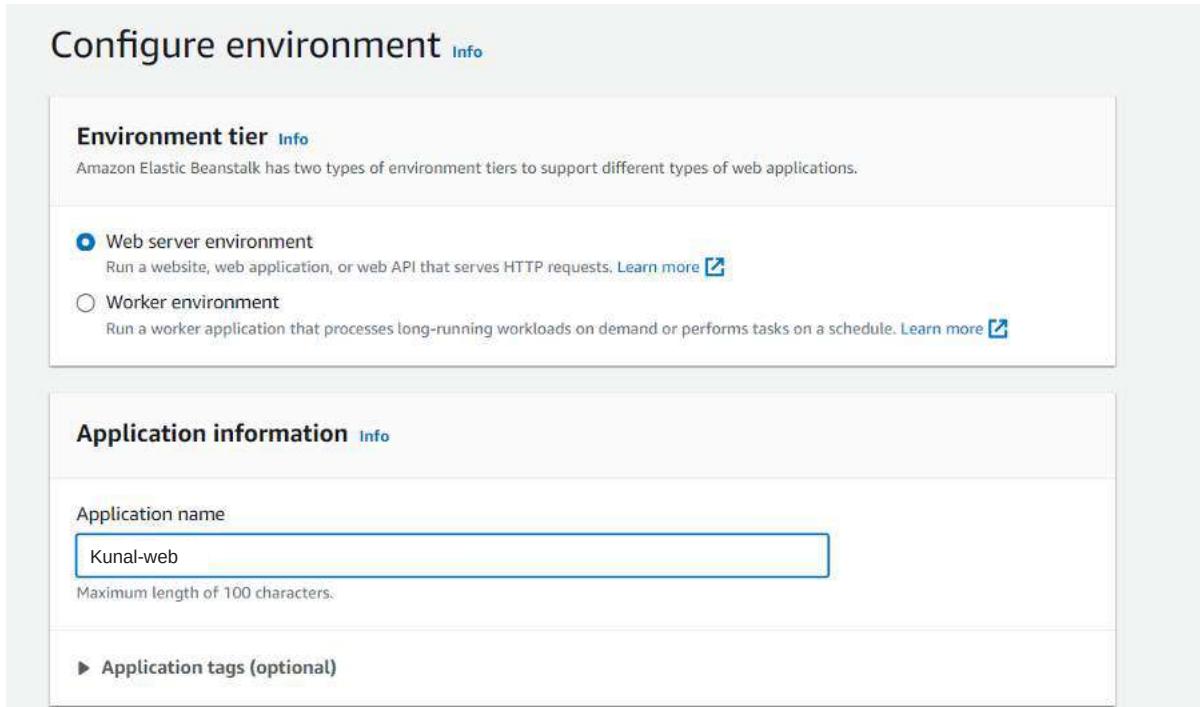
```
/etc/apache2/
|-- apache2.conf
|   '-- ports.conf
|-- mods-enabled
|   '-- *.load
|   '-- *.conf
|-- conf-enabled
```

Hi..Kunal here

## Elastic beanstalk



The screenshot shows the Amazon Elastic Beanstalk landing page. At the top left, it says "Compute". The main title is "Amazon Elastic Beanstalk" in large bold letters, followed by the subtitle "End-to-end web application management." Below the title, there is a brief description: "Amazon Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS." To the right, there is a "Get started" button with the subtext "Easily deploy your web application in minutes." and a "Create application" button.



The screenshot shows the "Configure environment" step of a wizard. It has two sections: "Environment tier" and "Application information".

**Environment tier** (Info):  
Amazon Elastic Beanstalk has two types of environment tiers to support different types of web applications.  
 Web server environment  
Run a website, web application, or web API that serves HTTP requests. [Learn more](#)   
 Worker environment  
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule. [Learn more](#) 

**Application information** (Info):  
Application name: Kunal-web  
Maximum length of 100 characters.  
▶ Application tags (optional)

## 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	Application name
Web server environment	Kunal-web
Environment name	Application code
Kunal-web-env	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	Log streaming	Allow URL fopen						
false	Deactivated	On						
Display errors	Document root	Max execution time						
Off	-	60						
Memory limit	Zlib output compression	Proxy server						
256M	Off	nginx						
Logs retention	Rotate logs	Update level						
7	Deactivated	minor						
X-Ray enabled								
Deactivated								
<b>Environment properties</b>								
<table border="1"> <thead> <tr> <th>Key</th> <th>▲   Value</th> </tr> </thead> <tbody> <tr> <td colspan="2">No environment properties</td> </tr> <tr> <td colspan="2">There are no environment properties defined</td> </tr> </tbody> </table>			Key	▲   Value	No environment properties		There are no environment properties defined	
Key	▲   Value							
No environment properties								
There are no environment properties defined								
<input type="button" value="Cancel"/> <input type="button" value="Previous"/> <input style="background-color: orange; color: white; border: none; padding: 2px 10px; border-radius: 5px;" type="button" value="Submit"/>								

Environment overview		Platform	<a href="#">Change version</a>
Health	Environment ID	Platform	
<span>⊖ Unknown</span>	<span>🔗 e-trkmirvjuZ</span>	Node.js 20 running on 64bit Amazon Linux 2023/6.2.0	
Domain	Application name	Running version	
-	Kunal-web	-	
		Platform state	
		<span>☑ Supported</span>	

# 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](#)

# ADVANCE DEVOPS EXPERIMENT - 3

Name:Kunal Punjabi

Class:D15A

Roll No:44

**Aim:** To understand the Kubernetes Cluster Architecture, install and Spin Up a Kubernetes Cluster on Linux Machines/Cloud Platforms.

## Step 1:Pre-requisites

1.1 Create 3 EC2 instances,one for the master node and two for the worker nodes.

The screenshot shows the 'Launch an instance' wizard on the AWS Management Console. The left pane displays the configuration steps, and the right pane shows the summary and configuration details.

**Left Pane (Launch an instance):**

- Name and tags:** A field where 'Master' is entered.
- Application and OS Images (Amazon Machine Image):** A search bar and a 'Quick Start' section with links to various AMIs: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE.

**Right Pane (Summary):**

- Number of Instances:** Set to 1.
- Software Image (AMI):** Canonical, Ubuntu, 22.04 LTS, ami-0c2af5f1e265bd5e0e.
- Virtual server type (instance type):** t2.medium.
- Firewall (security group):** New security group.
- Storage (volumes):** 1 volume(s) - 8 GiB.

A modal window is open on the right, providing information about the Free tier:

**Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month.

Buttons at the bottom of the modal include 'Cancel', 'Launch Instance' (highlighted in orange), and 'Review commands'.

1.2 Proceed with the following settings and create a new key pair as follows(use the same key pair for all the three nodes)

The screenshot shows the AWS Lambda 'Create Function' configuration interface. It includes sections for 'Instance type', 'Key pair (login)', and 'Network settings'.

**Instance type:** t2.medium (selected).  
Family: t2 - 2 vCPU - 4 GiB Memory - Current generation: true.  
On-Demand Linux base pricing: 0.0496 USD per Hour  
On-Demand Windows base pricing: 0.0676 USD per Hour  
On-Demand RHEL base pricing: 0.0784 USD per Hour  
On-Demand SUSE base pricing: 0.1496 USD per Hour

**Key pair (login):** two-tier-app-k8s (selected).  
Create new key pair

**Network settings:** Network: vpc-04007898e59a6979f  
Subnet: (Info)

## Create key pair

**Key pair name**  
Key pairs allow you to connect to your instance securely.

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

**Key pair type**

RSA  
RSA encrypted private and public key pair

ED25519  
ED25519 encrypted private and public key pair

**Private key file format**

.pem  
For use with OpenSSH

.ppk  
For use with PuTTY

**⚠ When prompted, store the private key in a secure and accessible location on**

**Cancel** **Create key pair**

Instances (1/3) <a href="#">Info</a>										
<a href="#">Last updated less than a minute ago</a> <span>C</span> <span>Connect</span> <span>Instance state ▾</span> <span>Actions ▾</span> <span>Launch instances</span> <span>▼</span>										
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/> <span>All states ▾</span>										
Name ▾	Instance ID	Instance state ▾	Instance type	Status check	Alarm status	Availability Zone ▾	Public IPv4 DNS ▾	Public IP ▾		
<input type="checkbox"/> Worker-2	i-0e3930ceb2d892d01	<span>Running</span> <span>Q</span> <span>Q</span>	t2.medium	<span>✓ Z/Z checks passed</span>	<span>View alarms +</span>	ap-south-1a	ec2-13-254-226-219.ap... 13.254.226.219	13.254.226.219		
<input type="checkbox"/> Worker-1	i-0d16e01d1824e0e3a	<span>Running</span> <span>Q</span> <span>Q</span>	t2.medium	<span>✓ Z/Z checks passed</span>	<span>View alarms +</span>	ap-south-1a	ec2-65-0-104-95.ap-so... 65.0.104.95	65.0.104.95		
<input checked="" type="checkbox"/> Master	i-01ae3d588db90ad73	<span>Running</span> <span>Q</span> <span>Q</span>	t2.medium	<span>✓ Z/Z checks passed</span>	<span>View alarms +</span>	ap-south-1a	ec2-13-252-56-34.ap-s... 13.252.56.34	13.252.56.34		

1.3 After the instances have been created, copy the text given in the example part of each of the three instances into git bash.

The screenshot shows the AWS CloudWatch Metrics interface with the 'SSH client' tab selected. It displays the following information:

- Instance ID:** i-0e3930ceb2d892d01 (Worker-2)
- Step-by-step instructions:**
  1. Open an SSH client.
  2. Locate your private key file. The key used to launch this instance is two-tier-app-k8s.pem
  3. Run this command, if necessary, to ensure your key is not publicly viewable.  
chmod 400 "two-tier-app-k8s.pem"
  4. Connect to your instance using its Public DNS:  
ec2-13-234-226-219.ap-south-1.compute.amazonaws.com
- Example command:** ssh -i "two-tier-app-k8s.pem" ubuntu@ec2-13-234-226-219.ap-south-1.compute.amazonaws.com

```
acer@TMP214-53 MINGW64 ~/Downloads
$ ssh -i "two-tier-app-k8s.pem" ubuntu@ec2-13-232-36-34.ap-south-1.compute.amazonaws.com
The authenticity of host 'ec2-13-232-36-34.ap-south-1.compute.amazonaws.com (13.232.36.34)' can't be established.
ED25519 key fingerprint is SHA256:uVGEO+FwYefj60j0ft70Sralv8NrzEi/IwxAtBY+EPE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-232-36-34.ap-south-1.compute.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1022-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Wed Sep 11 14:07:10 UTC 2024

System load: 0.0          Processes:           106
Usage of /: 20.7% of 7.57GB   Users logged in:      0
Memory usage: 5%           IPv4 address for eth0: 172.31.45.227
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
```

## Step 2: Prepare Nodes

### 2.1. Update the package manager on all nodes:

```
sudo apt-get update && sudo apt-get upgrade -y
```

The screenshot shows two terminal windows side-by-side. Both terminals are running on an Ubuntu system, indicated by the prompt `ubuntu@ip-172-31-28-127:~`. The left terminal displays the output of the command, which includes system status information like memory usage and swap usage, followed by the update and upgrade process. The right terminal shows the same command being run again. The output in both terminals is identical, showing the progress of the package download and installation.

```
ubuntu@ip-172-31-28-127:~
```

```
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
```

```
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
```

```
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
```

```
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
```

```
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
```

```
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
```

```
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
```

```
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2023 kB]
```

```
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [352 kB]
```

```
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [17.8 kB]
```

```
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [2437 kB]
```

```
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [419 kB]
```

```
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted a
```

This screenshot shows a single terminal window with the output of the update and upgrade command. The output is identical to the one shown in the previous screenshot, indicating that the package manager has successfully updated the system.

```
ubuntu@ip-172-31-28-127:~
```

```
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
```

```
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
```

```
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
```

```
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
```

```
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
```

```
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
```

```
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
```

```
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2023 kB]
```

```
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [352 kB]
```

```
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [17.8 kB]
```

```
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [2437 kB]
```

```
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [419 kB]
```

```
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted a
```

2.2. Disable Swap (Kubernetes requires swap to be off):

```
sudo swapoff -a
```

```
sudo sed -i '/ swap / s/^/#/' /etc/fstab
```

```
ubuntu@ip-172-31-22-29:~$ sudo swapoff -a  
sudo sed -i '/ swap / s/^/#/' /etc/fstab
```

2.3. Load necessary kernel modules for networking and iptables:

```
cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf
```

```
overlay
```

```
br_netfilter
```

```
EOF
```

```
sudo modprobe overlay
```

```
sudo modprobe br_netfilter
```

```
ubuntu@ip-172-31-22-29:~$ cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf  
overlay  
br_netfilter  
EOF  
sudo modprobe overlay  
sudo modprobe br_netfilter  
overlay  
br_netfilter
```

2.4. Configure sysctl settings for Kubernetes networking:

```
cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
```

```
net.bridge.bridge-nf-call-ip6tables = 1
```

```
net.bridge.bridge-nf-call-iptables = 1
```

```
EOF
```

```
sudo sysctl --system
```

```

ubuntu@ip-172-31-22-29:~$ cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf
overlay
br_netfilter
EOF
sudo modprobe overlay
sudo modprobe br_netfilter
overlay
br_netfilter
ubuntu@ip-172-31-22-29:~$ cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
EOF
sudo sysctl --system
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
* Applying /etc/sysctl.d/10-console-messages.conf ...
kernel.printk = 4 4 1 7
* Applying /etc/sysctl.d/10-ipv6-privacy.conf ...
net.ipv6.conf.all.use_tempaddr = 2
net.ipv6.conf.default.use_tempaddr = 2
* Applying /etc/sysctl.d/10-kernel-hardening.conf ...
kernel.kptr_restrict = 1

```

### Step 3: Install Docker

Kubernetes uses container runtimes like Docker. Install Docker on all nodes.

```

sudo apt-get update
sudo apt-get install -y apt-transport-https ca-certificates curl software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu
$(lsb_release -cs) stable"
sudo apt-get update
sudo apt-get install -y docker-ce docker-ce-cli containerd.io

```

```

ubuntu@ip-172-31-22-29:~$ sudo apt-get update
sudo apt-get install -y apt-transport-https ca-certificates curl software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu
$(lsb_release -cs) stable"
sudo apt-get update
sudo apt-get install -y docker-ce docker-ce-cli containerd.io
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Fetched 129 kB in 1s (241 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
curl is already the newest version (7.81.0-1ubuntu1.17).
curl set to manually installed.
software-properties-common is already the newest version (0.99.22.9).
software-properties-common set to manually installed

```

Configure Docker for Kubernetes:

```
cat <<EOF | sudo tee /etc/docker/daemon.json
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
EOF
```

```
sudo systemctl restart docker
```

```
ubuntu@ip-172-31-22-29:~$ cat <<EOF | sudo tee /etc/docker/daemon.json
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
EOF
sudo systemctl restart docker
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
```

## Step 4: Install kubeadm, kubelet, kubectl

Install Kubernetes tools on all nodes.

### 4.1. Add Kubernetes APT repository:

```
sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg
https://packages.cloud.google.com/apt/doc/apt-key.gpg
echo "deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg]
https://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee
/etc/apt/sources.list.d/kubernetes.list
```

```
ubuntu@ip-172-31-22-29:~$ sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg https://packages.cloud.google.com/apt/doc/apt-key.gpg
echo "deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/ kubernetes-xenial main
```

#### 4.2. Install kubeadm, kubelet, and kubectl:

```
sudo apt-get update
sudo apt-get install -y kubelet kubeadm kubectl
sudo apt-mark hold kubelet kubeadm kubectl
```

```
ubuntu@ip-172-31-22-29:~$ sudo apt-get update
sudo apt-get install -y kubelet kubeadm kubectl
sudo apt-mark hold kubelet kubeadm kubectl
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
```

### Step 5: Initialize the Kubernetes Cluster on Master Node

On the master node:

```
sudo kubeadm init --pod-network-cidr=10.244.0.0/16
```

```
ubuntu@ip-172-31-22-29:~$ sudo kubeadm init --pod-network-cidr=10.244.0.0/16 --v=5
Found multiple CRI endpoints on the host. Please define which one do you wish to
use by setting the 'criSocket' field in the kubeadm configuration file: unix://
/var/run/containerd/containerd.sock, unix:///var/run/crio/crio.sock
k8s.io/kubernetes/cmd/kubeadm/app/util/runtime.detectCRISocketImpl
    cmd/kubeadm/app/util/runtime/runtime.go:167
k8s.io/kubernetes/cmd/kubeadm/app/util/runtime.DetectCRISocket
    cmd/kubeadm/app/util/runtime/runtime.go:175
k8s.io/kubernetes/cmd/kubeadm/app/util/config.SetNodeRegistrationDynamicDefaults
    cmd/kubeadm/app/util/config/initconfiguration.go:118
k8s.io/kubernetes/cmd/kubeadm/app/util/config.SetInitDynamicDefaults
    cmd/kubeadm/app/util/config/initconfiguration.go:64
k8s.io/kubernetes/cmd/kubeadm/app/util/config.DefaultedInitConfiguration
    cmd/kubeadm/app/util/config/initconfiguration.go:248
k8s.io/kubernetes/cmd/kubeadm/app/util/config.LoadOrDefaultInitConfiguration
    cmd/kubeadm/app/util/config/initconfiguration.go:282
k8s.io/kubernetes/cmd/kubeadm/app/cmd.newInitData
    cmd/kubeadm/app/cmd/init.go:319
k8s.io/kubernetes/cmd/kubeadm/app/cmd.newCmdInit.func3
    cmd/kubeadm/app/cmd/init.go:170
k8s.io/kubernetes/cmd/kubeadm/app/cmd/phases/workflow.(*Runner).InitData
    cmd/kubeadm/app/cmd/phases/workflow/runner.go:183
k8s.io/kubernetes/cmd/kubeadm/app/cmd.newCmdInit.func1
```

5.1. Set up kubectl on the master node:

```
mkdir -p $HOME/.kube  
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config  
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

```
ubuntu@ip-172-31-22-29:~$ sudo kubeadm config images pull  
sudo kubeadm init  
mkdir -p "$HOME"/.kube  
sudo cp -i /etc/kubernetes/admin.conf "$HOME"/.kube/config  
sudo chown "$(id -u):$(id -g)" "$HOME"/.kube/config  
  
# Network Plugin = calico  
kubeadm apply -f https://raw.githubusercontent.com/projectcalico/calico/v3.26.0/manifests/calico.yaml  
  
kubeadm token create --print-join-command --v=5  
Found multiple CRI endpoints on the host. Please define which one do you wish to use by setting the 'criSocket' field in the kubeadm configuration file: unix:///var/run/containerd/containerd.sock, unix:///var/run/crio/crio.sock  
To see the stack trace of this error execute with --v=5 or higher  
Found multiple CRI endpoints on the host. Please define which one do you wish to use by setting the 'criSocket' field in the kubeadm configuration file: unix:///var/run/containerd/containerd.sock, unix:///var/run/crio/crio.sock
```

## Step 6: Install a Pod Network Add-on

To enable communication between pods, install a pod network plugin like Flannel or Calico.

### Install Flannel:

```
kubectl apply -f
```

<https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml>

```
ubuntu@ip-172-31-22-29:~$ kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml --validate=false  
E0913 15:35:04.261458 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
E0913 15:35:04.261902 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
E0913 15:35:04.263424 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
E0913 15:35:04.263795 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
E0913 15:35:04.265840 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
E0913 15:35:04.266524 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
unable to recognize "https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml": Get "http://localhost:8080/api?timeout=32s": dial
```

## Step 7: Join Worker Nodes to the Cluster

On the **worker nodes**, run the command provided by the master node during initialization . It looks something like this:

```
sudo kubeadm join <master-ip>:6443 --token <token> --discovery-token-ca-cert-hash sha256:<hash>
```

```
clusterrolebinding.rbac.authorization.k8s.io/calico-cni-plugin created  
daemonset.apps/calico-node created  
deployment.apps/calico-kube-controllers created  
kubeadm join 172.31.62.216:6443 --token br7fe5.hq2Badbw1mu17ky --discovery-token-ca-cert-hash sha256:2bc469a8d14fbef0f879328d2b416fad  
32b29a8505d3f448b98703ffff3b014d9
```

## Step 8: Verify the Cluster

Once the worker node joins, check the status on the **master node**

```
ubuntu@ip-172-31-45-227:~$ kubectl get nodes
NAME        STATUS   ROLES     AGE      VERSION
ip-172-31-43-211  Ready    <none>    50s     v1.29.0
ip-172-31-45-13   Ready    <none>    34s     v1.29.0
ip-172-31-45-227  Ready    control-plane  5m17s   v1.29.0
ubuntu@ip-172-31-45-227:~$ |
```

# **ADVANCE DEVOPS**

## **EXPERIMENT - 4**

**Name: Kunal Punjabi**

**Class:D15A**

**Roll No:44**

**Aim:** To install Kubectl and execute Kubectl commands to manage the Kubernetes cluster and deploy Your First Kubernetes Application.

### **Step 1: Install Kubectl on Ubuntu**

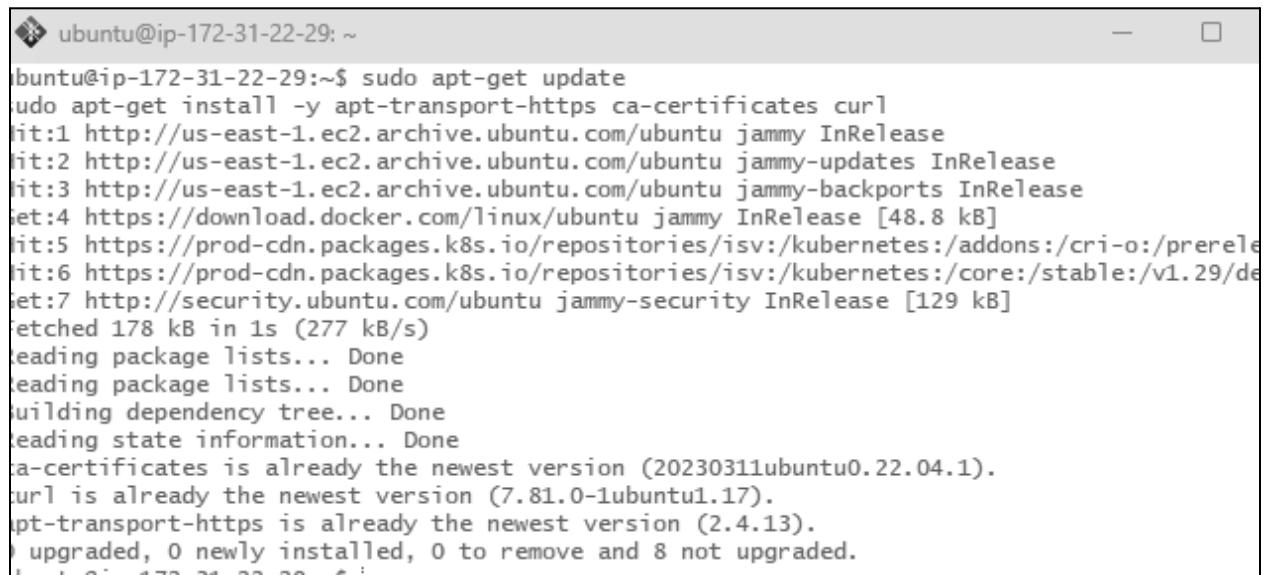
#### **1.1 Add Kubernetes APT repository**

First, add the Kubernetes repository to your system.

##### **1. Install prerequisites:**

```
sudo apt-get update
```

```
sudo apt-get install -y apt-transport-https ca-certificates curl
```



```
ubuntu@ip-172-31-22-29:~$ sudo apt-get update
[sudo] password for ubuntu:
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Get:5 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/addons:/cri-o:/prerelease
Get:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.29/de
Get:7 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Fetched 178 kB in 1s (277 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
curl is already the newest version (7.81.0-1ubuntu1.17).
apt-transport-https is already the newest version (2.4.13).
0 upgraded, 0 newly installed, 0 to remove and 8 not upgraded.
```

##### **2. Add the GPG key for Kubernetes:**

```
sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg
```

<https://packages.cloud.google.com/apt/doc/apt-key.gpg>

```
ubuntu@ip-172-31-22-29:~$ sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg
https://packages.cloud.google.com/apt/doc/apt-key.gpg
```

### 3. Add the Kubernetes repository:

```
echo "deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg]
https://apt.kubernetes.io/ kubernetes-focal main" | sudo tee
/etc/apt/sources.list.d/kubernetes.list
```

```
ubuntu@ip-172-31-22-29:~$ echo "deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring
.gpg] https://apt.kubernetes.io/ kubernetes-focal main" | sudo tee /etc/apt/sources.list.d/ku
ernetes.list
deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/
kubernetes-focal main
```

## 1.2 Install kubectl

Now install kubectl:

```
sudo apt-get update
```

```
sudo apt-get install -y kubectl
```

```
ubuntu@ip-172-31-22-29:~$ sudo apt-get update
sudo apt-get install -y kubectl
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/addons:/cri-o:/prerelease:/main/deb InRelease
Ign:7 https://packages.cloud.google.com/apt kubernetes-focal InRelease
Err:8 https://packages.cloud.google.com/apt kubernetes-focal Release
  404  Not Found [IP: 172.253.62.138 443]
Reading package lists... Done
E: The repository 'https://apt.kubernetes.io kubernetes-focal Release' does not have a Release file.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
kubectl is already the newest version (1.29.0-1.1).
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
```

Verify the installation(extra):

```
kubectl version --client
```

```
ubuntu@ip-172-31-22-29:~$ kubectl version --client
Client Version: v1.29.0
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
```

## **Step 2: Deploying Your Application on Kubernetes**

### **2.1 Set up Kubernetes Cluster**

1. If you haven't already set up a Kubernetes cluster (e.g., with kubeadm), use minikube or any managed Kubernetes service (like EKS, GKE, etc.) to get a cluster running.
2. Once your cluster is ready, verify the nodes:

```
kubectl get nodes
```

```
ubuntu@ip-172-31-45-227:~$ kubectl get nodes
NAME           STATUS   ROLES      AGE    VERSION
ip-172-31-43-211   Ready    <none>    50s   v1.29.0
ip-172-31-45-13   Ready    <none>    34s   v1.29.0
ip-172-31-45-227   Ready    control-plane   5m17s  v1.29.0
ubuntu@ip-172-31-45-227:~$ |
```

### **Step 3: Create the Deployment YAML file**

a) Create the YAML file: Use a text editor to create a file named nginx-deployment.yaml

```
ubuntu@ip-172-31-45-227:~$ nano nginx-deployment.yaml
```

b)Add the Deployment Configuration: Copy and paste the following YAML content into the file. Save and exit the editor (Press Ctrl+X, then Y, and Enter).

```
ubuntu@ip-172-31-45-227: ~          nginx-deployment.yaml
GNU nano 6.2
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.21.3
          ports:
            - containerPort: 80
```

#### Step 4:Create the Service YAML File

a)Create the YAML File: Create another file named nginx-service.yaml

```
ubuntu@ip-172-31-45-227:~$ nano nginx-service.yaml
```

b)Add the Service Configuration: Copy and paste the following YAML content into the file given below.

```
ubuntu@ip-172-31-45-227: ~          nginx-service.yaml *
GNU nano 6.2
apiVersion: v1
kind: Service
metadata:
  name: nginx-service
spec:
  selector:
    app: nginx
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
  type: LoadBalancer
```

## **Step 5:Apply the YAML Files**

a)Deploy the Application: Use kubectl to create the Deployment and Service from the YAML files.

```
ubuntu@ip-172-31-45-227:~$ kubectl apply -f nginx-deployment.yaml
kubectl apply -f nginx-service.yaml
deployment.apps/nginx-deployment created
service/nginx-service created
```

b)Verify the Deployment: Check the status of your Deployment,Pods and Services.

```
ubuntu@ip-172-31-45-227:~$ kubectl get deployments
kubectl get pods
kubectl get services
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   2/2     2           2          40s
NAME           READY   STATUS    RESTARTS   AGE
nginx-deployment-6b4d6fdbf-6k84m   1/1     Running   0          40s
nginx-deployment-6b4d6fdbf-9d8j6   1/1     Running   0          40s
NAME           TYPE      CLUSTER-IP      EXTERNAL-IP   PORT(S)      AGE
kubernetes     ClusterIP   10.96.0.1    <none>        443/TCP     40m
nginx-service   LoadBalancer   10.106.182.152  <pending>    80:32317/TCP  40s
```

## Describe the deployment(Extra)

```
ubuntu@ip-172-31-45-227:~$ kubectl get deployments
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment  1/1     1           1           14h
ubuntu@ip-172-31-45-227:~$ kubectl describe deployment
Name:            nginx-deployment
Namespace:       default
CreationTimestamp: Wed, 11 Sep 2024 17:16:17 +0000
Labels:          <none>
Annotations:    deployment.kubernetes.io/revision: 2
Selector:        app=nginx
Replicas:        1 desired | 1 updated | 1 total | 1 available | 0 unavailable
StrategyType:   RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nginx
  Containers:
    nginx:
      Image:      nginx:latest
      Port:       80/TCP
      Host Port:  0/TCP
      Environment: <none>
      Mounts:
        /usr/share/nginx/html from website-volume (rw)
  Volumes:
    website-volume:
      Type:      ConfigMap (a volume populated by a ConfigMap)
      Name:      nginx-website
      Optional:  false
Conditions:
  Type    Status  Reason
  ----  -----
  Available  True    MinimumReplicasAvailable
  Progressing  True    NewReplicaSetAvailable
OldReplicaSets: nginx-deployment-6b4d6fdbf (0/0 replicas created)
NewReplicaSet:  nginx-deployment-776b8fd845 (1/1 replicas created)
Events:  <none>
```

## Step 6:Ensure Service is Running

6.1 Verify Service: Run the following command to check the services running in your cluster:

```
kubectl get service
```

```
ubuntu@ip-172-31-45-227:~$ kubectl get service
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kubernetes  ClusterIP  10.96.0.1      <none>        443/TCP      16h
nginx     NodePort   10.106.0.176    <none>        80:32618/TCP  76m
nginx-service  NodePort   10.106.182.152  <none>        80:30007/TCP  15h
nginx2     NodePort   10.99.32.156    <none>        80:31421/TCP  8s
```

## Step 7:Forward the Service Port to Your Local Machine

kubectl port-forward allows you to forward a port from your local machine to a port on a service running in the Kubernetes cluster.

1. **Forward the Service Port:** Use the following command to forward a local port to the service's target port.

```
kubectl port-forward service/<service-name> <local-port>:<service-port>
```

```
ubuntu@ip-172-31-45-227:~$ kubectl port-forward service/nginx-service 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80
```

This command will forward local port 8080 on your machine to port 80 of the service nginx-service running inside the cluster.

2. This means port forwarding is now active, and any traffic to localhost:8080 will be routed to the nginx-service on port 80.

```
ubuntu@ip-172-31-45-227:~$ kubectl port-forward service/nginx-service 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80
^Cubuntu@ip-172-31-45-227:~$ kubectl port-forward service/nginx-service 8081:8080
Forwarding from 127.0.0.1:8081 -> 80
Forwarding from [::1]:8081 -> 80
^Cubuntu@ip-172-31-45-227:~$ kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
nginx-deployment-776b8fd845-k9cx4  1/1     Running   0          113m
ubuntu@ip-172-31-45-227:~$ kubectl logs nginx-deployment-776b8fd845-k9cx4
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2024/09/12 06:35:51 [notice] 1#1: using the "epoll" event method
2024/09/12 06:35:51 [notice] 1#1: nginx/1.27.1
2024/09/12 06:35:51 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2024/09/12 06:35:51 [notice] 1#1: OS: Linux 6.5.0-1022-aws
2024/09/12 06:35:51 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2024/09/12 06:35:51 [notice] 1#1: start worker processes
2024/09/12 06:35:51 [notice] 1#1: start worker process 24
2024/09/12 06:35:51 [notice] 1#1: start worker process 25
```

## Step 8: Access the Application Locally

1. **Open a Web Browser:** Now open your web browser and go to the following URL:

http://localhost:8080

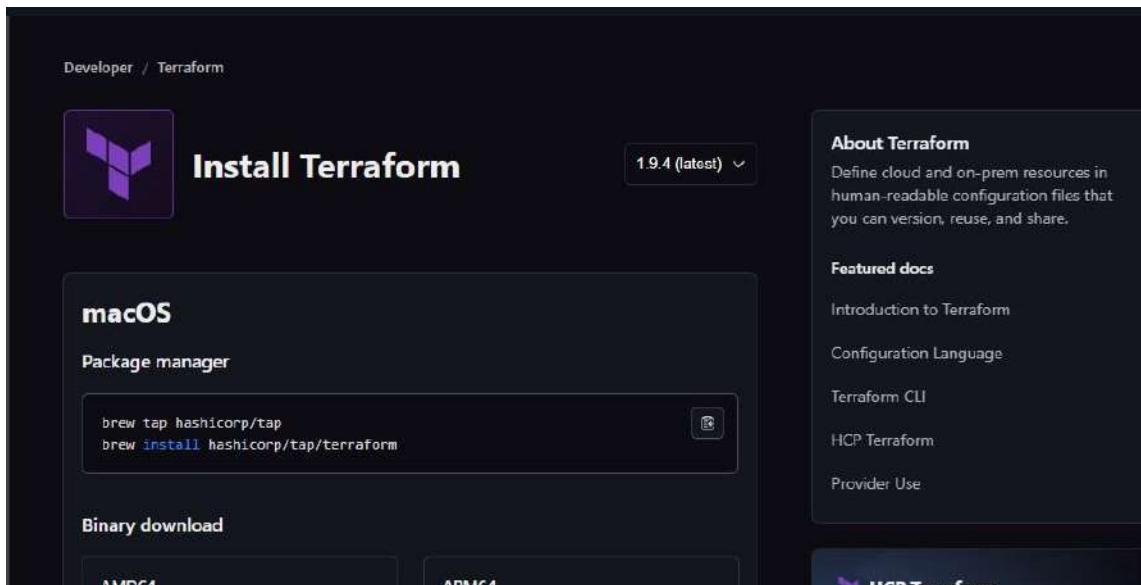
You should see the application (in this case, Nginx) that you have deployed running in the Kubernetes cluster, served locally via port 8080.

In case the port 8080 is unavailable, try using a different port like 8081

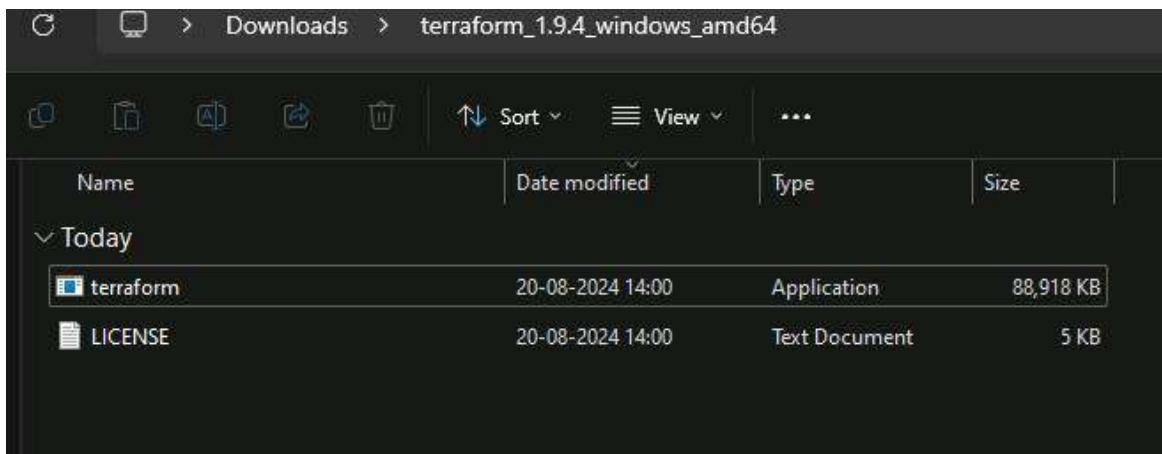


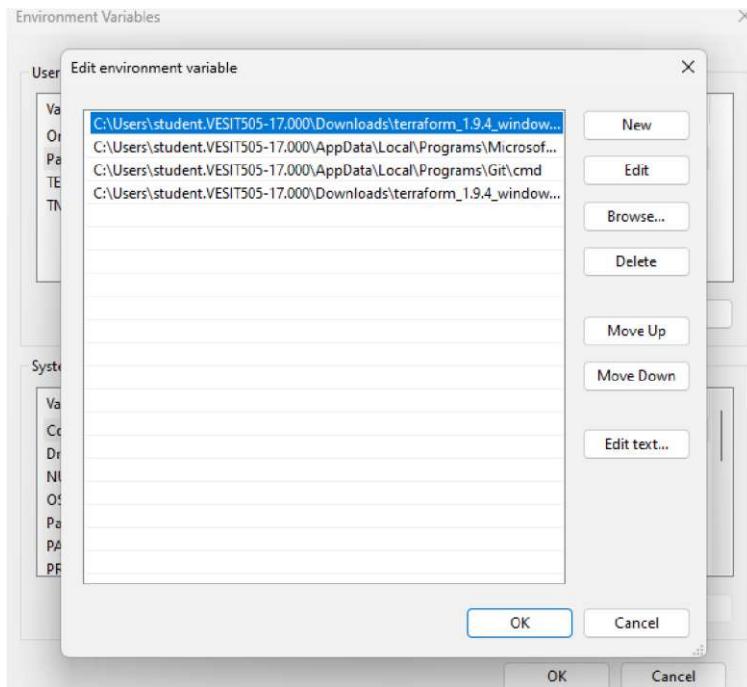
## AdvDevops Experiment 5

**AIM:** Installation and configuration of terraform on Windows



The screenshot shows the Terraform website's "Install Terraform" page. At the top, there's a purple logo and the text "Install Terraform". Below it, a dropdown menu shows "1.9.4 (latest)". To the right, there's a sidebar titled "About Terraform" with a brief description: "Define cloud and on-prem resources in human-readable configuration files that you can version, reuse, and share." Below that is a "Featured docs" section with links to "Introduction to Terraform", "Configuration Language", "Terraform CLI", "HCP Terraform", and "Provider Use". The main content area is titled "macOS" and contains two sections: "Package manager" and "Binary download". The "Package manager" section shows a terminal command: "brew tap hashicorp/tap" followed by "brew install hashicorp/tap/terraform". The "Binary download" section provides links for "AMD64" and "ARM64".





Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

```
PS C:\Users\student.VESIT505-17.000\Downloads\terraform_1.9.4_windows_amd64> terraform
Usage: terraform [global options] <subcommand> [args]
```

The available commands for execution are listed below.  
The primary workflow commands are given first, followed by  
less common or more advanced commands.

Main commands:

init	Prepare your working directory for other commands
validate	Check whether the configuration is valid
plan	Show changes required by the current configuration
apply	Create or update infrastructure
destroy	Destroy previously-created infrastructure

All other commands:

console	Try Terraform expressions at an interactive command prompt
fmt	Reformat your configuration in the standard style
force-unlock	Release a stuck lock on the current workspace
get	Install or upgrade remote Terraform modules
graph	Generate a Graphviz graph of the steps in an operation
import	Associate existing infrastructure with a Terraform resource
login	Obtain and save credentials for a remote host
logout	Remove locally-stored credentials for a remote host
metadata	Metadata related commands
output	Show output values from your root module
providers	Show the providers required for this configuration
refresh	Update the state to match remote systems
show	Show the current state or a saved plan
state	Advanced state management
taint	Mark a resource instance as not fully functional
test	Execute integration tests for Terraform modules
untaint	Remove the 'tainted' state from a resource instance

Kunal Punjabi D15A '44'

```
PS C:\Users\student.VESIT505-17.000\Downloads\terraform_1.9.4_windows_amd64> terraform --version
Terraform v1.9.4
on windows_amd64
PS C:\Users\student.VESIT505-17.000\Downloads\terraform_1.9.4_windows_amd64> |
```

## ADVANCE DEVOPS EXP 6

**Aim :** To Build, change, and destroy AWS / GCP /Microsoft Azure/ DigitalOcean infrastructure Using Terraform. (S3 bucket or Docker) fdp.

### Part A: Creating docker image using terraform

#### Step 1:Check Docker functionality

```
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.

C:\Users\student>docker

Usage: docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Common Commands:
  run      Create and run a new container from an image
  exec     Execute a command in a running container
  ps       List containers
  build    Build an image from a Dockerfile
  pull     Download an image from a registry
  push     Upload an image to a registry
  images   List images
  login    Log in to a registry
  logout   Log out from a registry
  search   Search Docker Hub for images
  version  Show the Docker version information
  info     Display system-wide information

Management Commands:
  builder  Manage builds
  buildx*  Docker Buildx
  checkpoint  Manage checkpoints
  compose*  Docker Compose
  container  Manage containers
  context   Manage contexts
  debug*   Get a shell into any image or container
  desktop* Docker Desktop commands (Alpha)
  dev*     Docker Dev Environments
  extension* Manages Docker extensions
  feedback* Provide feedback, right in your terminal!
```

Check for the docker version with the following command.

```
C:\Users\student>docker --version
Docker version 27.1.1, build 6312585

C:\Users\student>
```

Create a folder named ‘Terraform Scripts’ in which we save our different typesof scripts which will be further used in this experiment.

## Step 2:

Creating a new folder named ‘Docker’ in the ‘TerraformScripts’ folder.

Creating a new docker.tf file using Atom editor and write the following contents into.

This will create a Ubuntu Linux container

```
"# docker.tf  X
  docker.tf
  1  terraform {
  2    required_providers {
  3      docker = {
  4        source  = "kreuzwerker/docker"
  5        version = "2.21.0"
  6      }
  7    }
  8  }
  9
 10 provider "docker" {
 11   host = "npipe:///./pipe/docker_engine"
 12 }
 13
 14 # Pull the image
 15 resource "docker_image" "ubuntu" {
 16   name = "ubuntu:latest"
 17 }
 18
 19 # Create a container
 20 resource "docker_container" "foo" {
 21   image = docker_image.ubuntu.image_id
 22   name  = "foo"
 23   command = ["sleep", "3600"]
 24 }
```

### Step 3: Execute Terraform Init command to initialize the resources

```
● PS C:\Users\Admin\TerraformScripts> cd Docker
● PS C:\Users\Admin\TerraformScripts\ Docker> terraform init
Initializing the backend...
Initializing provider plugins...
  - Finding kreuzwerker/docker versions matching "2.21.0"...
  - Installing kreuzwerker/docker v2.21.0...
○ - Installed kreuzwerker/docker v2.21.0 (self-signed, key ID BD080C4571C6104C)
  Partner and community providers are signed by their developers.
  If you'd like to know more about provider signing, you can read about it here:
  https://www.terraform.io/docs/cli/plugins/signing.html
  Terraform has created a lock file .terraform.lock.hcl to record the provider
  selections it made above. Include this file in your version control repository
  so that Terraform can guarantee to make the same selections by default when
  you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

## Step 4: Execute Terraform plan to see the available resources

```
PS C:\Users\Admin\TerraformScripts\Docker> terraform plan

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the fol
+ create

Terraform will perform the following actions:
```

```
# docker_container.foo will be created
+ resource "docker_container" "foo" {
    + attach          = false
    + bridge          = (known after apply)
    + command         = [
        + "sleep",
        + "3600",
    ]
    + container_logs = (known after apply)
    + entrypoint      = (known after apply)
    + env             = (known after apply)
    + exit_code       = (known after apply)
    + gateway         = (known after apply)
    + hostname        = (known after apply)
    + id              = (known after apply)
    + image           = (known after apply)
    + init            = (known after apply)
    + ip_address      = (known after apply)
    + ip_prefix_length = (known after apply)
    + ipc_mode        = (known after apply)
    + log_driver      = (known after apply)
    + logs            = false
    + must_run        = true
    + name            = "foo"
    + network_data   = (known after apply)
    + read_only       = false
    + remove_volumes = true
    + restart         = "no"
    + rm              = false
}
```

```
+ runtime          = (known after apply)
+ security_opts    = (known after apply)
+ shm_size         = (known after apply)
+ start            = true
+ stdio_open       = false
+ stop_signal      = (known after apply)
+ stop_timeout     = (known after apply)
+ tty               = false

+ healthcheck (known after apply)

+ labels (known after apply)
}

# docker_image.ubuntu will be created
+ resource "docker_image" "ubuntu" {
    + id              = (known after apply)
    + image_id        = (known after apply)
    + latest          = (known after apply)
    + name            = "ubuntu:latest"
    + output          = (known after apply)
    + repo_digest     = (known after apply)
}
```

Plan: 2 to add, 0 to change, 0 to destroy.

**Step 5:** Execute Terraform apply to apply the configuration, which will automatically create and run the Ubuntu Linux container based on our configuration. Using command : “**terraform apply**”

```
● PS C:\Users\Admin\TerraformScripts\Docker> terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

# docker_container.foo will be created
+ resource "docker_container" "foo" {
    + attach           = false
    + bridge           = (known after apply)
    + command          = [
        + "sleep",
        + "3600",
    ]
    + container_logs   = (known after apply)
    + entrypoint       = (known after apply)
    + env              = (known after apply)
    + exit_code         = (known after apply)
    + gateway          = (known after apply)
    + hostname         = (known after apply)
    + id               = (known after apply)
    + image             = (known after apply)
    + init              = (known after apply)
    + ip_address        = (known after apply)
    + ip_prefix_length = (known after apply)
    + ipc_mode          = (known after apply)
    + log_driver        = (known after apply)
    + logs              = false
    + must_run          = true
    + name              = "foo"
    + network_data      = (known after apply)
    + read_only          = false
}
```

```
+ remove_volumes  = true
+ restart         = "no"
+ rm              = false
+ runtime         = (known after apply)
+ security_opts   = (known after apply)
+ shm_size         = (known after apply)
+ start            = true
+ stdin_open       = false
+ stop_signal      = (known after apply)
+ stop_timeout     = (known after apply)
+ tty              = false

+ healthcheck (known after apply)

+ labels (known after apply)
}

# docker_image.ubuntu will be created
+ resource "docker_image" "ubuntu" {
    + id               = (known after apply)
    + image_id         = (known after apply)
    + latest           = (known after apply)
    + name              = "ubuntu:latest"
    + output            = (known after apply)
    + repo_digest      = (known after apply)
}
```

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?  
Terraform will perform the actions described above.  
Only 'yes' will be accepted to approve.

Enter a value: yes

```
● docker_image.ubuntu: Creating...
● docker_image.ubuntu: Creation complete after 9s [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
● docker_container.foo: Creating...
● docker_container.foo: Creation complete after 2s [id=01adf07e5918931fee9b90073726a03671037923dd92032ce0e15bbb764a6f24]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```

Before Executing Apply step:

```
● PS C:\Users\Admin\TerraformScripts\Docker> docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
ubuntu              latest   edbfe74c41f8  3 weeks ago  78.1MB
```

After Executing Apply step:

```
● PS C:\Users\Admin\TerraformScripts\Docker> docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
ubuntu              latest   edbfe74c41f8  3 weeks ago  78.1MB
```

**Step 6:** Execute Terraform destroy to delete the configuration, which will automatically delete the Ubuntu Container.

```
● PS C:\Users\Admin\TerraformScripts\Docker> terraform destroy
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_container.foo: Refreshing state... [id=01adf07e5918931fee9b90073726a03671037923dd92032ce0e15bbb764a6f24]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
- destroy

Terraform will perform the following actions:

# docker_container.foo will be destroyed
- resource "docker_container" "foo" {
    - attach           = false -> null
    - command          = [
        - "sleep",
        - "3600",
    ] -> null
    - cpu_shares       = 0 -> null
    - dns               = [] -> null
    - dns_opts          = [] -> null
    - dns_search         = [] -> null
    - entrypoint         = [] -> null
    - env               = [] -> null
    - gateway           = "172.17.0.1" -> null
    - group_add          = [] -> null
    - hostname          = "01adf07e5918" -> null
    - id               = "01adf07e5918931fee9b90073726a03671037923dd92032ce0e15bbb764a6f24" -> null
    - image             = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
    - init              = false -> null
    - ip_address         = "172.17.0.2" -> null
    - ip_prefix_length  = 16 -> null
    - ipc_mode          = "private" -> null
    - links              = [] -> null
    - log_driver         = "json-file" -> null
    - log_opts            = {} -> null
    - logs              = false -> null
    - max_retry_count    = 0 -> null
}
```

```

- memory          = 0 -> null
- memory_swap    = 0 -> null
- must_run        = true -> null
- name            = "foo" -> null
- network_data    = [
  {
    - gateway           = "172.17.0.1"
    - global_ipv6_prefix_length = 0
    - ip_address        = "172.17.0.2"
    - ip_prefix_length   = 16
    - network_name       = "bridge"
    # (2 unchanged attributes hidden)
  },
  ] -> null
- network_mode     = "default" -> null
- privileged       = false -> null
- publish_all_ports = false -> null
- read_only         = false -> null
- remove_volumes   = true -> null
- restart          = "no" -> null
- rm                = false -> null
- runtime          = "runc" -> null
- security_opts    = [] -> null
- shm_size          = 64 -> null
- start             = true -> null
- stdin_open        = false -> null
- stop_timeout      = 0 -> null
- storage_opts     = {} -> null
- sysctls           = {} -> null
- tmpfs             = {} -> null
- tty               = false -> null
# (8 unchanged attributes hidden)
}

```

```

# docker_image.ubuntu will be destroyed
- resource "docker_image" "ubuntu" {
  - id      = "sha256:edbf74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest" -> null
  - image_id = "sha256:edbf74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
  - latest   = "sha256:edbf74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
  - name     = "ubuntu:latest" -> null
  - repo_digest = "ubuntu@sha256:8a37d68f4f73ebf3d4efafbcf66379bf3728902a8038616800f04e34a9ab63ee" -> null
}


```

Plan: 0 to add, 0 to change, 2 to destroy.

**Do you really want to destroy all resources?**

Terraform will destroy all your managed infrastructure, as shown above.  
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

```

docker_container.foo: Destroying... [id=01adf07e5918931fee9b90073726a03671037923dd92032ce0e15bbb764a6f24]
docker_container.foo: Destruction complete after 0s
docker_image.ubuntu: Destroying... [id=sha256:edbf74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_image.ubuntu: Destruction complete after 1s

```

Destroy complete! Resources: 2 destroyed.

## Docker images After Executing Destroy step

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
------------	-----	----------	---------	------

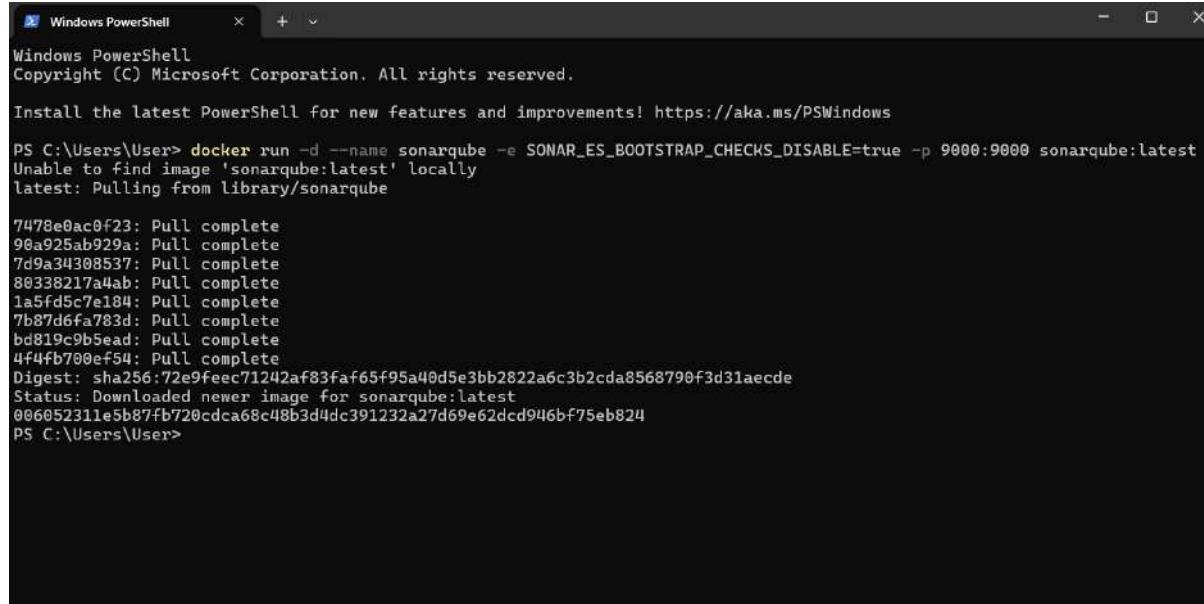
## EXPERIMENT 7

Aim: To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

Step 1: Open Windows PowerShell and run the following command –

```
docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
```

WARNING: Run the following command only once

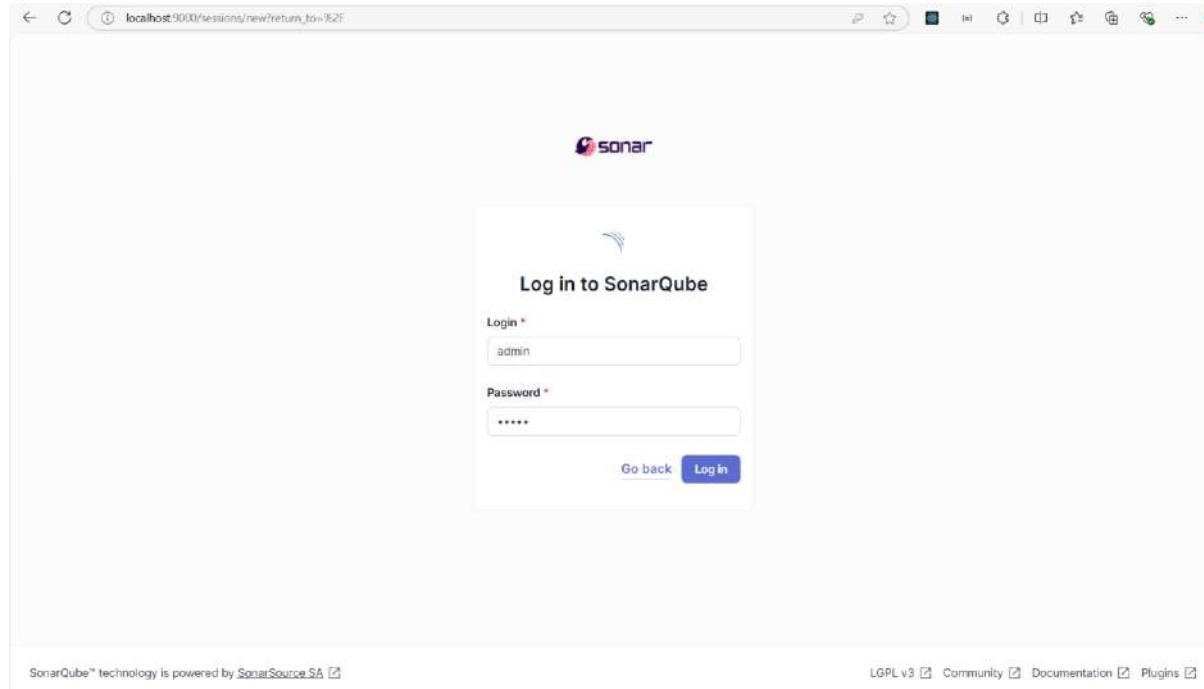


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\User> docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
Unable to find image 'sonarqube:latest' locally
latest: Pulling from library/sonarqube
7478s0ac0f23: Pull complete
90a925ab929a: Pull complete
7d9a34308537: Pull complete
88338217a4ab: Pull complete
1a5fd5c7e184: Pull complete
7b87d6fa783d: Pull complete
bd819c9b5ead: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
Status: Downloaded newer image for sonarqube:latest
006052311e5b87fb720cdca68c48b3d4dc391232a27d69e62cd946bf75eb824
PS C:\Users\User>
```

Step 2: Visit <http://localhost:9000/> to open SonarQube. Login with username: admin and password: admin.



Step 3: Click on create a local project and name the project as sonarqube-test and key as sonarqube-test and click on the next button. In the next step select the “Use the global setting” option and click on create project.

localhost:9000/projects/create

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More Q

### How do you want to create your project?

Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)? Create your project from your favorite DevOps platform.

First, you need to set up a DevOps platform configuration.

Import from Azure DevOps Setup  
 Import from Bitbucket Cloud Setup  
 Import from Bitbucket Server Setup  
  
 Import from GitHub Setup  
 Import from GitLab Setup

Are you just testing or have an advanced use-case? Create a local project.

Create a local project

**⚠️ Embedded database should be used for evaluation purposes only**  
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

SonarQube™ technology is powered by SonarSource SA Community Edition v10.6 (92116) ACTIVE LGPL v3 Community Documentation Plugins Web API

1 of 2

### Create a local project

Project display name sonarqube-test

Project key sonarqube-test

Main branch name \* main

The name of your project's default branch [Learn More](#)

Cancel Next

**⚠️ Embedded database should be used for evaluation purposes only**  
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

SonarQube™ technology is powered by SonarSource SA Community Edition v10.6 (92116) ACTIVE LGPL v3 Community Documentation Plugins Web API

2 of 2

## Set up project for Clean as You Code

The new code definition sets which part of your code will be considered new code. This helps you focus attention on the most recent changes to your project, enabling you to follow the Clean as You Code methodology. Learn more: [Defining New Code](#)

Choose the baseline for new code for this project

Use the global setting

Previous version

Any code that has changed since the previous version is considered new code.

Recommended for projects following regular versions or releases.

Define a specific setting for this project

Previous version

Any code that has changed since the previous version is considered new code.

Recommended for projects following regular versions or releases.

Number of days

Any code that has changed in the last x days is considered new code. If no action is taken on a new issue after x days, this issue will become part of the overall code.

Recommended for projects following continuous delivery.

Reference branch

Choose a branch as the baseline for the new code.

Step 4: Open Jenkins using <http://localhost:8080/> and select Manage Jenkins, then select the Plugins and select available plugins from sidebar and search for SonarQube Scanner and install it. Once installed you can view the installed plugin in installed plugins section in sidebar.

The screenshot shows the Jenkins Manage Jenkins page. The left sidebar includes links for New Item, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins (which is selected), My Views, and Build Queue. The main area displays system status (Build Executor Status: 1 idle, 2 idle, 1 Slave (offline)), System Configuration (with links to System, Tools, Nodes, Clouds, and Appearance), and a warning about Jenkins 2.452.3 core and libraries. A prominent message at the top right says "New version of Jenkins (2.462.2) is available for download (changelog)." Below it is a button to "Upgrade Automatically".

Dashboard > Manage Jenkins > Plugins

The screenshot shows the Jenkins 'Plugins' management interface. A search bar at the top contains the text 'sonar'. On the left, a sidebar lists 'Updates', 'Available plugins', 'Installed plugins' (which is selected and highlighted in blue), and 'Advanced settings'. In the main area, a table lists the 'SonarQube Scanner for Jenkins' plugin, version 2.17.2. The table includes columns for 'Name', 'Version', 'Description', 'Status' (Enabled), and 'Actions' (checkboxes for 'Install' and 'Uninstall'). At the bottom right of the main area, it says 'REST API Jenkins 2.452.3'.

Step 5: Select Manage Jenkins, then select the System and then scroll down to SonarQube Server. Name the server as sonarqube and set the server url as <http://localhost:9000/> then click on save.

Dashboard > Manage Jenkins

The screenshot shows the Jenkins 'Manage Jenkins' configuration page. The left sidebar includes links for 'New Item', 'Build History', 'Project Relationship', 'Check File Fingerprint', 'Manage Jenkins' (selected and highlighted in blue), and 'My Views'. Under 'Manage Jenkins', there are sections for 'Build Queue', 'Build Executor Status' (with a table showing 1 idle, 2 idle, and 1 Slave (offline)), and 'System Configuration' (with links for System, Tools, Nodes, Clouds, Plugins, and Appearance). A central message box indicates a new version of Jenkins (2.462.2) is available for download, with options to 'Or Upgrade Automatically' or 'Manage' the update. Another message box about multiple security vulnerabilities in Jenkins 2.470 and earlier, LTS 2.452.3, and earlier, with a link to fix the issue. The URL 'localhost:8080/manage/configure' is visible at the bottom left.

**SonarQube servers**

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

Environment variables

**SonarQube installations**

List of SonarQube installations:

Name	<input type="text" value="sonarqube"/>
Server URL	Default is <a href="http://localhost:9000">http://localhost:9000</a>
<input type="text" value="http://localhost:9000"/>	
Server authentication token	SonarQube authentication token. Mandatory when anonymous access is disabled.
<input type="text" value="- none -"/> <a href="#">+ Add +</a>	
<a href="#">Advanced</a>	

[Save](#)[Apply](#)

Step 6: Go to Jenkins Dashboard and select Manage Jenkins, then select the Tools and then scroll down to SonarQube Scanner installations. Name the sonarqube scanner as sonarqubescanner and select install automatically then click on save.

[+ New Item](#)[Build History](#)[Project Relationship](#)[Check File Fingerprint](#)[Manage Jenkins](#)[My Views](#)[Build Queue](#)[Build Executor Status](#)[Build In Node](#)

1 Idle

2 Idle

[Slave1](#)**Manage Jenkins**[Search settings](#)[/](#)

New version of Jenkins (2.462.2) is available for download ([changeslog](#)).

[Or Upgrade Automatically](#)

Building on the built-in node can be a security issue. You should set the number of executors on the built-in node to 0. See the [documentation](#).

[Manage](#)[Dismiss](#)

Warnings have been published for the following currently installed components:

[Configure which of these warnings are shown](#)

Jenkins 2.452.3 core and libraries:

[Multiple security vulnerabilities in Jenkins 2.470 and earlier, LTS 2.452.3 and earlier](#)

A fix for this issue is available. Update Jenkins now.

**System Configuration**[!\[\]\(b51766659593a870be97c2e9ade41f18\_img.jpg\) System](#)

Configure global settings and paths.

[!\[\]\(a0eafdbb980ce723d1bfbc2f84b9cf35\_img.jpg\) Tools](#)

Configure tools, their locations and automatic installers.

[!\[\]\(fdfd3bca5d414f27790c3ef1a28251f0\_img.jpg\) Plugins](#)

Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

[!\[\]\(5a6f487eeb3aedc7006648f6b63d3f8b\_img.jpg\) Nodes](#)

Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

[!\[\]\(f395fc3421d5dd5822afb01169b0e8be\_img.jpg\) Clouds](#)

Add, remove, and configure cloud instances to provision agents on-demand.

[!\[\]\(300e119c4213a466599a160e162647c6\_img.jpg\) Appearance](#)

Configure the look and feel of Jenkins.

## SonarScanner for MSBuild installations

[Add SonarScanner for MSBuild](#)

## SonarQube Scanner installations

[Add SonarQube Scanner](#)

## SonarQube Scanner

Name

sonarqubescanner

 Install automatically ?

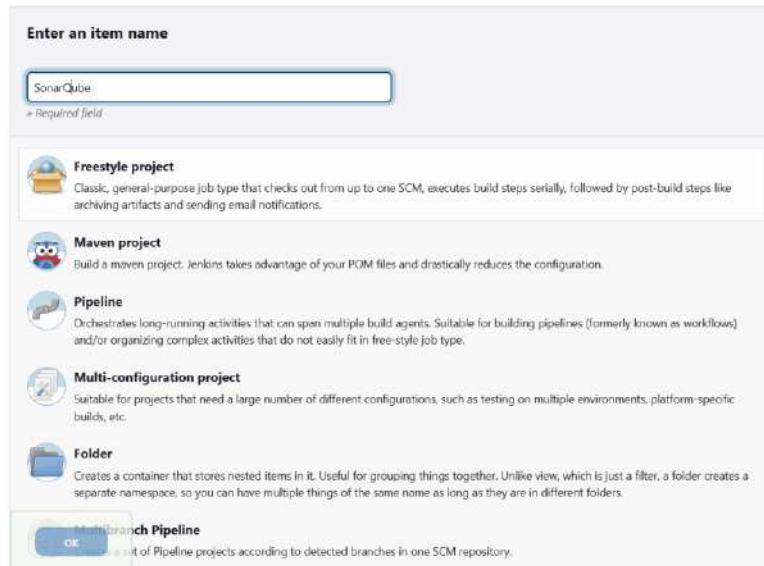
## Install from Maven Central

Version

SonarQube Scanner 6.2.0.4584

[Add Installer](#)[Save](#)[Apply](#)

Step 7: Go to Jenkins dashboard and click on New Item and select Freestyle project and name it as SonarQube and then click on ok.



Step 8: For configuration, Select git and paste the following git repository in the repository url.

[https://github.com/shazforiot/MSBuild\\_firstproject](https://github.com/shazforiot/MSBuild_firstproject)

This is a simple Hello world project

Dashboard > SonarQube > Configuration

### Configure

#### Source Code Management

General

Source Code Management **Git**

Build Triggers

Build Environment

Build Steps

Post-build Actions

None

Git

Repositories

Repository URL: https://github.com/shazforce/MSBuild\_firstproject.git

Credentials: none

Add

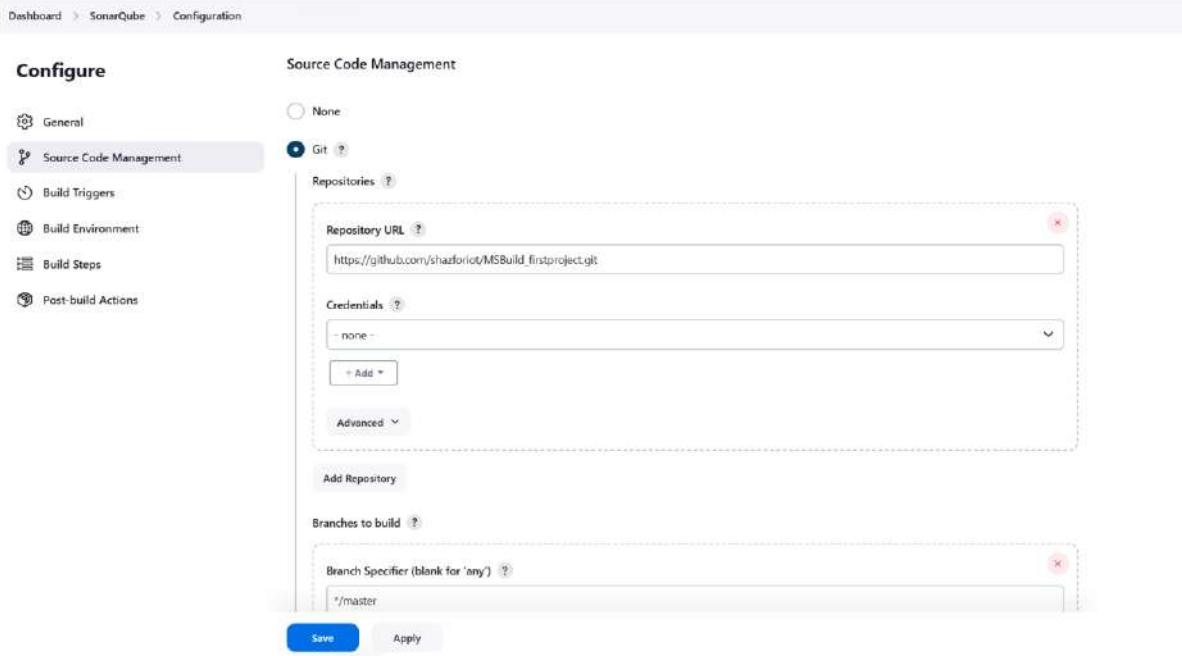
Advanced

Add Repository

Branches to build

Branch Specifier (blank for 'any'): \*/master

Save Apply



Step 9: Under the Build steps select “Execute SonarQube Scanner” option and under Analysis Properties write the following -

sonar.projectKey=sonarqube-test

sonar.login=admin

sonar.password=sonarqube

sonar.hosturl=http://sonarqube:9000

Then click on the save button.

Dashboard > SonarQube > Configuration

### Configure

#### Build Steps

General

Source Code Management

Build Triggers

Build Environment

Build Steps **Execute SonarQube Scanner**

Path to project properties

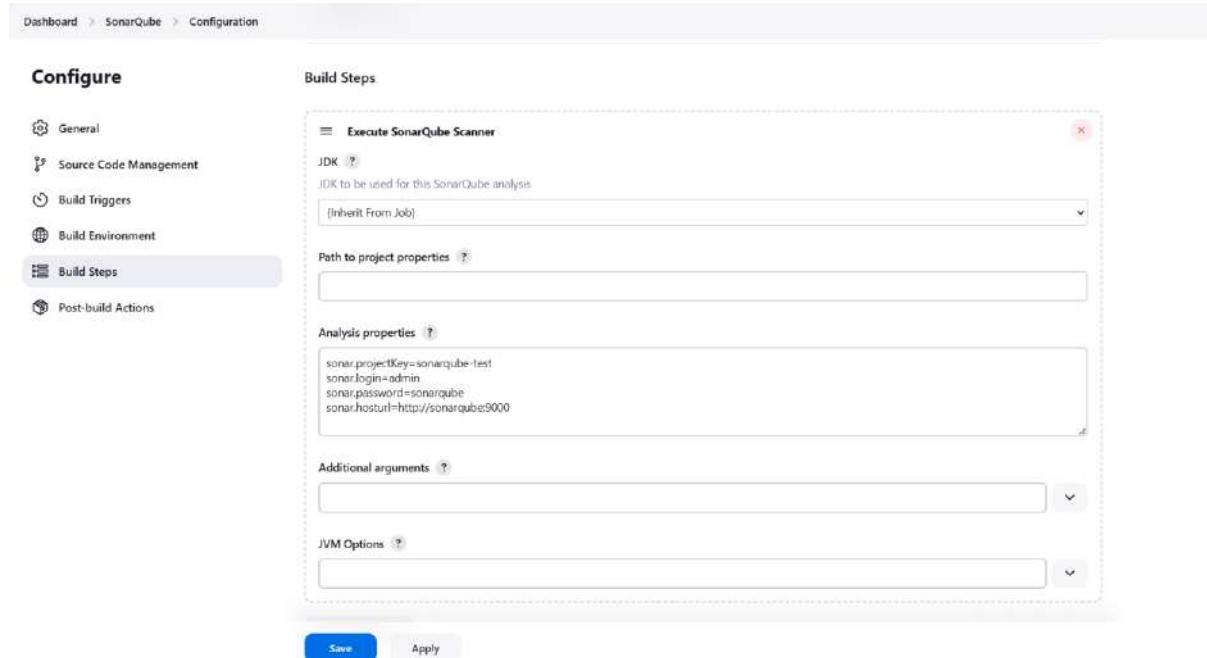
Analysis properties

```
sonar.projectKey=sonarqube-test
sonar.login=admin
sonar.password=sonarqube
sonar.hosturl=http://sonarqube:9000
```

Additional arguments

JVM Options

Save Apply



Step 10: Visit <http://localhost:9000/admin/permissions> and select the Users tab and for Administrator select the checkbox Execute Analysis.

Step 11: Now, come back to Jenkins and click on Build Now. The build is success.

**Console Output**

```

Started by user Anuprita Mhapankar
Running as SYSTEM
Building on the built-in node in workspace C:\ProgramData\Jenkins\jenkins\workspace\SonarQube
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\jenkins\workspace\SonarQube\.git # timeout=10
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/shafiqist/M5Build_firstproject.git # timeout=10
Fetching upstream changes from https://github.com/shafiqist/M5Build_firstproject.git
> git.exe --version # timeout=10
> git.exe --version # 'git version 2.41.0.windows.3'
> git.exe fetch --tags --progress -- https://github.com/shafiqist/M5Build_firstproject.git +refs/heads/*:refs/remotes/origin/*
timeout=10
> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision f2bc042c04c6e72427c380bcae6d6fee7b49adf (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f f2bc042c04c6e72427c380bcae6d6fee7b49adf # timeout=10
Commit message: "updated"
> git.exe rev-list --no-walk f2bc042c04c6e72427c380bcae6d6fee7b49adf # timeout=10
[SonarQube] $ C:\ProgramData\Jenkins\jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqubescanner\bin\sonar-scanner.bat -
-Dsonar.host.url=http://localhost:9000 -Dsonar.projectKey=sonarqube-test -Dsonar.login=admin -Dsonar.hostUrl=http://sonarqube:9000 -
Dsonar.password=sonarqube -Dsonar.projectBaseDir=C:\ProgramData\Jenkins\jenkins\workspace\SonarQube
18:40:04.147 INFO Scanner configuration file:
C:\ProgramData\Jenkins\jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqubescanner\bin..\conf\sonar-scanner.properties
18:40:04.152 INFO Project root configuration file: NONE
18:40:04.175 INFO SonarScanner CLI 6.2.0.4584
18:40:04.177 INFO Java 21.0.4 Eclipse Adoptium (64-bit)
18:40:04.184 INFO Windows 11 10.0 amd64

```

Dashboard > SonarQube > #4 > Console Output

```

18:40:41.286 INFO ----- Run sensors on project
18:40:41.484 INFO Sensor C# [csharp]
18:40:41.485 WARN Your project contains C# files which cannot be analyzed with the scanner you are using. To analyze C# or VB.NET, you must use the SonarScanner for .NET 5.x or higher, see https://redirect.sonarsource.com/doc/install-configure-scanner-msbuild.html
18:40:41.485 INFO Sensor C# [csharp] (done) | time=2ms
18:40:41.486 INFO Sensor Analysis Warnings import [csharp]
18:40:41.488 INFO Sensor Analysis Warnings import [csharp] (done) | time=2ms
18:40:41.488 INFO Sensor C# File Caching Sensor [csharp]
18:40:41.489 WARN Incremental PR analysis: Could not determine common base path, cache will not be computed. Consider setting 'sonar.projectBaseDir' property.
18:40:41.490 INFO Sensor C# File Caching Sensor [csharp] (done) | time=1ms
18:40:41.491 INFO Sensor Zero Coverage Sensor
18:40:41.508 INFO Sensor Zero Coverage Sensor (done) | time=19ms
18:40:41.514 INFO SCM Publisher SCM provider for this project is: git
18:40:41.517 INFO SCM Publisher 4 source files to be analyzed
18:40:42.309 INFO SCM Publisher 4/4 source files have been analyzed (done) | time=791ms
18:40:42.317 INFO CPD Executor Calculating CPD for 0 files
18:40:42.318 INFO CPD Executor CPD calculation finished (done) | time=0ms
18:40:42.326 INFO SCM revision ID 'f2bc042c04c6e72427c380bcae6d6fee7b49adf'
18:40:42.522 INFO Analysis report generated in 181ms, dir size=201.1 kB
18:40:42.588 INFO Analysis report compressed in 63ms, zip size=22.3 kB
18:40:42.876 INFO Analysis report uploaded in 283ms
18:40:42.880 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=sonarqube-test
18:40:42.881 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
18:40:42.882 INFO More about the report processing at http://localhost:9000/api/ce/task?id=d10eb30d-ebb2-b564-0aa4ea71bf2
18:40:42.916 INFO Analysis total time: 25.189 s
18:40:42.926 INFO SonarScanner Engine completed successfully
18:40:43.027 INFO EXECUTION SUCCESS
18:40:43.029 INFO Total time: 38.885s
Finished: SUCCESS

```

[REST API](#) [Jenkins 2.452.3](#)

Step 12: Visit the following URL to see the result - <http://localhost:9000/dashboard?id=sonarqube-test&codeScope=overall>

localhost:9000/dashboard?id=sonarqube-test&codeScope=overall

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More

main / main

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

main Version not provided Set as homepage Last analysis 14 minutes ago

Quality Gate \* Passed

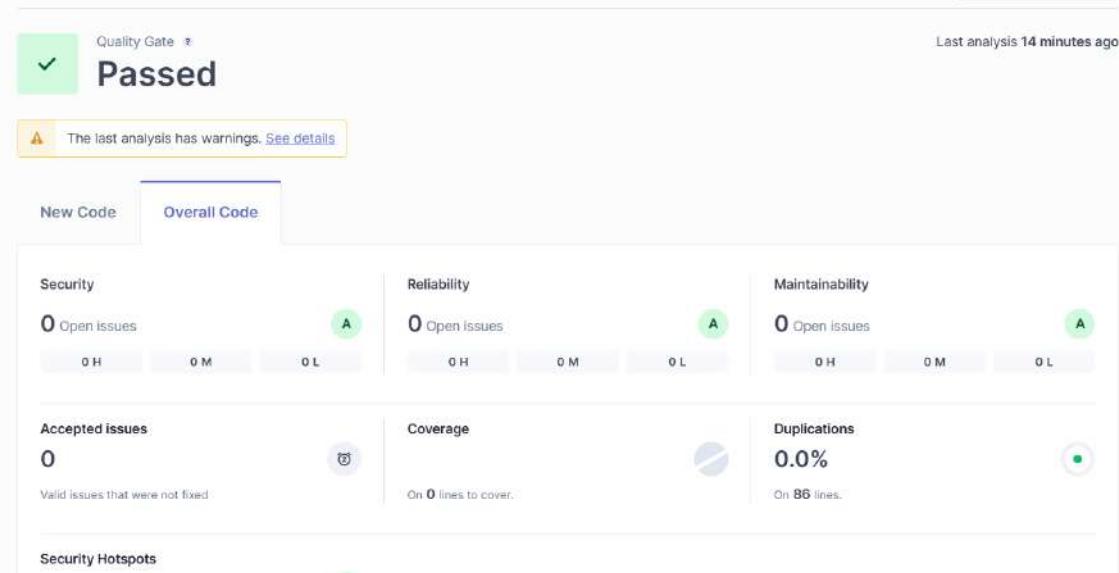
The last analysis has warnings. See details

New Code Overall Code

Security	Reliability	Maintainability
0 Open issues 0 H 0 M 0 L	0 Open issues 0 H 0 M 0 L	0 Open issues 0 H 0 M 0 L

Accepted issues	Coverage	Duplications
0 Valid issues that were not fixed	On 0 lines to cover.	0.0% On 86 lines.

Security Hotspots



## Advance DevOps

### EXPERIMENT -8

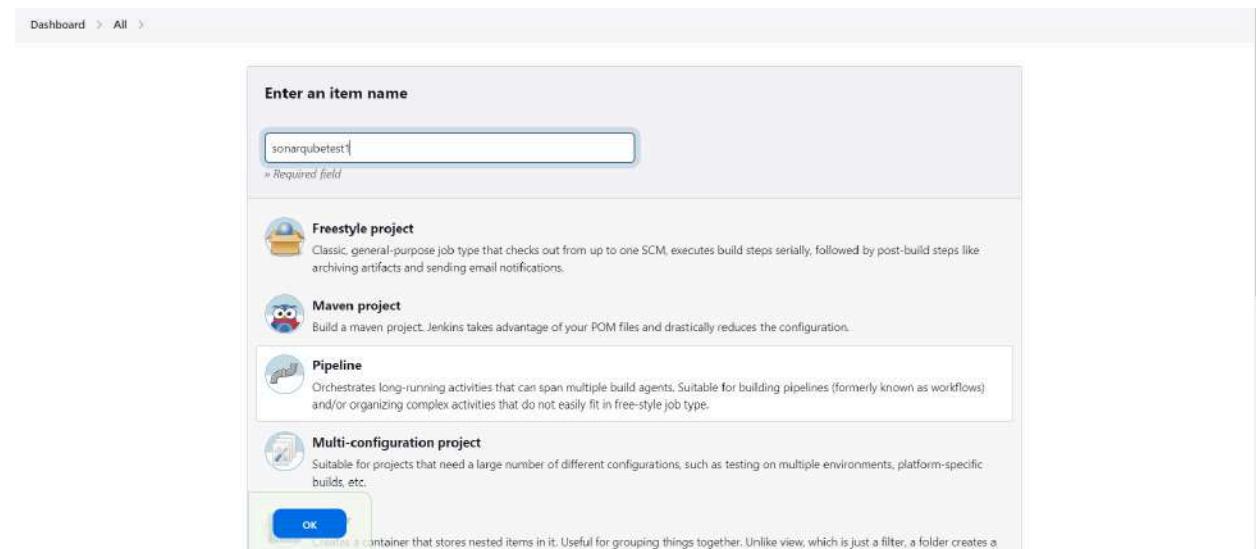
**Kunal Punjabi**  
**D15A**  
**ROLL NO - 44**

Step 1: Open Windows PowerShell and run the following command – docker run -d --name sonarqube-test1 -e SONAR\_ES\_BOOTSTRAP\_CHECKS\_DISABLE=true -p 9000:9000 sonarqube:latest

```
erShell does not load commands from the current location by default. If you trust this command, instead type: ".\sonar-s
canner.bat". See "get-help about_Command_Precedence" for more details.
PS C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin> .\sonar-scanner.bat
11:02:02.120 INFO Scanner configuration file: C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin\..\conf\sonar-s
canner.properties
11:02:02.124 INFO Project root configuration file: NONE
11:02:02.140 INFO SonarScanner CLI 6.2.0.4584
11:02:02.142 INFO Java 17.0.12 Eclipse Adoptium (64-bit)
11:02:02.142 INFO Windows 11 10.0 amd64
11:02:02.160 INFO User cache: C:\Users\navan\.sonar\cache
11:02:02.644 INFO JRE provisioning: os[windows], arch[amd64]
11:02:06.241 INFO EXECUTION FAILURE
11:02:06.243 INFO Total time: 4.126s
11:02:06.244 ERROR Error during SonarScanner CLI execution
java.lang.IllegalStateException: Error status returned by url [https://api.sonarcloud.io/analysis/jres?os=windows&arch=a
md64]: 401
        at org.sonarsource.scanner.lib.internal.http.ServerConnection.callUrl(ServerConnection.java:182)
        at org.sonarsource.scanner.lib.internal.http.ServerConnection.callApi(ServerConnection.java:145)
        at org.sonarsource.scanner.lib.internal.http.ServerConnection.callRestApi(ServerConnection.java:123)
        at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.getJreMetadata(JavaRunnerFactory.java:159)
        at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.getJreFromServer(JavaRunnerFactory.java:138)
        at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.createRunner(JavaRunnerFactory.java:85)
        at org.sonarsource.scanner.lib.internal.ScannerEngineLauncherFactory.createLauncher(ScannerEngineLauncherFactory
.java:53)
        at org.sonarsource.scanner.lib.ScannerEngineBootstrapper.bootstrap(ScannerEngineBootstrapper.java:118)
        at org.sonarsource.scanner.cli.Main.analyze(Main.java:75)
        at org.sonarsource.scanner.cli.Main.main(Main.java:63)
11:02:06.246 ERROR
11:02:06.246 ERROR Re-run SonarScanner CLI using the -X switch to enable full debug logging.
PS C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin> |
```

- Login to SonarQube using username admin and password admin.
- Create a manual project in SonarQube with the name sonarqube-test1

Step2: go to the jenkins and create new item select pipeline:



Under Pipeline Script, enter the following -

```
node {
    stage('Cloning the GitHub Repo') {
        git 'https://github.com/shazforiot/GOL.git'
    }
    stage('SonarQube analysis') {
        withSonarQubeEnv('sonarqube') {
            sh "<PATH_TO SONARQUBE FOLDER>/bin//sonar-scanner \
                -D sonar.login=<SonarQube_USERNAME> \
                -D sonar.password=<SonarQube_PASSWORD> \
                -D sonar.projectKey=<Project_KEY> \
                -D sonar.exclusions=vendor/**,resources/**,/**/.java \
                -D sonar.host.url=http://127.0.0.1:9000/""
        }
    }
}
```

The screenshot shows the Jenkins Pipeline Configuration screen. The pipeline definition is set to 'Pipeline script'. The script content is identical to the one provided above, detailing the cloning of a GitHub repository and performing a SonarQube analysis. A 'Use Groovy Sandbox' checkbox is checked. At the bottom, there are 'Save' and 'Apply' buttons.

```
node {
    stage('Cloning the GitHub Repo') {
        git 'https://github.com/shazforiot/GOL.git'
    }
    stage('SonarQube analysis') {
        withSonarQubeEnv('sonarqube') {
            sh "<PATH_TO SONARQUBE FOLDER>/bin//sonar-scanner \
                -D sonar.login=<SonarQube_USERNAME> \
                -D sonar.password=<SonarQube_PASSWORD> \
                -D sonar.projectKey=<Project_KEY> \
                -D sonar.exclusions=vendor/**,resources/**,/**/.java \
                -D sonar.host.url=http://127.0.0.1:9000/""
        }
    }
}
```

Save the changes and go to build now:



Console output:

### Console Output

```

Started by user jai navani
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in C:\ProgramData\Jenkins\.jenkins\workspace\SonarQube Pipeline
[Pipeline]
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Cloning the Github Repo)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\SonarQube Pipeline\.git # timeout=10
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/shazforiot/MSBuild_firstproject.git # timeout=10
Fetching upstream changes from https://github.com/shazforiot/MSBuild_firstproject.git
> git.exe --version # timeout=10
> git --version # 'git version 2.45.2.windows.1'
> git.exe fetch --tags --force --progress -- https://github.com/shazforiot/MSBuild_firstproject.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision f2bc042c04c6e72427c380bcae6d6fee7b49adf (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f f2bc042c04c6e72427c380bcae6d6fee7b49adf # timeout=10
> git.exe branch -a -v --no-abbrev # timeout=10
> git.exe branch -D master # timeout=10

```

```

11:22:18.236 INFO Sensor C# File Caching Sensor [csharp]
11:22:18.237 WARN Incremental PR analysis: Could not determine common base path, cache will not be computed. Consider setting 'sonar.projectBaseDir' property.
11:22:18.237 INFO Sensor C# File Caching Sensor [csharp] (done) | time=1ms
11:22:18.237 INFO Sensor Zero Coverage Sensor
11:22:18.251 INFO Sensor Zero Coverage Sensor (done) | time=14ms
11:22:18.256 INFO SCM Publisher SCM provider for this project is: git
11:22:18.257 INFO SCM Publisher 4 source files to be analyzed
11:22:18.789 INFO SCM Publisher 4/4 source files have been analyzed (done) | time=531ms
11:22:18.793 INFO CPD Executor Calculating CPD for 0 files
11:22:18.795 INFO CPD Executor CPD calculation finished (done) | time=0ms
11:22:18.810 INFO SCM revision ID 'f2bc042c04c6e72427c380bcace6d6fee7b49adf'
11:22:19.074 INFO Analysis report generated in 134ms, dir size=201.0 kB
11:22:19.137 INFO Analysis report compressed in 45ms, zip size=22.5 kB
11:22:19.351 INFO Analysis report uploaded in 212ms
11:22:19.353 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=sonarqube-test1
11:22:19.354 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
11:22:19.354 INFO More about the report processing at http://localhost:9000/api/ce/task?id=971ae2f2-4e0f-49a7-88c4-ad4a6cceddf8
11:22:19.366 INFO Analysis total time: 24.819 s
11:22:19.368 INFO SonarScanner Engine completed successfully
11:22:19.454 INFO EXECUTION SUCCESS
11:22:19.455 INFO Total time: 29.696s
[Pipeline]
[Pipeline] // withSonarQubeEnv
[Pipeline]
[Pipeline] // stage
[Pipeline]
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS

```

## Output:

The screenshot shows the SonarQube interface for a project named 'sonarqube-test1'. The main view is the 'main' branch. A prominent green checkmark indicates the analysis has passed. Below it, a yellow warning icon with the text 'The last analysis has warnings. See details' is visible. The dashboard provides a summary of code quality across several dimensions:

- Security:** 0 Open issues
- Reliability:** 0 Open issues
- Maintainability:** 0 Open issues
- Accepted issues:** 0
- Coverage:** 0.0%
- Duplications:** 0.0%

The navigation bar at the top includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, More, and a search bar. Below the main content, there are tabs for Overview, Issues, Security Hotspots, Measures, Code, and Activity, as well as Project Settings and Project Information.

## Experiment 9

**Aim:** To Understand Continuous monitoring and Installation and configuration of Nagios Core, Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine.

Step 1: Create an EC2 Instance and name it as nagios-host

The screenshot shows the AWS EC2 Dashboard. In the left sidebar, under 'Instances', there is a list including 'Instances', 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', and 'Capacity Reservations'. Under 'Images', there are 'AMIs' and 'AMI Catalog'. Under 'Elastic Block Store', there are 'Volumes', 'Snapshots', and 'Lifecycle Manager'. Under 'Network & Security', there are 'Security Groups' and 'Elastic IPs'. At the bottom of the sidebar, there are links for 'CloudShell' and 'Feedback'. The main content area displays the 'Instances (1/1)' section. A search bar at the top says 'Find instance by attribute or tag (case-sensitive)'. Below it, a filter bar shows 'Instance state = running' and 'All states'. A table lists one instance: 'nagios-host' (Instance ID: i-033ee56b96fef8322), which is 'Running' (Status check: initializing), t2.micro instance type, in us-east-1c availability zone, with a public IP of ec2-44-211-225-236.compute-1.amazonaws.com. A detailed view for 'i-033ee56b96fef8322 (nagios-host)' is open, showing 'Details' tab selected. It provides information like Public IPv4 address (44.211.225.236), Private IP4 addresses (172.31.83.53), Public IPv6 DNS (ec2-44-211-225-236.compute-1.amazonaws.com), and Elastic IP addresses.

Step 2: Under the security groups, click on edit inbound rules and set as shown in the figure below

The screenshot shows the AWS EC2 Dashboard. The left sidebar includes 'Instances', 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', 'Capacity Reservations', 'Images', 'AMIs', 'AMI Catalog', 'Elastic Block Store', 'Volumes', 'Snapshots', 'Lifecycle Manager', and 'Network & Security' sections. At the bottom are 'CloudShell' and 'Feedback' links. The main area shows a security group named 'launch-wizard-13' with details: Security group name (sg-0a1c694292da367bd), Owner (856746069793), Description (launch-wizard-13 created 2024-09-30T16:57:21.185Z), and VPC ID (vpc-0ec7dea56d46f7acf). The 'Inbound rules' tab is selected, showing 7 permission entries. The table lists the following rules:

Name	Security group rule...	IP version	Type	Protocol	Port range
-	sgr-0cbb8fb0697d678	IPv4	Custom TCP	TCP	5666
-	sgr-0214f72ad5a70a602	IPv4	HTTPS	TCP	445
-	sgr-070fbe0339e680801	IPv4	SSH	TCP	22
-	sgr-0685a56749b5e8...	IPv4	HTTP	TCP	80
-	sgr-044962adæ76c80...	IPv4	All traffic	All	All
-	sgr-02126894299u1c4...	IPv6	All ICMP - IPv6	IPv6 ICMP	All
-	sgr-0da3debdaf8755e	IPv4	All ICMP - IPv4	ICMP	All

Step 3: Then select the instance nagios-host and then connect the instance.

EC2 > Instances > i-025f1d18f7c8a8cda > Connect to instance

### Connect to instance Info

Connect to your instance i-025f1d18f7c8a8cda (nagios-host) using any of these options.

**EC2 Instance Connect** Session Manager SSH client EC2 serial console

**All ports are open to all IPv4 addresses in your security group**

All ports are currently open to all IPv4 addresses, indicated by **All** and **0.0.0.0/0** in the inbound rule in **your security group**. For increased security, consider restricting access to only the EC2 Instance Connect service IP addresses for your Region: 18.206.107.24/29. [Learn more](#).

Instance ID: **i-025f1d18f7c8a8cda (nagios-host)**

Connection Type:

- Connect using EC2 Instance Connect: Connect using the EC2 Instance Connect browser-based client, with a public IPv4 or IPv6 address.
- Connect using EC2 Instance Connect Endpoint: Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IPv4 address: **3.86.198.73**

IPv6 address:

Username: **ec2-user**

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 3: Now, run the following commands -

```
sudo su
sudo yum update
sudo yum install httpd php
sudo yum install gcc glibc glibc-common
sudo yum install gd gd-devel
```

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-172-31-93-157 ~]# sudo su
[root@ip-172-31-93-157 ec2-user]# sudo yum update
Last metadata expiration check: 0:11:30 ago on Mon Sep 30 16:39:07 2024.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-93-157 ec2-user]# sudo yum install httpd php
Last metadata expiration check: 0:11:51 ago on Mon Sep 30 16:39:07 2024.
Dependencies resolved.

Package           Architecture Version       Repository   Size
Installing:
httpd             x86_64      2.4.62-1.amzn2023    amazonlinux  48 k
php8_3            x86_64      8.3.10-1.amzn2023.0.1  amazonlinux  10 k
Installing dependencies:
apr               x86_64      1.7.2-2.amzn2023.0.2  amazonlinux  129 k
aprutil           x86_64      1.6.3-1.amzn2023.0.1  amazonlinux  98 k
generic-logos-httd noarch     19.0.0-12.amzn2023.0.3  amazonlinux  19 k
httpd-core        x86_64      2.4.62-1.amzn2023    amazonlinux  1.4 M
httpd-filesystem noarch     2.4.62-1.amzn2023    amazonlinux  14 k
httpd-tools       x86_64      2.4.62-1.amzn2023    amazonlinux  81 k
libbrotli         x86_64      1.0.9-4.amzn2023.0.2  amazonlinux  315 k
libeodium          x86_64      1.0.19-4.amzn2023    amazonlinux  176 k

i-025f1d18f7c8a8cda (nagios-host)
PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157
```

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```

php8.3-x86_64-8.3.10-1.amzn2023.0.1.x86_64
Complete!
[root@ip-172-31-93-157 ec2-user]# sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:12:22 ago on Mon Sep 30 16:39:07 2024.
package glibc-2.34-52.amzn2023.0.11.x86_64 is already installed.
package glibc-common-2.34-52.amzn2023.0.11.x86_64 is already installed.
Dependencies resolved.

Transaction summary
Install 13 Packages

Total download size: 52 M
Installed size: 169 M
Is this ok [y/N]: y

i-025f1d18f7c8a8cda (nagios-host)
PublicIP: 3.86.198.73 PrivateIP: 172.31.93.157

```

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```

Complete!
[root@ip-172-31-93-157 ec2-user]# sudo yum install gd gd-devel
Last metadata expiration check: 0:13:10 ago on Mon Sep 30 16:39:07 2024.
Dependencies resolved.

Transaction summary
Install 2 Packages

Total download size: 139 k
Installed size: 38 k
Is this ok [y/N]: y

i-025f1d18f7c8a8cda (nagios-host)
PublicIP: 3.86.198.73 PrivateIP: 172.31.93.157

```

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 4: Create a new nagios user with its password.

`sudo adduser -m nagios`

`sudo passwd nagios`

`sudo groupadd nagcmd`

`sudo usermod -a -G nagcmd nagios`

`sudo usermod -a -G nagcmd apache`

```
root@ip-172-31-93-157 ec2-user]# sudo adduser -m nagios
root@ip-172-31-93-157 ec2-user]# sudo passwd nagios
Changing password for user nagios.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
root@ip-172-31-93-157 ec2-user]# sudo groupadd nagcmd
root@ip-172-31-93-157 ec2-user]# sudo usermod -a -G nagcmd nagios
root@ip-172-31-93-157 ec2-user]# sudo usermod -a -G nagcmd apache
root@ip-172-31-93-157 ec2-user]#
```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 5: Now, run the following commands -

```
mkdir ~/downloads
cd ~/downloads
wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz
wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
tar zxvf nagios-4.0.8.tar.gz
```

```
root@ip-172-31-93-157 ec2-user]# mkdir ~/downloads
root@ip-172-31-93-157 ec2-user]# cd ~/downloads
root@ip-172-31-93-157 downloads]# wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz
wget: missing URL
Usage: wget [OPTION]... [URL]...
Try 'wget --help' for more options.
bash: http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz: No such file or directory
bash: gz: command not found
(root@ip-172-31-93-157 downloads]# wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
--2024-09-30 17:00:06-- http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org) (45.56.123.251):80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2659772 (2.5M) [application/x-gzip]
Saving to: "nagios-plugins-2.0.3.tar.gz"

nagios-plugins-2.0.3.tar.gz      100%[=====] 2.54M 6.16MB/s   in 0.4s
2024-09-30 17:00:07 (6.16 MB/s) - "nagios-plugins-2.0.3.tar.gz" saved [2659772/2659772]

(root@ip-172-31-93-157 downloads]# tar zxvf nagios-4.0.8.tar.gz
tar (child): nagios-4.0.8.tar.gz: Cannot open: No such file or directory
tar (child): Error is not recoverable: exiting now
tar: Child returned status 2
tar: Error is not recoverable: exiting now
(root@ip-172-31-93-157 downloads]#
```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

To resolve the error run the following commands -

```
wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz
wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
tar zxvf nagios-4.0.8.tar.gz
tar zxvf nagios-plugins-2.0.3.tar.gz
cd nagios-4.0.8
```

```
[root@ip-172-31-93-147 downloads]# wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz
--2024-09-30 17:03:04 -- http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz
Resolving prdownloads.sourceforge.net (prdownloads.sourceforge.net)... 204.68.111.105
Connecting to prdownloads.sourceforge.net (prdownloads.sourceforge.net)|204.68.111.105|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://downloadsource.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz [following]
--2024-09-30 17:03:04 -- http://downloadsource.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz
Resolving downloadsource.sourceforge.net (downloadsource.sourceforge.net)... 204.68.111.105
Resolving downloadsource.sourceforge.net (downloadsource.sourceforge.net)... 204.68.111.105
Resolving existing connection to prdownloads.sourceforge.net:80.
HTTP request sent, awaiting response... 302 Found
Location: http://versaweb.dl.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz?viafs=1 [following]
--2024-09-30 17:03:04 -- http://versaweb.dl.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz?viafs=1
Resolving versaweb.dl.sourceforge.net (versaweb.dl.sourceforge.net)... 162.251.232.173
Connecting to versaweb.dl.sourceforge.net (versaweb.dl.sourceforge.net)|162.251.232.173|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1805059 (1.78M) [application/x-gzip]
Saving to: 'nagios-4.0.8.tar.gz'

nagios-4.0.8.tar.gz          100%[=====]  1.72M  2.21MB/s   in 0.8s

2024-09-30 17:03:05 (2.21 MB/s) - 'nagios-4.0.8.tar.gz' saved [1805059/1805059]

--2024-09-30 17:03:05 -- http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2659772 (2.5M) [application/x-gzip]
Saving to: 'nagios-plugins-2.0.3.tar.gz.l1'

nagios-plugins-2.0.3.tar.gz.l1      100%[=====]  2.54M  7.26MB/s   in 0.3s

2024-09-30 17:03:05 (7.26 MB/s) - 'nagios-plugins-2.0.3.tar.gz.l1' saved [2659772/2659772]
```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
nagios-plugins-2.0.3/plugin-scripts/check_ifoperstatus.pl
nagios-plugins-2.0.3/plugin-scripts/Makefile.am
nagios-plugins-2.0.3/plugin-scripts/subst.in
nagios-plugins-2.0.3/plugin-scripts/check_breeze.pl
nagios-plugins-2.0.3/plugin-scripts/check_log.sh
nagios-plugins-2.0.3/plugin-scripts/check_flexim.pl
nagios-plugins-2.0.3/plugin-scripts/check_rpc.pl
nagios-plugins-2.0.3/plugin-scripts/check_oracle.sh
nagios-plugins-2.0.3/plugin-scripts/utils.pm.in
nagios-plugins-2.0.3/plugin-scripts/check_disk_smb.pl
nagios-plugins-2.0.3/plugin-scripts/t/
nagios-plugins-2.0.3/plugin-scripts/t/check_ifoperstatus.t
nagios-plugins-2.0.3/plugin-scripts/t/check_rpc.t
nagios-plugins-2.0.3/plugin-scripts/t/check_file_aget
nagios-plugins-2.0.3/plugin-scripts/t/check_disk_mbt
nagios-plugins-2.0.3/plugin-scripts/t/check_lsstatus.t
nagios-plugins-2.0.3/plugin-scripts/t/utils.t
nagios-plugins-2.0.3/plugin-scripts/check_mailq.pl
nagios-plugins-2.0.3/plugin-scripts/check_wave.pl
nagios-plugins-2.0.3/plugin-scripts/check_irccd.pl
nagios-plugins-2.0.3/plugin-scripts/utils.sh.in
nagios-plugins-2.0.3/plugin-scripts/check_ifstatus.pl
nagios-plugins-2.0.3/plugin-scripts/check_sensors.sh
nagios-plugins-2.0.3/pkg/
nagios-plugins-2.0.3/pkg/fedora/
nagios-plugins-2.0.3/pkg/fedora/requirements
nagios-plugins-2.0.3/pkg/solaris/
nagios-plugins-2.0.3/pkg/solaris/preinstall
nagios-plugins-2.0.3/pkg/solaris/solpkg
nagios-plugins-2.0.3/pkg/solaris/pkginfo.in
nagios-plugins-2.0.3/pkg/solaris/pkginfo
nagios-plugins-2.0.3/pkg/redhat/
nagios-plugins-2.0.3/pkg/redhat/requirements
[root@ip-172-31-93-157 downloads]# ]
```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
nagios-plugins-2.0.3/plugin-scripts/Makefile.am
nagios-plugins-2.0.3/plugin-scripts/subst.in
nagios-plugins-2.0.3/plugin-scripts/check_breeze.pl
nagios-plugins-2.0.3/plugin-scripts/check_log.sh
nagios-plugins-2.0.3/plugin-scripts/check_flexim.pl
nagios-plugins-2.0.3/plugin-scripts/check_rpc.pl
nagios-plugins-2.0.3/plugin-scripts/check_oracle.sh
nagios-plugins-2.0.3/plugin-scripts/utils.pm.in
nagios-plugins-2.0.3/plugin-scripts/check_disk_smb.pl
nagios-plugins-2.0.3/plugin-scripts/t/
nagios-plugins-2.0.3/plugin-scripts/t/check_ifoperstatus.t
nagios-plugins-2.0.3/plugin-scripts/t/check_rpc.t
nagios-plugins-2.0.3/plugin-scripts/t/check_file_aget
nagios-plugins-2.0.3/plugin-scripts/t/check_disk_mbt
nagios-plugins-2.0.3/plugin-scripts/t/check_lsstatus.t
nagios-plugins-2.0.3/plugin-scripts/t/utils.t
nagios-plugins-2.0.3/plugin-scripts/check_mailq.pl
nagios-plugins-2.0.3/plugin-scripts/check_wave.pl
nagios-plugins-2.0.3/plugin-scripts/check_irccd.pl
nagios-plugins-2.0.3/plugin-scripts/utils.sh.in
nagios-plugins-2.0.3/plugin-scripts/check_ifstatus.pl
nagios-plugins-2.0.3/plugin-scripts/check_sensors.sh
nagios-plugins-2.0.3/pkg/
nagios-plugins-2.0.3/pkg/fedora/
nagios-plugins-2.0.3/pkg/fedora/requirements
nagios-plugins-2.0.3/pkg/solaris/
nagios-plugins-2.0.3/pkg/solaris/preinstall
nagios-plugins-2.0.3/pkg/solaris/solpkg
nagios-plugins-2.0.3/pkg/solaris/pkginfo.in
nagios-plugins-2.0.3/pkg/solaris/pkginfo
nagios-plugins-2.0.3/pkg/redhat/
nagios-plugins-2.0.3/pkg/redhat/requirements
[root@ip-172-31-93-157 downloads]# cd nagios-4.0.8
[root@ip-172-31-93-157 nagios-4.0.8]# ]
```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 6: Now to run the configuration script run the following command.

```
./configure --with-command-group=nagcmd
```

```
[root@ip-172-31-93-157 nagios-4.0.8]# ./configure --with-command-group=nagcmd
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-unknown-linux-gnu
checking host system type... x86_64-unknown-linux-gnu
checking for gcc... gcc
checking for C compiler default output file name... a.out
checking whether the C compiler works... yes
checking whether we are cross compiling... no
checking for suffix of executables...
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking whether make sets $MAKE... yes
checking for strip... /usr/bin/strip
checking how to run the C preprocessor... gcc -E
checking for grep that handles long lines and -e... /usr/bin/grep
checking for egrep... /usr/bin/grep -E
checking for ANSI C header files... yes
checking whether time.h and sys/time.h may both be included... yes
checking for sys/wait.h that is POSIX.1 compatible... yes
checking for sys/types.h... yes
checking for sys/stat.h... yes
checking for stdlib.h... yes
checking for string.h... yes
checking for memory.h... yes
checking for strings.h... yes
checking for inttypes.h... yes
checking for stdint.h... yes
checking for unistd.h... yes
checking arpa/inet.h usability... yes
checking arpa/inet.h presence... yes
checking for arpa/inet.h... yes
checking ctype.h usability... yes

i-025f1d18f7c8a8cda (nagios-host)

Public IPs: 3.86.198.73 | Private IPs: 172.31.93.157
```

Step 7: Now, to compile the source code run the following command -  
`make all`

```
root@ip-172-31-93-157 nagios-4.0.8]# make all
cd ./base && make
gcc [1]: Entering directory '/root/downloads/nagios-4.0.8/base'
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o nagios.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o broker.o broker.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nomhmds.o nomhmds.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o ../common/shared.o ../common/shared.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nerd.o nerd.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o query-handler.o query-handler.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o workers.o workers.c
In function 'get_proc_list':
  inlined from 'get_worker' at workers.c:224:12:
workers.c:209:17: warning: '%s' directive argument is null [-Wformat-overflow=]
  209 |         log_debug_info(DERUNK_CHECKS, 1, "Found specialized worker(s) for '%s', (slash @@ *slash != '/') ? slash : cmd_name");
|-----|
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o checks.o checks.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o config.o config.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o commands.o commands.c
commands.c: In function 'process_passive_service_check':
commands.c:224:19: warning: assignment discards 'const' qualifier from pointer target type [-Wdiscarded-qualifiers]
  224 |     cr.source = command_worker.source_name;
|-----|
commands.c: In function 'process_passive_host_check':
commands.c:239:19: warning: assignment discards 'const' qualifier from pointer target type [-Wdiscarded-qualifiers]
  239 |     cr.source = command_worker.source_name;
|-----|
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o events.o events.c
gcc -Wall -I... -g -O2 -DHAVS_CONFIG_H -DNSCORE -c -o flapping.o flapping.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o logging.o logging.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o macros-base.o ../common/macros.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o netutils.o netutils.c
netutils.c: In function 'my_tcp_connect':
netutils.c:50:47: warning: '#d' directive output may be truncated writing between i and ll bytes into a region of size 6 [-Wformat-truncation=]
  50 |     saoprintf(port_st, sizeof(port_st), "%d", port);
|-----|
```

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
[root@ip-172-31-93-157 nagios-4.0.8]# sudo make install
cd ./base; make install
make[1]: Entering directory '/root/downloads/nagios-4.0.8/base'
make install-basic
make[2]: Entering directory '/root/downloads/nagios-4.0.8/base'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/bin
/usr/bin/install -c -m 774 -o nagios -g nagios nagios /usr/local/nagios/bin
/usr/bin/install -c -m 774 -o nagios -g nagios nagiosstats /usr/local/nagios/bin
make[2]: Leaving directory '/root/downloads/nagios-4.0.8/base'
make strip-post-install
make[2]: Entering directory '/root/downloads/nagios-4.0.8/base'
/usr/bin/strip /usr/local/nagios/bin/nagios
/usr/bin/strip /usr/local/nagios/bin/nagiosstats
make[2]: Leaving directory '/root/downloads/nagios-4.0.8/base'
make[1]: Leaving directory '/root/downloads/nagios-4.0.8/base'
cd ./cgi; make install
make[1]: Entering directory '/root/downloads/nagios-4.0.8/cgi'
make install-basic
make[2]: Entering directory '/root/downloads/nagios-4.0.8/cgi'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/sbin
for file in *.cgi; do \
    /usr/bin/install -c -m 775 -o nagios -g nagios $file /usr/local/nagios/sbin; \
done
/usr/bin/install: cannot stat '**.cgi': No such file or directory
make[2]: *** [Makefile:205: install-basic] Error 1
make[2]: Leaving directory '/root/downloads/nagios-4.0.8/cgi'
make[1]: *** [Makefile:197: install] Error 2
make[1]: Leaving directory '/root/downloads/nagios-4.0.8/cgi'
make: *** [Makefile:235: install] krror 2
[root@ip-172-31-93-157 nagios-4.0.8]# sudo make install-init
/usr/bin/install -c -m 755 -d -o root -g root /etc/rc.d/init.d/nagios
/usr/bin/install -c -m 755 -o root -g root daemon-init /etc/rc.d/init.d/nagios
*** init script installed ***
*** init script installed ***
```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

```
[root@ip-172-31-93-157 nagios-4.0.8]# sudo make install-init
/usr/bin/install -c -m 755 -d -o root -g root /etc/rc.d/init.d
/usr/bin/install -c -m 755 -o root -g root daemon-init /etc/rc.d/init.d/nagios
*** Init script installed ***
*** Init script installed ***

[root@ip-172-31-93-157 nagios-4.0.8]# sudo make install-config
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr/local/nagios/etc/nagios.cfg
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/cgi.cgi /usr/local/nagios/etc/cgi.cgi
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /usr/local/nagios/etc/resource.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/timersperiods.cfg /usr/local/nagios/etc/objects/timersperiods.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/switche.cfg /usr/local/nagios/etc/objects/switche.cfg
*** config files installed ***
Remember, these are *SAMPLE* config files. You'll need to read the documentation for more information on how to actually define services, hosts, etc. to fit your particular needs.
```

```
[root@ip-172-31-93-157 nagios-4.0.8]# sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagcmd -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw
```

\*\*\* External command directory configured \*\*\*

```
[root@ip-172-31-93-157 nagios-4.0.8]#
```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

To resolve the errors run the following commands -

sudo yum install -y gcc glibc glibc-common gd gd-devel make net-snmp openssl-devel

rm -rf nagios-4.0.8

cd ~/downloads/nagios-4.4.6

./configure --with-command-group=nagcmd

make all

sudo make install

```

web interface
make install-classicui
- This installs the classic theme for the Nagios
web interface

*** Support Notes *****
If you have questions about configuring or running Nagios,
please make sure that you:
- Look at the sample config files
- Read the documentation on the Nagios Library at:
  https://library.nagios.com

before you post a question to one of the mailing lists.
Also make sure to include pertinent information that could
help others help you. This might include:
- What version of Nagios you are using
- What version of the plugins you are using
- Relevant snippets from your config files
- Relevant error messages from the Nagios log file

For more information on obtaining support for Nagios, visit:
  https://support.nagios.com

*****
Enjoy.

i-025f1d18f7c8a8cda (nagios-host)
PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

```

X

[CloudShell](#) [Feedback](#)

© 2024 Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

## Step 8: Edit the config file and change the email address.

sudo nano /usr/local/nagios/etc/objects/contacts.cfg

```

GNU nano 5.8
/usr/local/nagios/etc/objects/contacts.cfg
This contact definition inherits a lot of default values from the 'generic-contact'
template which is defined elsewhere.

define contact{
    contact_name          nagiosadmin      ; Short name of user
    use                   generic-contact   ; Inherit default values from generic-contact template (defined above)
    alias                Nagios Admin     ; Full name of user
    email                2022.anuprita.mhapankar@ves.ac.in ; <***** CHANGE THIS TO YOUR EMAIL ADDRESS *****

}

CONTACT DEFINED
#####
# We only have one contact in this simple configuration file, so there is
# no need to create more than one contact group.

define contactgroup{
    contactgroup_name    admins
    alias                Nagios Administrators
    members              nagiosadmin
}

i-025f1d18f7c8a8cda (nagios-host)
PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

```

[CloudShell](#) [Feedback](#)

© 2024 Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

## Step 9: Now run the following commands –

sudo make install-webconf  
 sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin  
 sudo service httpd restart  
 cd ~/downloads  
 tar zxvf nagios-plugins-2.0.3.tar.gz

```
- Read the documentation on the Nagios Library at:  
  https://library.nagios.com  
  
before you post a question to one of the mailing lists.  
Also make sure to include pertinent information that could  
help others help you. This might include:  
- What version of Nagios you are using  
- What version of the plugins you are using  
- Relevant snippets from your config files  
- Relevant error messages from the Nagios log file  
  
For more information on obtaining support for Nagios, visit:  
  https://support.nagios.com  
*****  
Enjoy.  
  
[root@ip-172-31-93-157 nagios-4.4.6]# sudo nano /usr/local/nagios/etc/objects/contacts.cfg  
[root@ip-172-31-93-157 nagios-4.4.6]# sudo make install-webconf  
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/httpd/conf.d/nagios.conf  
if ! 0 -eq 1 ]; then \  
    in -s /etc/httpd/conf.d/nagios.conf /etc/apache2/sites-enabled/nagios.conf; \  
fi  
*** Nagios/Apache conf file installed ***  
  
[root@ip-172-31-93-157 nagios-4.4.6]# sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin  
New password:  
Re-type new password:  
Adding password for user nagiosadmin  
[root@ip-172-31-93-157 nagios-4.4.6]# [ ]  
  
i-025f1d18f7c8a8cda (nagios-host)  
PublicIP: 3.86.198.73 PrivateIP: 172.31.93.157
```

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
[root@ip-172-31-93-157 nagios-4.4.6]# sudo service httpd restart  
Redirecting to /bin/systemctl restart httpd.service  
[root@ip-172-31-93-157 nagios-4.4.6]# cd ~/downloads  
tar xvzf nagios-plugins-2.0.3.tar.gz  
nagios-plugins-2.0.3/  
nagios-plugins-2.0.3/perlmods/  
nagios-plugins-2.0.3/perlmods/Config-Tiny-2.14.tar.gz  
nagios-plugins-2.0.3/perlmods/parent-0.226.tar.gz  
nagios-plugins-2.0.3/perlmods/test-Simple-0.98.tar.gz  
nagios-plugins-2.0.3/perlmods/Makefile.in  
nagios-plugins-2.0.3/perlmods/version-0.9903.tar.gz  
nagios-plugins-2.0.3/perlmods/Makefile.am  
nagios-plugins-2.0.3/perlmods/Module-Runtime-0.013.tar.gz  
nagios-plugins-2.0.3/perlmods/Module-Metadata-1.000014.tar.gz  
nagios-plugins-2.0.3/perlmods/Params-Validate-1.08.tar.gz  
nagios-plugins-2.0.3/perlmods/Class-Accessor-0.34.tar.gz  
nagios-plugins-2.0.3/perlmods/try-Tiny-0.18.tar.gz  
nagios-plugins-2.0.3/perlmods/Module-Implementation-0.07.tar.gz  
nagios-plugins-2.0.3/perlmods/Makefile  
nagios-plugins-2.0.3/perlmods/Perl-OSType-1.003.tar.gz  
nagios-plugins-2.0.3/perlmods/Install_order  
nagios-plugins-2.0.3/perlmods/Nagios-Plugin-0.36.tar.gz  
nagios-plugins-2.0.3/perlmods/Math-Calc-Units-1.07.tar.gz  
nagios-plugins-2.0.3/perlmods/Module-Build-0.4007.tar.gz  
nagios-plugins-2.0.3/ABOUT-NLS  
nagios-plugins-2.0.3/configure.ac  
nagios-plugins-2.0.3/Makefile.in  
nagios-plugins-2.0.3/config.h.in  
nagios-plugins-2.0.3/ChangeLog  
nagios-plugins-2.0.3/AUTHORS  
nagios-plugins-2.0.3/lib/  
nagios-plugins-2.0.3/lib/parse_ini.h  
nagios-plugins-2.0.3/lib/extr_opts.c  
nagios-plugins-2.0.3/lib/Makefile.in
```

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## Step 10: Compile and install plugins

```
cd nagios-plugins-2.0.3  
.configure --with-nagios-user=nagios --with-nagios-group=nagios  
make  
sudo make install
```

Step 11: To start nagios run the following commands –  
`sudo chkconfig --add nagios`

```
sudo chkconfig nagios on
```

Verify using the following command -

```
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

```
root@ip-172-31-93-157 nagios-plugins-2.0.3# sudo chkconfig --add nagios
sudo chkconfig nagios on
root@ip-172-31-93-157 nagios-plugins-2.0.3# sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

nagios Core 4.4.6
copyright (c) 2009-present Nagios Core Development Team and Community Contributors
copyright (c) 1999-2009 Ethan Galstad
last Modified: 2020-04-28
license: GPL

website: https://www.nagios.org
reading configuration data...
  Read main config file okay...
WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please use check_interval instead.
WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please use retry_interval instead.
WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please use check_interval instead.
WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please use retry_interval instead.
  Read object config files okay...

running pre-flight check on configuration data...

Checking objects...
  Checked 8 services.
  Checked 1 hosts.
  Checked 1 host groups.
  Checked 0 service groups.
  Checked 1 contacts.
  Checked 1 contact groups.
  Checked 24 commands.
  Checked 5 time periods.
  Checked 0 host escalations.
  Checked 0 service escalations.
Checking for circular paths...
  Checked 1 hosts
```

If there are no errors run the following command –  
`sudo service nagios start`

```
WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please use retry_interval instead.
WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please use check_interval instead.
WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please use retry_interval instead.
Read object config files okay...
```

```
Running pre-flight check on configuration data...
```

```
Checking objects...
    Checked 8 services.
    Checked 1 hosts.
    Checked 1 host groups.
    Checked 0 service groups.
    Checked 1 contacts.
    Checked 1 contact groups.
    Checked 24 commands.
    Checked 5 time periods.
    Checked 0 host escalations.
    Checked 0 service escalations.
Checking for circular paths...
    Checked 1 hosts
    Checked 0 service dependencies
    Checked 0 host dependencies
    Checked 5 timperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...
```

```
Total Warnings: 0
```

```
Total Errors: 0
```

```
Things look okay - no serious problems were detected during the pre-flight check
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# sudo service nagios start
Starting nagios (via systemctl):
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# [OK]
```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.75 PrivateIPs: 172.31.93.157

Check status using the following command -  
sudo systemctl status nagios

```
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# sudo systemctl status nagios
● nagios.service - LSB: starts and stops the Nagios monitoring server
   Loaded: loaded (/etc/init.d/nagios; generated)
   Active: active (running) since Mon 2024-09-30 18:02:27 UTC; 42s ago
     Docs: man:sysvinit(8)
   Process: 79969 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)
   Tasks: 6 (limit: 1112)
   Memory: 2.2M
      CPU: 52ms
     CpuTime: 0us
    Cgroup: /system.slice/nagios.service
           ├─$0009 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           ├─$0011 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
           ├─$0012 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
           ├─$0013 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
           ├─$0014 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
           └─$0037 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Successfully registered manager as @wproc with query handler
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80011;pid=80011
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80014;pid=80014
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80017;pid=80013
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80012;pid=80012
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please use check_interval instead.
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please use retry_interval instead.
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please use check_interval instead.
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: successfully launched command file worker with pid 80037
lines= 1-26/26 (END)
```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.75 PrivateIPs: 172.31.93.157

Step 12: Go to EC2 instance and copy the public IP address of the instance

Screenshot of the AWS CloudWatch Instances page showing a single EC2 instance named "nagios-host".

**Instances (1/1) Info**

Last updated about 1 hour ago | Connect | Instance state | Actions | Launch instances

Instance state = running | Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPs
nagios-host	i-025f1d18f7c8a8cda	Running	t2.micro	2/2 checks passed	View alarm	us-east-1c	ec2-3-86-

**i-025f1d18f7c8a8cda (nagios-host)**

**Details** | Status and alarms | Monitoring | Security | Networking | Storage | Tags

**Instance summary**

- Instance ID: i-025f1d18f7c8a8cda (nagios-host)
- IPv6 address: -
- Hostname type: IP name: ip-172-31-93-157.ec2.internal
- Answer private resource DNS name: -
- Public IPv4 address copied: 3.86.198.73 | open address
- Private IP DNS name (IPv4 only): ip-172-31-93-157.ec2.internal
- Instance state: Running
- Instance type: t2.micro
- Private IPv4 addresses: 172.31.93.157
- Public IPv4 DNS: ec2-3-86-198-73.compute-1.amazonaws.com | open address
- Elastic IP addresses: -

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 13: Now visit [http://<your\\_public\\_ip\\_address>/nagios](http://<your_public_ip_address>/nagios) Enter correct credentials and then you will see this page.

Screenshot of the Nagios Core web interface.

**Nagios® Core**

General | Home | Documentation | Current Status | Reports | System

**Current Status**

- Tactical Overview
- Map (Legacy)
- Hosts
- Services
- Host Groups
- Summary
- Grid
- Service Groups
- Summary
- Problems
- Services (Unhandled)
- Hosts (Unhandled)
- Network Outages
- Quick Search

**Reports**

- Availability
- Trends (Legacy)
- Alerts
- History
- Summary
- Histogram (Legacy)
- Notifications
- Event Log

**System**

- Comments
- Downtime
- Process Info
- Performance Info
- Scheduling Queue
- Configuration

**Get Started**

- Start monitoring your infrastructure
- Change the look and feel of Nagios
- Extend Nagios with hundreds of add-ons
- Get support
- Get training
- Get certified

**Quick Links**

- Nagios Library (tutorials and docs)
- Nagios Libre (development blog)
- Nagios Exchange (plugins and add-ons)
- Nagios Support (tech support)
- Nagios.com (company)
- Nagios.org (project)

**A new version of Nagios Core is available!** Visit [nagios.org](http://nagios.org) to download Nagios 4.5.5

**Nagios® Core™ Version 4.4.6**  
April 28, 2020 | Check for updates

**Latest News**

**Don't Miss...**

Copyright © 2019-2026 Nagios Core Development Team and Community Contributors. Copyright © 1999-2009 Ethan Galstad. See the THANKS file for more information on contributors.

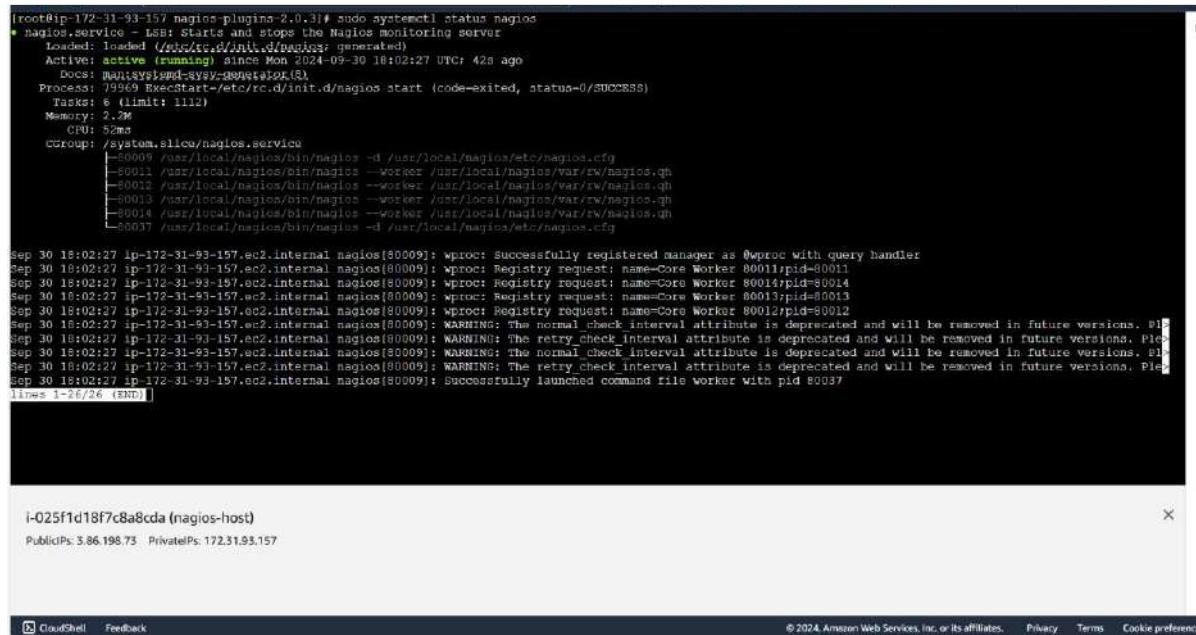
Nagios Core is licensed under the GNU General Public License and is provided AS IS with NO WARRANTY OF ANY KIND, INCLUDING THE WARRANTY OF DESIGN, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Nagios, Nagios Core and the Nagios logo are trademarks, service marks, registered trademarks or registered service marks owned by Nagios Enterprises, LLC. Use of the Nagios marks is governed by the trademark user restrictions.

**Nagios® Core™** SOURCEFORGE.NET

## Experiment 10

**Aim:** To perform Port, Service monitoring, Windows/Linux server monitoring using Nagios.

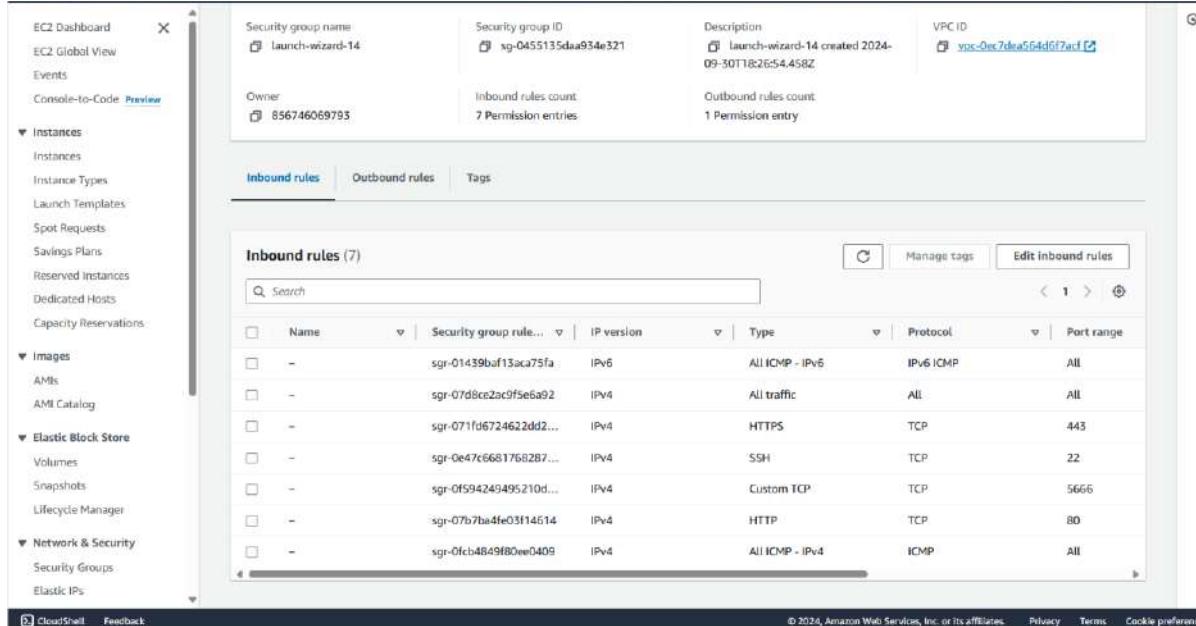
**Step 1:** Initially confirm that Nagios is running on the server side. For this run the following command -  
**sudo systemctl status nagios**  
on the nagios-host instance.



```
root@ip-172-31-93-157 nagios-plugins-2.0.3]# sudo systemctl status nagios
nagios.service - LSB: starts and stops the Nagios monitoring server
   Loaded: loaded (/etc/init.d/nagios; generated)
   Active: active (running) since Mon 2024-09-30 10:02:27 UTC; 42s ago
     Docs: man:lsb_start-stop(8)
   Process: 79969 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)
   Tasks: 6 (limit: 1112)
   Memory: 2.2M
      CPU: 52ms
     CpuUsage: 0.000us
    Cgroup: /system.slice/nagios.service
            └─$0009 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
              ├─$0011 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
              ├─$0012 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
              ├─$0013 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
              ├─$0014 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
              └─$0037 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproot: Successfully registered manager as @mpmc with query handler
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproot Registry request: name=Core Worker 80011/pid=80011
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproot Registry request: name=Core Worker 80014/pid=80014
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproot Registry request: name=Core Worker 80015/pid=80013
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproot Registry request: name=Core Worker 80012/pid=80012
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please see https://nagios-plugins.org/doc/troubleshooting.html#normal_check_interval_deprecated
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please see https://nagios-plugins.org/doc/troubleshooting.html#retry_check_interval_deprecated
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please see https://nagios-plugins.org/doc/troubleshooting.html#retry_check_interval_deprecated
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: successfully launched command file worker with pid 80037
lines: 1-26 (END)
```

**Step 2:** Once confirmed, make another instance with the same security group as that of nagios-host.  
**For now, leave this machine as it is, and go back to your nagios-host machine.**



Name	Security group rule...	IP version	Type	Protocol	Port range
sgr-01439bf13aca75fa	IPv6	All ICMP - IPv6	IPv6 ICMP	All	
sgr-07d8ce2ac9f5e6a92	IPv4	All traffic	All	All	
sgr-071fd6724622dd2...	IPv4	HTTPS	TCP	443	
sgr-0e47c6681768287...	IPv4	SSH	TCP	22	
sgr-0f594249495210d...	IPv4	Custom TCP	TCP	5666	
sgr-07b7ba4fe03f14614	IPv4	HTTP	TCP	80	
sgr-0fcfb4849f80ee0409	IPv4	All ICMP - IPv4	ICMP	All	

**Step 3:** Now run the following command -  
**ps -ef | grep nagios**

```

Active: active (running) since Mon 2024-09-30 10:02:27 UTC; 42s ago
  Docs: man:systemd-sysv-generator(8)
Process: 79969 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)
 Tasks: 6 (limit: 1112)
Memory: 2.2M
 CPU: 52ms
 CGroup: /system.slice/nagios.service
           ├─80009 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           ├─80011 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─80012 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─80013 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─80014 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           └─80037 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Successfully registered manager as $wproc with query handler
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80011;pid=80011
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80014;pid=80014
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80013;pid=80013
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80012;pid=80012
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please note that the check_interval attribute is now used for both normal and passive checks.
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please note that the check_interval attribute is now used for both normal and passive checks.
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please note that the check_interval attribute is now used for both normal and passive checks.
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please note that the check_interval attribute is now used for both normal and passive checks.
Sep 30 10:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: Successfully launched command file worker with pid 80037

[root@ip-172-31-93-157 nagios-plugins-2.0.3]# ps -ef | grep nagios
nagios   80009      1  0 10:02 ?        0:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios   80011  80009  0 10:02 ?        0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   80012  80009  0 10:02 ?        0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   80013  80009  0 10:02 ?        0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   80014  80009  0 10:02 ?        0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   80037  80009  0 10:02 ?        0:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
root     81960    3110  0 10:02 pts/1    0:00:00 grep --color=auto nagios
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# []

```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.95.157

Step 4: Now, run the following commands -

**sudo su**

```
mkdir /usr/local/nagios/etc/objects/monitorhosts
```

```
mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
```

cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

```
root@ip-172-31-93-157 nagios-plugins-2.0.3]# sudo su
root@ip-172-31-93-157 nagios-plugins-2.0.3]# mkdir /usr/local/nagios/etc/objects/monitorhosts
root@ip-172-31-93-157 nagios-plugins-2.0.3]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
root@ip-172-31-93-157 nagios-plugins-2.0.3]# cp /usr/local/nagios/etc/objects/localhost.cfg
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
cp: missing destination file operand after '/usr/local/nagios/etc/objects/localhost.cfg'
try 'cp --help' for more information.
root@ip-172-31-93-157 nagios-plugins-2.0.3]# rm /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg: No such file or directory
root@ip-172-31-93-157 nagios-plugins-2.0.3]# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
root@ip-172-31-93-157 nagios-plugins-2.0.3]# cp /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg: No such file or directory
root@ip-172-31-93-157 nagios-plugins-2.0.3]# cp /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
cp: missing destination file operand after '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg'
try 'cp --help' for more information.
root@ip-172-31-93-157 nagios-plugins-2.0.3]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
root@ip-172-31-93-157 nagios-plugins-2.0.3]# [
```

i-025f1d18f7c8a8cda (naqios-host)

PublicIP: 3.86.198.73 PrivateIP: 172.31.93.157

Step 5: Open linuxserver.cfg using the the following command -

```
nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

**Change the hostname to linuxserver (EVERYWHERE ON THE FILE)**

**Change address to the public IP address of your LINUX CLIENT.**

**Change hostgroup\_name under hostgroup to linux-servers1**

```
GNU nano 5.2                               /usr/local/nagios/etc/objects/monitrohosts/linuxserver.cfg                         Modified
example of how you can create configuration entities to monitor
the local Linux machine.

#####
# Host DEFINITION
#####

# Define a host for the local machine.

define host{
    use          linux-server           ; Name of host template to use
    ; This host definition will inherit all variables that are defined
    ; in (or inherited by) the linux-server host template definition.

    host_name    linux-server
    alias        linux-server
    address      3.95.202.23[]
}

#####

i-025f1d18f7c8a8cda (nagios-host)
PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157
```

```
CloudShell Feedback                                         © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

GNU nano 5.2                               /usr/local/nagios/etc/objects/monitrohosts/linuxserver.cfg                         Modified
check_command    check_local_swap!20!10

}

# Define a service to check SSH on the local machine.
# Disable notifications for this service by default, as not all users may have SSH enabled.

define service{
    use          local-service           ; Name of service template to use
    host_name    linuxserver
    service_description  SSH
    check_command   check_ssh
    notifications_enabled  0
}

# Define a service to check HTTP on the local machine.
# Disable notifications for this service by default, as not all users may have HTTP enabled.

define service{
    use          local-service           ; Name of service template to use
    host_name    linuxserver
    service_description  HTTP
    check_command   check_http
    notifications_enabled  0
}

i-025f1d18f7c8a8cda (nagios-host)
PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157
```

Step 6: Open Nagios config file and add the following line -  
nano /usr/local/nagios/etc/nagios.cfg

Then add this line -  
cfg\_dir=/usr/local/nagios/etc/objects/monitorhosts/

```

GND nano 5.8                                         /usr/local/nagios/etc/nagios.cfg                                         Modified
--cfg_file=/usr/local/nagios/etc/objects/commands.cfg
--cfg_file=/usr/local/nagios/etc/objects/contacts.cfg
--cfg_file=/usr/local/nagios/etc/objects/timperiods.cfg
--cfg_file=/usr/local/nagios/etc/objects/templates.cfg

# Definitions for monitoring the local (Linux) host
--cfg_file=/usr/local/nagios/etc/objects/localhost.cfg

# Definitions for monitoring a Windows machine
--cfg_file=/usr/local/nagios/etc/objects/Windows.cfg

# Definitions for monitoring a router/switch
--cfg_file=/usr/local/nagios/etc/objects/switch.cfg

# Definitions for monitoring a network printer
--cfg_file=/usr/local/nagios/etc/objects/printer.cfg

# You can also tell Nagios to process all config files (with a .cfg
# extension) in a particular directory by using the cfg_dir
# directive as shown below:

--cfg_dir=/usr/local/nagios/etc/servers
--cfg_dir=/usr/local/nagios/etc/printers
--cfg_dir=/usr/local/nagios/etc/switches
--cfg_dir=/usr/local/nagios/etc/routers

cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/[]

  Help     Write Out   Where Is   Cut   Execute   Location   Undo   Set Mark   To Bracket   Previous
  Exit     Read File   Replace   Paste   Justify   Go To Line   Redo   Copy   Where Was   Next

i-025f1d18f7c8a8cda (nagios-host)                                     X
PublicIPs: 3.86.198.73  PrivateIPs: 172.31.93.157

CloudShell  Feedback

```

Step 8: Verify configuration files using the following command -  
 sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

If there are no errors, run the following command -  
 sudo service nagios start

```

[root@ip-172-31-93-157 nagios-plugins-2.0.3]# nano /usr/local/nagios/etc/nagios.cfg
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# nano /usr/local/nagios/etc/nagios.cfg
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.4.6
copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
License: GPL

Website: https://www.nagios.org
Reading configuration data...
  Read main config file okay...
WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please use check_interval instead.
WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please use retry_interval instead.
WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please use check_interval instead.
WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please use retry_interval instead.
Error: Could not find any host matching 'linuxserver' (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 45)
Error: Could not expand members specified in hostgroup (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 45)
  Error processing object config files!

**** One or more problems was encountered while processing the config files...

Check your configuration file(s) to ensure that they contain valid
directives and data definitions. If you are upgrading from a previous
version of Nagios, you should be aware that some variables/definitions
may have been removed or modified in this version. Make sure to read
the HTML documentation regarding the config files, as well as the
'Whats New' section to find out what has changed.

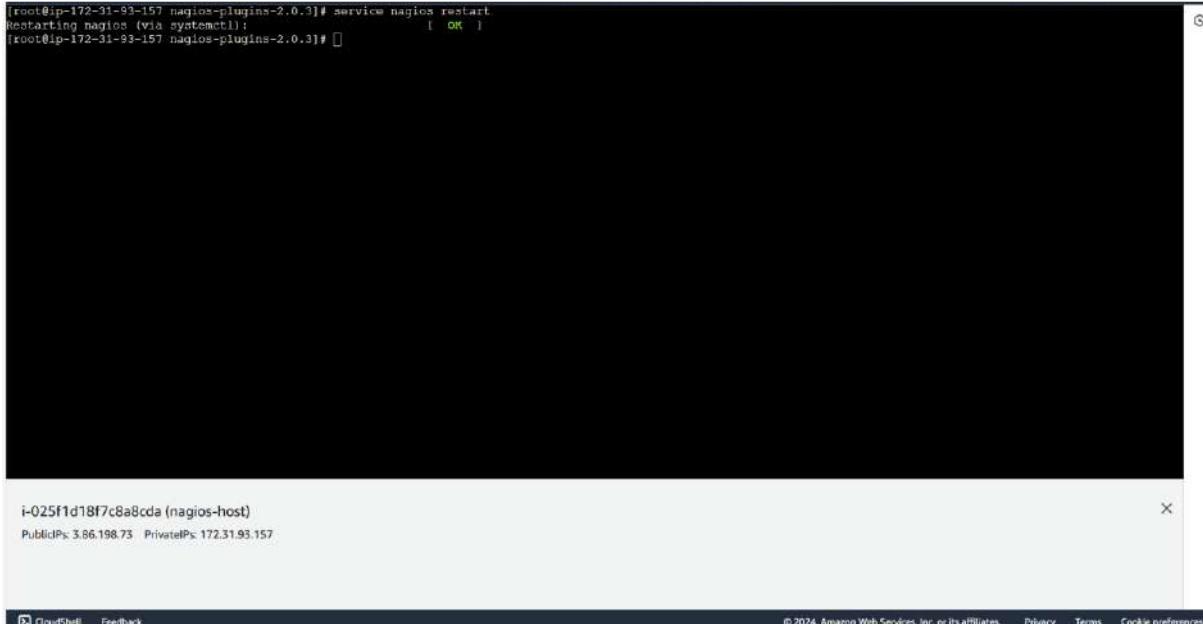
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# []

i-025f1d18f7c8a8cda (nagios-host)                                     X
PublicIPs: 3.86.198.73  PrivateIPs: 172.31.93.157

CloudShell  Feedback

```

```
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# service nagios restart
Restarting nagios (via systemctl): [ OK ]
[root@ip-172-31-93-157 nagios-plugins-2.0.3]#
```



Step 9: After entering the correct credentials, you will see this page.

EC2 Dashboard EC2 Global View Events Console-to-Code Preview Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Capacity Reservations Images AMIs AMI Catalog Elastic Block Store Volumes Snapshots Lifecycle Manager Network & Security Security Groups Elastic IPs

Instances (1/1) [Info](#) Last updated about 1 hour ago Connect Instance state Actions Launch instances

Find instance by attribute or tag (case-sensitive) All states Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
<a href="#">nagios-host</a>	i-025f1d18f7c8a8cda	<span>Running</span>	t2.micro	<span>2/2 checks passed</span>	<a href="#">View alarms</a> +	us-east-1c	ec2-3-86-198-73.compute-1.amazonaws.com

**i-025f1d18f7c8a8cda (nagios-host)**

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary Public IPv4 address copied  
Instance ID: i-025f1d18f7c8a8cda (nagios-host) Instance state: Running  
IPv6 address: - Private IP DNS name (IPv4 only): ip-172-31-93-157.ec2.internal  
Hostname type: IP name: ip-172-31-93-157.ec2.internal Answer private resource DNS name  
Private IP4 addresses: 172.31.93.157  
Public IP4 DNS: ec2-3-86-198-73.compute-1.amazonaws.com  
Elastic IP addresses:

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Not secure 3.86.198.73/nagios/

# Nagios\*

**Current Network Status**

Last Updated: Mon Sep 30 19:13:49 UTC 2024  
Nagios Core v4.4.6 - www.nagios.org  
Logged in as nrogersden

[View Service Status Details For All Host Groups](#)  
[View Status Overview For All Host Groups](#)  
[View Status Summary For All Host Groups](#)  
[View Status Grid For All Host Groups](#)

**Host Status Totals**

Up	Down	Unreachable	Pending
2	0	0	0
All Problems	All Types		
0	2		

**Service Status Totals**

Ok	Warning	Unknown	Critical	Pending
6	1	0	1	8
All Problems	All Types			
2	14			

**Host Status Details For All Host Groups**

Host	Status	Last Check	Duration	Status Information
BraceServer	UP	09-30-2024 19:13:16	0d 0h 0m 33ms	PING OK - Packet loss = 0%, RTA = 1.02 ms
located	UP	09-30-2024 19:01:49	0d 1h 11m 22s	PING OK - Packet loss = 0%, RTA = 0.94 ms

Results 1 - 2 of 2 Matching Hosts

**Current Status**

- [General Overview](#)
- [Map \(Legacy\)](#)
- [Hosts](#)
- [Services](#)
- [Host Groups](#)
- [Summary](#)
- [Grid](#)
- [Service Groups](#)
- [Summary](#)
- [Grid](#)
- [Problems](#)
- [Services \(Unhandled\)](#)
- [Hosts \(Unhandled\)](#)
- [Network Outages](#)
- [Quick Search](#)

**Reports**

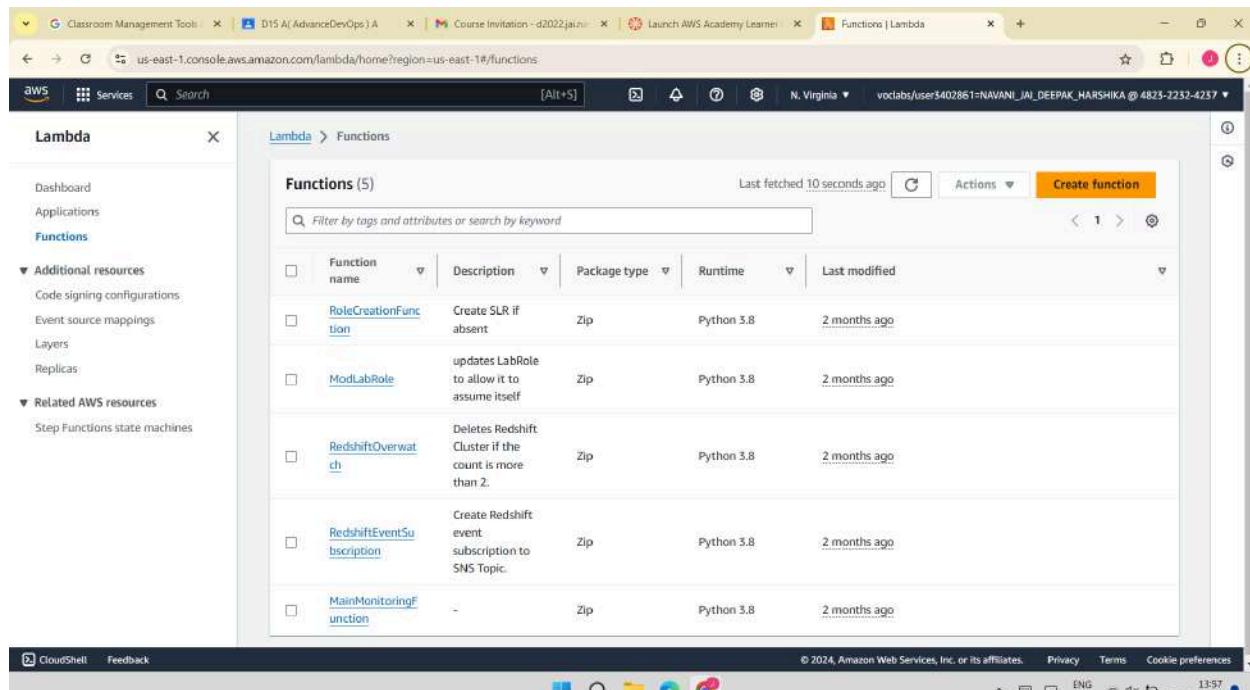
- [Availability](#)
- [Trends \(Legacy\)](#)
- [Alerts](#)
- [History](#)
- [Summary](#)
- [Histogram \(Legacy\)](#)
- [Notifications](#)
- [Event Log](#)

**System**

- [Comments](#)
- [Downtime](#)
- [Process Info](#)
- [Performance Info](#)
- [Scheduling Queue](#)
- [Configuration](#)

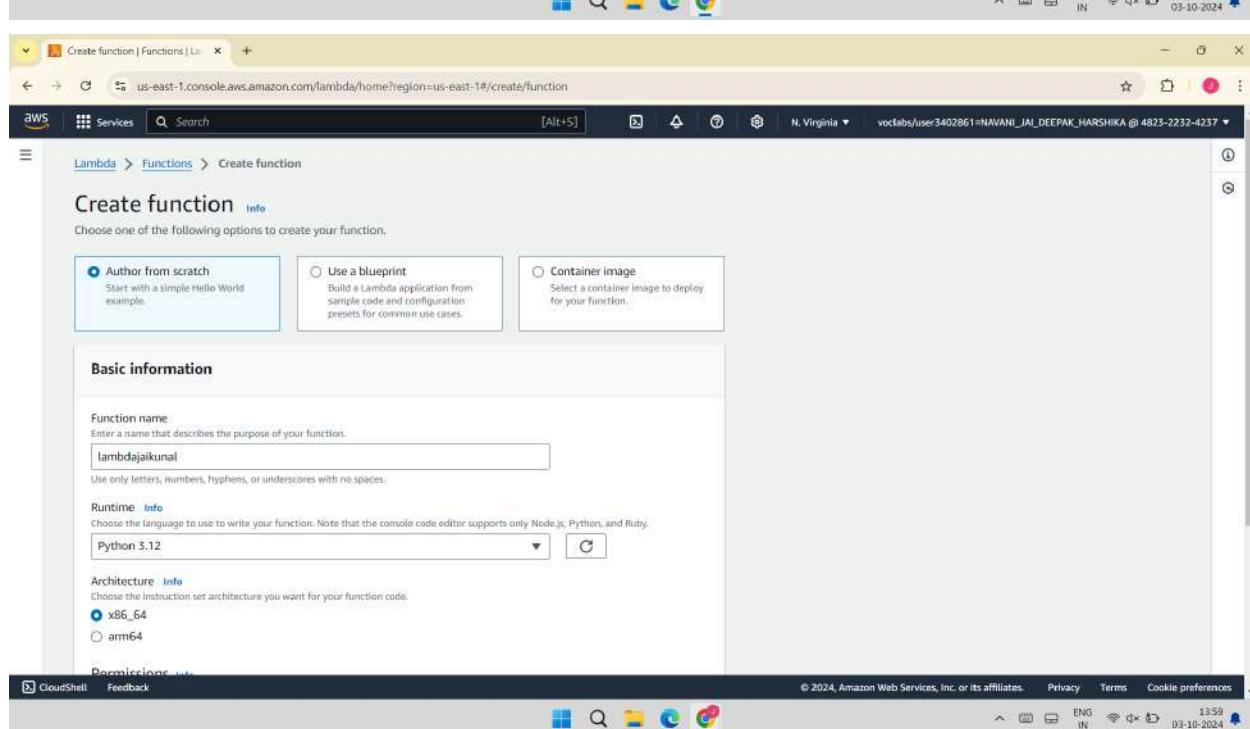
Page: 1

**Advance devops- experiment 11**  
**Kunal Punjabi**  
**D15-A(44)**



The screenshot shows the AWS Lambda Functions page with a list of five existing functions:

Function name	Description	Package type	Runtime	Last modified
<a href="#">RoleCreationFunction</a>	Create SLR if absent	Zip	Python 3.8	2 months ago
<a href="#">ModLabRole</a>	updates LabRole to allow it to assume itself	Zip	Python 3.8	2 months ago
<a href="#">RedshiftOverwatch</a>	Deletes Redshift Cluster if the count is more than 2.	Zip	Python 3.8	2 months ago
<a href="#">RedshiftEventSubscription</a>	Create Redshift event subscription to SNS Topic.	Zip	Python 3.8	2 months ago
<a href="#">MainMonitoringFunction</a>	-	Zip	Python 3.8	2 months ago

The screenshot shows the AWS Lambda Create function page. The "Author from scratch" option is selected under "Choose one of the following options to create your function".

**Basic information**

Function name: lambdaajikunal

Runtime: Python 3.12

Architecture: x86\_64

Permissions: [Select]

Screenshot of the AWS Lambda 'Create function' wizard.

**Runtime:** Python 3.12

**Architecture:** x86\_64

**Permissions:**

- Execution role: Use an existing role (LabRole)
- Existing role: LabRole

**Advanced settings:**

Successfully created the function lambdaJaikunal. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

**Function overview:**

- Diagram tab (selected): Shows the function name lambdaJaikunal and a placeholder for triggers and destinations.
- Template tab: Placeholder for the Lambda function template.
- Description: Last modified 2 seconds ago.
- Function ARN: arnaws:lambda:us-east-1:482322324237:function:lambdaJaikunal
- Function URL: Info

**Code, Test, Monitor, Configuration, Aliases, Versions:**

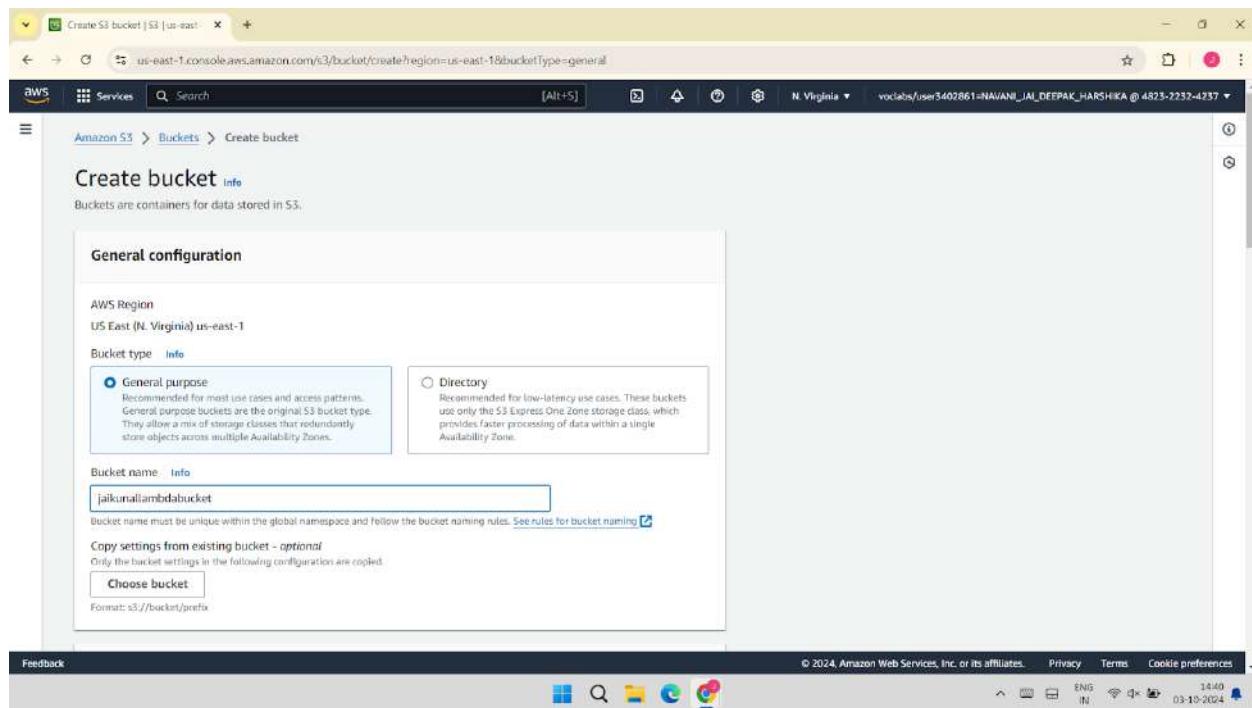
# Advanced DevOps Exp-12

Kunal Punjabi

D15A 44

## Procedure:-

1. Create an S3 bucket of the same location as that of the Lambda function



Upload objects - S3 bucket jail

aws Services Search [Alt+S] N. Virginia vocabs/user3402861=NAVANJAI\_DEPPAK\_HARSHIK @ 4823-2232-4237

Upload succeeded  
View details below.

Upload: status Close

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

**Summary**

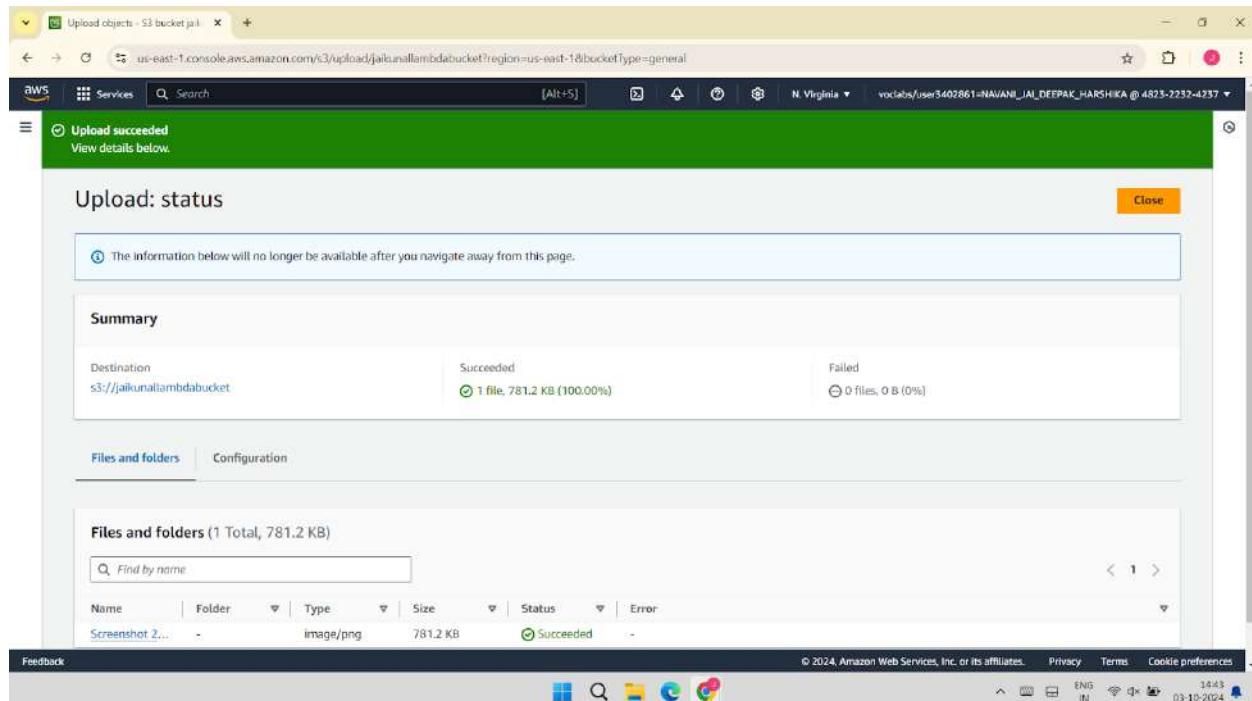
Destination	Succeeded	Failed
s3://jalkunaliambabucket	1 file, 781.2 KB (100.00%)	0 files, 0 B (0%)

[Files and folders](#) [Configuration](#)

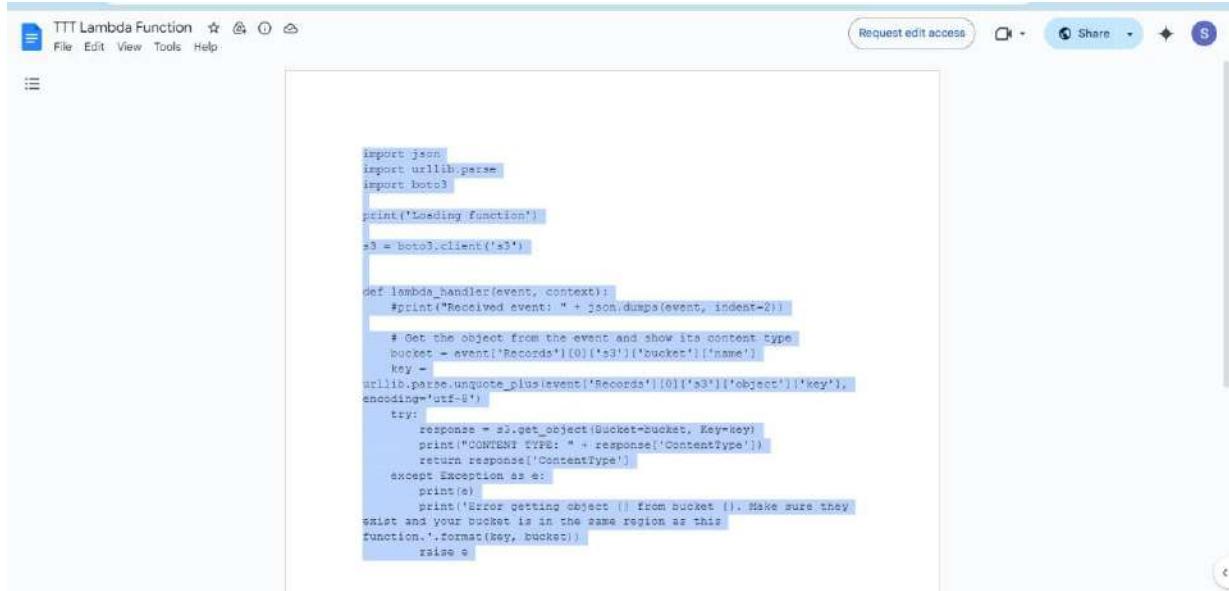
**Files and folders (1 Total, 781.2 KB)**

Name	Folder	Type	Size	Status	Error
Screenshot 2...	-	image/png	781.2 KB	Succeeded	-

Feedback © 2024 Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 14:43 03-10-2024



2. After creating the Lambda function copy a code available on the internet which allows the Lambda function to access the S3 bucket contents.



The screenshot shows the AWS Lambda function editor interface. The title bar says "TTT Lambda Function". The menu bar includes "File", "Edit", "View", "Tools", and "Help". On the right side, there are buttons for "Request edit access", "Share", and a save icon. The main area contains the following Python code:

```
import json
import urllib.parse
import boto3

print('Loading Function')
s3 = boto3.client('s3')

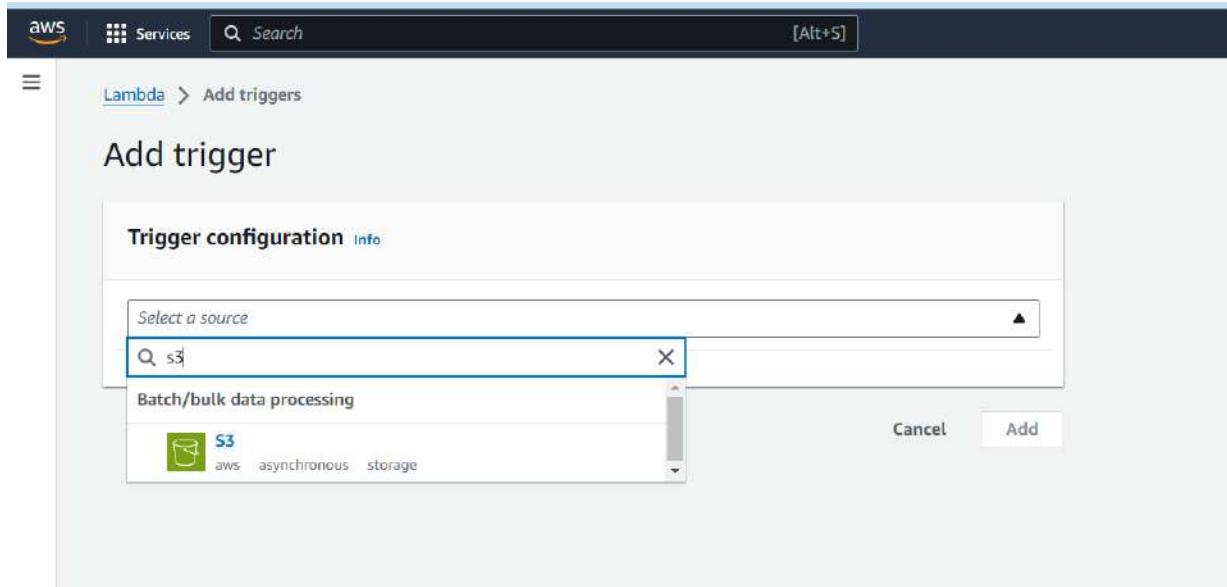
def lambda_handler(event, context):
    #print("Received event: " + json.dumps(event, indent=2))

    # Get the object from the event and show its content type
    bucket = event['Records'][0]['s3']['bucket']['name']
    key = urllib.parse.unquote_plus(event['Records'][0]['s3']['object']['key'],
                                    encoding='utf-8')
    try:
        response = s3.get_object(Bucket=bucket, Key=key)
        print("CONTENT TYPE: " + response['ContentType'])
        return response['ContentType']
    except Exception as e:
        print(e)
        print('Error getting object {} from bucket {}. Make sure they exist and your bucket is in the same region as this function.'.format(key, bucket))
        raise e
```

The screenshot shows the AWS Lambda code editor interface. At the top, there are tabs for Code, Test, Monitor, Configuration, Aliases, and Versions. The Code tab is selected. Below the tabs, there are buttons for Code source (with an info link), Upload from, and Test. The main area contains a code editor with the following Python code:

```
1 import json
2 import urllib.parse
3 import boto3
4
5 print('Loading function')
6
7 s3 = boto3.client('s3')
8
9
10 def lambda_handler(event, context):
11     #print("Received event: " + json.dumps(event, indent=2))
12
13     # Get the object from the event and show its content type
14     bucket = event['Records'][0]['s3']['bucket']['name']
15     key = urllib.parse.unquote_plus(event['Records'][0]['s3']['object']['key'], encoding='utf-8')
16     try:
17         response = s3.get_object(Bucket=bucket, Key=key)
18         print("CONTENT TYPE: " + response['ContentType'])
19         return response['ContentType']
20     except Exception as e:
21         print(e)
22         print("Error getting object {} from bucket {}. Make sure they exist and your bucket is in the same region as the Lambda function.".format(key, bucket))
23         raise e
24
```

3. Add a trigger to the Lambda function so any changes in the S3 bucket will be first visible to the user.



aws | Services | Search [Alt+S]

Lambda > Add triggers

## Add trigger

**Trigger configuration** Info

**S3** aws asynchronous storage

**Bucket**  
Choose or enter the ARN of an S3 bucket that serves as the event source. The bucket must be in the same region as the function.

Bucket region: eu-north-1

**Event types**  
Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual events cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.

All object create events

**Prefix - optional**  
Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters. Any special characters  must be URL encoded.

**Recursive invocation**  
If your function writes objects to an S3 bucket, ensure that you are using different S3 buckets for input and output. Writing to the same bucket increases the risk of creating a recursive invocation, which can result in increased Lambda usage and increased costs. [Learn more !\[\]\(1a0fb58894a69802a4d8f90a264d4c87\_img.jpg\)](#)

I acknowledge that using the same S3 bucket for both input and output is not recommended and that this configuration can cause recursive invocations, increased Lambda usage, and increased costs.

Lambda will add the necessary permissions for AWS S3 to invoke your Lambda function from this trigger. [Learn more !\[\]\(8fa3fcaa246ece64e9b2a0354c969a4b\_img.jpg\)](#) about the Lambda permissions model.

4. In the event notification of the S3 bucket we can see that it has been connected to the Lambda function .

The screenshot shows the 'Event notifications' section of the AWS S3 console. At the top, there is a message 'No data events to display.' and a 'Configure in CloudTrail' button. Below this, the 'Event notifications (1)' section is displayed. It contains one entry for a Lambda function. The table columns are 'Name', 'Event types', 'Filters', and 'Destination type'. The entry shows '905f180d-6a25-4474-941b-66671d74e4cd' as the Name, 'All object create events' as the Event types, no filters applied, and 'Lambda function' as the Destination type. There are 'Edit' and 'Delete' buttons for this entry. Below the table, there is a section titled 'Amazon EventBridge' with a note about using it for event-driven applications. At the bottom, there is a switch labeled 'Off' for sending notifications to Amazon EventBridge.

Name	Event types	Filters	Destination type
905f180d-6a25-4474-941b-66671d74e4cd	All object create events	-	Lambda function

**Amazon EventBridge**  
For additional capabilities, use Amazon EventBridge to build event-driven applications at scale using S3 event notifications. [Learn more](#) or [see EventBridge pricing](#).

Send notifications to Amazon EventBridge for all events in this bucket  
Off

Managed policy AWSLambdaBasicExecutionRole-Sa94e815-c025-4185-8c68-157a8a145ce0.statement

### Resource-based policy document

```
1 Version: "2012-10-17",
2   "Id": "default",
3   "Statement": [
4     {
5       "Sid": "lambda-f873ff0-bb23-44ff-a3a8-08ebd4e381d2",
6       "Effect": "Allow",
7       "Principal": {
8         "Service": "s3.amazonaws.com"
9       },
10      "Action": "lambda:InvokeFunction",
11      "Resource": "arn:aws:lambda:eu-north-1:869935102438:function:sanketlambda123",
12      "Condition": {
13        "StringEquals": {
14          "AWS:SourceAccount": "869935102438"
15        },
16        "ArnLike": {
17          "AWS:SourceArn": "arn:aws:s3:::sanketbucket123"
18        }
19      }
20    }
21  ]
22 ]
23
```

1:1 JSON Spaces: 2

[Close](#)

## 5. Upload a photo to the S3 bucket

Amazon S3 > Buckets > [sanketbucket123](#) > Upload

### 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**.

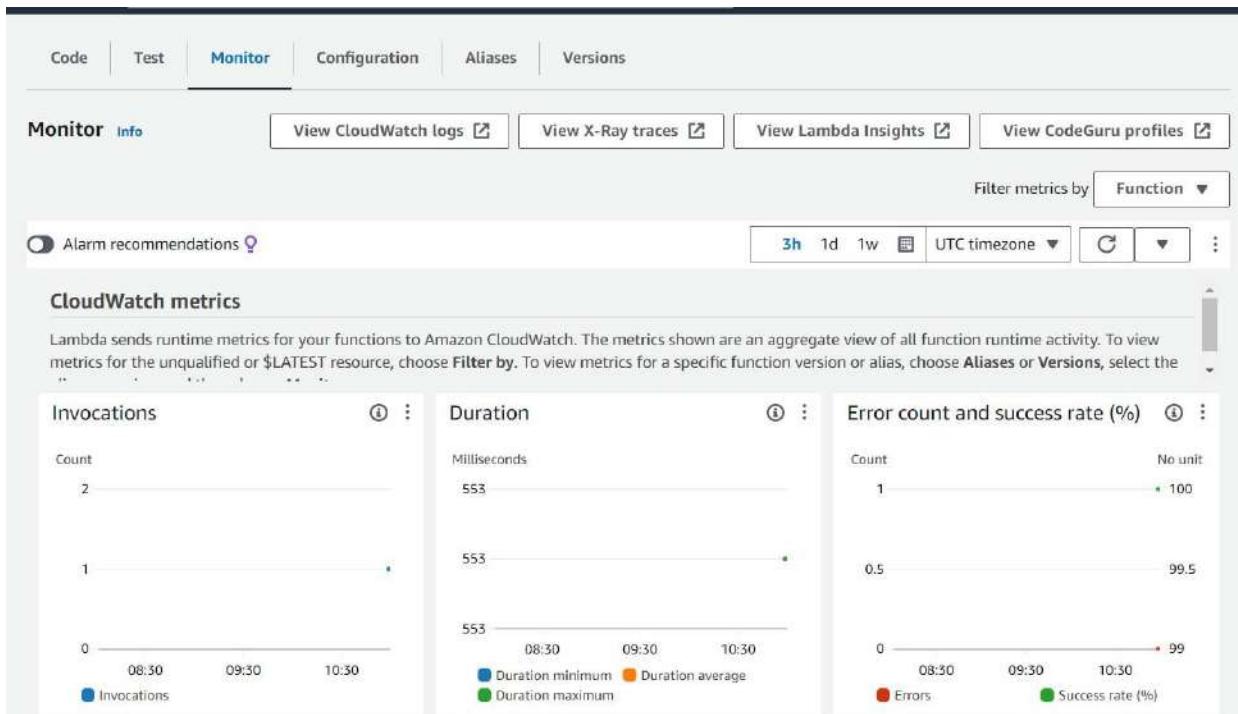
Files and folders (1 Total, 78.6 KB)		<a href="#">Remove</a>	<a href="#">Add files</a>	<a href="#">Add folder</a>
All files and folders in this table will be uploaded.				
<input type="text"/> <a href="#">Find by name</a>		< 1 >		
<input type="checkbox"/>	Name	Folder	Type	
<input type="checkbox"/>	photo 1.jpeg	-	image/jpeg	

### Destination [Info](#)

Destination  
[s3://sanketbucket123](#)

The screenshot shows the AWS Lambda console after a file has been uploaded. At the top, a green banner indicates "Upload succeeded" with a link to "View details below". Below the banner, a message states: "The information below will no longer be available after you navigate away from this page." The main section is titled "Summary" and shows the destination as "s3://sanketbucket123". It displays two rows: "Succeeded" with "1 file, 78.6 KB (100.00%)" and "Failed" with "0 files, 0 B (0%)". Below this, there are tabs for "Files and folders" (selected) and "Configuration". The "Files and folders" tab shows a table with one item: "sanket more ..." (image/jpeg, 78.6 KB, Succeeded). There is also a search bar labeled "Find by name".

6. Now run the function and in the cloud watch logs of AWS you can see the message printed and all the other details of the working of the Lambda function.



The screenshot shows the AWS Lambda console interface. At the top, there are several tabs: 'Upload objects - S3 bucket jai...', 'Course Invitation - d002j.json...', 'Launch AWS Academy Learner...', 'lambdaJaikunral | Functions | L...', and '+'. Below the tabs, the URL is 'us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/functions/lambdaJaikunral?tab=configure'. The main navigation bar includes 'Services' and 'Search' with a search bar containing '[Alt+S]'. The region is set to 'N. Virginia'. On the far right, the user information is 'vocabs/user3402861-NAVANI\_JAI\_DEEPAK\_HARSHIK @ 4823-2232-4237'.

The main content area shows the 'lambdaJaikunral' function. It has a 'Function overview' section with a 'Diagram' tab selected, showing a single node labeled 'LambdaJaikunral'. To the right, there is a 'Description' section with 'Last modified 15 minutes ago' and a 'Function ARN' field containing 'arn:aws:lambda:us-east-1:482322324237:function:lambdaJaikunral'. Below this is a 'Function URL' field with the value 'arn:aws:lambda:us-east-1:482322324237:lambdaJaikunral'. On the left, there are buttons for 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. On the right, there is a 'Tutorials' sidebar with a section titled 'Create a simple web app' and a link to 'Learn more'.

The screenshot shows the AWS CloudWatch Logs interface. The left sidebar has sections for 'Favorites and recent', 'Dashboards', 'Alarms', 'Logs' (selected), 'Log groups' (selected), 'Log Anomalies', 'Live Tail', 'Logs Insights', 'Contributor Insights', 'Metrics', 'X-Ray traces', 'Events', 'Application Signals', 'Network monitoring', 'Insights', and 'Settings'. The main content area shows the log group '/aws/lambda/sanketlambda123' with the log stream '2024/10/02[\$LATEST]8ed57b1dccf54ab8b05688935ed748db'. The 'Log events' section displays the following log entries:

Timestamp	Message
2024-10-02T10:59:56.489Z	INIT START Runtime Version: python:3.12.v36 Runtime Version ARN: arn:aws:lambda:eu-north-1:runtime:186d9ca2e3714ff5637bd2b...
2024-10-02T10:59:56.801Z	Loading function...
2024-10-02T10:59:57.172Z	START RequestId: df929631-f73a-46eb-8a07-56f2f4a810c8 Version: \$LATEST
2024-10-02T10:59:57.718Z	CONTENT TYPE: image/jpeg
2024-10-02T10:59:57.725Z	END RequestId: df929631-f73a-46eb-8a07-56f2f4a810c8
2024-10-02T10:59:57.725Z	REPORT RequestId: df929631-f73a-46eb-8a07-56f2f4a810c8 Duration: 552.91 ms Billed Duration: 553 ms Memory Size: 128 MB Max H...

No newer events at this moment. Auto retry paused. [Resume](#)

Kunal Mahesh Punjab.

DISA

Roll No: 44

Adv DevOps: Assignment No 1

(04/05)

Q.1

use S3 buckets and host video streaming.

→ To Host video streaming using S3 bucket on AWS, you can follow these steps.

Steps:

1) Login to AWS console

- Go to AWS management console
- Enter your login credentials.

2) Create an S3 bucket.

- In the console, Search for S3 in the search bar and select S3 from the result.
- Click Create bucket.
- Give your bucket a unique name ("My-video-streaming-video-bucket").
- Choose a region (closer to your audience).
- Scroll down and uncheck Block all Public access.
- Confirm by checking the acknowledgement box.
- Click Create bucket.

3) Upload your video file to S3.

- Click on your newly created bucket.
- Click the upload button.
- Add your video file from your computer.
- Click Upload to start the upload process.

#### 4) Set Permissions for Public Access.

- Once the video file is uploaded
- Select your video file in the S3 bucket.
- Use the Actions dropdown and choose make public
- Confirm the action by clicking make public again.

#### 5) Get the video URL

- After making the file Public, click on the video file.
- You will see a URL for the video under Object URL. This is direct link to your video.
- Copy this URL

Conclusion: Above method for bucket policy does not work

I personally used this method:-

- Go to Permission tab of your bucket.
- Click on the permission tab for my bucket.
- Edit bucket Policy
- Add the following Policy.

json

{

"Version": "2012-10-17",

"Statement": [ {

"Sid": "Public Read Get Object",

"Effect": "Allow",

"Principal": "\*",

"Action": "S3: Get Object",

"Resources": "arn:aws:s3:::my-video-streaming-bucket/\*"

}

}

Testing Public Access → final Step : After making ur <sup>video</sup> file public.  
Paste the URL into web browser to see video plays.

Personalized Recommendations: Hotstar uses AWS's machine learning tools to suggest content to users, making their viewing experience more enjoyable.

Key AWS Services used by BMW and Hotstar.

- Compute: Amazon EC2, Amazon Lambda
- Storage: Amazon S3, Amazon EBS
- Database: Amazon RDS, Amazon DynamoDB
- Networking: Amazon VPC, AWS Direct Connect
- Analytics: Amazon SageMaker, Amazon Rekognition

Born BMW and Hotstar Show how AWS can be valuable resources for different types of business. By using AWS, they have been able to innovate, grow and provide great experiences for their customers.

- Q.3  
- why Kubernetes and advantages and disadvantages  
Ob Kubernetes . Explain How adidas uses Kubernetes.

Answer:

Kubernetes is popular because it simplifies the management of containerized application. It automates tasks such as deployment, scaling and monitoring making it easier for organizations to manage their applications in a cloud environment.

Advantages of Kubernetes.

- 1) Portability
- 2) Scalability
- 3) Reliability
- 4) Efficiency.

Disadvantages of Kubernetes.

- 1) Complexity
- 2) Steep learning curve
- 3) Resource intensive
- 4) Management overhead.

Q. How adidas uses Kubernetes.

adidas has adopted Kubernetes to enhance its IT infrastructure and improve its resource utilization needs.

- 1) faster Application Development
- 2) Operational Efficiency.

3) Scalability for Demand.

4) Encouraging Innovation.

Q.4 what are Nagios and Explain How Nagios are used in E-Services.

→ Nagios is an open-source monitoring tool that helps organizations keep track of their IT infrastructure including it provides a way to ensure the system are running smoothly and alert us if any issues arise.

Key feature of Nagios.

1) Monitoring

2) Alerts

3) Reporting.

How Nagios is used in E-Services.

Nagios plays a vital role in the operation of e-services by ensuring the online systems are reliable and efficient.

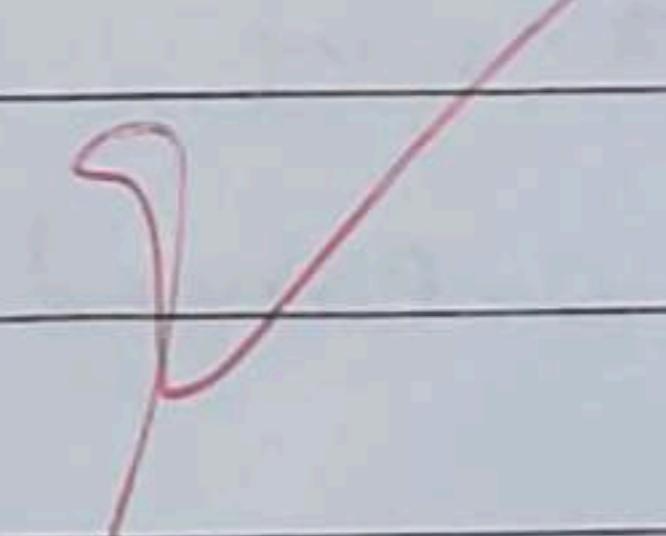
1) Infrastructure Monitoring

2) Service Availability

3) Performance Management

4) Incident Management

5) User Experience Monitoring



Pdu DevOps - Assignment 2,

04  
05

18

Create a Rest API with Serverless framework.  
Creating Rest APIs with Serverless framework is an efficient  
way to deploy Serverless application that can scale  
automatically without managing servers.

### (i) Serverless framework:

A powerful tool that deployment of services and serverless  
application across various cloud providers such as AWS,  
Azure and Google Cloud.

(ii) Serverless architecture : This design model allows developers  
to build application without worrying about underlying  
infrastructure enabling focus on code and business logic.

(iii) Rest API : Representation State Transfer is architecture style for  
designing application.

Steps for creating REST API for Serverless framework.

#### 1) Install Serverless framework:

You start by installing Serverless framework globally using  
Node Package Manager (NPM) this allow you to manage  
Serverless application directly from your terminal.

#### 2) Creating a Node.js Serverless project.

A directory is created for your project where you initialize a  
Serverless service using the command Serverless and you  
set up a template for AWS node.js middware that will  
eventually deploy to AWS Lambda.

Project Structure : The Project contains essential files  
like handler.js (which contains) code for Lambda  
functions and serverless.yml.

Create a Rest API Response.

### 5) Deploy the Service:

With the SIS deploy command, Serverless framework packages your application uploads, all necessary resources or AWS on. Set up the infrastructure.

### 6) Testing the API:

Once deployed you can test REST API using tools like curl or Postman by making POST requests to generated API.

### 7) Storing data in DynamoDB:

To store submitted candidate data you integrate AWS DynamoDB as a database.

### 8) Adding more functionalities like list all candidates or candidate by ID

### a) AWS IAM Permission

You need to ensure that Serverless framework is given right permission to interact with AWS resources like DynamoDB.

### 10) Monitoring and maintenance.

After Deployment Serverless framework provides service information like deployed endpoint, API key, log structure.

case Study for SonarQube.

SonarQube is an open source platform used for continuous inspection of quality in code bugs, risks, smells and security vulnerabilities in projects across various programming languages.

### Profile creation in SonarQube

Quality profiles in SonarQube are essential configuration that defines rules applied during code analysis. Each project has a quality profile for every supported language with default being 'Sonar way' profile comes built in for all languages. Custom profiles can be created by copying or extending existing ones.

Copying creates an independent profile, while extending existing ones. Copying creates an independent profile, while extending inherits rules from parent profile and reflects future changes automatically. You can activate or deactivate rules prioritized down rules to specific projects. Permissions to manage quality profile are restricted. To ensure

profiles include new rules it's important to check against uploaded built in profiles or use SonarQube rules page.

### using Sonar Cloud to analyze GitHub code!

Sonar Cloud is cloud-based counterpart of SonarQube that integrates directly with GitHub, Bitbucket, Azure and GitHub repositories. To get started with SonarCloud via GitHub Signup via Sonar Cloud product page and connect your GitHub organization or personal account once connected, Sonar Cloud mirrors your GitHub setup with each project corresponding to GitHub repository. After setting up the organization choose subscription plan and import repositories into your SonarCloud organization where each

5) If your project becomes a SonarCloud project. Define 'new code' to focus on recent changes and choose between automatic analysis or CI-based analysis. Automatic analysis happens directly in SonarCloud, while CI-based analysis integrates with your build process once the analysis is complete. Results can be viewed in both SonarCloud and GitHub including security important issues.

### 3) Sonarlint in JAVA IDE:

Sonarlint is an IDE that performs on-the-fly code analysis as you write code. It helps developers detect bugs, security vulnerabilities and code smells directly in the development environment such as IntelliJ IDEA or Eclipse. To set it up install the Sonarlint plugin, configure the connection with SonarQube or SonarCloud and select the project profile to analyze Java code. This approach ensures immediate feedback on code quality, promoting clean and maintainable code from beginning.

### 4) Analyzing Python Projects with SonarQube:

SonarQube supports Python for coverage reporting but requires third-party tools like CoveragePy to generate coverage reports in XML format. The build process can also be automated using GitHub Actions, which install dependencies, run tests and invoke SonarQube Scan. Ensure report in Cobertura XML format and place where Scanner can access it.

Analyse Node.js Project with SonarQube  
 For Node.js projw SonarQube can analyze JavaScript and  
 TypeScript code. Similar to the Python Script, you can  
 for configure SonarQube to analyze Node.js projects  
 by installing the appropriate plugin and using Sonar  
 Scanner to Scan the Projects. SonarQube will check  
 the code against industry Standard rules and best  
 practices, flagging issues related to security vulnerabilities  
 bugs and performance optimization.

At a large organization your centralized operation team may  
 get many repetitive infrastructure requests. You can  
 use Terraform to build a self-service infrastructure  
 model that lets produce them manage their own  
 infrastructure independently. You can create and use Terraform  
 modules that codify the standards for deploying  
 Terraform Cloud can also integrate with ticketing system  
 like ServiceNow to automatically generate new  
 infrastructure requests.

Implementing a self-service infrastructure model using Terraform  
 can transform how large organization manage their  
 infrastructure. Independent organization can enhance efficiency  
 reduce bottlenecks and comply with established needs.

The need for self-service infrastructure: In large organization  
 centralizing operations teams often face an overwhelming number  
 of repetitive requests. This can lead to delay in  
 delivery and move quickly.

A self-service model allows teams to provision and manage their infrastructure without an operator to handle every request.

Benefits of using Terraform:

1) Modularity and Reusability:

Terraform modules encapsulate standard configuration for various infrastructure components like networks, databases, computer resources.

Teams can reuse these modules across different projects, reducing redundancy and minimizing the risk of errors.

2, Standardization:

By defining best practices within modular organization, you can ensure that all deployments comply with organizational policies and standards.

This consistency helps maintain security and operational integrity across the organization.

3) Integration with DevOps Systems: Terraform Cloud can integrate with DevOps systems like GitHub, Jenkins, to automate the generation of their delivery platform, reducing manual intervention.

# Implementation Steps

Identifying infrastructure components within your infrastructure group, load balancers.

Develop Terraform modules.

Create & Use + Modules  
And resources.

desire configuration

Ensure each module includes input variables for  
customization and output for integration with other

## Establish Growth and Best Product:

Define product: guidelines for modular way, versioning and documentation to ensure cloning and maintaining.

# Testing and Validation

~~Implement a factory for no Validators medium function before the IPmer.~~

# BEST practices for Module management

Utilize the transform region

learning existing communities  
and Terraform Registry  
sure on best practices  
to avoid reusing Sutin's  
modules from the  
learning existing communities  
and Terraform Registry  
sure on best practices  
to avoid reusing Sutin's  
modules from the

### Version control:

Implement versioning for your modules to track changes. Over time, this helps manage dependencies efficiently, minimizing disruptions during updates.

Encouraging collaboration fosters a culture of collaboration. Sharing modules across teams. This promotes consistency in deployments and by adopting a self-service infrastructure model within your organization can empower product teams to efficiency more. An approach not only streamlines processing but also agility in responding to changing business needs. Ultimately, it leads to a more responsive environment that supports innovation and growth within the organization.

