

Jai Navani 31 D15A

Experiment-1

Screenshots for hosting webpage on EC2 instance :-

Scanning candidates...
scanning linux images...

Pending kernel upgrade!
Running kernel version:
6.8.0-1009-aws
Diagnostics:
The currently running kernel version is not the expected kernel version 6.8.0-1013-aws.
Restarting the system to load the new kernel will not be handled automatically, so you should consider rebooting.

Restarting services...
systemctl restart multipathd.service polkit.service rsyslog.service udisks2.service

Service restarts being deferred:
systemctl restart ModemManager.service
/etc/needrestart/restart.d/dbus.service
systemctl restart networkd-dispatcher.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
ubuntu @ session #1: apt[1706], bash[1374], sshd[857]
ubuntu @ user manager service: systemd[1204]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-42-7:~#

CloudShell Feedback

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

E: Package 'yum' has no installation candidate
root@ip-172-31-42-7:~# apt install apache2

root@ip-172-31-42-7:~# 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-sqlite3 libaprutil1-ldap libaprutil1t64 liblua5.4-0 ssl-cert
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-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 liblua5.4-0 ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 0 not upgraded.
Need to get 2083 kB of archives.
After this operation, 8094 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libapr1t64 amd64 1.7.2-3.1build2 [107 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1t64 amd64 1.6.3-1.1ubuntu7 [91.9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.3-1.1ubuntu7 [111.2 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-ldap amd64 1.6.3-1.1ubuntu7 [9116 B]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 liblua5.4-0 amd64 5.4.6-3build2 [166 kB]
Progress: [98%] #####
...
.1 :7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-data all 2.4.58-1ubuntu8.4 [163 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-utils amd64 2.4.58-1ubuntu8.4 [97.1 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2 amd64 2.4.58-1ubuntu8.4 [90.2 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 ssl-cert all 1.1.2ubuntul [17.8 kB]
Fetched 2083 kB in 0s (25.5 MB/s)
Preconfiguring packages ...
Scanning processes...
root@ip-172-31-42-7:~# outdated hypervisor (qemu) binaries on this host.ally, so you should consider rebooting.

CloudShell Feedback

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

AWS CloudShell terminal window showing the download of `index.html` from `https://d15a-jai-61.github.io/IP-Exp2/`. The terminal output includes the command `wget`, the progress bar for the download, and the final message indicating the file was saved.

```

root@ip-172-31-42-7:~# mkdir temp
root@ip-172-31-42-7:~# ls
snap temp
root@ip-172-31-42-7:~# cd temp
root@ip-172-31-42-7:~/temp# wget https://d15a-jai-61.github.io/IP-Exp2/
--2024-08-12 11:00:29- https://d15a-jai-61.github.io/IP-Exp2/
Resolving d15a-jai-61.github.io (d15a-jai-61.github.io)... 185.199.108.153, 185.199.109.153, 185.199.110.153, ...
Connecting to d15a-jai-61.github.io (d15a-jai-61.github.io)|185.199.108.153|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 4359 (4.3K) [text/html]
Saving to: 'index.html'

index.html          100%[=====] 4.26K --.-KB/s   in 0s

2024-08-12 11:00:29 (41.9 MB/s) - 'index.html' saved [4359/4359]

root@ip-172-31-42-7:~/temp# 

```

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

AWS CloudShell terminal window showing the contents of the `temp` directory. It contains a single file, `index.html`.

```

root@ip-172-31-42-7:~/temp# ls -lrt
total 8
-rw-r--r-- 1 root root 4359 Aug 12 09:23 index.html
root@ip-172-31-42-7:~/temp# 

```

AWS CloudShell terminal window showing the start and status of the Apache2 service. The service starts successfully, and the log shows the server starting and being active.

```

root@ip-172-31-42-7:~# cd temp
root@ip-172-31-42-7:~/temp# ls
index.html
root@ip-172-31-42-7:~/temp# service apache2 start
root@ip-172-31-42-7:~/temp# service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Mon 2024-08-12 10:47:34 UTC; 51min ago
     Docs: http://httpd.apache.org/docs/2.4/
   Process: 493 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 519 (apache2)
    Tasks: 55 (limit: 1130)
   Memory: 7.8M (peak: 8.0M)
      CPU: 203ms
     CGroup: /system.slice/apache2.service
             └─519 /usr/sbin/apache2 -k start
                  ├─520 /usr/sbin/apache2 -k start
                  ├─521 /usr/sbin/apache2 -k start
                  └─522 /usr/sbin/apache2 -k start

Aug 12 10:47:33 ip-172-31-42-7 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Aug 12 10:47:34 ip-172-31-42-7 systemd[1]: Started apache2.service - The Apache HTTP Server.
root@ip-172-31-42-7:~/temp# 

```

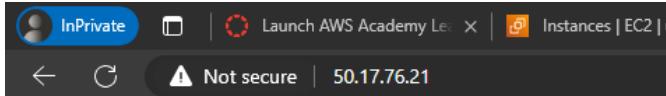
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS Cloud interface for managing security group inbound rules. The top navigation bar includes the AWS logo, 'Services' (with a dropdown menu), a search bar, keyboard shortcuts [Alt+S], and account information for 'N. Virginia'. Below the navigation is a breadcrumb trail: EC2 > Security Groups > sg-0234926154e2078d6 > Edit inbound rules. The main title 'Edit inbound rules' is displayed with an 'Info' link. A descriptive text below states: 'Inbound rules control the incoming traffic that's allowed to reach the instance.' The central area is titled 'Inbound rules' with its own 'Info' link. It contains a table with columns: Security group rule ID, Type, Protocol, Port range, Source, and Description - optional. Two rules are listed:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0a05c7d3f2d659b92	SSH	TCP	22	Cust... ▾	<input type="text" value="0.0.0.0/0"/> X
sgr-00956191191b304fc	HTTP	TCP	80	Cust... ▾	<input type="text" value="0.0.0.0/0"/> X

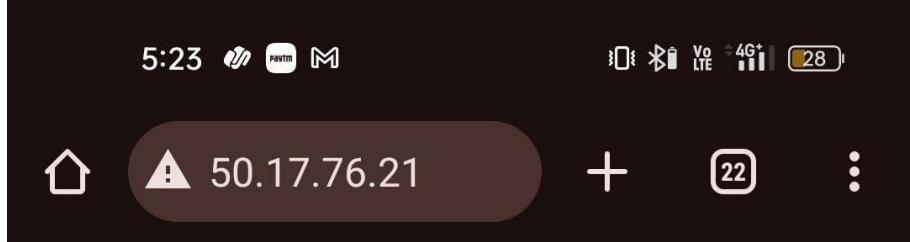
A large 'Add rule' button is located at the bottom left. A warning message at the bottom right states: '⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.' with a close button.

A screenshot of a web browser showing a Petco website. The page features a large banner with the text "Your pet deserves the best treatment in the world !". Below the banner is a photograph of a man in a white t-shirt and grey shorts kneeling on a wooden chair, petting a brown and white dog. The background shows a textured wall and a dark door with a white curtain to the right.



Petco

Your pet deserves the beat treatment in t...



Petco

Your pet deserves the best treatment in the world !



About us

Petco is a category-defining health and wellness company focused on improving the lives of pets, pet parents and our own Petco partners. Since our founding in 1965, we've been trailblazing new standards in pet care, delivering comprehensive wellness solutions through our products and services, and creating communities that deepen the pet-pet parent bond.

We employ more than 29,000 partners nationwide and operate more than 1,500 Petco locations across the U.S., Mexico and Puerto Rico — including a growing network of more than 200 in-store veterinary hospitals — and offer a complete online resource for pet health and wellness at petco.com and on the Petco app.

In tandem with Petco Love, an independent nonprofit organization, we have helped find homes for more than 7 million animals through in-store adoption events.

Working at Petco



Experiment-2

Screenshots for setting up and deploying Elastic Beanstalk application :-

Configure environment

Environment tier

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

Application name

Maximum length of 100 characters.

▶ Application tags (optional)

Step 1 Configure environment

Step 2 Configure service access

Step 3 - optional Set up networking, database, and tags

Step 4 - optional Configure instance traffic and scaling

Step 5 - optional Configure updates, monitoring, and logging

Step 6 Review

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

Platform branch

Platform version

Application code

Sample application

Existing version
Application versions that you have uploaded.

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

Presets Info

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

Configuration presets

- Single instance (free tier eligible)
- Single instance (using spot instance)
- High availability
- High availability (using spot and on-demand instances)
- Custom configuration

Cancel**Next****Service access**

IAM roles, assumed by Elastic Beanstalk as a service role, and EC2 instance profiles allow Elastic Beanstalk to create and manage your environment. Both the IAM role and instance profile must be attached to IAM managed policies that contain the required permissions. [Learn more](#) 

Service role

- Create and use new service role
- Use an existing service role

Existing service roles

Choose an existing IAM role for Elastic Beanstalk to assume as a service role. The existing IAM role must have the required IAM managed policies.




EC2 key pair

Select an EC2 key pair to securely log in to your EC2 instances. [Learn more](#) 




EC2 instance profile

Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.




View permission details**Cancel****Skip to review****Previous****Next**

Set up networking, database, and tags - optional [Info](#)

Virtual Private Cloud (VPC)

VPC

Launch your environment in a custom VPC instead of the default VPC. You can create a VPC and subnets in the VPC management console.

[Learn more](#)

vpc-0b4df95bddc923aea | (172.31.0.0/16)

-

vpc-0b4df95bddc923aea | (172.31.0.0/16)

Instance settings

Choose a subnet in each AZ for the instances that run your application. To avoid exposing your instances to the Internet, run your instances in private subnets and load balancer in public subnets. To run your load balancer and instances in the same public subnets, assign public IP addresses to the instances. [Learn more](#)

Public IP address

Assign a public IP address to the Amazon EC2 instances in your environment.

Activated

Instance subnets

Instance settings

Choose a subnet in each AZ for the instances that run your application. To avoid exposing your instances to the Internet, run your instances in private subnets and load balancer in public subnets. To run your load balancer and instances in the same public subnets, assign public IP addresses to the instances. [Learn more](#)

Public IP address

Assign a public IP address to the Amazon EC2 instances in your environment.

Activated

Instance subnets

Filter instance subnets

	Availability Zone	Subnet	▲	CIDR	Name
<input checked="" type="checkbox"/>	us-east-1e	subnet-0098f3bf4...		172.31.48.0/20	
<input type="checkbox"/>	us-east-1f	subnet-03b751cf5...		172.31.64.0/20	
<input checked="" type="checkbox"/>	us-east-1a	subnet-05ff6de98...		172.31.16.0/20	
<input type="checkbox"/>	us-east-1c	subnet-089dbc2d1...		172.31.0.0/20	
<input type="checkbox"/>	us-east-1b	subnet-0d4591b4f...		172.31.32.0/20	
<input type="checkbox"/>	us-east-1d	subnet-0f341fff62...		172.31.80.0/20	

	Name	Subnet ID	CIDR Block
<input type="checkbox"/>	us-east-1f	subnet-03b751cf5...	172.31.64.0/20
<input type="checkbox"/>	us-east-1a	subnet-05ff6de98...	172.31.16.0/20
<input type="checkbox"/>	us-east-1c	subnet-089dbc2d1...	172.31.0.0/20
<input type="checkbox"/>	us-east-1b	subnet-0d4591b4f...	172.31.32.0/20
<input type="checkbox"/>	us-east-1d	subnet-0f341fff62...	172.31.80.0/20

Enable database

Restore a snapshot - *optional*

Restore an existing snapshot from a previously used database.

Snapshot

None

Database settings

Choose an engine and instance type for your environment's database.

Engine

Engine version

Tags

Apply up to 50 tags. You can use tags to group and filter your resources. A tag is a key-value pair. The key must be unique within the resource and is case-sensitive. [Learn more](#) 

No tags associated with the resource.

[Add new tag](#)

You can add 50 more tags.

[Cancel](#)

[Skip to review](#)

[Previous](#)

[Next](#)

Configure instance traffic and scaling - optional Info

▼ Instances Info

Configure the Amazon EC2 instances that run your application.

Root volume (boot device)

Root volume type

General Purpose (SSD) ▲

(Container default)

Magnetic

hed to each instance.

General Purpose (SSD) ✓ GB

General Purpose 3(SSD)

General Purpose (SSD)

Provisioned IOPS (SSD)

eded IOPS (SSD) volume.

▼ IOPS

Throughput

The desired throughput to provision for the Amazon EBS root volume attached to your environment's EC2 instance

125

MiB/s

Root volume type

General Purpose (SSD) ▼

Size

The number of gigabytes of the root volume attached to each instance.

8

GB

IOPS

Input/output operations per second for a provisioned IOPS (SSD) volume.

▼

IOPS

Throughput

The desired throughput to provision for the Amazon EBS root volume attached to your environment's EC2 instance

125

▼

MiB/s

Amazon CloudWatch monitoring

The time interval between when metrics are reported from the EC2 instances

Monitoring interval

5 minute ▼

Instance metadata service (IMDS)

Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. [Learn more](#)

IMDSv1

With the current setting, the environment enables only IMDSv2.

Deactivated

EC2 security groups

Select security groups to control traffic.

EC2 security groups (3)

C

Filter security groups

<input type="checkbox"/>	Group name	▲	Group ID	▼	Name	▼
<input checked="" type="checkbox"/>	default		sg-0b6f2684b3eea9987			
<input type="checkbox"/>	launch-wizard-1		sg-0234926154e2078d6			
<input type="checkbox"/>	launch-wizard-2		sg-0f76834772eb69439			

▼ Capacity [Info](#)

Configure the compute capacity of your environment and auto scaling settings to optimize the number of instances used.

Auto scaling group

Environment type

Select a single-instance or load-balanced environment. You can develop and test an application in a single-instance environment to save costs and then upgrade to a load-balanced environment when the application is ready for production. [Learn more](#)

Single instance



Instances

1



Min

1



Max

Fleet composition

Spot Instances are launched at the lowest available price. [Learn more](#)

- On-Demand instance
- Spot instance

Maximum spot price

The maximum price per instance-hour, in USD, that you're willing to pay for a Spot Instance. Setting a custom price limits your chances to fulfill your target capacity using Spot instances.

- Default
- Set your maximum price

Instance types

Add instance types for your fleet. Change the order that the instances are in to set the preferred launch order. This only affects On-Demand instances. We recommend you include at least two instance types. [Learn more](#)

Choose x86 instance types ▾

t2.small X

AMI ID

Elastic Beanstalk selects a default Amazon Machine Image (AMI) for your environment based on the Region, platform version, and processor architecture that you choose. [Learn more](#)

ami-0b4a9cc2fba693a25

Availability Zones

Number of Availability Zones (AZs) to use.

Any ▾

Placement

Specify Availability Zones (AZs) to use.

Choose Availability Zones (AZs) ▾

Scaling cooldown

360 ▾ seconds

Cancel

Skip to review

Previous

Next

Root volume (boot device)**Root volume type**

General Purpose (SSD) ▾

Size

The number of gigabytes of the root volume attached to each instance.

8 ▾ GB

✖ Size must be between 10 and 16384.

Root volume type

General Purpose (SSD) ▾

Size

The number of gigabytes of the root volume attached to each instance.

16 ▾ GB

Configure updates, monitoring, and logging - optional Info

▼ Monitoring Info

Health reporting

Enhanced health reporting provides free real-time application and operating system monitoring of the instances and other resources in your environment. The **EnvironmentHealth** custom metric is provided free with enhanced health reporting. Additional charges apply for each custom metric. For more information, see [Amazon CloudWatch Pricing](#)

System

- Basic
- Enhanced

Health event streaming to CloudWatch Logs

Configure Elastic Beanstalk to stream environment health events to CloudWatch Logs. You can set the retention up to a maximum of ten years and configure Elastic Beanstalk to delete the logs when you terminate your environment.

Log streaming

- Activated (standard CloudWatch charges apply.)

Retention

7



Lifecycle

▼ Managed platform updates Info

Activate managed platform updates to apply platform updates automatically during a weekly maintenance window that you choose. Your application stays available during the update process.

Managed updates

- Activated

Weekly update window

Tuesday at 01 : 14 UTC

Update level

Minor and patch

Instance replacement

If enabled, an instance replacement will be scheduled if no other updates are available.

- Activated

Activated

Instance log streaming to CloudWatch logs

Configure the instances in your environment to stream logs to CloudWatch logs. You can set the retention to up to 10 years and configure Elastic Beanstalk to delete the logs when you terminate your environment. [Learn more](#)

Log streaming

(standard CloudWatch charges apply.)

Activated

Retention

7



Lifecycle

Keep logs after terminating envir...



Environment properties

The following properties are passed in the application as environment properties. [Learn more](#)

No environment properties have been configured.

[Add environment property](#)

[Cancel](#)

[Previous](#)

[Next](#)

Elastic Beanstalk application created, review screen before finalization :-

Review [Info](#)

Step 1: Configure environment

[Edit](#)

Environment information

Environment tier	Application name
Web server environment	D15A-Jai-61
Environment name	Application code
D15A-Jai-61-env	Sample application
Platform	arn:aws:elasticbeanstalk:us-east-1::platform/Node.js 20
	running on 64bit Amazon Linux 2023/6.2.0

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.

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.

No options configured

Step 3: Set up networking, database, and tags

[Edit](#)

Networking, database, and tags [Info](#)

Configure VPC settings, and subnets for your environment's EC2 instances and load balancer. Set up an Amazon RDS database that's integrated with your environment.

Network

VPC	Public IP address	Instance subnets
vpc-0b4df95bddc923aea	true	subnet-0098f3bf430c41040,subnet-05ff6de9850aa5577

Tags

Step 4: Configure instance traffic and scaling**Edit****Instance traffic and scaling** [Info](#)

Customize the capacity and scaling for your environment's instances. Select security groups to control instance traffic. Configure the software that runs on your environment's instances by setting platform-specific options.

Instances

Root volume type	Instance size	IMDSv1
gp2	16	Deactivated

EC2 Security Groups

sg-0b6f2684b3eea9987

Capacity

Environment type	Fleet composition	On-demand base
Single instance	On-Demand instance	0
On-demand above base	Capacity rebalancing	Scaling cooldown
0	Deactivated	360
Processor type	Instance types	AMI ID
x86_64	t2.small	ami-0b4a9cc2fba693a25

Step 5: Configure updates, monitoring, and logging**Edit****Updates, monitoring, and logging** [Info](#)

Define when and how Elastic Beanstalk deploys changes to your environment. Manage your application's monitoring and logging settings, instances, and other environment resources.

Monitoring

System	Cloudwatch custom metrics - instance	Cloudwatch custom metrics - environment
basic	—	—
Log streaming	Retention	Lifecycle
Deactivated	7	false

Updates

Managed updates	Deployment batch size	Deployment batch size type
Deactivated	100	Percentage
Command timeout	Deployment policy	Health threshold
600	AllAtOnce	Ok

Ignore health check

Instance replacement

Ignore health check	Instance replacement							
false	false							
Platform software								
Lifecycle	Log streaming	Proxy server						
false	Deactivated	nginx						
Logs retention	Rotate logs	Update level						
7	Deactivated	minor						
X-Ray enabled								
Deactivated								
Environment properties								
<table border="1"> <thead> <tr> <th>Key</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td colspan="2" style="text-align: center;">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 type="button" value="Submit"/>								

AWS Services Search [Alt+S] Mumbai Supra003

WebServerPipe Pipeline type: V2 Execution mode: QUEUED

Source Succeeded Pipeline execution ID: [ae246baaf-df87-463f-925d-7d5d2ee71640](#)

Source GitHub (Version 2) Succeeded - 1 minute ago View details

Deploy Succeeded Pipeline execution ID: [ae246baaf-df87-463f-925d-7d5d2ee71640](#)

AWS Elastic Load Balancer Succeeded - Just now View details

Notify Edit Stop execution Clone pipeline Release change

Developer Tools CodePipeline

- Source • CodeCommit
- Artifacts • CodeArtifact
- Build • CodeBuild
- Deploy • CodeDeploy
- Pipeline • CodePipeline

Getting started Pipelines History Settings

Go to resource Feedback

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

AWS Services Search [Alt+S] Mumbai Supra003 ⓘ

Elastic Beanstalk

Elastic Beanstalk > Applications > WebServer

Application WebServer environments (1) Info

Actions Create new environment

Filter environments < 1 > ⚙️

Environment name	Health	Date created	Domain	Running vers
WebServer-env	Green	August 17, 2024 22:...	WebServer-env.eba-227p9xyx...	code-pipeline

Application: WebServer

- Application versions
- Saved configurations

Recent environments

- WebServer-env
- WebApp02-env
- SupraApp-env-1
- MyFirstApp-env

https://ap-south-1.console.aws.amazon.com/elasticbeanstalk/home?region=ap-south-1# © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Environment successfully launched. ⌂ X ⓘ

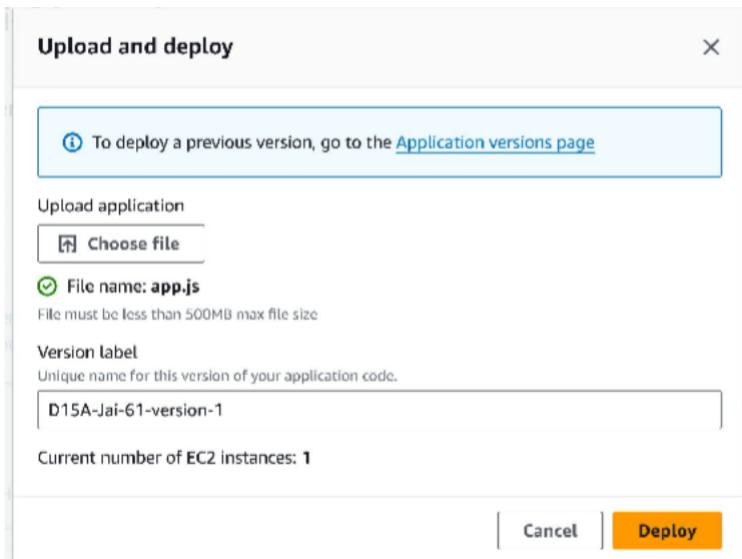
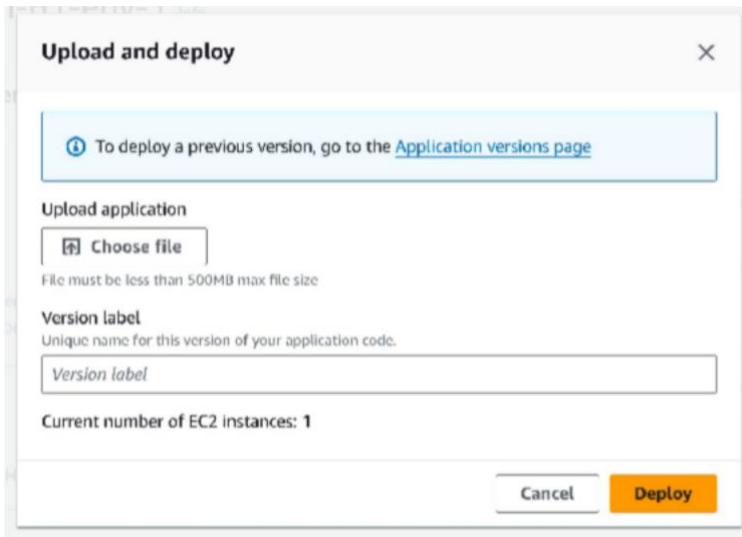
Events Health Logs Monitoring Alarms Managed updates Tags

Events (7) Info

Filter events by text, property or value < 1 > ⚙️

Time	Type	Details
August 20, 2024 20:01:18 (UTC+5:30)	INFO	Successfully launched environment: D15A-Jai-61-env-1
August 20, 2024 20:00:14 (UTC+5:30)	INFO	Instance deployment completed successfully.
August 20, 2024 19:59:01 (UTC+5:30)	INFO	Waiting for EC2 instances to launch. This may take a few minutes.
August 20, 2024 19:57:58 (UTC+5:30)	INFO	Created EIP: 23.21.64.185
August 20, 2024 19:57:43 (UTC+5:30)	INFO	Created security group named: sg-003edb017065a12ed
August 20, 2024 19:57:22 (UTC+5:30)	INFO	Using elasticbeanstalk-us-east-1-567270636093 as Amazon S3 storage bucket for environment data.
August 20, 2024 19:57:21 (UTC+5:30)	INFO	createEnvironment is starting.

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



The screenshot shows a confirmation page with a green header bar. The main content area has a large 'Congratulations' heading. Below it, a message states: 'Your first AWS Elastic Beanstalk Node.js application is now running on your own dedicated environment in the AWS Cloud'. A smaller note at the bottom says: 'This environment is launched with Elastic Beanstalk Node.js Platform'. To the right, a dark sidebar titled 'What's Next?' lists several links:

- [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](#)

Experiment no:3
Advance-devops
Jai navani
D15-A

Step 1: create instances (1 master and 2 nodes)

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with links like EC2 Dashboard, EC2 Global View, Events, Console-to-Code Preview, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, Reservations (New), Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes, Snapshots). The main content area displays a table titled 'Instances (3) Info' with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IP. The instances listed are: node-2 (i-0045fc812ca071756, Running, t2.medium, 2/2 checks passed, View alarms, us-east-1a, ec2-44-20-123-45-67-89), Master (i-0e812dd807ae2d49, Running, t2.medium, 2/2 checks passed, View alarms, us-east-1a, ec2-54-21-123-45-67-89), and node-1 (i-0384c7c6bd2e91fb8, Running, t2.medium, 2/2 checks passed, View alarms, us-east-1a, ec2-54-84-123-45-67-89).

Connect the instances:

```
Amazon Linux 2
AL2 End of Life is 2025-06-30.

A newer version of Amazon Linux is available!
Amazon Linux 2023, GA and supported until 2028-03-15.
https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-81-4 ~]$ sudo -i
[root@ip-172-31-81-4 ~]# yum update
```

Install docker in all instances

```
[root@ip-172-31-88-48 ec2-user]# yum install docker -y
Last metadata expiration check: 0:04:33 ago on Sun Sep 22 07:37:42 2024.
Dependencies resolved.

=====
| Package           | Architecture | Version      | Repository | Size   |
|=====             |=====         |=====        |=====       |=====  |
| Installing:      |              |              |            |        |
| docker           | x86_64       | 25.0.6-1.amzn2023.0.2 | amazonlinux | 44 M  |
|=====             |              |              |            |        |
| Installing dependencies: |
| containerd        | x86_64       | 1.7.20-1.amzn2023.0.1 | amazonlinux | 35 M  |
| iptables-libs     | x86_64       | 1.8.8-3.amzn2023.0.2 | amazonlinux | 401 k |
| iptables-nft      | x86_64       | 1.8.8-3.amzn2023.0.2 | amazonlinux | 103 k |
| libcgroup         | x86_64       | 3.0-1.amzn2023.0.1   | amazonlinux | 75 k  |
| libnetfilter_conntrack | x86_64 | 1.0.8-2.amzn2023.0.2 | amazonlinux | 58 k  |
| libnftnl          | x86_64       | 1.0.1-19.amzn2023.0.2 | amazonlinux | 30 k  |
| libnftnl          | x86_64       | 1.2.2-2.amzn2023.0.2 | amazonlinux | 84 k  |
| pigz              | x86_64       | 2.5-1.amzn2023.0.3   | amazonlinux | 83 k  |
| runc              | x86_64       | 1.1.13-1.amzn2023.0.1 | amazonlinux | 3.2 M |
|=====             |              |              |            |        |

Transaction Summary
=====
Install 10 Packages

Total download size: 84 M
Installed size: 317 M
```

After installation start the docker:

```
Installed:
  containerd-1.7.20-1.amzn2023.0.1.x86_64      docker-25.0.6-1.amzn2023.0.2.x86_64      iptables-libs-1.8.8-3.amzn2023.0.2.x86_64
  iptables-nft-1.8.8-3.amzn2023.0.2.x86_64    libcgroup-3.0-1.amzn2023.0.1.x86_64      libnftnl-1.2.2-2.amzn2023.0.2.x86_64
  libnftnl-1.0.1-19.amzn2023.0.2.x86_64        libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64
  runc-1.1.13-1.amzn2023.0.1.x86_64            pigz-2.5-1.amzn2023.0.3.x86_64

Complete!
[root@ip-172-31-88-48 ec2-user]# systemctl start docker
[root@ip-172-31-88-48 ec2-user]#
[root@ip-172-31-88-48 ec2-user]#
[root@ip-172-31-88-48 ec2-user]#
[root@ip-172-31-88-48 ec2-user]# systemctl start docker
```

Code for installation of kubernetes:

```
cat <<EOF | sudo tee /etc/yum.repos.d/kubernetes.repo

[kubernetes]
name=Kubernetes

baseurl=https://pkgs.k8s.io/core:/stable:/v1.31/rpm/
enabled=1
gpgcheck=1
gpgkey=https://pkgs.k8s.io/core:/stable:/v1.31/rpm/repo/repodata/repomd.xml.key
exclude=kubelet kubeadm kubectl cri-tools kubernetes-cni
EOF

# Set SELinux in permissive mode (effectively disabling it)

sudo setenforce 0
```

```

sudo sed -i 's/^SELINUX=enforcing$/SELINUX=permissive/'
/etc/selinux/config

sudo yum install -y kubelet kubeadm kubectl --disableexcludes=kubernetes

sudo systemctl enable --now kubelet

```

Output (in all instances):

```

Installed:
  conntrack-tools-1.4.6-2.amzn2023.0.2.x86_64          cri-tools-1.31.1-150500.1.1.x86_64          kubeadm-1.31.1-150500.1.1.x86_64
  kubelet-1.31.1-150500.1.1.x86_64          libnetfilter_cthelper-1.0.0-21.amzn2023.0.2.x86_64  kubernetes-cni-1.5.1-150500.1.1.x86_64
                                             libnetfilter_cttimeout-1.0.0-19.amzn2023.0.2.x86_64  libnetfilter_queue-1.0.5-2.amzn2023.0.2.x86_64

Complete!
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /usr/lib/systemd/system/kubelet.service.

[root@ip-172-31-88-48 ec2-user]# yum repolist
repo id                                repo name
amazonlinux                             Amazon Linux 2023 repository
kernel-livepatch                         Amazon Linux 2023 Kernel Livepatch repository
kubernetes                             Kubernetes

```

```

Complete!
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /usr/lib/systemd/system/kubelet.service.

[root@ip-172-31-88-241 ec2-user]# yum repolist
repo id                                repo name
amazonlinux                             Amazon Linux 2023 repository
kernel-livepatch                         Amazon Linux 2023 Kernel Livepatch repository
kubernetes                             Kubernetes

```

i-0384c7c6bd2e91fb8 (node-1)
 PublicIPs: 54.84.57.3 PrivateIPs: 172.31.88.241

```

libnetfilter_cthelper-1.0.0-21.amzn2023.0.2.x86_64  libnetfilter_cttimeout-1.0.0-19.amzn2023.0.2.x86_64  libnetfilter_
Complete!
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /usr/lib/systemd/system/kubelet.service.

[root@ip-172-31-89-83 ec2-user]# yum repolist
repo id                                repo name
amazonlinux                             Amazon Linux 2023 repository
kernel-livepatch                         Amazon Linux 2023 Kernel Livepatch repository
kubernetes                             Kubernetes

```

i-0045fc812ca071756 (node-2)
 PublicIPs: 44.201.84.150 PrivateIPs: 172.31.89.83

Kubeadm init:
 (after intialization):

```
Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

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

Alternatively, if you are the root user, you can run:

export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 172.31.88.48:6443 --token 2nrq25.jqlp7gerx8nty6ys \
    --discovery-token-ca-cert-hash sha256:d969d1bb086d72a8c952b2b6904c7c6f8c7e42a1d12f6ef1a82a46935363e411
```

```
[root@ip-172-31-88-241 ec2-user]# sudo yum install iproute
last metadata expiration check: 0:29:35 ago on Sun Sep 22 07:54:40 2024.
Package iproute-5.10.0-2.amzn2023.0.5.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-88-241 ec2-user]# kubeadm join 172.31.88.48:6443 --token 2nrq25.jqlp7gerx8nty6ys --discovery-token-ca-cert-hash sha256:d969d1bb086d72a8c952b2b6904c7c6f8c7e42a1d12f6ef1a82a46935363e411
[preflight] Running pre-flight checks
[WARNING FileExisting-tc]: tc not found in system path
```

Step 1: Deploying Your Application on Kubernetes

1.1 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-172-31-46-220:~$ kubectl get nodes
NAME           STATUS    ROLES      AGE     VERSION
ip-172-31-36-212   Ready    <none>    47s    v1.29.0
ip-172-31-46-220   Ready    control-plane   16m    v1.29.0
ip-172-31-47-26   Ready    <none>    29s    v1.29.0
```

Step 2: Create the Deployment YAML file

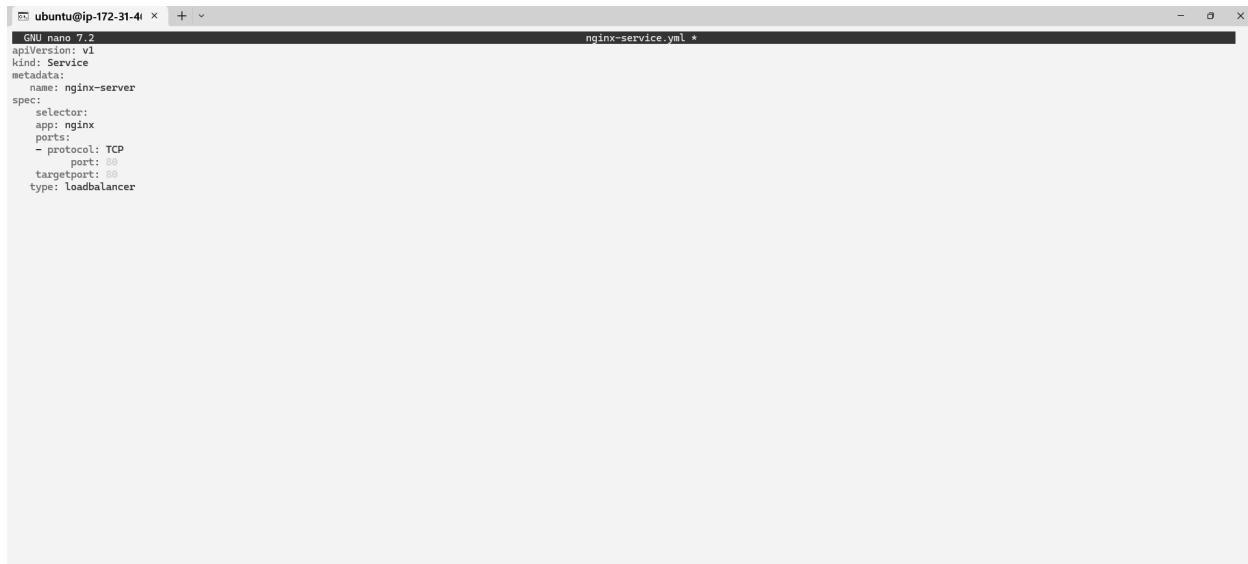
a) Create the YAML file: Use a text editor to create a file named nginx-deployment.yaml

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).

```
name: nginx-deployment
labels:
  app: nginx
spec:
  replicas: 3
  selector:
    app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.21.3
          ports:
            - containerPort: 80
```

Step 3:Create the Service YAML File

a) Create the YAML File: Create another file named nginx-service.yaml Add the Service Configuration: Copy and paste the following YAML content into the file given below



```
GNU nano 7.2
apiVersion: v1
kind: Service
metadata:
  name: nginx-server
spec:
  selector:
    app: nginx
  ports:
    - protocol: TCP
      port: 80
      targetport: 80
  type: loadbalancer
```

Step 4:Apply the YAML Files a) Deploy the Application: Use kubectl to create the Deployment and Service from the YAML files. Verify the Deployment: Check the status of your Deployment,Pods and Services. Describe the deployment(Extra)

```
ubuntu@ip-172-31-46-220:~$ kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx-deployment created
```

```
ubuntu@ip-172-31-46-220:~$ kubectl apply -f nginx-service.yaml
service/nginx-server created
```

Step 5:Ensure Service is Running 6.1 Verify Service: Run the following command to check the services running in your cluster: Kubectl get deployment Kubectl get pods kubectl get service

```
error: the server doesn't have a resource type "deployments"
ubuntu@ip-172-31-46-220:~$ kubectl get deployments
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment  3/3     3           3           7m27s
```

```
ubuntu@ip-172-31-46-220:~$ kubectl get services
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kubernetes  ClusterIP  10.96.0.1      <none>        443/TCP      85m
nginx-server  LoadBalancer  10.111.218.213  <pending>      80:30798/TCP  110s
```

Step 6: 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/ :

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

```
ubuntu@ip-172-31-46-220:~$ kubectl describe deployments
Name:           nginx-deployment
Namespace:      default
CreationTimestamp: Sat, 21 Sep 2024 12:30:54 +0000
Labels:          app=nginx
Annotations:    deployment.kubernetes.io/revision: 1
Selector:        app=nginx
Replicas:       3 desired | 3 updated | 3 total | 3 available | 0 unavailable
StrategyType:   RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nginx
  Containers:
    nginx:
      Image:      nginx:1.16
      Port:       80/TCP
      Host Port:  80/TCP
      Environment: <none>
      Mounts:     <none>
      Volumes:    <none>
  Conditions:
    Type     Status  Reason
    ----  -----
    Available  True    MinimumReplicasAvailable
    Progressing  True    NewReplicaSetAvailable
  OldReplicaSets: <none>
  NewReplicaSet:  nginx-deployment-854bc88786 (3/3 replicas created)
Events:
  Type     Reason          Age   From            Message
  ----  -----  --  --  -----
  Normal  ScalingReplicaSet 11m  deployment-controller  Scaled up replica set nginx-deployment-854bc88786 to 3
```

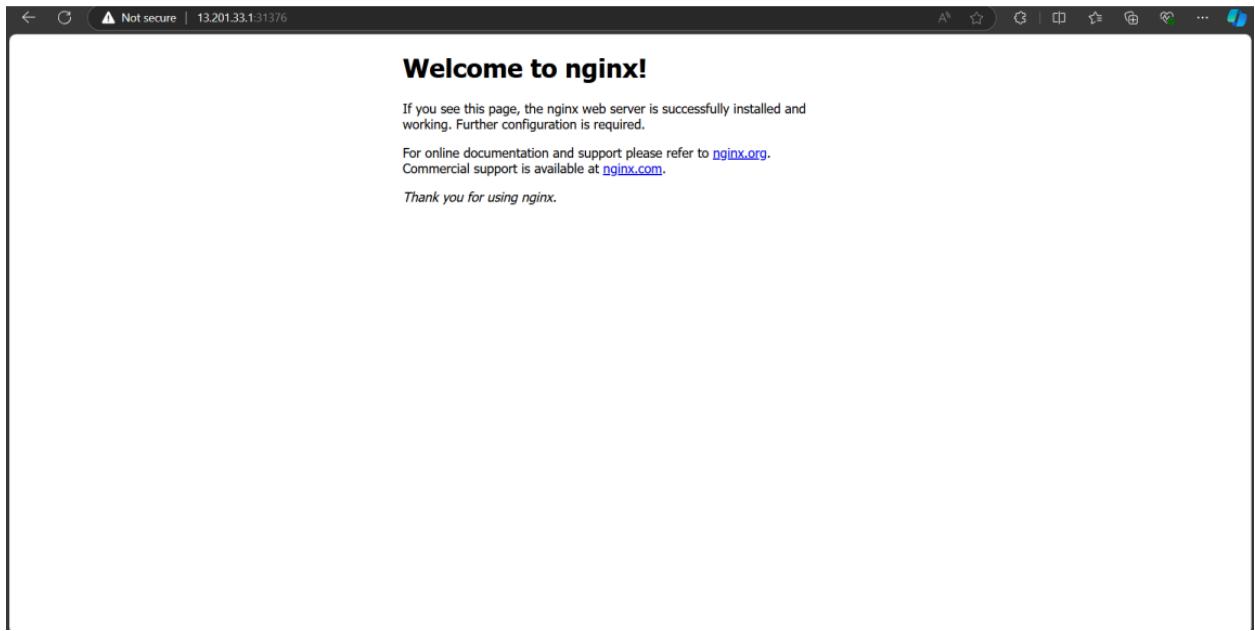
2. This means port forwarding is now active, and any traffic to localhost:8080 will be routed to the nginx-service on port 80.

```
ubuntu@ip-172-31-46-220:~$ kubectl port-forward service/nginx-server 8080:80
```

Step 7:

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



Advance devops Exp:5

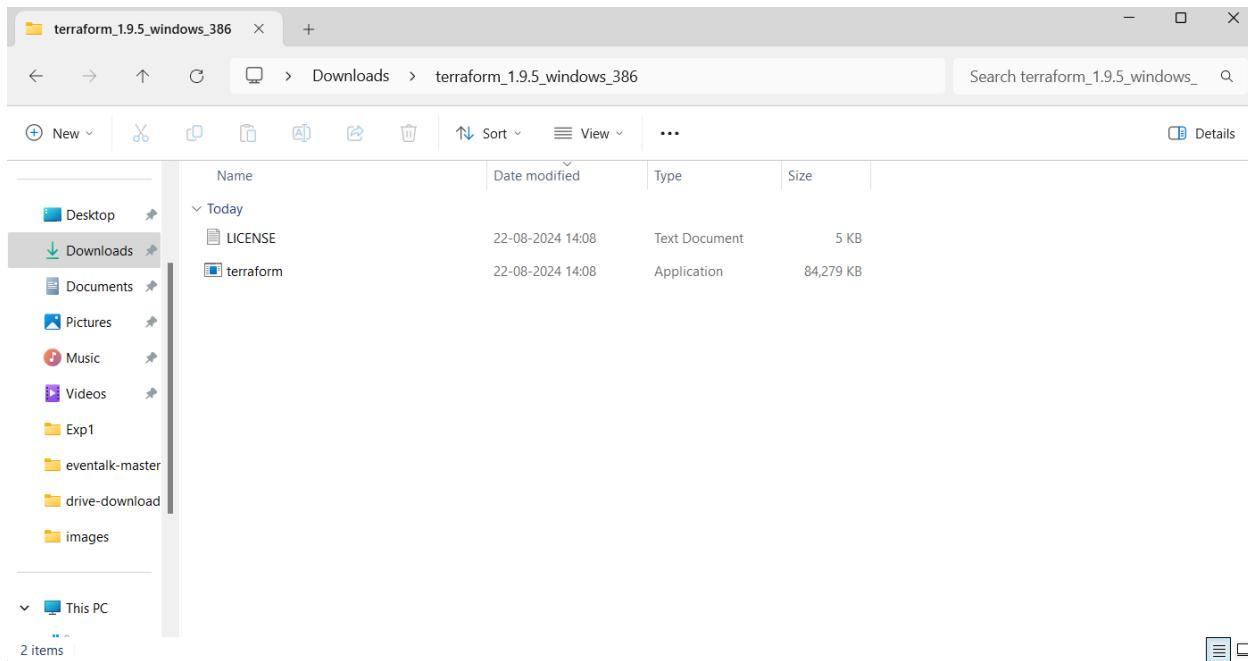
Jai Navani

D15 A-31

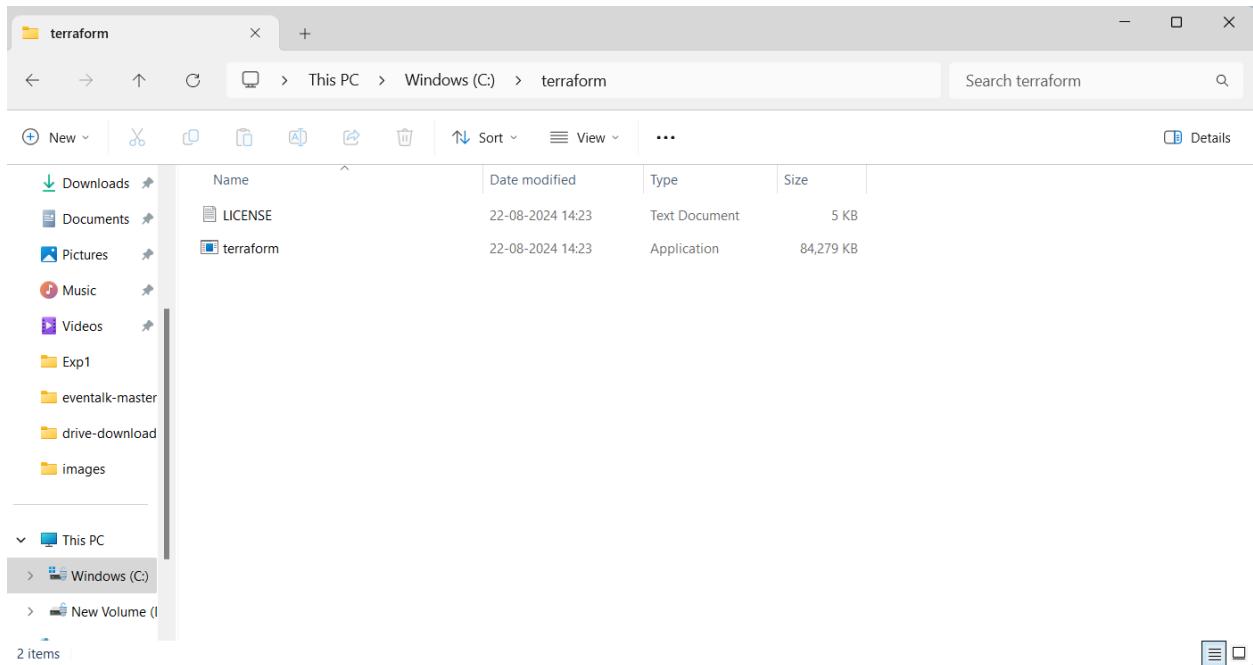
Aim: To understand terraform lifecycle, core concepts/terminologies and install it on a linux machine and windows

Step1: Download Terraform from the official website

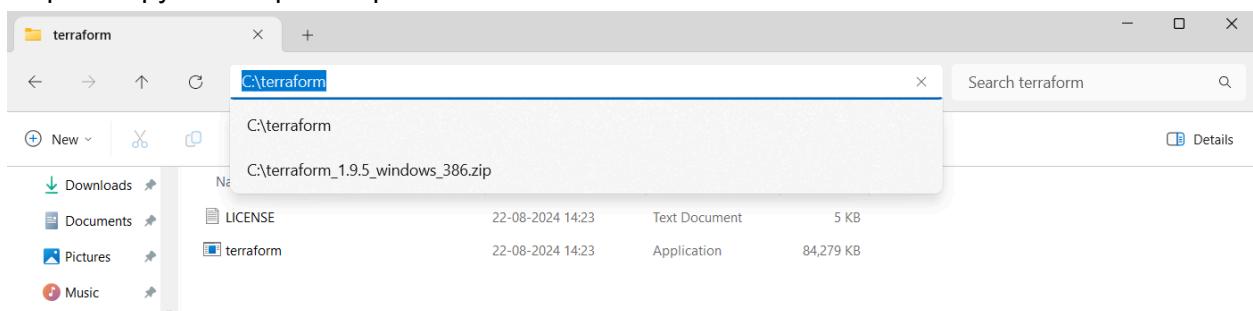
The screenshot shows the HashiCorp Terraform website's 'Install Terraform' page for macOS. On the left, there's a sidebar with links for 'Install Terraform', 'Operating Systems' (with 'macOS' selected), 'Release information', 'Next steps', and 'Resources'. The main content area has a title 'Install Terraform' with a version dropdown set to '1.9.5 (latest)'. Below it, under 'macOS', there's a 'Package manager' section with a terminal command: `brew tap hashicorp/tap
brew install hashicorp/tap/terraform`. Under 'Binary download', there are two options: 'AMD64 Version: 1.9.5' and 'ARM64 Version: 1.9.5', each with a 'Download' button. To the right, there's an 'About Terraform' sidebar with a brief description and links to 'Featured docs' like 'Introduction to Terraform' and 'Configuration Language'. At the bottom, there's a cookie consent banner.



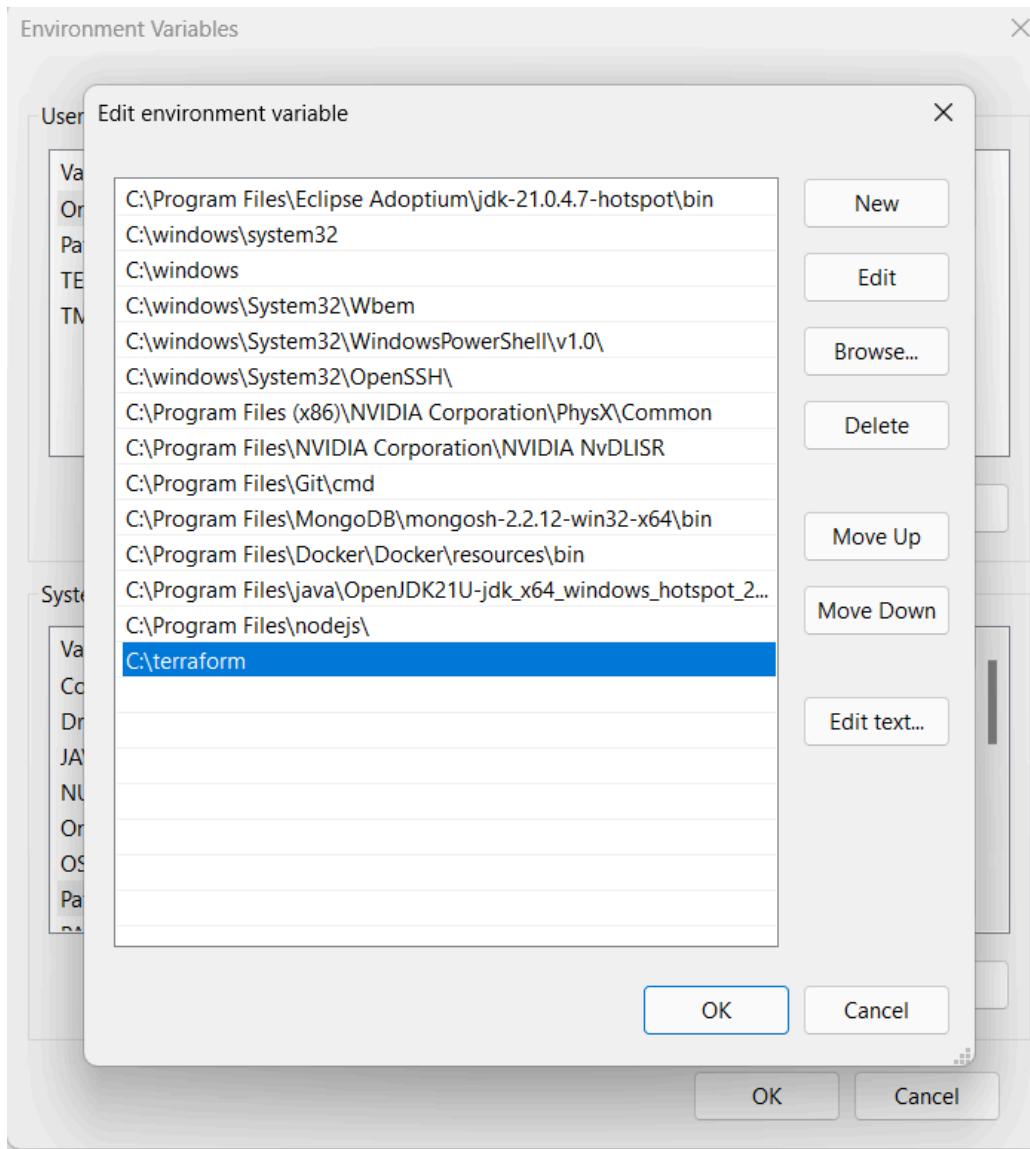
Step 2: Copy and extract Terraform from the downloads and paste it in the C drive



Step 3: Copy the file path to paste in the environment variables



Step 4: Set the environment variables for terraform



Step 5: Check whether the terraform is installed

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

C:\Users\navan>terraform --version
Terraform v1.9.5
on windows_386
```

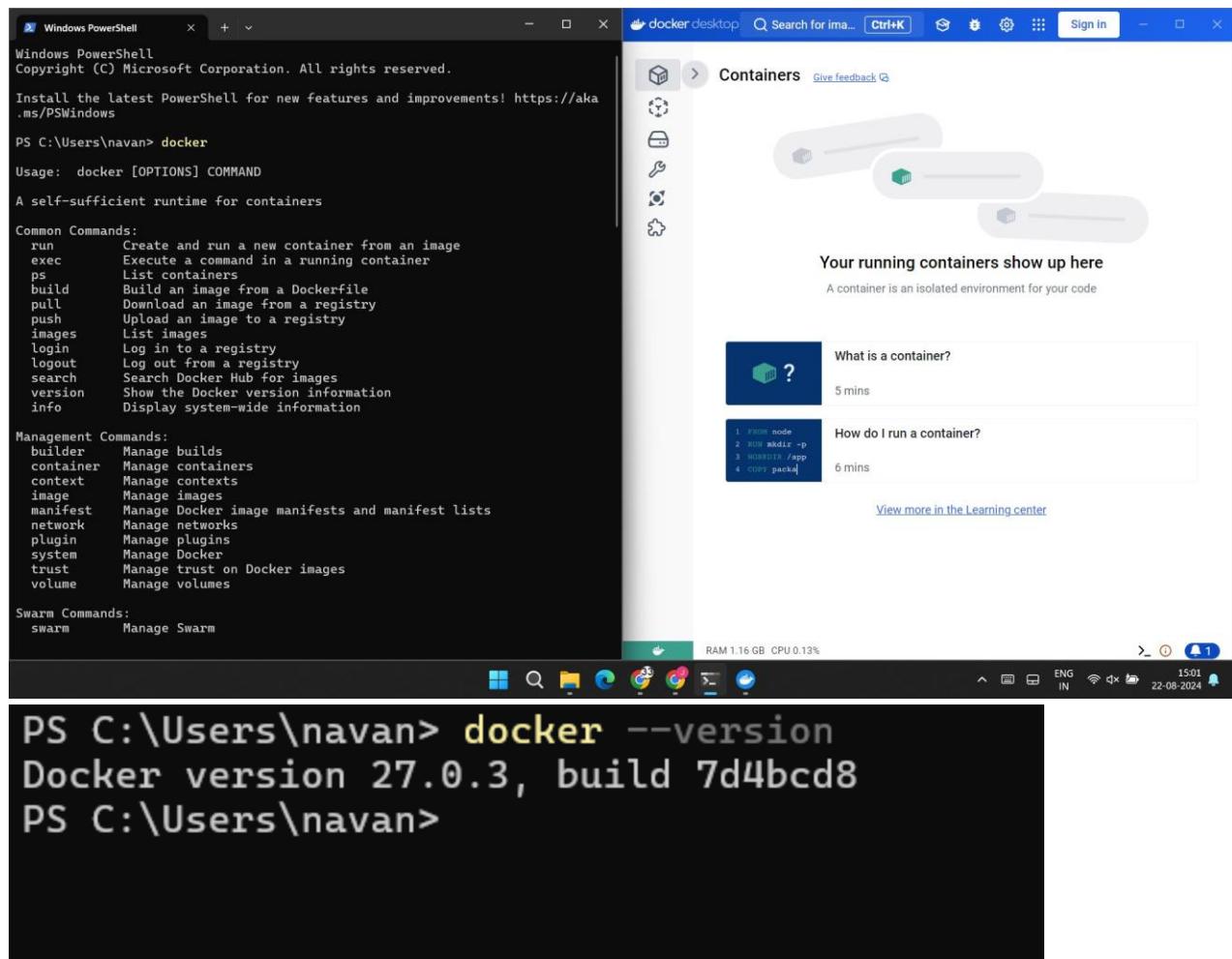
Advance devops Exp:6

Jai Navani

D15 A-31

Aim: Creating docker image using Terraform

Step 1: Install docker Desktop after installation check the functionality

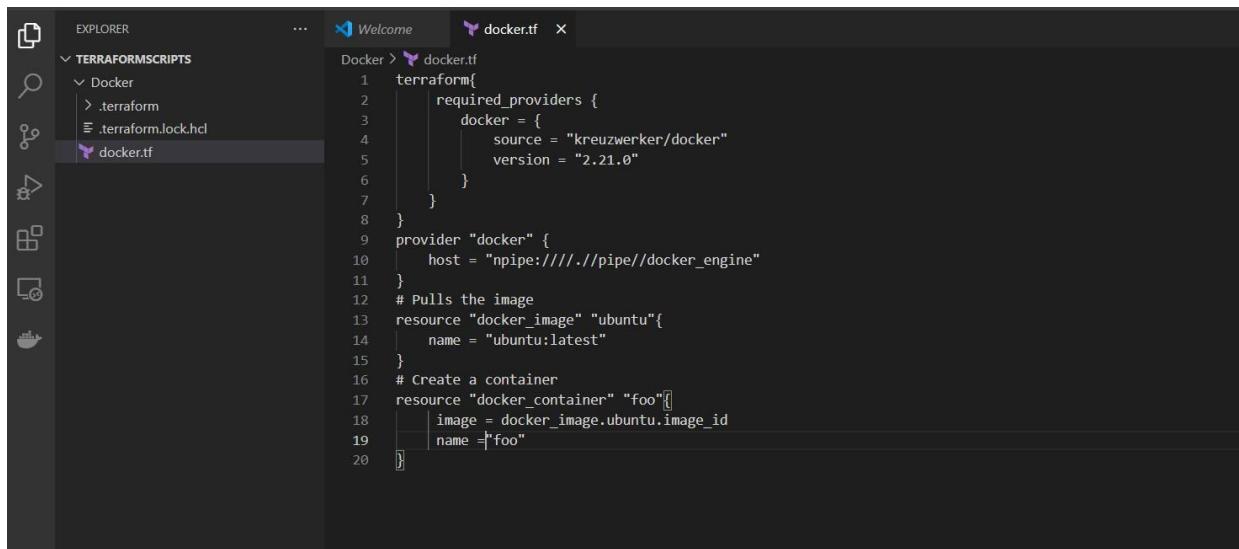


Now, create a folder named 'Terraform Scripts' in which we save our different types of scripts which will be further used in this experiment.

Step 2: Firstly create a new folder named 'Docker' in the 'TerraformScripts' folder. Then create a new docker.tf file using Atom editor and write the following contents into it to create a Ubuntu Linux container.

Script:

```
terraform{
    required_providers {
        docker = {
            source = "kreuzwerker/docker"
            version = "2.21.0"
        }
    }
}
provider "docker" {
    host = "npipe:///./pipe//docker_engine"
}
# Pulls the image
resource "docker_image" "ubuntu"{
    name = "ubuntu:latest"
}
# Create a container
resource "docker_container" "foo"{
    image = docker_image.ubuntu.image_id
    name = "foo"
}
```



Step 3: Execute terraform init command to initialize the resources

```
C:\Users\navan\Desktop\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.

C:\Users\navan\Desktop\TerraformScripts\Docker>
```

Step 4: Execute Terraform plan to see the available resources

```
C:\Users\navan\Desktop\TerraformScripts\Docker>terraform plan
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         = (known after apply)
    + 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.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.

C:\Users\navan\Desktop\TerraformScripts\Docker>

```

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”

```

}
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 10s [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_container.foo: Creating...

```

Docker images,before Executing Apply step:

```

C:\Users\navan\Desktop\TerraformScripts\Docker>docker images
REPOSITORY      TAG          IMAGE ID      CREATED        SIZE
ubuntu          latest        edbfe74c41f8  2 weeks ago   78.1MB
node            20-alpine    e2997a3fdff8  4 weeks ago   133MB

```

```
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28e  
3e6df8c9d66519b6ad761c2598aubuntu:latest]
```

Note: Objects have changed outside of Terraform

Terraform detected the following changes made outside of Terraform since the last "terraform apply" which may have affected this plan:

```
# docker_image.ubuntu has been deleted
- resource "docker_image" "ubuntu" {
    id          = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6  
2598aubuntu:latest"
    - image_id   = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6  
2598a" -> null
    name        = "ubuntu:latest"
    # (2 unchanged attributes hidden)
}
```

Unless you have made equivalent changes to your configuration, or ignored the relevant attributes using ignore_changes, the following plan may include actions to undo or respond to these changes.

Step 6: Execute Terraform destroy to delete the configuration, which will automatically delete the ubuntu container.

```
C:\Users\navan\Desktop\TerraformScripts\Docker>terraform destroy
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]

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_image.ubuntu will be destroyed
- resource "docker_image" "ubuntu" {
    - id          = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest" -> null
    - image_id   = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
    - latest     = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
    - name       = "ubuntu:latest" -> null
    - repo_digest = "ubuntu@sha256:8a37d68f4f73ebf3d4efafbcf66379bf3728902a8038616808f04e34a9ab63ee" -> null
}

Plan: 0 to add, 0 to change, 1 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:
```

```
C:\Windows\System32\cmd.exe > terraform destroy
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
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_image.ubuntu will be destroyed
- resource "docker_image" "ubuntu" {
  - id          = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
  - image_id    = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
  - latest      = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
  - name        = "ubuntu:latest" -> null
  - repo_digest = "ubuntu@sha256:8a37d68f4f73ebf3d4efafbcf66379bf3728902a8038616808f04e34a9ab63ee" -> null
}

Plan: 0 to add, 0 to change, 1 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_image.ubuntu: Destroying... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_image.ubuntu: Destruction complete after 0s

Destroy complete! Resources: 1 destroyed.
```

Docker images after executing destroy step

```
C:\Users\navan\Desktop\TerraformScripts\Docker>docker images
REPOSITORY      TAG          IMAGE ID      CREATED       SIZE
node            20-alpine    e2997a3fdff8   5 weeks ago   133MB
```

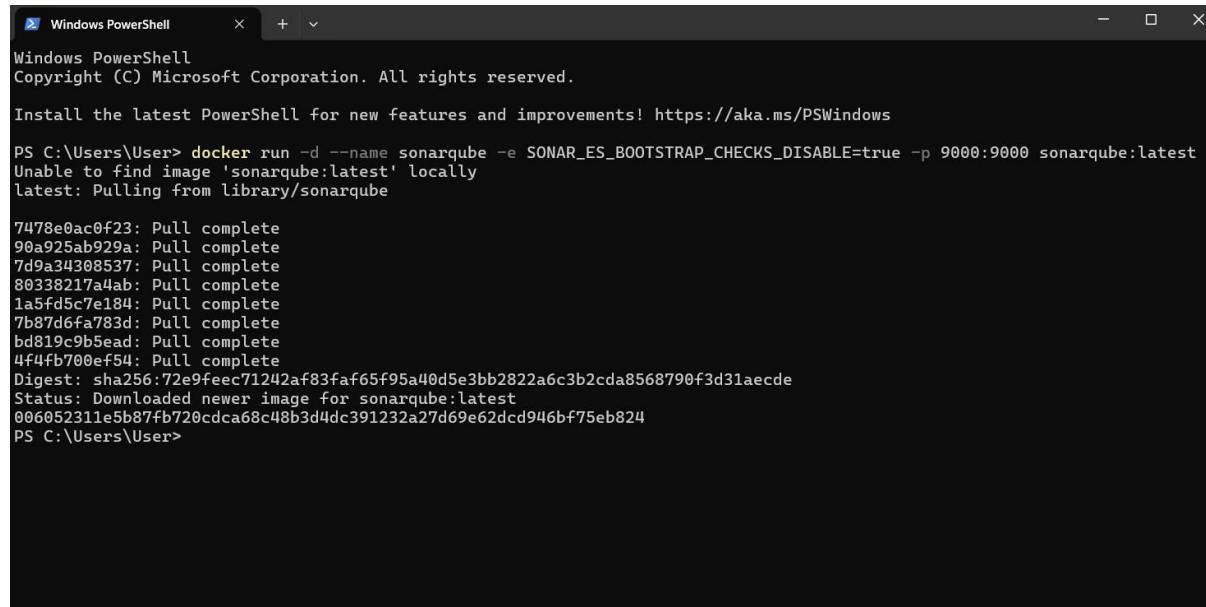
EXPERIMENT 7

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

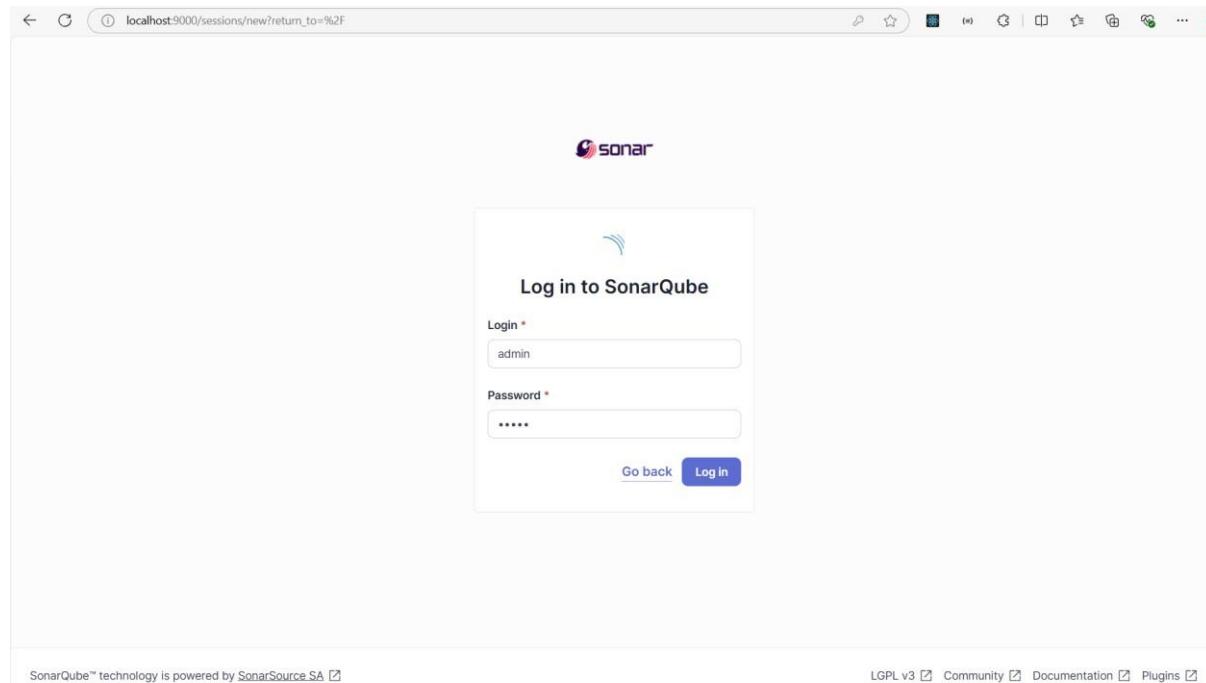


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

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
7478e0ac0f23: Pull complete
90a925ab929a: Pull complete
7d9a34308537: Pull complete
80338217a4ab: Pull complete
1a5fd5c7e184: Pull complete
7b87d6fa783d: Pull complete
bd819c9b5ead: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
Status: Downloaded newer image for sonarqube:latest
0060952311e5b87fb720cdca68c48b3d4dc391232a27d69e62cd946bf75eb824
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 Import from Bitbucket Cloud Import from Bitbucket Server Import from GitHub Import from GitLab

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

Projects Issues Rules Quality Profiles Quality Gates Administration More Q

1 of 2

Create a local project

Project display name *

Project key *

Main branch name *

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

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

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.

Recommended for projects using feature branches.

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.

Dashboard > Manage Jenkins

Manage Jenkins

New version of Jenkins (2.462.2) is available for download (changelog). [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:

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.

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

System Configuration

- System**: Configure global settings and paths.
- Tools**: Configure tools, their locations and automatic installers.
- Nodes**: Add, remove, control and monitor the various nodes that Jenkins runs jobs on.
- CLOUDS**: Add, remove, and configure cloud instances to provision agents on-demand.
- Plugins**: Add, remove, disable or enable plugins that can extend the functionality of Jenkins. (29)
- Appearance**: Configure the look and feel of Jenkins.

localhost:8080/manage/pluginManager

Dashboard > Manage Jenkins > Plugins

The screenshot shows the Jenkins 'Plugins' page. A search bar at the top contains the text 'sonar'. Below it, a sidebar on the left lists 'Updates', 'Available plugins', 'Installed plugins', and 'Advanced settings'. The main area displays a single result for the 'SonarQube Scanner for Jenkins 2.17.2' plugin. The card includes the plugin name, version, a brief description stating it allows for easy integration with SonarQube for code quality inspection, and two buttons: a blue 'Enabled' button with a checkmark and a red 'Disabled' button with a crossed-out circle. At the bottom right of the page are links for 'REST API' and '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 'Manage Jenkins' page. On the left, a sidebar lists 'New Item', 'Build History', 'Project Relationship', 'Check File Fingerprint', 'Manage Jenkins' (which is selected and highlighted in grey), and 'My Views'. Below this is a 'Build Queue' section. The main content area is titled 'Manage Jenkins' and features several status cards. One card for 'New version of Jenkins (2.462.2) is available for download (changelog)' has a 'Or Upgrade Automatically' button. Another card for 'Building on the built-in node can be a security issue' has 'Manage' and 'Dismiss' buttons. A third card for 'Warnings have been published for the following currently installed components' (Jenkins 2.452.3 core and libraries) has a 'Configure which of these warnings are shown' button. The 'System Configuration' section at the bottom contains links for 'System', 'Tools', 'Nodes', 'Clouds', 'Plugins' (with a note about 29 available), and 'Appearance'.

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
sonarqube

Server URL
Default is <http://localhost:9000>

Server authentication token
SonarQube authentication token. Mandatory when anonymous access is disabled.

[+ Add](#)

[Advanced](#)

[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

 Built-In Node

1 Idle

2 Idle

 Slave1

Manage Jenkins

Search settings /

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

[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:

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.

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

System Configuration System

Configure global settings and paths.

 Tools

Configure tools, their locations and automatic installers.

 Plugins

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

 Nodes

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

 Clouds

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

 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.

Enter an item name

SonarQube Required field

Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

Multibranch Pipeline
OK
Creates a set of Pipeline projects according to detected branches in one SCM repository.

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

https://github.com/shazforiot/MSBuild_firstproject

This is a simple Hello world project

Configure

Source Code Management

None

Git [?](#)

Repositories [?](#)

Repository URL [?](#)
https://github.com/shazforiot/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.

Configure

Build Steps

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

Execute SonarQube Scanner

JDK [?](#)
JDK to be used for this SonarQube analysis
(Inherit From Job)

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**

Dashboard > SonarQube >

SonarQube

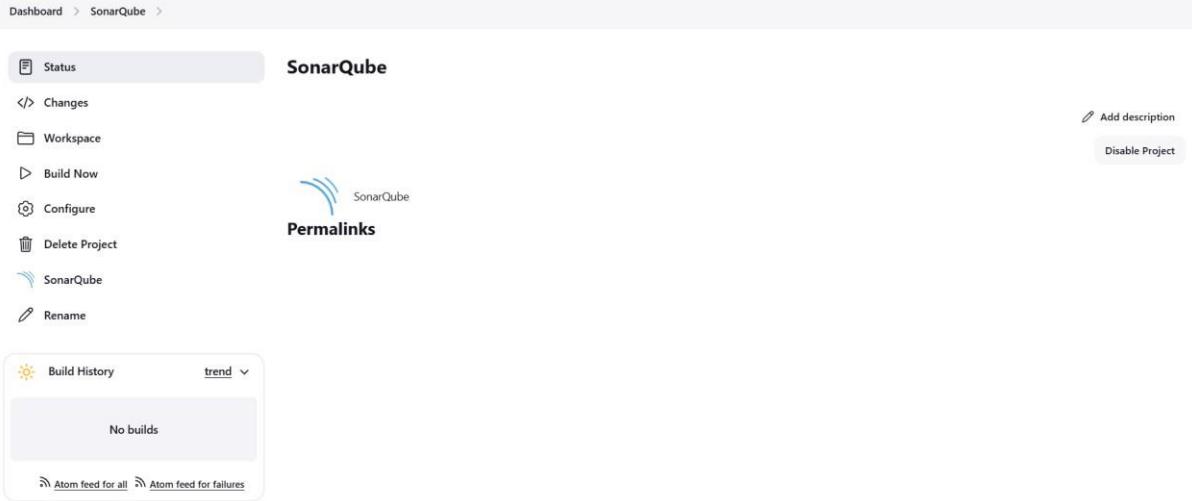
Status Changes Workspace Build Now Configure Delete Project SonarQube Rename

Add description Disable Project

SonarQube Permalinks

Build History trend ▾ No builds Atom feed for all Atom feed for failures

REST API Jenkins 2.452.3



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

sonarQube Projects Issues Rules Quality Profiles Quality Gates Administration More Q

Administration Configuration Security Projects System Marketplace

Global Permissions

Grant and revoke permissions to make changes at the global level. These permissions include editing Quality Profiles, executing analysis, and performing global system administration.

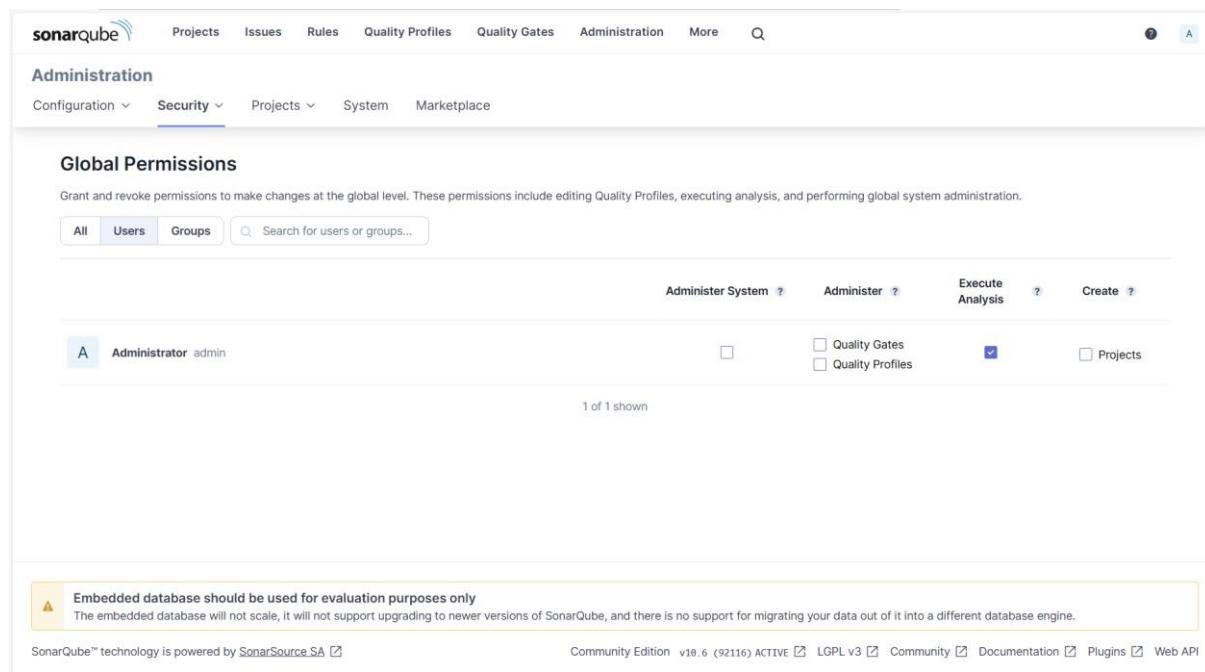
All Users Groups Search for users or groups...

	Administer System	Administer	Execute Analysis	Create
A Administrator admin	<input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

1 of 1 shown

⚠️ 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 GPL v3 Community Documentation Plugins Web API



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

Status

Changes

Console Output

View as plain text

Edit Build Information

Delete build '#4'

Timings

Git Build Data

Previous Build

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/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 2.41.0.windows.3'
> 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 f2bc042c04c6e72427c380bcaee6d6fee7b49adf (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f f2bc042c04c6e72427c380bcaee6d6fee7b49adf # timeout=10
Commit message: "updated"
> git.exe rev-list -no-walk f2bc042c04c6e72427c380bcaee6d6fee7b49adf # 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=..\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
18:40:41.488 INFO Sensor Analysis Warnings import [csharp]
18:40:41.488 INFO Sensor Analysis Warnings import [csharp] (done) | time=2ms
18:40:41.489 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 'f2bc042c04c6e72427c380bcaee6d6fee7b49adf'
18:40:42.524 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-ebdd-4bb2-b564-0aa4ea71b0f2
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

```

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 ?

sonarqube-test / main ?

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

main Version not provided · Set as homepage

Quality Gate Passed Last analysis 14 minutes ago

The last analysis has warnings. See details

New Code Overall Code

Security	Reliability	Maintainability
0 Open issues (A)	0 Open issues (A)	0 Open issues (A)
0 H 0 M 0 L	0 H 0 M 0 L	0 H 0 M 0 L

Accepted issues	Coverage	Duplications
0 Valid issues that were not fixed (2)	On 0 lines to cover.	0.0% On 86 lines.

Security Hotspots

This screenshot shows the SonarQube dashboard for the 'main' branch of the 'sonarqube-test' project. The top navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, More, and a search bar. The main content area displays the 'main' branch status, which is currently 'Passed'. A note indicates that the last analysis had warnings, with a link to see details. Below this, there are three main sections: Security, Reliability, and Maintainability, each showing 0 open issues with an 'A' rating. There are also sections for Accepted issues (0 valid issues), Coverage (0 lines to cover), and Duplications (0.0% on 86 lines). The bottom section is labeled 'Security Hotspots'.

Advance-Devops

Experiment no:8

D15-A

Jai navani-31

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:

The screenshot shows the Jenkins interface for creating a new item. At the top, there's a navigation bar with 'Dashboard > All >'. Below it is a search bar with the placeholder 'Enter an item name'. A red box highlights the input field where 'sonarqubetest' is typed. Below the search bar, there's a 'Required field' message. A list of project types is displayed in a scrollable area:

- Freestyle project**: Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
- Maven project**: Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.
- Pipeline**: Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
- Multi-configuration project**: Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

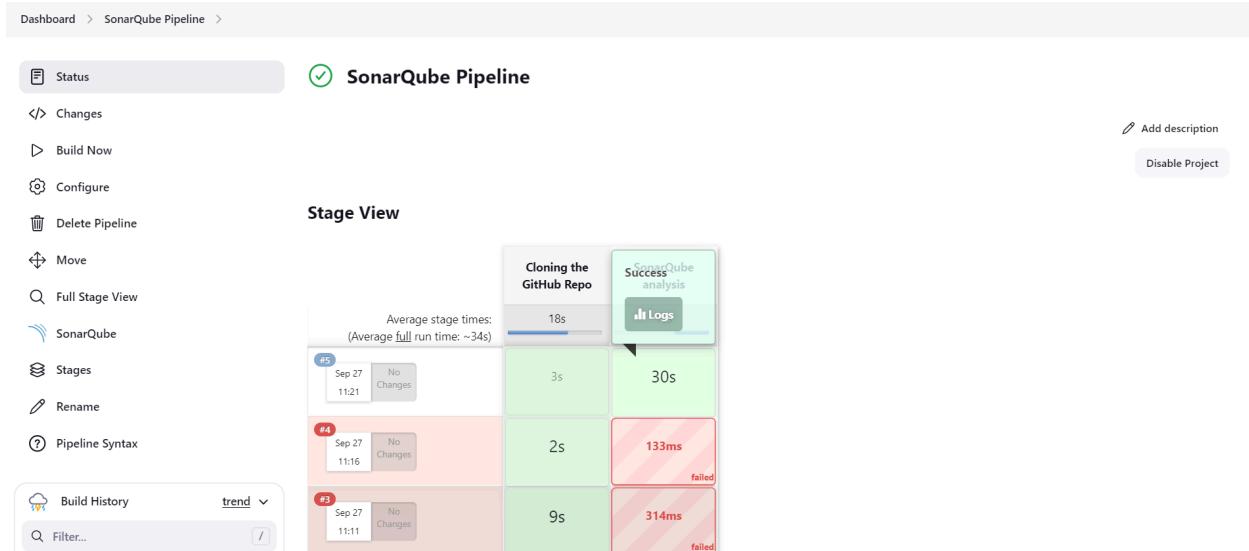
A blue 'OK' button is visible at the bottom of the list. A small note at the bottom right says 'Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a'.

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 interface. At the top, there's a breadcrumb navigation: Dashboard > SonarQube Pipeline > Configuration. Below that, the title is "Pipeline". Under "Definition", it says "Pipeline script". The "Pipeline" tab is selected, while "General" and "Advanced Project Options" are other tabs available. The main area contains a code editor with the pipeline script pasted in. The script is identical to the one provided above. Below the code editor is a checkbox labeled "Use Groovy Sandbox" which is checked. At the bottom of the page are two buttons: "Save" and "Apply".

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] 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 f2bc042c04c6e72427c380bcaee6d6fee7b49adf (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f f2bc042c04c6e72427c380bcaee6d6fee7b49adf # 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 'f2bc042c04c6e72427c380bc4ee6d6fee7b49adf'
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-ad4a6ccedd8
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 the 'main' project. At the top, there's a navigation bar with links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, More, and a search bar. Below the navigation, the project name 'sonarqube-test1' is shown with a dropdown menu containing 'main' and other options.

The main content area is titled 'main'. It features a large green 'Passed' status indicator with a checkmark icon. A yellow warning box below it says 'The last analysis has warnings. See details'. The dashboard is divided into several sections: Security (0 Open issues), Reliability (0 Open issues), Maintainability (0 Open issues), Accepted issues (0), Coverage (0.0%), and Duplications (0.0%). Each section includes a progress bar and a letter grade (A or B).

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 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, EC2 Global View, Events, and Instances. Under Instances, it shows sub-links for Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, and AMI Catalog. Below these are Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), Network & Security (Security Groups, Elastic IPs), and CloudShell/Feedback buttons. The main content area displays 'Instances (1/1) Info'. It lists one instance: 'nagios-host' (Instance ID: i-033ee56b96fef8322). The instance is 'Running' (Status check: Initializing), has an 't2.micro' instance type, and is in 'us-east-1c' availability zone with a public IP of 'ec2-44-211-225-236'. A detailed view for 'i-033ee56b96fef8322 (nagios-host)' is open, showing the Details tab. In the 'Instance summary' section, it shows the instance ID (i-033ee56b96fef8322), Public IPv4 address (44.211.225.236), Private IP4 addresses (172.31.83.53), and other details like Hostname type (IP name: ip-172-31-83-53.ec2.internal) and Instance type (t2.micro).

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

The screenshot shows the AWS Security Groups page. The sidebar includes links for EC2 Dashboard, EC2 Global View, Events, Instances, Images, Elastic Block Store, Network & Security, and CloudShell/Feedback. The main area shows a security group named 'launch-wizard-13' with details like Security group ID (sg-0a1c694292da367bd), Description (launch-wizard-13 created 2024-09-30T16:37:21.185Z), Owner (856746069793), and VPC ID (vpc-0ec7dea564d6f7acf). The 'Inbound rules' tab is selected, displaying a table with seven entries. The columns include Name, Security group rule..., IP version, Type, Protocol, and Port range. The rules are: 1. sgr-0ccb8fb06a97d678 (Custom TCP, TCP, 5666). 2. sgr-0214f72ad5a70a602 (HTTPS, TCP, 443). 3. sgr-070fbe0339e680801 (SSH, TCP, 22). 4. sgr-0685a56749b9e8... (HTTP, TCP, 80). 5. sgr-044962adae76c80... (All traffic, All). 6. sgr-02126894299e1c4... (All ICMP - IPv6, IPv6 ICMP, All). 7. sgr-0da3debdasfb755e (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
 Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default 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: 0:11:38 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
php8_3           x86_64          8.3.10-1.amzn2023.0.1
Installing dependencies:
apr              x86_64          1.7.2-2.amzn2023.0.2
apr-util         x86_64          1.6.3-1.amzn2023.0.1
generic-logos-httpd noarch        18.0.0-12.amzn2023.0.3
httpd-core       x86_64          2.4.62-1.amzn2023
httpd-filesystem noarch        2.4.62-1.amzn2023
httpd-tools      x86_64          2.4.62-1.amzn2023
libbrotli        x86_64          1.0.9-4.amzn2023.0.2
libsodium         x86_64          1.0.19-4.amzn2023
                                         48 k
                                         10 k
                                         129 k
                                         98 k
                                         19 k
                                         1.4 M
                                         14 k
                                         81 k
                                         315 k
                                         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-xml-0.3.10-1.amzn2023.0.1.x86_64
=====
Complete!
root@ip-172-31-93-157:~# 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.
=====

```

Package	Architecture	Version	Repository	Size
installing:				
gcc	x86_64	11.4.1-2.amzn2023.0.2	amazonlinux	32 M
installing dependencies:				
annobin-docs	noarch	10.93-1.amzn2023.0.1	amazonlinux	92 k
annobin-plugin-gcc	x86_64	10.93-1.amzn2023.0.1	amazonlinux	887 k
cpp	x86_64	11.4.1-2.amzn2023.0.2	amazonlinux	10 M
gc	x86_64	8.0.4-5.amzn2023.0.2	amazonlinux	105 k
glibc-devel	x86_64	2.34-52.amzn2023.0.11	amazonlinux	27 k
glibc-headers-x86	noarch	2.34-52.amzn2023.0.11	amazonlinux	427 k
guile22	x86_64	2.2.7-2.amzn2023.0.3	amazonlinux	6.4 M
kernel-headers	x86_64	6.1.109-118.189.amzn2023	amazonlinux	1.4 M
libmpc	x86_64	1.2.1-2.amzn2023.0.2	amazonlinux	62 k
libtool-ltdl	x86_64	2.4.7-1.amzn2023.0.3	amazonlinux	38 k
libcrypt-devel	x86_64	4.4.33-7.amzn2023	amazonlinux	32 k
make	x86_64	1:4.3-5.amzn2023.0.2	amazonlinux	534 k

Transaction Summary

Install 13 Packages
Total download size: 52 M
Installed size: 168 M
Is this ok [y/N]: y

i-025f1d18f7c8a8cda (nagios-host)

Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

```

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

```

Package	Architecture	Version	Repository	Size
Installing:				
gd	x86_64	2.3.3-5.amzn2023.0.3	amazonlinux	139 k
gd-devel	x86_64	2.3.3-5.amzn2023.0.3	amazonlinux	38 k
Installing dependencies:				
brotli	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	314 k
brotli-devel	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	31 k
bzip2-devel	x86_64	1.0.8-6.amzn2023.0.2	amazonlinux	214 k
cairo	x86_64	1.17.6-2.amzn2023.0.1	amazonlinux	684 k
cmake-fs	x86_64	3.22.2-1.amzn2023.0.4	amazonlinux	16 k
fontconfig	x86_64	2.13.94-2.amzn2023.0.2	amazonlinux	273 k
fontconfig-devel	x86_64	2.13.94-2.amzn2023.0.2	amazonlinux	128 k
fnts-fs	noarch	1:2.0.5-12.amzn2023.0.2	amazonlinux	9.5 k
freetype	x86_64	2.13.2-5.amzn2023.0.1	amazonlinux	423 k
freetype-devel	x86_64	2.13.2-5.amzn2023.0.1	amazonlinux	912 k
glib2-devel	x86_64	2.74.7-689.amzn2023.0.2	amazonlinux	486 k
google-noto-fonts-common	noarch	20201206-2.amzn2023.0.2	amazonlinux	15 k
google-noto-sans-vf-fonts	noarch	20201206-2.amzn2023.0.2	amazonlinux	492 k
graphite2	x86_64	1.3.14-7.amzn2023.0.2	amazonlinux	97 k
graphite2-devel	x86_64	1.3.14-7.amzn2023.0.2	amazonlinux	21 k
harfbuzz	x86_64	7.0.0-2.amzn2023.0.1	amazonlinux	868 k
harfbuzz-devel	x86_64	7.0.0-2.amzn2023.0.1	amazonlinux	404 k
harfbuzz-icu	x86_64	7.0.0-2.amzn2023.0.1	amazonlinux	18 k
jbigkit-libs	x86_64	2.1-21.amzn2023.0.2	amazonlinux	54 k
langpacks-core-font-en	noarch	3.0-21.amzn2023.0.4	amazonlinux	10 k
libtce	x86_64	1.0.10-6.amzn2023.0.2	amazonlinux	71 k
libSM	x86_64	1.2.3-8.amzn2023.0.2	amazonlinux	42 k

i-025f1d18f7c8a8cda (nagios-host)

Public IPs: 3.86.198.73 Private IPs: 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:~# sudo adduser -m nagios
root@ip-172-31-93-157:~# sudo passwd nagios
Changing password for user nagios.
New password:
Re-type new password:
passwd: all authentication tokens updated successfully.
root@ip-172-31-93-157:~# sudo groupadd nagcmd
root@ip-172-31-93-157:~# sudo usermod -a -G nagcmd nagios
root@ip-172-31-93-157:~# sudo usermod -a -G nagcmd apache
root@ip-172-31-93-157:~#
```

i-025f1d18f7c8a8cda (nagios-host)

Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

© 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:~# mkdir ~/downloads
root@ip-172-31-93-157:~# 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: q: 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)

Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

© 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-172-31-93-157 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://downloads.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://downloads.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz
Resolving downloads.sourceforge.net (downloads.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 (1.7M) [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)

Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

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

```
nagios-plugins-2.0.3/plugins-scripts/check_ifoperstatus.pl
nagios-plugins-2.0.3/plugins-scripts/Makefile.am
nagios-plugins-2.0.3/plugins-scripts/subst.in
nagios-plugins-2.0.3/plugins-scripts/check_breeze.pl
nagios-plugins-2.0.3/plugins-scripts/check_log.sh
nagios-plugins-2.0.3/plugins-scripts/check_flexlm.pl
nagios-plugins-2.0.3/plugins-scripts/check_rpc.pl
nagios-plugins-2.0.3/plugins-scripts/check_oracle.sh
nagios-plugins-2.0.3/plugins-scripts/utils.pm.in
nagios-plugins-2.0.3/plugins-scripts/check_disk_smb.pl
nagios-plugins-2.0.3/plugins-scripts/t/
nagios-plugins-2.0.3/plugins-scripts/t/check_ifoperstatus.t
nagios-plugins-2.0.3/plugins-scripts/t/check_rpc.t
nagios-plugins-2.0.3/plugins-scripts/t/check_file_age.t
nagios-plugins-2.0.3/plugins-scripts/t/check_disk_smb.t
nagios-plugins-2.0.3/plugins-scripts/t/check_ifstatus.t
nagios-plugins-2.0.3/plugins-scripts/t/utils.t
nagios-plugins-2.0.3/plugins-scripts/check_mailq.pl
nagios-plugins-2.0.3/plugins-scripts/check_wave.pl
nagios-plugins-2.0.3/plugins-scripts/check_ircd.pl
nagios-plugins-2.0.3/plugins-scripts/utils.sh.in
nagios-plugins-2.0.3/plugins-scripts/check_ifstatus.pl
nagios-plugins-2.0.3/plugins-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/solpkgs
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)

Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

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

```
nagios-plugins-2.0.3/plugins-scripts/Makefile.am
nagios-plugins-2.0.3/plugins-scripts/subst.in
nagios-plugins-2.0.3/plugins-scripts/check_breeze.pl
nagios-plugins-2.0.3/plugins-scripts/check_log.sh
nagios-plugins-2.0.3/plugins-scripts/check_flexlm.pl
nagios-plugins-2.0.3/plugins-scripts/check_rpc.pl
nagios-plugins-2.0.3/plugins-scripts/check_oracle.sh
nagios-plugins-2.0.3/plugins-scripts/utils.pm.in
nagios-plugins-2.0.3/plugins-scripts/check_disk_smb.pl
nagios-plugins-2.0.3/plugins-scripts/t/
nagios-plugins-2.0.3/plugins-scripts/t/check_ifoperstatus.t
nagios-plugins-2.0.3/plugins-scripts/t/check_rpc.t
nagios-plugins-2.0.3/plugins-scripts/t/check_file_age.t
nagios-plugins-2.0.3/plugins-scripts/t/check_disk_smb.t
nagios-plugins-2.0.3/plugins-scripts/t/check_ifstatus.t
nagios-plugins-2.0.3/plugins-scripts/t/utils.t
nagios-plugins-2.0.3/plugins-scripts/check_mailq.pl
nagios-plugins-2.0.3/plugins-scripts/check_wave.pl
nagios-plugins-2.0.3/plugins-scripts/check_ircd.pl
nagios-plugins-2.0.3/plugins-scripts/utils.sh.in
nagios-plugins-2.0.3/plugins-scripts/check_ifstatus.pl
nagios-plugins-2.0.3/plugins-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/solpkgs
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)

Public IPs: 3.86.198.73 Private IPs: 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 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

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

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
make[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 nemods.o nemods.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_wproc_list':
  inlined from 'get_worker' at workers.c:224:12:
workers.c:209:17: warning: '*' directive argument is null [-Wformat-overflow=]
  209 |     log_debug_info(DNSCORE_DEBUG_CHECKS, 1, "Found specialized worker(s) for '%s'", (slash && *slash != '/') ? slash : cmd_name);
  |     ^
  |
  |     log_debug_info(DNSCORE_DEBUG_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:2247:19: warning: assignment discards 'const' qualifier from pointer target type [-Wdiscarded-qualifiers]
  2247 |     cr.source = command_worker.source_name;
  |     ^
  |
  |     cr.source = command_worker.source_name;
commands.c: In function 'process_passive_host_check':
commands.c:2339:19: warning: assignment discards 'const' qualifier from pointer target type [-Wdiscarded-qualifiers]
  2339 |     cr.source = command_worker.source_name;
  |     ^
  |
  |     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 -DHAVE_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 1 and 11 bytes into a region of size 6 [-Wformat-truncation=]
  50 |     sprintf(port_str, sizeof(port_str), "%d", port);
```

i-025f1d18f7c8a8cda (nagios-host)

Public IPs: 3.86.198.73 Private IPs: 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
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[1]: *** [Makefile:197: install] Error 2
make[1]: Leaving directory '/root/downloads/nagios-4.0.8/cgi'
make: *** [Makefile:235: install] Error 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
/usr/bin/install -c -m 755 -o root -g root daemon-init /etc/rc.d/init.d/nagios
*** 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 ***

[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
/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 664 -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/timerperiods.cfg /usr/local/nagios/etc/objects/timerperiods.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/switc.cfg /usr/local/nagios/etc/objects/switc.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 nagcmd -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

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.

[root@ip-172-31-93-157 nagios-4.4.6]#

```

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 GROUPS
#####
# 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
}

G Help      ⌘C Write Out   ⌘W Where Is   ⌘R Cut           ⌘T Execute   ⌘G Location   M-U Undo   M-A Set Mark   M-J To Bracket   M-V Previous
X Exit      ⌘F Read File   ⌘N Replace   ⌘U Paste   ⌘J Justify   ⌘I Go To Line   M-E Redo   M-B Copy   ⌘Q Where Was   M-W Next

```

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 \  
    ln -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)  
PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157
```

```
[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 zxf 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
```

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 10: Compile and install plugins

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

```

/usr/bin/install -c -o nagios -g nagios check_dhcp /usr/local/nagios/libexec/check_dhcp
chmod root /usr/local/nagios/libexec/check_dhcp
chmod ug-rx,ufs /usr/local/nagios/libexec/check_dhcp
/usr/bin/install -c -o nagios -g nagios check_icmp /usr/local/nagios/libexec/check_icmp
chmod root /usr/local/nagios/libexec/check_icmp
chmod ug-rx,ufs /usr/local/nagios/libexec/check_icmp
chmod [2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/root/downloads/nagios-plugins-2.0.3/plugins-root'
make[1]: Leaving directory '/root/downloads/nagios-plugins-2.0.3/plugins-root'
taking install in po
make[1]: Entering directory '/root/downloads/nagios-plugins-2.0.3/po'
/usr/bin/mkdir -p /usr/local/nagios/share/gettext/po; \
for file in Makefile.in.in remove-potcdate.sin Makevars.template; do \
/usr/bin/install -c -o nagios -g nagios -m 644 ./${file} \
/usr/local/nagios/share/gettext/po/${file}; \
done; \
for file in Makevars; do \
rm -f /usr/local/nagios/share/gettext/po/${file}; \
done; \
else \
:; \
fi
make[1]: Leaving directory '/root/downloads/nagios-plugins-2.0.3/po'
make[1]: Entering directory '/root/downloads/nagios-plugins-2.0.3'
make[2]: Entering directory '/root/downloads/nagios-plugins-2.0.3'
make[2]: Nothing to be done for 'install-exec-am'.
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/root/downloads/nagios-plugins-2.0.3'
make[1]: Leaving directory '/root/downloads/nagios-plugins-2.0.3'
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# []

```

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 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

```

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

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

```

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](#)

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/rc.d/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
   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 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 80013;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]: 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]: Successfully launched command file worker with pid 80037
lines 1-26/26 (END)

```

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

Screenshot of the AWS EC2 Instances page showing a single instance named "nagios-host" (i-025f1d18f7c8a8cda) in the "Running" state. The instance type is t2.micro, located in us-east-1c with a public IP of ec2-3-86-. The sidebar on the left shows navigation links for EC2 Dashboard, Global View, Events, Console-to-Code, 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, and Elastic IPs.

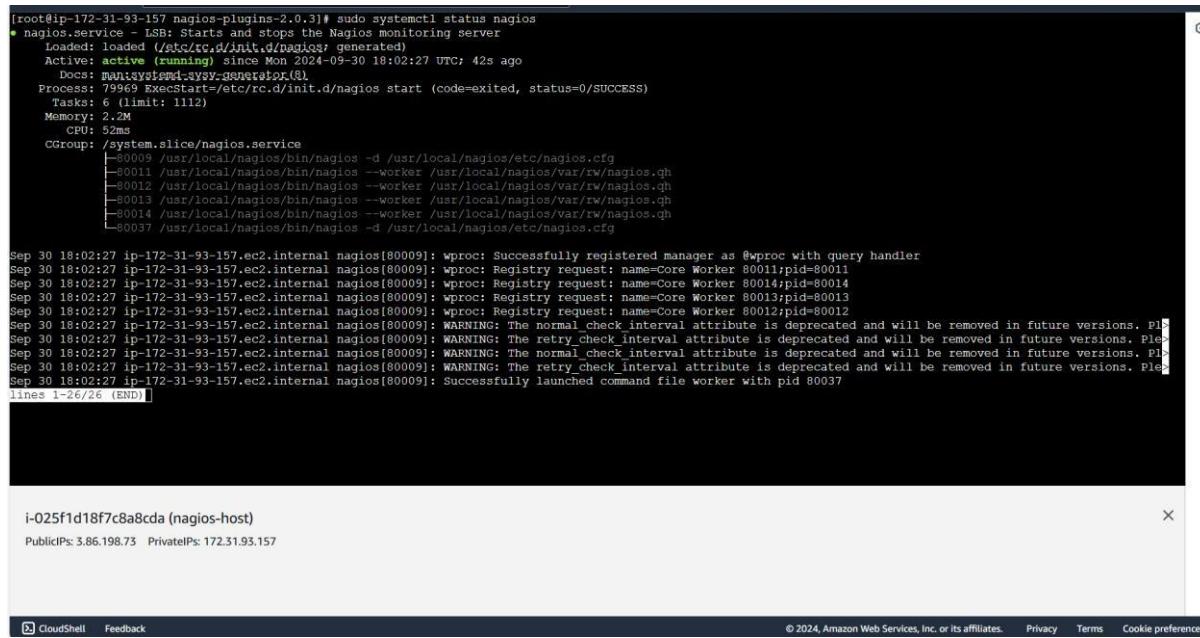
Step 13: Now visit http://<your_public_ip_address>/nagios Enter correct credentials and then you will see this page.

Screenshot of the Nagios Core web interface. The top header shows the URL as "Not secure | 3.86.198.73/nagios/". The main content area displays the Nagios Core logo and the message "Daemon running with PID 80009". Below this, it shows the version information: "Nagios® Core™ Version 4.4.6 April 28, 2020 Check for updates". A blue box at the top right says "A new version of Nagios Core is available! Visit nagios.org to download Nagios 4.5.5.". The left sidebar contains navigation links for General (Home, Documentation), Current Status (Tactical Overview, Map (Legacy), Hosts, Services, Host Groups, Problems, Services (Unhandled), Hosts (Unhandled), Network Outages, Quick Search), Reports (Availability, Trends (Legacy), Alerts, History, Summary, Histogram (Legacy), Notifications, Event Log), and System (Comments, Downtime, Process Info, Performance Info, Scheduling Queue, Configuration). The bottom footer includes copyright information, a link to the GNU General Public License, and logos for Nagios Core and 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.



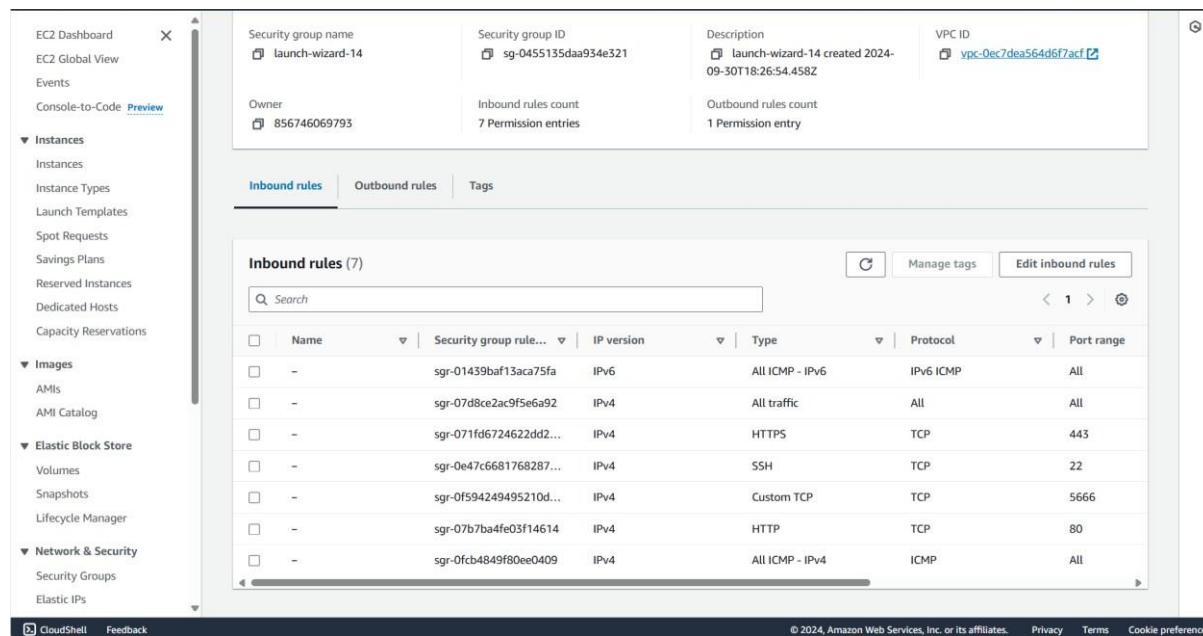
```
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# sudo systemctl status nagios
● nagios.service - LSB: Starts and stops the Nagios monitoring server
   Loaded: loaded (/etc/rc.d/init.d/nagios)
   Active: active (running) since Mon 2024-09-30 18:02:27 UTC; 42s ago
     Docs: man:sysvinit(8)
   Process: 10959 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 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 80013;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 retry_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 normal_check_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 retry_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: 3.86.198.73 PrivateIPs: 172.31.93.157

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

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-01439baf13aca75fa	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-07b7ba4fe0f14614	IPv4	HTTP	TCP	80	
sgr-0fcba4849f80ee0409	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 18:02:27 UTC; 42s ago
  Docs: man:systemd-SYSTEMD-GENERAL-SIG(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 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 80013;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 note that this warning was suppressed by the --no-warn-deprecated option.
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 note that this warning was suppressed by the --no-warn-deprecated option.
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 note that this warning was suppressed by the --no-warn-deprecated option.
Sep 30 18:02:27 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-2.0.31]# ps -ef | grep nagios
nagios 80009 1 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios 80011 80009 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 80012 80009 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 80013 80009 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 80014 80009 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 80037 80009 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
root 81960 3110 0 18:35 pts/1 0:00:00 grep --color=auto nagios
[root@ip-172-31-93-157 nagios-plugins-2.0.31]# 
```

i-025f1d18f7c8a8cda (nagios-host)
 Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

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

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-2.0.31# sudo su
root@ip-172-31-93-157 nagios-plugins-2.0.31# mkdir /usr/local/nagios/etc/objects/monitorhosts
root@ip-172-31-93-157 nagios-plugins-2.0.31# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
root@ip-172-31-93-157 nagios-plugins-2.0.31# 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.
ash: /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg: No such file or directory
root@ip-172-31-93-157 nagios-plugins-2.0.31# nano
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
ash: /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg: No such file or directory
root@ip-172-31-93-157 nagios-plugins-2.0.31# 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-2.0.31# 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-2.0.31# 
```

i-025f1d18f7c8a8cda (nagios-host)
 Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

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

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 5.8                               /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
example of how you can Create configuration entries 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
}

#####

^G Help      ^C Write Out   ^W Where Is   ^K Cut        ^E Execute   ^C Location   M-U Undo   M-Z Set Mark   M-J To Bracket   M-Q Previous
^X Exit      ^R Read File    ^V Replace    ^U Paste     ^J Justify    ^Y Go To Line  M-E Redo   M-B Copy      M-Q Where Was   M-W Next
```

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 5.8                               /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
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
}

#####

^G Help      ^C Write Out   ^W Where Is   ^K Cut        ^E Execute   ^C Location   M-U Undo   M-Z Set Mark   M-J To Bracket   M-Q Previous
^X Exit      ^R Read File    ^V Replace    ^U Paste     ^J Justify    ^Y Go To Line  M-E Redo   M-B Copy      M-Q Where Was   M-W Next
```

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 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/

```

GNU nano 5.8                               /usr/local/nagios/etc/nagios.cfg                         Modified [Q]
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/timerperiods.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[]

^C Help          ^C Write Out     ^M Where Is      ^M Cut        ^M Execute      ^M Location    M-U Undo      M-D Set Mark   M-[ To Bracket M-[ Previous
^X Exit          ^R Read File     ^V Replace       ^U Paste       ^J Justify      ^I Go To Line  M-E Redo      M-C Copy      M-] Where Was   M-] Next

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 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

Webpage: 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 file...

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]#

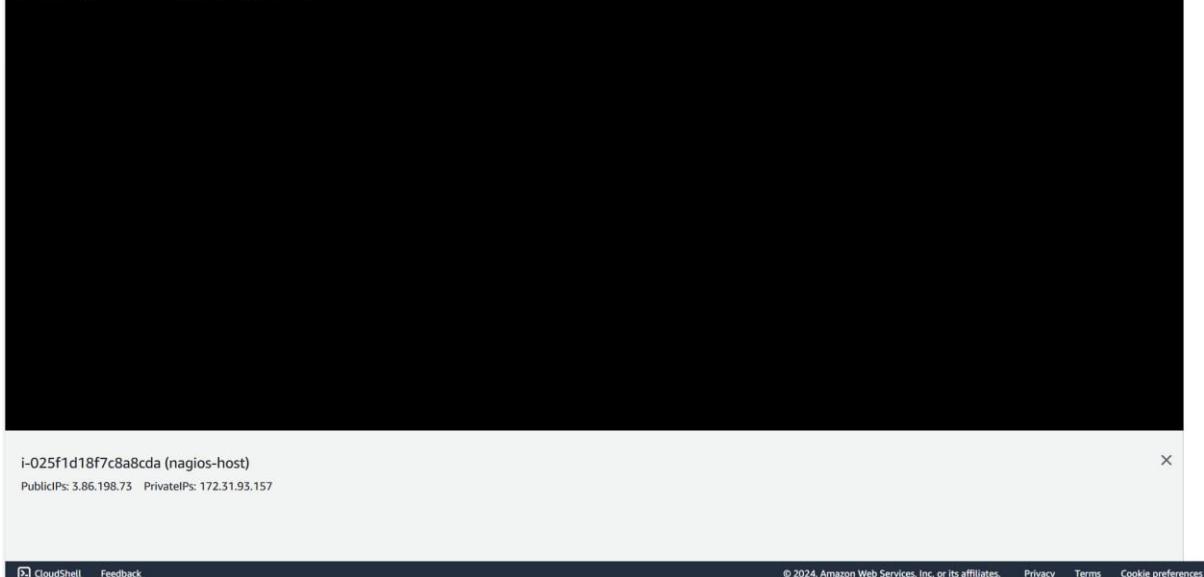
```

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

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

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



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

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

Instance state = running Clear filters

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

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)
IPv6 address: -
Hostname type: IP name: ip-172-31-93-157.ec2.internal
Answer private resource DNS name

Private IP DNS name (IPv4 only): ip-172-31-93-157.ec2.internal
Instance type: t2.micro

Private IPv4 addresses: 172.31.93.157
Public IPv4 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: Sep 30 19:13:49 UTC 2024
Updated every 30 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as nagiosadmin

[View Service Status Detail 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	16			

Host Status Details For All Host Groups

Host	Status	Last Check	Duration	Status Information
linuxserver	UP	09-30-2024 19:13:16	0d 0h 0m 33s+	PING OK - Packet loss = 0%, RTA = 1.82 ms
localhost	UP	09-30-2024 19:01:49	0d 1h 11m 22s	PING OK - Packet loss = 0%, RTA = 0.04 ms

Results 1 - 2 of 2 Matching Hosts

General

[Home](#) [Documentation](#)

Current Status

- [Tactical Overview Map \(Legacy\)](#)
- [Hosts](#)
- [Services](#)
- [Host Groups](#)
- [Summary](#)
- [Service Groups](#)
- [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 Top](#)

Advance devops- experiment 11

Jai navani

D15-A(31)

The screenshot shows the AWS Lambda Functions page with the following details:

Functions (5)

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

Create function

Choose one of the following options to create your function.

- Author from scratch
Start with a simple Hello World example.
- Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.
- Container image
Select a container image to deploy for your function.

Basic information

Function name: lambdaajikunal

Runtime: Python 3.12

Architecture: x86_64

Permissions: (not visible)

Screenshot of the AWS Lambda 'Create function' wizard:

Runtime: Python 3.12

Architecture: x86_64

Permissions: LabRole

Advanced settings

Success message: 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 details:

- Description: -
- Last modified: 2 seconds ago
- Function ARN: arn:aws:lambda:us-east-1:482322524237:function:lambdaJaikunal
- Function URL: Info

Function tabs: Code (selected), Test, Monitor, Configuration, Aliases, Versions

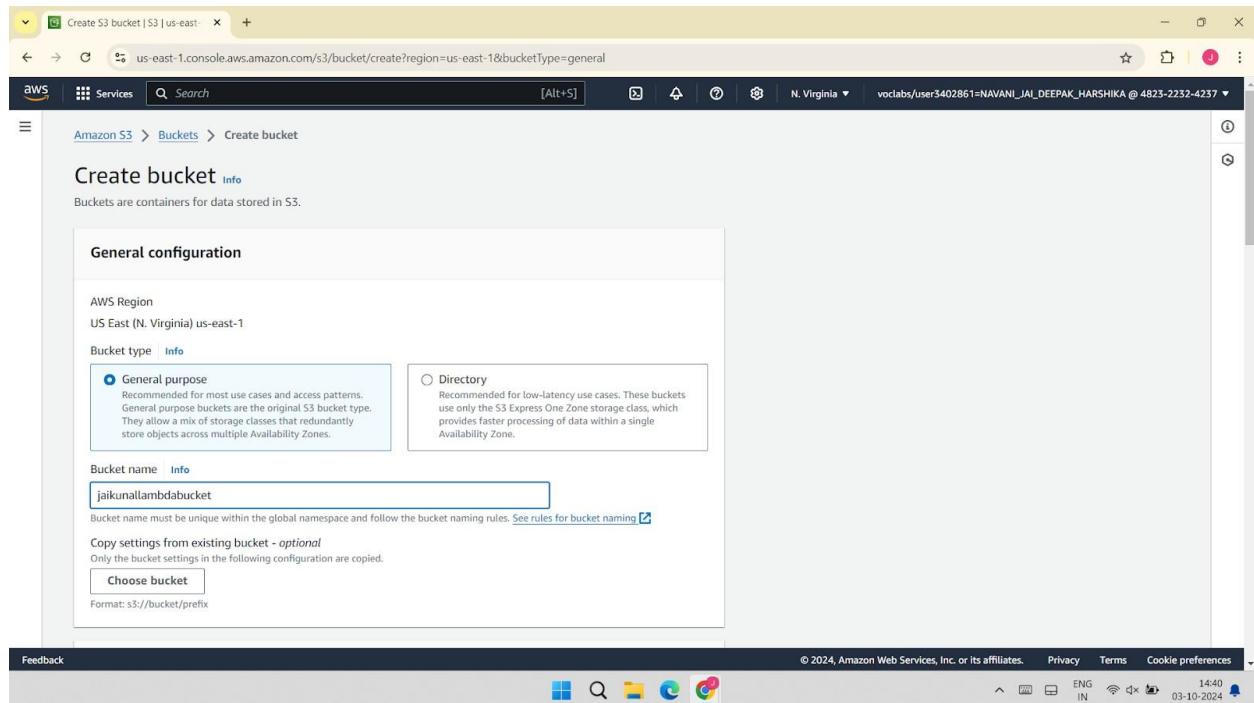
Advanced DevOps Exp-12

Jai Navani

D15A 31

Procedure:-

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



Upload objects - S3 bucket jailkunallambdabucket

us-east-1.console.aws.amazon.com/s3/upload/jailkunallambdabucket?region=us-east-1&bucketType=general

AWS Services Search [Alt+S] N. Virginia v vocabs/user3402861=NAVANI_JAI_DEEPAK_HARSHIKA @ 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

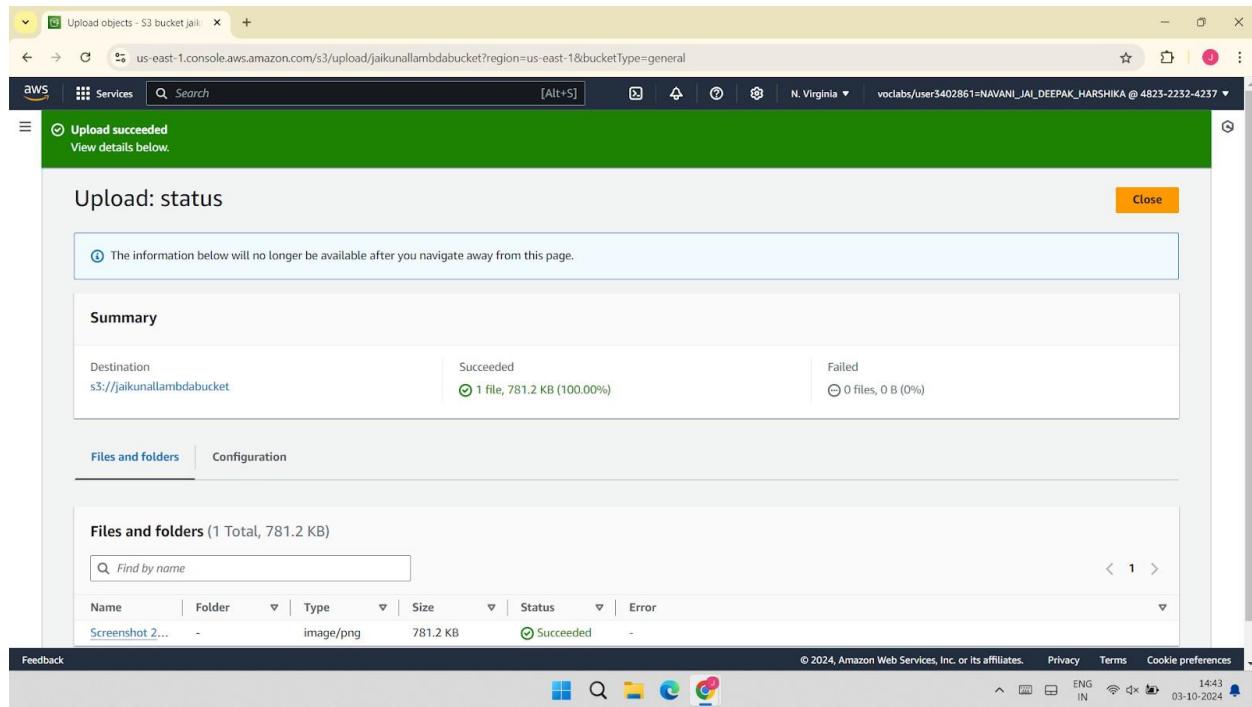
Destination	Succeeded	Failed
s3://jailkunallambdabucket	1 file, 781.2 KB (100.00%)	0 files, 0 B (0%)

Files and folders Configuration

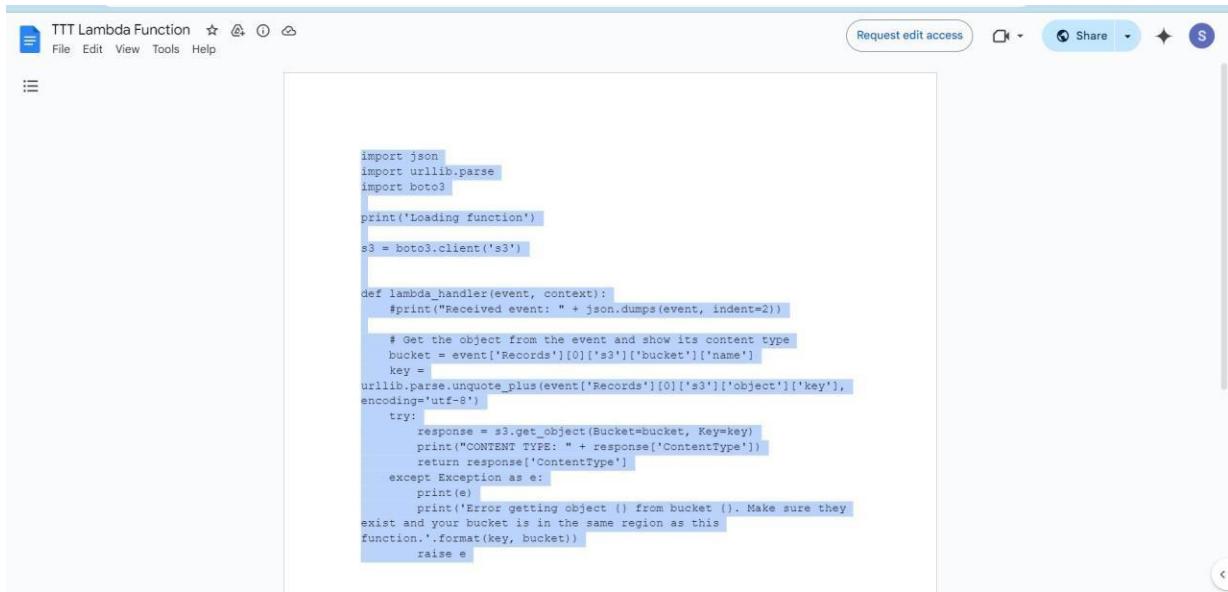
Files and folders (1 Total, 781.2 KB)

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 top bar displays the function name "TTT Lambda Function" and various navigation options like File, Edit, View, Tools, Help, Request edit access, Share, and a settings icon. The main area contains the following Python code:

```
import json
import urllib.parse
import boto3

print('Loading function')

s3 = boto3.client('s3')

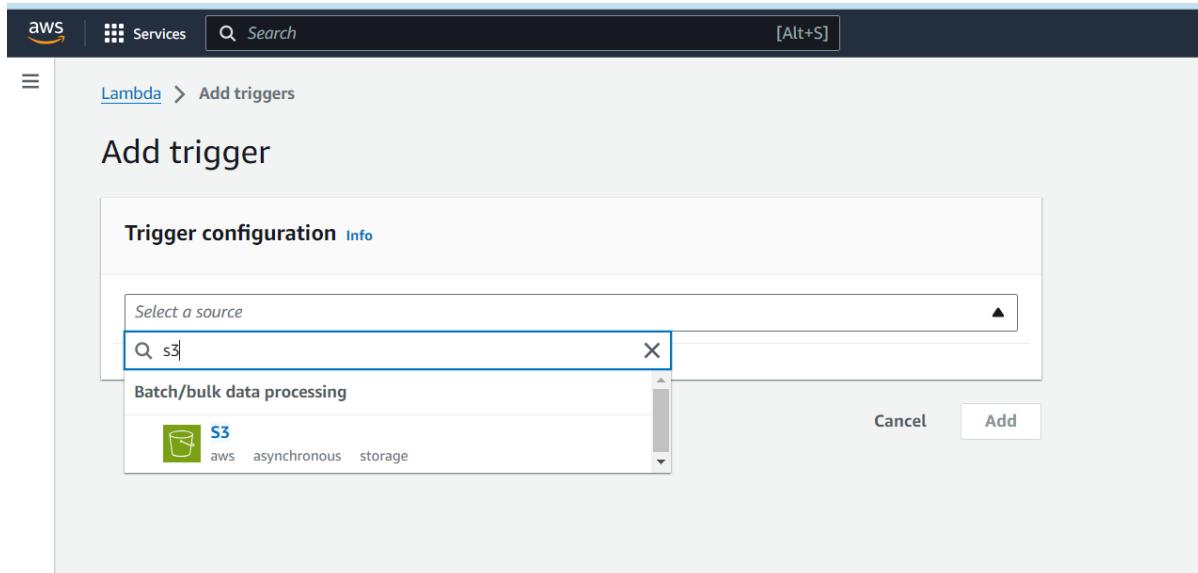
def lambda_handler(event, context):
    #print("Received event: " + json.dumps(event, indent=2))

    # 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
```

Screenshot of the AWS Lambda function code editor showing the Python script for the lambda_function:

```
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 this 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](#)

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](#) 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 .

No data events to display.

Configure in CloudTrail [\[Edit\]](#)

Event notifications (1)

Send a notification when specific events occur in your bucket. [Learn more \[?\]](#)

Name	Event types	Filters	Destination type
<input type="checkbox"/> 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-8a94e813-c025-4185-8c68-137a8a145ce0.statement.1

Resource-based policy document

```

1 Version: "2012-10-17",
2   "Id": "default",
3   "Statement": [
4     {
5       "Sid": "lambda-f873ffb0-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)		
Remove Add files Add folder		
All files and folders in this table will be uploaded.		
<input style="width: 150px; margin-right: 10px;" type="text"/> Find by name < 1 >		
<input type="checkbox"/>	Name	Folder
<input type="checkbox"/>	photo 1.jpeg	-
Remove Add files Add folder		

Destination [Info](#)

Destination

[s3://sanketbucket123](#)

☰ **Upload succeeded**
View details below.

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

Summary

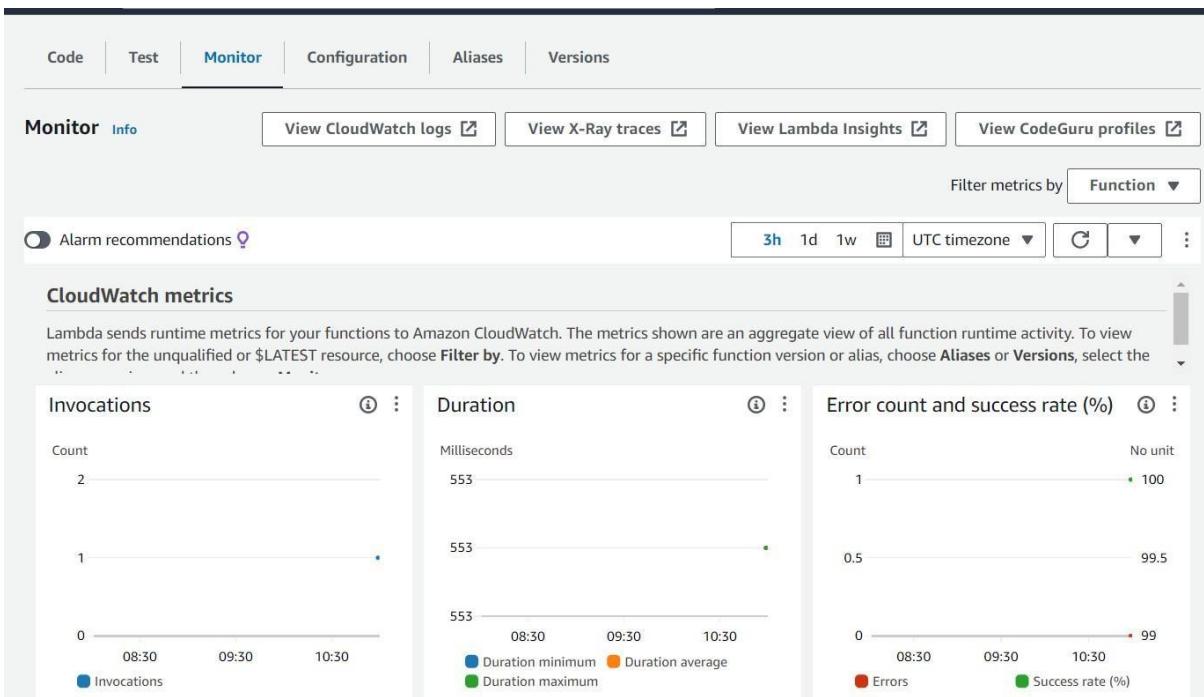
Destination s3://sanketbucket123	Succeeded 1 file, 78.6 KB (100.00%)	Failed 0 files, 0 B (0%)
---	--	-----------------------------

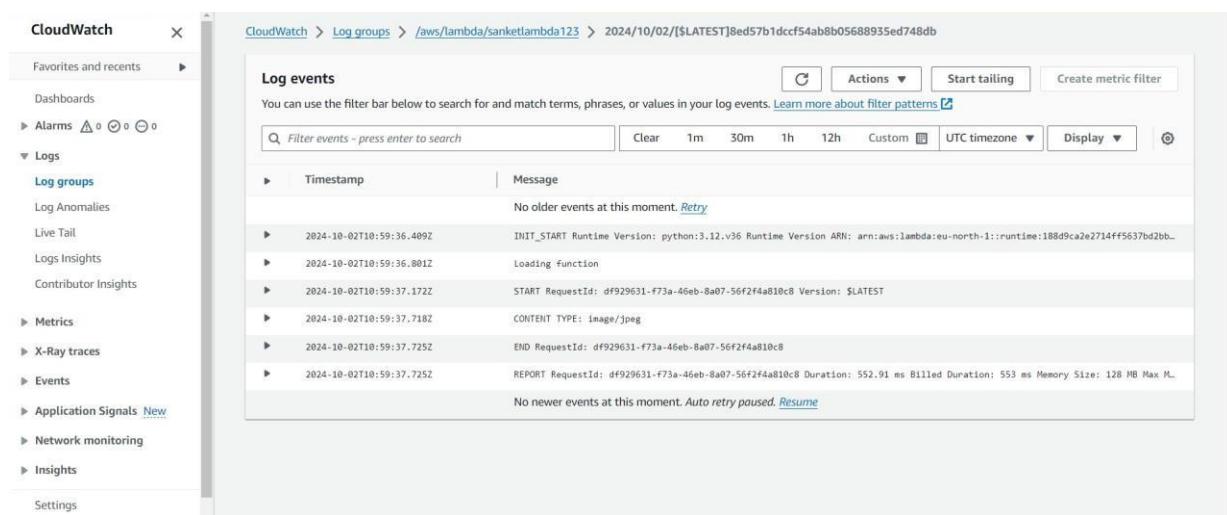
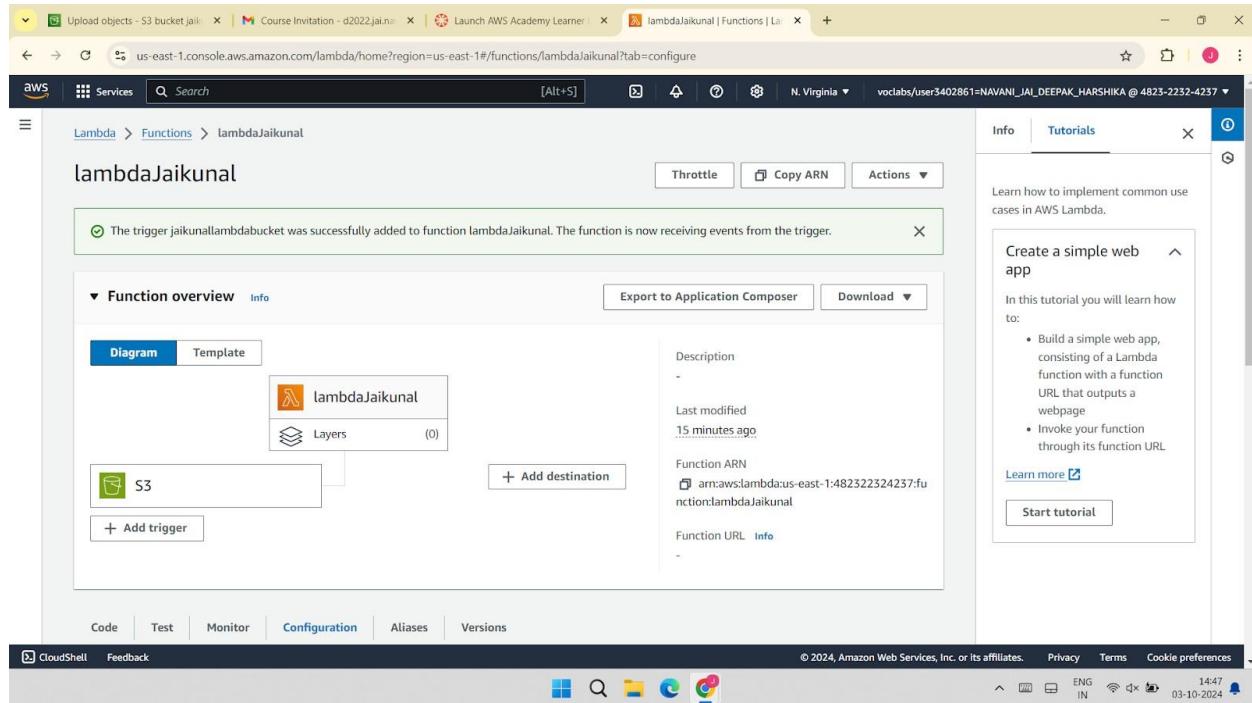
[Files and folders](#) [Configuration](#)

Files and folders (1 Total, 78.6 KB)

Name	Folder	Type	Size	Status	Error
sanket more ...	-	image/jpeg	78.6 KB	SUCCEEDED	-

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.





(05)
/05/2

Advance devops assignment no. 4

Q.1) Use S3 bucket and host video streaming.

Ans:

1) Log into 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 results.
- Click "Create Bucket"
- Give your bucket a unique name.
- Choose a region
- 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 on the upload button.
- Add your video file from your computer.
- Click "Upload" to start the upload process.

Set Permission for public Access.

- Confirm the action by clicking Make public again.

g. 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.
- Copy the URL

6. Edit the Bucket policy

{

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

 "Statement": [

 "Sid": "PublicReadGetObject",

 "Effect": "Allow",

 "Principal": "*",

 "Action": "S3:GetObject",

 "Resource": "arn:aws:s3:::video-bucket/*"

]

}

(Q.2) Discuss BMW and Hotstar case studies using AWS.

- BMW: A connected driving Experience BMW is a leader in car technology and AWS has helped them a lot in creating connected cars.
 - Connected cars features: BMW has developed features like remote diagnostic, over-the-air-updates, and real-time traffic info. AWS provides the necessary tools and infrastructure to make the features work.
 - Improving customer experience: They use AWS to enhance the customers interact with their cars for example ~~BMW~~ has added voice activated assistance and personalized suggestions using services like Amazon Lex.
 - Better manufacturing: BMW uses AWS to improve its manufacturing. By analyzing data from machines, they can find problem early and increase efficiency.
 - A streaming revolution: Hotstar is India's biggest video streaming platform offering a huge range of movies, TV shows and sports. AWS has been key of Hotstar success helping them.

- Manage huge traffic: During big events like IPL many people watch at once. AWS helps Hotstar handle this stage in traffic without slowing down.
- Quality streaming: AWS services like Amazon Kinesis ensures that users highly stream quality.
- Personalized Recommendation: Hotstar uses AWS machine learning tools to suggest contents to users making their viewing experience more enjoyable

Key AWS services:

- Compute : Amazon EC2
- Storage : Amazon S3, Amazon E PS

3) Why Kubernetes, and advantages and disadvantages of Kubernetes. Explain How adidas uses Kubernetes.

Kubernetes is popular because it simplifies the management of containerized applications. 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: Application can be moved easily between different environment without major changes.
- 2) Scalability: Kubernetes can automatically scale applications up or down based on traffic and demand.
- 3) Reliability: It features self-healing capabilities, meaning it can restart failed containers and balance workload.

Disadvantages of Kubernetes:

- 1) Complexity: It can be complicated to set up and manage, especially for those new to containerization.
- 2) Steep learning curve: Requires time and knowledge to fully understand and utilize its features.
- 3) Resource intensive: It may require more computing resources than simpler solutions.

Adidas has adopted Kubernetes to enhance its IT infrastructure and improve its ability to respond to market needs.

Q.4) What are Nagios and explain how Nagios are used in E-services?

Ans Nagios is an open source monitoring tool that helps the organization keeps track of their IT infrastructure, including servers, networks and applications. It provides a way to ensure that systems are running smoothly and alerts users if any issues arise.

- Key features of Nagios:
- Monitoring: Checks the performance and availability of servers, application and network devices.
 - Alerts: sends notification via email or SMS when problems occur, so teams can respond quickly.

How Nagios is used in E-services:

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

- Infrastructure Monitoring: Nagios monitors services, databases and other IT components to ensure they are operational. If a server goes down, Nagios alerts the IT team immediately.

Advance devops (Assignment no - 2)

Create a REST API with serverless framework
 Creating REST API with serverless framework
 efficient way to deploy serverless application that
 can scale automatically without messaging server
 A powerful tool that deployment of services and
 serverless application across various cloud providers
 such as AWS, and google cloud.

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

iii) REST API: Representational state transfer is
 architecture style for designing network application
 directly from your terminal.

Steps for creating REST API for serverless framework

~~Install Serverless framework:~~

You start by installing Serverless framework CLI
 globally using node modules. This allows you to
 manage Serverless application directly from your
 terminal.

Creating a node.js serverless project:
 A directory is created for your project, where you will
 initialize a serverless service. This service will house all
 your lambda functions configuration and cloud
 services. Using the command `serverless create` your AWS

Node.js microservices that will eventually deploy to AWS Lambda.

3) Project structure:-

The project scaffold creates essential files, like hand.js (which contains code for lambda function and serverless.yml).

4) Create a REST API Resource:

In the serverless.yml file you define function that handles port request of HTTP.

5) Deploy the service:

With the SLS deploy command, serverless framework packages your application, uploads necessary resources to AWS and set up the infrastructure.

6) Testing the API: Once deployed you can test REST API using tools like Postman by making POST request to generated API.

7) Storing Data in Dynamodb: To store submitted candidate data you integrate AWS' DynamoDB as database.

8) Adding more functionality: Adding functionalities like candidates get candidates by ID.

9) AWS IAM permissions

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

➤ Monitoring and maintenance.

This deployment serverless framework provides service information like deployed endpoints, API key, log streams.

Case study for sonarqube

Creating your own profile in Sonarqube for testing project quality. Use Sonarqube to analyze your code. Install Java plugin and analyze java code.

Sonarqube is an open source platform used for continuous inspection of quality. It detects bug, code smells and security vulnerabilities in project across programming languages.

➤ Profile creation in Sonarqube

Quality profiles in Sonarqube are essentials configuration that define 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 you can activate or deactivate rules, prioritize certain rules and configure parameters to profile to specific projects.

2) Using SonarCloud to analyse github code:
 SonarCloud is a cloud based counter part of sonarqube that triggers directly with github, Bit Bucket, and github repositories. To get started with sonarcloud via github signup product page and connect your github organization or personal account. Once connect, sonarcloud mirrors your github setup with each project corresponding to the github repos:- After setting up organization choose subscription plan. Next input repositories into your sonarcloud organization where each github repository is a sonarcloud project. Define new cloud 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 results can be viewed in both sonarcloud and github including security & import issue.

3) Sonarlint in Java IDE:

~~Sonarlint~~ is an IDE that performs on the fly code analysis as you write code, it helps developers in the developing 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 analyse Java code in code quality; promoting clean & maintainable code from beginning.

Analyzing Python projects with SonarQube

SonarQube supports Python test coverage reporting but it requires third party tools like coverage to enable and adjust your build process so that coverage tools runs before Sonar scanner and ensures report file is saved in diff path. For set up you can use TUX and coverage to py configuration for pytest and coverage to generate report in XML format. The build process can also be automated using GitHub Actions which install dependencies, runs test and invokes SonarQube scan. Ensure report in XML format and place where scanner can access it.

Analyzing Node.js projects with SonarQube

For node.js project SonarQube can analyse Java script and typescript code. Similar to the Python setup you can configure SonarQube to analyze node.js project 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.

3) At a large organization your centralized operation team may get many repeatable infrastructure requests you can use Terraform to build a self-service infrastructure model that lets product team manage their own infrastructure independently you can create and use Terraform modules that codify the standards for deploying & managing services in your organization, allowing teams to efficiently deploy services.

Thus implementing a self-service infrastructure model using Terraform can transform how large organizations manage their infrastructure independently. Organizations can enhance efficiency, reduce benefits and ensure compliance with established needs.

The need for self-service infrastructure:

In large organization, centralized operations teams often face an overwhelming number of repetitive requests. This can lead to delay in service delivery and frustration among the product teams who need to move quickly.

A self-service model allows teams to provision and manage their infrastructure without relying on the operations team for every request.

Benefits of using Terraform:

1. Modularity & Reusability:

Terraform modules encapsulate standard configurations for various infrastructure components. Teams can reuse those modules across different projects, reducing redundancy and minimizing the risks of errors.

Standardizations
By identifying and practices within modules, organization can ensure that all deployment comply with internal policies and standards. This consistency helps maintain security and operational integrity across the organization.

Increased Efficiency:
Product teams can deploy services quickly by using predefined modules, significantly reducing the time spent on infrastructure setup. This allows them to focus on developing features rather than managing infrastructure.

Integration with ticketing systems:
Terraform cloud can integrate with ticketing system like ServiceNow, to automate the generation of infrastructure requests. This integration streamlines workflow by allowing teams to initiate requests directly from their ticketing platform, reducing manual intervention.

Implementation steps:

1. Identify infrastructure components

Begin by identifying which components of your infrastructure can be modularized.

2. Develop Terraform modules.

- Create reusable modules that define the desired configuration and resources.
- Ensure each module includes input variables for customization and outputs for integration with other modules.

3. Establish governance and Best Practices

- Define guidelines for module usage, versions, and documentation to ensure clarity and maintainability.
- Encourage teams to contribute to module development and share improvements.

4. Testing and validation

- Implement a testing framework to validate module functionality before development.
- Best particular for module management.
- Utilize the Terraform registry.
- Leverage existing community modules from the Terraform registry.