

DevOps-Project-BankApp-CICD-BlueGreen-Monitoring- LogAggregartion Multi-Cloud (AWSandAzure)



By Ritesh Kumar Singh

Email Address: - riteshkumarsingh9559@gmail.com

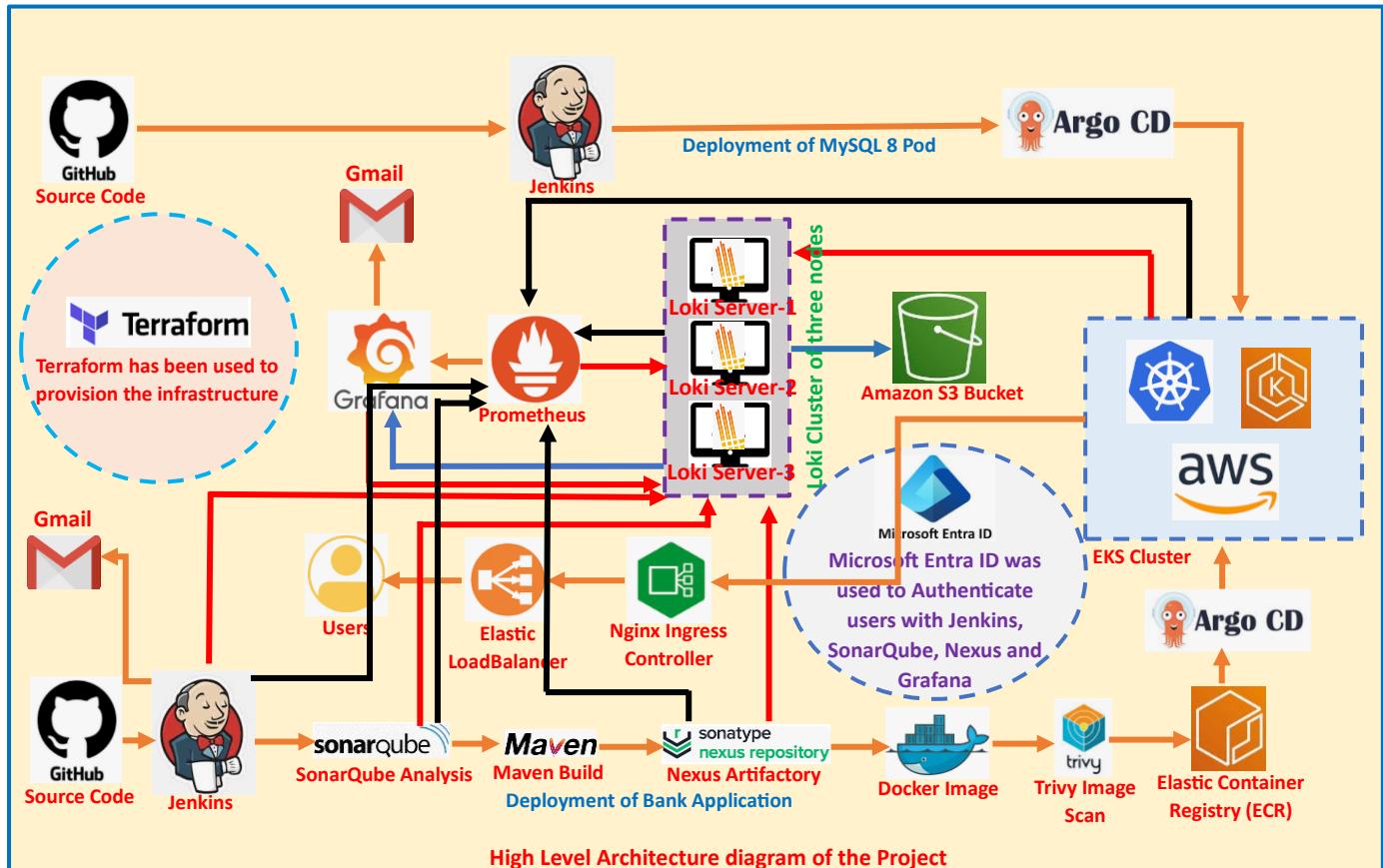
LinkedIn: - <https://www.linkedin.com/in/ritesh-kumar-singh-41113128b/>

GitHub: - <https://github.com/singhritesh85>



या कुन्दनतुषारहारधवला या शुभ्रवस्त्रावृता
या वीणावरदण्डमण्डितकरा या श्वेतपद्मासना।
या ब्रह्माच्युत शंकरप्रभृतिभिर्देवैः सदा वन्दिता
सा मां पातु सरस्वती भगवती निःशेषजाङ्घापहा ॥

DevOps-Project-BankApp-CICDBlueGreen-Monitoring-LogAggregation



This DevOps Project deals with creation of Infrastructure using Terraform and setup of CICD Pipeline using Jenkins, Monitoring using Prometheus and Grafana and Log Aggregation using Loki, Promtail and Grafana. SonarQube was used for Code-Analysis and Maven was used as the Build Tool. Nexus Artifactory was used to keep the Artifacts as shown in the Architecture diagram above. Trivy was used for Docker Image Scan. The Docker Image was kept in the Elastic Container Registry (ECR) and which was deployed to EKS Cluster using the ArgoCD as shown in the high-level architecture diagram above. User was able to access the Application through the Ingress and hence the Kubernetes Service. The source code was present in the GitHub Repository

<https://github.com/singhritesh85/Bank-App.git>. For this project to deploy a new Release of code I had used Blue-Green deployment with the help of ArgoCD Rollout. Promtail and Node Exporter was installed on all the Loki Servers, Grafana, Prometheus, Jenkins (Master and Slave Nodes) using the boot-strapping script. For the EKS cluster the Promtail and Node Exporter was installed with the help of helm.

Promtail is Log Aggregation Agent for Loki and Node Exporter agent collects the metrics and forward to Loki and Prometheus respectively. The high-level architecture diagram of the project is as shown above. Installation of Jenkins-Master, Jenkins-Slave, Nexus, SonarQube, Loki Servers, Prometheus and Grafana had been done using the bootstrapping Script.

To validate the SSL Certificate generated from AWS certificate manager I used DNS validation and did the entry for Azure DNS Zone record set of type CNAME as shown in the screenshot attached below.

Renewal status	Type	CNAME name	CNAME value
-	CNAME	_acm-validation.singhritesh85.com	_0.acm-validations.aws.

TTL	Value
172800	(Redacted)
3600	(Redacted)
3600	_0.acm-validations.aws.

Then I wait for around 20 seconds and after that SSL Certificate was issued as shown in the first screenshot attached above.

In my Azure Active Directory, I created a custom domain singhritesh85.com then made this custom domain as primary domain as shown in the screenshot attached below.

singhritesh85.com ...

Custom domain name

 Delete |  Got feedback?

 To use singhritesh85.com with your Microsoft Entra tenant, create a new TXT record with your domain name registrar using the info below.

Record type

TXT

MX

Alias or host name

@

C Copied

Destination or points to address

[REDACTED]



TTL

3600



[Share these settings via email](#)

Verification will not succeed until you have configured your domain with your registrar as described above.

[Verify](#)

Do the entry in Azure DNS Zone to create the record set with Name and value and type TXT as shown in the screenshot attached below.

Add record set

singhritesh85.com

Name

@

.singhritesh85.com

Type

TXT – Text records

TTL *

1

TTL unit

Hours

Value



The quick brown fox jumps over the lazy dog.

Add

Cancel

Give feedback

Then verified it and found it was verified successfully as shown in the screenshot attached below.

Home > Default Directory > Custom domain names >
singhritesh85.com

Custom domain name
Delete Got feedback!

To use singhritesh85.com with your Microsoft Entra tenant, create a new TXT record with your domain name registrar using the info below.

Record type: TXT MX
Alias or host name:
Destination or points to address:
TTL: 3600

Share these settings via email
Verification will not succeed until you have configured your domain with your registrar as described above.

Verify domain name
Successfully verified domain name singhritesh85.com for use within Default Directory

Then made this as a primary domain as shown in the screenshot attached below.

Home > Default Directory | Custom domain names >

singhritesh85.com

Custom domain name

<input checked="" type="checkbox"/> Make primary			
Type	Custom		
Status	Verified		
Federated	No		
Primary domain	No		
In use	No		
<hr/>			
Name	Status	Federated	Primary
singhritesh85.com			

Verified from above attached screenshot, in my Azure Entra ID, the primary domain is **singhritesh85.com**.

For this project I am creating AWS Resources and the Azure Resources using the single terraform script present in the GitHub Repo <https://github.com/singhritesh85/DevOps-Project-Bank-Application-Blue-Green-Deployment-Aws.git> at the path **terraform-bankapp-multicloud**. In my case the terraform was installed on an EC2 Installed and a sufficient Role was attached (RBAC) to the EC2 and installed azure-cli and provided the access as shown in the screenshot attached below. The state file and sate lock in terraform was achieved using the Azure Storage Account's container.

```
[root@REDACTED main]# echo -e "[azure-cli]
> name=Azure CLI
> baseurl=https://packages.microsoft.com/yumrepos/azure-cli
> enabled=1
> gpgcheck=1
> gpgkey=https://packages.microsoft.com/keys/microsoft.asc" | sudo tee /etc/yum.repos.d/azure-cli.repo
[azure-cli]
name=Azure CLI
baseurl=https://packages.microsoft.com/yumrepos/azure-cli
enabled=1
gpgcheck=1
gpgkey=https://packages.microsoft.com/keys/microsoft.asc
[root@REDACTED main]# yum install azure-cli -y
[root@REDACTED main]# az login
To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code REDACTED to authenticate
[
  {
    "cloudName": "AzureCloud",
    "homeTenantId": "REDACTED",
    "id": "REDACTED",
    "isDefault": true,
    "managedByTenants": [],
    "name": "REDACTED",
    "state": "Enabled",
    "tenantId": "REDACTED",
    "user": {
      "name": "REDACTED",
      "type": "user"
    }
  }
]
```

To create the Resources (AWS and Azure Resources) using the terraform script first switch the directory to **terraform-bankapp-multicloud/main** and then run the commands in below given orders.

terraform init -----> **Initialize the working directory by installing plugins and required providers**

terraform validate -----> **check the terraform script is correct or not**

terraform plan -----> **To verify what are the resources to be created**

terraform apply -auto-approve -----> **To create the resources**

terraform destroy -auto-approve -----> **Run this command only when you want to destroy all the resources.**

```
module.eks_cluster.azurerm_active_directory_domain_service.entra_ds: Still creating... [1h11m41s elapsed]
module.eks_cluster.azurerm_active_directory_domain_service.entra_ds: Still creating... [1h11m51s elapsed]
module.eks_cluster.azurerm_active_directory_domain_service.entra_ds: Still creating... [1h12m1s elapsed]
module.eks_cluster.azurerm_active_directory_domain_service.entra_ds: Creation complete after 1h12m4s [id=/subscriptions/[REDACTED]/resourceGroups/bankapp-rg/providers/Microsoft.AAD/domainServices/dexter-domainservices/initialReplicaSetId/[REDACTED]]
Releasing state lock. This may take a few moments...

Apply complete! Resources: 143 added, 0 changed, 0 destroyed.

Outputs:

acr_ec2_private_ip_alb_dns = {
  "EC2_Instance_Bloxbox_Exporter_Server_Private_IP_Address" = "[REDACTED]"
  "EC2_Instance_Grafana_Server_Private_IP_Address" = "[REDACTED]"
  "EC2_Instance_Jenkins_Master_Server_Private_IP_Address" = "[REDACTED]"
  "EC2_Instance_Jenkins_Slave_Server_Private_IP_Address" = "[REDACTED]"
  "EC2_Instance_Loki_Servers_Private_IP_Addresses" = [
    "[REDACTED]"
    "[REDACTED]"
    "[REDACTED]"
  ]
  "EC2_Instance_Prometheus_Server_Private_IP_Address" = "[REDACTED]"
  "EC2_Instance_SonarQube_Server_Private_IP_Address" = "[REDACTED]"
  "Grafana_ALB_DNS_Name" = "Grafana-[REDACTED].us-east-2.elb.amazonaws.com"
  "Jenkins_ALB_DNS_Name" = "jenkins-ms-[REDACTED].us-east-2.elb.amazonaws.com"
  "Loki_ALB_DNS_Name" = "Loki-[REDACTED].us-east-2.elb.amazonaws.com"
  "SonarQube_ALB_DNS_Name" = "SonarQube-[REDACTED].us-east-2.elb.amazonaws.com"
  "registry_id" = "[REDACTED]"
  "repository_url" = "02-[REDACTED].dkr.ecr.us-east-2.amazonaws.com/bankapp"
}
```

This terraform script took total 1 hour 15 minutes in all to got executed successfully as shown in the screenshot attached above. After that All the resources in AWS and Azure will be created. The Microsoft Entra Domain Services gets created and I did below change in the Microsoft Entra Domain Services as shown in the screenshot attached below.

The screenshot shows the Microsoft Entra Domain Services blade in the Azure portal. The 'Overview' tab is active. A yellow box highlights a warning message: 'Configuration issues for your managed domain were detected. Run configuration diagnostics to see a detailed diagnosis.' A red arrow points from this message to a red box labeled 'Click here'. Below the warning, there's a section titled 'Trust Relationship' with a 'Create a Trust' button.

The screenshot shows the 'dexter-domainservices' configuration diagnostics page. At the top, there's a 'Run' button highlighted with an orange arrow pointing to it, labeled 'Click here to Run it.' Below the run button, there's a warning icon and the text 'Warning'. Under the 'DNS records' section, there's a 'Fix' button highlighted with an orange arrow pointing to it, labeled 'Click here to fix the issue'.

Issues found

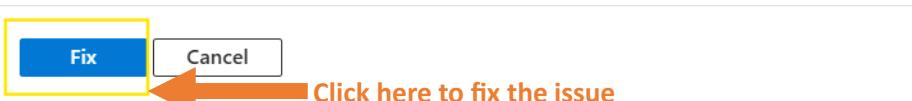
- DNS server settings for managed domain service IPs 192.168.1.5,192.168.1.4 need to be configured for virtual networks East US/bankapp-vnet

Resolution

According to Microsoft Entra Domain Services network configuration guidelines, the following fixes are proposed:

- Add DNS server settings for managed domain service IPs 192.168.1.5,192.168.1.4 on East US/bankapp-vnet.

The fixes can be carried out manually, or by clicking "Fix" below. By clicking "Fix" below, you agree the proposed fixes are carried out on your behalf.



By fixing this issue through click this button will change the DNS Servers for VNet from Default (Azure-Provided) to Custom as shown in the screenshot attached below.

The screenshot shows two Azure portal pages. The top page is 'dexter-domainservices | Properties' under 'Microsoft Entra Domain Services'. It displays the following details:

- DNS domain name:** singhritesh85.com
- Locations:** East US
- Virtual Networks/Subnets:** East US/bankapp-vnet/bankapp-vnet/default
- Network security groups:** East US/domain-services-nsg
- IP addresses:** East US/192.168.1.4 192.168.1.5 (highlighted)
- Secure LDAP:** Enabled
- Secure LDAP external IP addresses:** East US/172. [REDACTED] 41
- Synchronization:** All
- Admin group:** AAD DC Administrators

The bottom page is 'bankapp-vnet | DNS servers' under 'Virtual network'. It shows the following configuration:

- DNS servers:** Custom (radio button selected, highlighted)
- IP Address:** 192.168.1.4
192.168.1.5 (highlighted)
- Add DNS server:** Add DNS server

The 'DNS servers' option in the left navigation bar is also highlighted.

Then go to Security Setting of Microsoft Entra Domain Services and enable LDAP Signing and LDAP Channel Binding then save this setting as shown in the screenshot attached below.

dexter-domainservices | Security settings

Microsoft Entra Domain Services

Search Save Discard

Overview Activity log Access control (IAM) Tags Resource visualizer Settings Properties Secure LDAP Synchronization Custom attributes Replica sets Trusts Health Notification settings SKU Security settings

Enable or disable Kerberos RC4 encryption for your managed domain. When Kerberos RC4 encryption is disabled, all Kerberos requests that use RC4 encryption will fail.

Disable Enable

Kerberos Armoring

Enable or disable Kerberos Armoring for your managed domain. This will provide a protected channel between the Kerberos client and the KDC.

Disable Enable

LDAP Signing

Require all LDAP clients to request signing during bind time. Any bind request that does not request signing will fail.

Disable Enable

LDAP Channel Binding

Require all LDAP clients to provide channel binding information when communicating with the directory. Any client that does not provide this information will fail.

Disable Enable

LDAP Signing

Require all LDAP clients to request signing during bind time. Any bind request that does not request signing will fail.

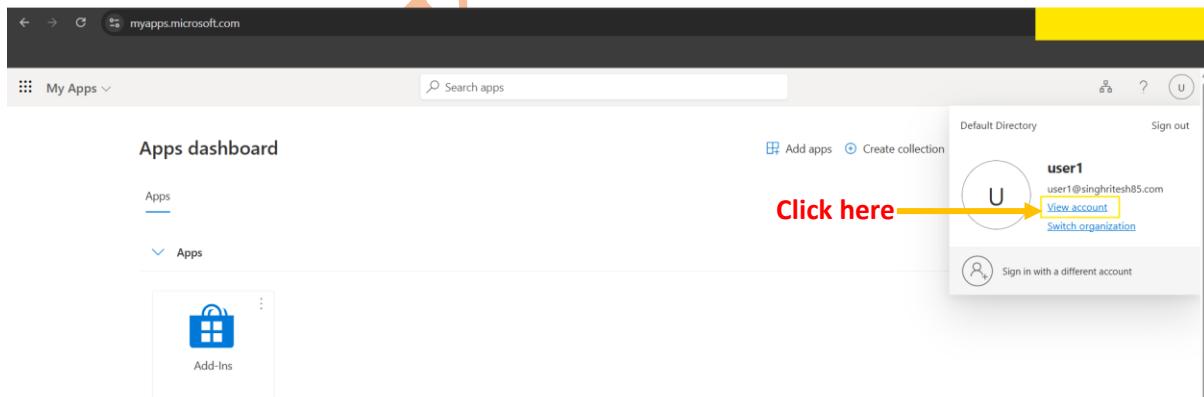
Disable Enable

LDAP Channel Binding

Require all LDAP clients to provide channel binding information when communicating with the directory. Any client that does not provide this information will fail.

Disable Enable

After successfully creating the Microsoft Entra Domain Services it is mandatory to change the password of user **user1** which was the member of Group **AAD DC Administrators**. To change the password of user **user1** I logged-in into the URL <https://myapps.microsoft.com> as shown in the screenshot attached below and changed the password.



The screenshot shows the Microsoft My Account interface at myaccount.microsoft.com/?ref=MeControl. The left sidebar has 'Security info' highlighted with a yellow box and a red arrow pointing to it. The main area displays 'user1' and 'user1@singhritesh85.com'. Below this is a navigation bar with 'Overview', 'Devices', 'Password', 'Organizations', 'Settings & Privacy', 'My sign-ins', 'My Apps', 'My Groups', and 'Give feedback'. A large central box contains sections for 'Security info', 'Devices', and 'Organizations'. The 'Security info' section includes a 'Keep your verification methods and security info up to date.' link and a 'UPDATE INFO >' button. The 'Devices' section shows a computer icon and a 'Disable a lost device and review your connected devices.' link. The 'Organizations' section shows a briefcase icon and a 'See all the organizations that you're a part of.' link.

The screenshot shows the Microsoft My Sign-Ins interface at mysignins.microsoft.com/security-info. The left sidebar has 'Security info' highlighted with a yellow box and a red arrow pointing to it. The main area displays 'Security info' with a note: 'These are the methods you use to sign into your account or reset your password.' It shows two sign-in methods: 'Password' (last updated) and 'Microsoft Authenticator' (Push multi-factor authentication (MFA)). A 'Change' button is highlighted with a yellow box and a red arrow pointing to it. A 'Lost device?' link is also visible.

The screenshot shows the Microsoft My Sign-Ins interface at mysignins.microsoft.com/security-info. The left sidebar has 'Security info' highlighted. A modal dialog box titled 'Change your password' is open. It shows the 'User ID' as 'user1@singhritesh85.com'. The 'New password' and 'Confirm new password' fields are empty and highlighted with yellow boxes. A 'Delete' link is visible next to the 'Confirm new password' field. At the bottom of the dialog are 'Cancel' and 'Submit' buttons, with 'Submit' highlighted with a yellow box and a red arrow pointing to it.

After running the terraform script LoadBalancers for SonarQube, Jenkins, Nexus, Loki and Grafana had been created as shown in the screenshot attached below. I created the record set of type CNAME for SonarQube, Jenkins, Nexus and Grafana in Azure DNS Zone using the DNS Name of the LoadBalancers for example I had shown below the creation of record set for SonarQube.

Load balancers (5)

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Name	DNS name	State	VPC ID	Availability Zones	Type
Loki	Loki-[REDACTED].us-east-2....	Active	vpc-[REDACTED]	3 Availability Zones	application
Grafana	Grafana-[REDACTED].us-east-2....	Active	vpc-[REDACTED]	3 Availability Zones	application
SonarQube	SonarQube-[REDACTED].us-...[REDACTED]	Active	vpc-[REDACTED]	3 Availability Zones	application
jenkins-ms	jenkins-ms-[REDACTED].us-...[REDACTED]	Active	vpc-[REDACTED]	3 Availability Zones	application
Nexus-ALB	Nexus-ALB-[REDACTED].us-...[REDACTED]	Active	vpc-[REDACTED]	3 Availability Zones	application

singhritesh85.com | Recordsets

A record set is a collection of records in a zone that have the same name and type. They are used to map a domain or subdomain to another record. If you don't see what you're looking for, try refreshing the page or using the search bar.

Name	Type
@	NS
@	SOA
sonarqube	CNAME

Add record set

Name: sonarqube

Type: CNAME – Link your subdomain to another record

Alias record set: No

TTL: 1

TTL unit: Hours

Alias: SonarQube-[REDACTED].us-east-2.elb.amazonaws.com

Add Cancel Give feedback

After doing the entry for all the DNS Names (of the LoadBalancers SonarQube, Jenkins, Nexus and Grafana) as shown in the screenshot attached below to create the record set. The final screenshot is as shown in the screenshot attached below.

singhritesh85.com | Recordsets

DNS zone

Search @ SOA 3600

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Resource visualizer
- Settings
- DNS Management
- Recordsets**
- DNSSEC
- Monitoring
- Automation
- Help

	[REDACTED]	CNAME	3600	[REDACTED]
grafana	grafana	CNAME	3600	Grafana-[REDACTED].us-east-2.elb.amazonaws.com
	jenkins-ms	CNAME	3600	jenkins-ms-[REDACTED].us-east-2.elb.amazonaws.com
	nexus	CNAME	3600	Nexus-ALB-[REDACTED].us-east-2.elb.amazonaws.com
	sonarqube	CNAME	3600	SonarQube-[REDACTED].us-east-2.elb.amazonaws.com

It is possible to monitor Jenkins Job using Prometheus and Grafana and for that I installed **Prometheus Metrics plugin in Jenkins, after its installation I restarted the Jenkins as shown in the screenshot attached below.**

jenkins-ms.singhritesh85.com/manage/pluginManager/available

Jenkins

Ritesh Kumar Singh log out

Dashboard > Manage Jenkins > Plugins

Plugins

Available plugins

Updates

Installed plugins

Advanced settings

Download progress

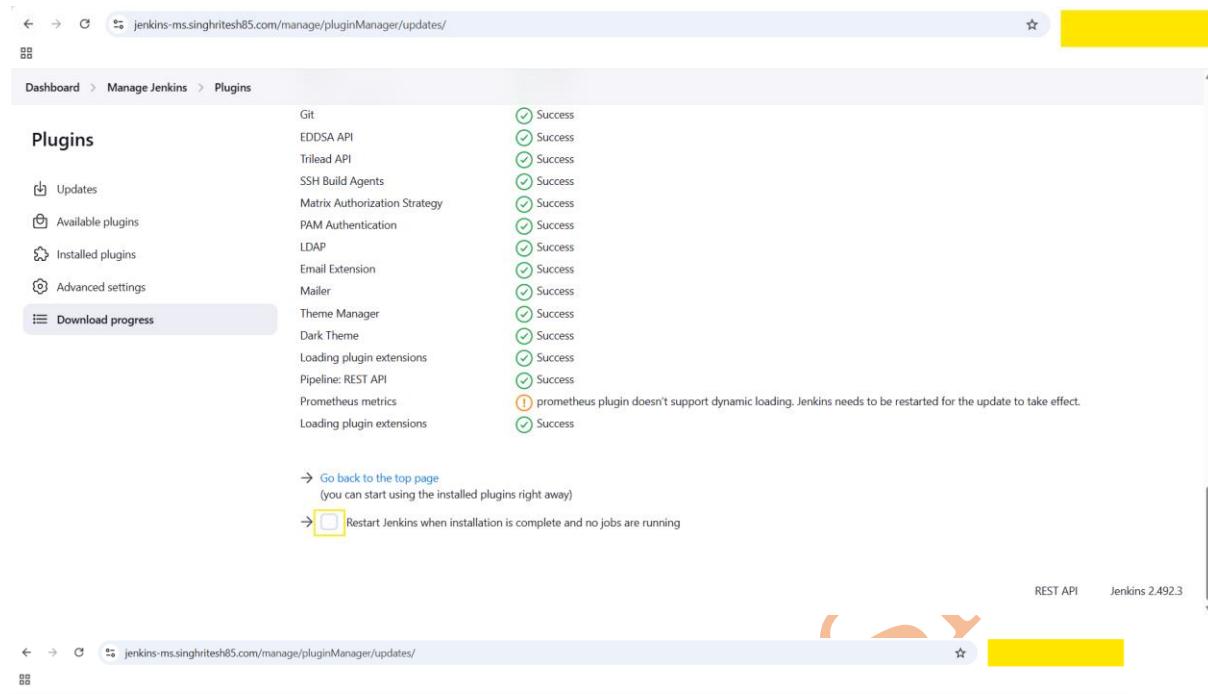
prometheus metrics

Install Name Released

Prometheus metrics [REDACTED] monitoring Miscellaneous 1 mo 16 days ago

Jenkins Prometheus Plugin expose an endpoint (default /prometheus) with metrics where a Prometheus Server can scrape.

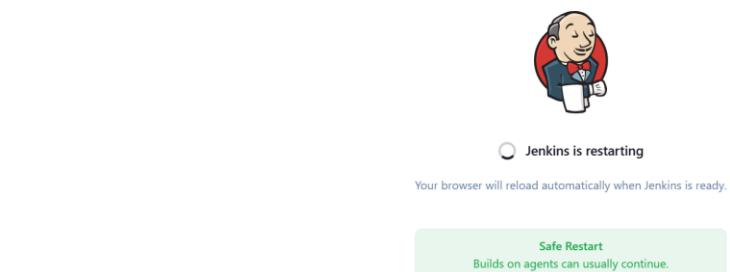
REST API Jenkins 2.492.3



The screenshot shows the Jenkins plugin manager interface. The left sidebar has options like 'Updates', 'Available plugins', 'Installed plugins', 'Advanced settings', and 'Download progress' (which is selected). The main table lists various Jenkins plugins with their status:

Plugin	Status
Git	Success
EDDSA API	Success
Trilead API	Success
SSH Build Agents	Success
Matrix Authorization Strategy	Success
PAM Authentication	Success
LDAP	Success
Email Extension	Success
Mailer	Success
Theme Manager	Success
Dark Theme	Success
Loading plugin extensions	Success
Pipeline: REST API	Success
Prometheus metrics	Info: prometheus plugin doesn't support dynamic loading. Jenkins needs to be restarted for the update to take effect.
Loading plugin extensions	Success

Below the table are two links: 'Go back to the top page' and 'Restart Jenkins when installation is complete and no jobs are running'. At the bottom right, it says 'REST API' and 'Jenkins 2.492.3'.



Now, login into the Grafana for the first time and update admin password then login into Grafana and created two data sources each for prometheus and loki as shown in the screenshot attached below.

The image contains three separate screenshots of the Grafana interface, each showing the configuration of a different data source:

- Prometheus Data Source Configuration:**
 - General Settings:** Shows the 'prometheus' data source with 'Type: Prometheus'. The 'Name' field is set to 'prometheus'.
 - Connection:** The 'Prometheus server URL' is set to 'http://10.10.4.87:9090'.
 - Authentication:** Shows options for 'Incremental querying (beta)', 'Disable recording rules (beta)', 'Custom query parameters', 'HTTP method' (set to 'POST'), and 'Use series endpoint'.
 - Success Message:** A green box indicates 'Successfully queried the Prometheus API.'
 - Action Buttons:** Includes 'Delete' and 'Save & test' buttons.
- Loki Data Source Configuration:**
 - General Settings:** Shows the 'loki' data source with 'Type: Loki'. The 'Name' field is set to 'loki'.
 - Connection:** The 'URL' is set to 'http://Loki[REDACTED]us-east-2.elb.amazonaws.com'.
 - Authentication:** Shows the 'Authentication methods' section, which is currently empty.
- Grafana General Interface:**
 - The top navigation bar shows 'Connections > Data sources > loki'.
 - Header tabs include 'Type: Loki', 'Alerting: Supported', 'Explore data', and 'Build a dashboard'.
 - A search bar at the top right is present.

Alerting
Manage alert rules for the Loki data source. [Learn more about alerting](#)

Manage alert rules in Alerting UI

Queries
Additional options to customize your querying experience. [Learn more about query settings](#)

Maximum lines

Derived fields
Derived fields can be used to extract new fields from a log message and create a link from its value. [Learn more about derived fields](#)

+ Add

✓ Data source successfully connected.
Next, you can start to visualize data by building a dashboard, or by querying data in the Explore view.

Delete Save & test

Now I will configure the Integration of Azure Entra ID with Jenkins, SonarQube and Grafana for Authentication

Configuration of Integration of Azure Entra ID with Jenkins

Search from the list of Available plugins for **Microsoft Entra ID Plugin** in Jenkins and install it as shown in the screenshot attached below.

jenkins-ms.singhritesh85.com/manage/pluginManager/available

Jenkins

Ritesh Kumar Singh log out

Dashboard > Manage Jenkins > Plugins

Plugins

Updates

Available plugins Installed plugins

Advanced settings

Search: entra id

Install	Name	Released
<input checked="" type="checkbox"/>	Microsoft Entra ID (previously Azure AD) <small>5.1.0</small>	12 days ago
	Security Authentication and User Management <small>azure</small>	

A Jenkins authentication & authorization plugin for Microsoft Entra ID (a.k.a. Azure Active Directory, Azure AD)

REST API Jenkins 2.492.3

After installation of Azure Entra ID Plugin in Jenkins restart Jenkins and go to the Azure Portal and open Entra ID and create a new App Registration as shown in the screenshot attached below.

Home > Default Directory | App registrations >
Register an application ...

* Name
The user-facing display name for this application (this can be changed later).

Supported account types

Who can use this application or access this API?

- Accounts in this organizational directory only (Default Directory only - Single tenant)
- Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)
- Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
- Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be

By proceeding, you agree to the Microsoft Platform Policies

Register



Home > Default Directory | App registrations >

Register an application

Supported account types

Who can use this application or access this API?

- Accounts in this organizational directory only (Default Directory only - Single tenant)
- Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)
- Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
- Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from [Enterprise applications](#).

By proceeding, you agree to the Microsoft Platform Policies

Register

After App Registration in Azure go to Jenkins and **Manage Jenkins > Security > Security Realm and Select the Azure Active Directory** and provide the Client ID, Client Secret and Tenant ID then Save it as shown in screenshot attached below.

The screenshot shows the Jenkins 'Manage Jenkins > Security' page. Under 'Authentication', there is a section for 'Azure Active Directory'. It includes fields for 'Client ID' (redacted) and 'Secret' (redacted). Below these are 'Save' and 'Apply' buttons.

In the Registered Application go to Application and under **Implicit grant and hybrid flows** and select **ID Tokens** as shown in the screenshot attached below.

The screenshot shows the 'Authentication' section of the Microsoft Azure App Registration. Under 'Implicit grant and hybrid flows', the 'ID tokens (used for implicit and hybrid flows)' checkbox is checked and highlighted with a yellow box. Other options like 'Access tokens (used for implicit flows)' are also present. At the bottom are 'Save' and 'Discard' buttons.

Now select API Permission > Add a permission > Microsoft Graph > Application Permissions People.Read.All, Group.Read.All, User.Read.All and Directory.Read.All in the created Registered App. Then Click on Grant admin consent as shown in the screenshot attached below

Home > jenkins-login

jenkins-login | API permissions

[Search](#) [Refresh](#) [Got feedback?](#)

[Overview](#) [Quickstart](#) [Integration assistant](#) [Diagnose and solve problems](#) [Manage](#) [Branding & properties](#) [Authentication](#) [Certificates & secrets](#) [Token configuration](#) [API permissions](#) [Expose an API](#) [App roles](#) [Owners](#) [Roles and administrators](#) [Manifest](#)

Configured permissions

Granting tenant-wide consent may revoke permissions that have already been granted tenant-wide for that application. Permissions that users have already granted on their own behalf aren't affected. [Learn more](#)

The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect the value in your organization, or in organizations where this app will be used. [Learn more](#)

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (1)				
User.Read	Delegated	Sign in and read user profile	No	...

To view and manage consented permissions for individual apps, as well as your tenant's consent settings, try [Enterprise applications](#).

Home > jenkins-login

jenkins-login | API permissions

[Search](#) [Refresh](#) [Got feedback?](#)

[Overview](#) [Quickstart](#) [Integration assistant](#) [Diagnose and solve problems](#) [Manage](#) [Branding & properties](#) [Authentication](#) [Certificates & secrets](#) [Token configuration](#) [API permissions](#) [Expose an API](#) [App roles](#) [Owners](#) [Roles and administrators](#) [Manifest](#)

Configured permissions

Granting tenant-wide consent may revoke permissions that have already been granted tenant-wide for that application. Permissions that users have already granted on their own behalf aren't affected. [Learn more](#)

The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect the value in your organization, or in organizations where this app will be used. [Learn more](#)

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (5)				
Directory.Read.All	Application	Read directory data	Yes	Granted for Default Dire...
Group.Read.All	Application	Read all groups	Yes	Granted for Default Dire...
People.Read.All	Application	Read all users' relevant people lists	Yes	Granted for Default Dire...
User.Read	Delegated	Sign in and read user profile	No	Granted for Default Dire...
User.Read.All	Application	Read all users' full profiles	Yes	Granted for Default Dire...

To view and manage consented permissions for individual apps, as well as your tenant's consent settings, try [Enterprise applications](#).

For Authorization select Azure Active Directory Matrix based Authorization as shown in the screenshot attached below, to login for the very first time provide Authenticated users as overall access and the login with a user and add other users if needed.

← → ⌂ [jenkins-ms.singhritesh85.com/manage/configureSecurity/](#)

Dashboard > Manage Jenkins > Security

Authorization

Azure Active Directory Matrix-based security

User/group	Overall	This permission allows users to run jobs as them on agents.										Job	Run	View	SCM	Metrics
		Credentials	Agent	Administrator	Read	Create	Delete	Update	View	Build	Configure					
Anonymous	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Authenticated Users	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Auth User/group to add

Q. Search for a name

Add

Markup Formatter

Markup Formatter ?

Save Apply

Below screenshot shows the user and group in Azure Entra ID Which I had created initially using the terraform.

The screenshot shows the Azure Entra ID portal. At the top, there is a card for 'user1' with details: Email - user1@singhritesh85.com, Type - Member, and Object ID - [REDACTED]. Below this, there is a card for the 'AAD DC Administrators' group with details: Type - Security, and Status - Assigned. The main area shows the 'AAD DC Administrators' group members page. The left sidebar has 'Members' selected under 'Manage'. The table lists one member: 'user1' (Type: User, User type: Member, Object Id: [REDACTED]).

For the very first time I logged-in with a user then I provided user1 as administrator privileges and removed the Authenticated user overall administrator access (and provided overall Read access) which I provided before first time logged-in as shown in the screenshot attached below.

The screenshot shows the Jenkins security matrix configuration page. The matrix is organized by User/group (Anonymous, Authenticated Users, user1) across various Jenkins features (Overall, Credentials, Agent, Job, Run, View, SCM, Metrics). For 'user1', the 'Overall' row has 'Administrator' checked. In the 'Agent' row, 'Configure' and 'Create' are checked. In the 'Job' row, 'Discover' and 'Create' are checked. In the 'Run' row, 'Delete' is checked. In the 'View' row, 'Read' is checked. In the 'Metrics' row, 'Read' is checked. At the bottom, there is a 'Save' button highlighted with a yellow box.

Below screenshot shows the access which I had when I logged-in with user1.

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job +

Set up a distributed build

Set up an agent

Configure a cloud

Learn more about distributed builds ?

REST API Jenkins 2.492.3

Integration of Azure Entra ID with SonarQube

To Integrate Azure Active Directory with SonarQube I installed Azure Active Directory (AAD) Authentication Plug-in for SonarQube as shown in the screenshot attached below.

Plugin Name	Version	Description	Action
Ansible Lint EXTERNAL ANALYZERS	2.5.1	Support for SonarQube 9.2 ...	Install
Apigee EXTERNAL ANALYZERS	3.0.2	Support for SQ 9.7+ ...	Install
Azure Active Directory (AAD) Authentication Plug-in for SonarQube INTEGRATION	1.3.2	Updates commons-text to fix CVE-2022-42889. ...	Install
CVS INTEGRATION	1.1.1	Fix classnotfound error ...	Install
Checkstyle EXTERNAL ANALYSERS	10.21.1	Upgrade to Checkstyle 10.21.1 ...	Install
Chinese Pack LOCALIZATION	9.9	Support SonarQube 9.9 ...	Install
Clover COVERAGE	4.1	Makes the plugin compatible with SQ 7.9 ...	Install
Codehawk Java EXTERNAL ANALYSERS	1.6	Added new rules ...	Install

Then restarted the sonarqube server as shown in the screenshot attached below.

Screenshot of the SonarQube Marketplace page. The URL is sonarqube.singhritesh85.com/admin/marketplace. The page shows a list of available plugins:

- 1C (BSL) Community Plugin** (Code Analyzer for 1C (BSL)) - Version 1.16.2, Support SQ 25+ ...
- AEM Rules for SonarQube** (EXTERNAL ANALYZERS) - Adds rules for AEM Java development - Version 1.6, SonarQube 8.9 LTS compatibility release due to underlying Java plugin API changes ...
- Ansible Lint** (EXTERNAL ANALYZERS) - Analyze Ansible playbooks - Version 2.5.1, Support for SonarQube 9.2 ...
- Apigee** (EXTERNAL ANALYZERS) - Adds XML rules test Apigee apiproxies - Version 3.0.2, Support for SQ 9.7+ ...
- Azure Active Directory (AAD) Authentication** (Plug-in for SonarQube) (INTEGRATION) - Allows the use of Azure Active Directory as an authentication source for SonarQube - Version 1.3.2, Updates commons-text to fix CVE-2022-42889 ...
- CVS INTEGRATION** - Provides SCM CVS integration - Version 1.1.1, Fix classnotfound error ...

Each plugin entry includes a "Homepage" and "Issue Tracker" link, license information, and an "Install" button. A message at the top says "SonarQube needs to be restarted in order to install 1 plugins" with "Restart Server" and "Revert" buttons.

Go to SonarQube UI then to Administration > Configuration > General Settings > General and Add the Server Base URL as shown in the screenshot attached below.

Screenshot of the SonarQube General Settings page under Administration > Configuration > General. The URL is sonarqube.singhritesh85.com/admin/settings. The "General" tab is selected. In the "Server base URL" section, the input field contains `https://sonarqube.singhritesh85.com/`. The sidebar on the left shows other settings categories like Analysis Scope, Authentication, Azure Active Directory, DevOps Platform Integrations, External Analyzers, and Languages.

Create the App Registration in Azure Entra ID as shown in the screenshot attached below.

Home > Default Directory | App registrations >
Register an application ...

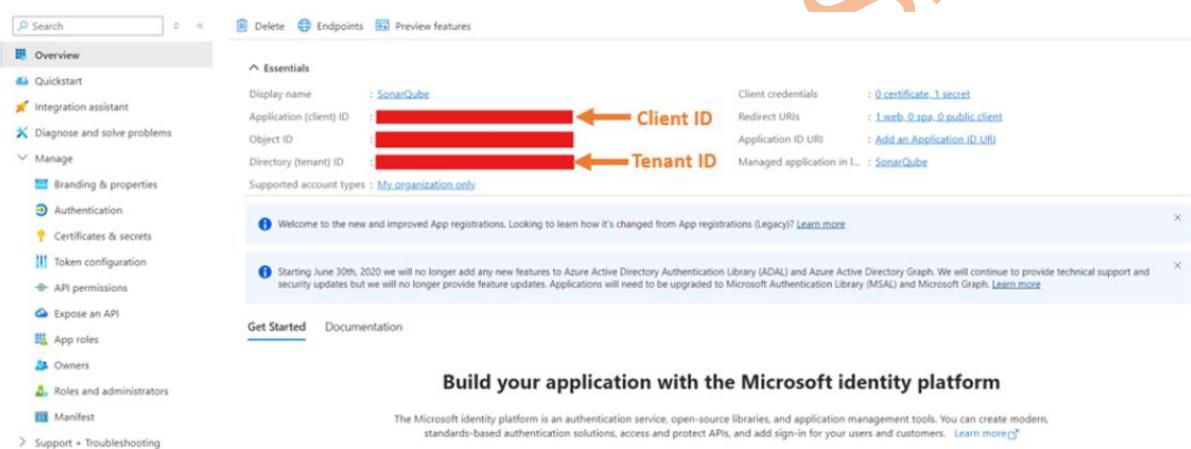
Supported account types
 Who can use this application or access this API?
 Accounts in this organizational directory only (Default Directory only - Single tenant)
 Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)
 Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
 Personal Microsoft accounts only
[Help me choose...](#)

Redirect URI (optional)
 We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

By proceeding, you agree to the Microsoft Platform Policies [Learn more](#)

 Register

Create the client secret as shown in the screenshot attached below.



The screenshot shows the Azure App Registrations Overview page. On the left, there's a sidebar with various navigation items like Overview, Quickstart, Integration assistant, etc. The main area is titled 'Overview' and contains sections for 'Essentials' and 'Supported account types'. In the 'Essentials' section, the 'Display name' is 'SonarQube', and the 'Client credentials' field shows '0_certificate_1 secret'. Below this, 'Client ID' and 'Tenant ID' are listed, both of which are redacted with a large red arrow pointing to them. Other fields shown include 'Redirect URIs' (1 web, 0 spa, 0 public client), 'Application ID URI' (Add an Application ID URI), and 'Managed application in ...' (SonarQube). At the bottom, there are two informational banners about the new app registration experience and the end of ADAL support.

Now go to Sonarqube UI and then to **Administration > Configuration > General Settings > Azure Active Directory** as shown in the screenshot attached below and enable Azure AD users to login then provide the **client id**, **client secret** and **tenant id** as shown in the screenshot attached below.

The screenshots show the 'General Settings' section of the SonarQube administration interface, specifically for configuring Azure Active Directory (AAD) authentication.

Screenshot 1: Configuration - General Settings - AAD Authentication

- Enabled:** A toggle switch is set to **Enabled**. A tooltip indicates: "Enable Azure AD users to login. Value is ignored if client ID and secret are not defined." Below it, a key value is listed: "Key: sonar.auth.aad.enabled".
- Client ID:** A field showing the Client ID provided by Azure AD during application registration, with a **Change** button.

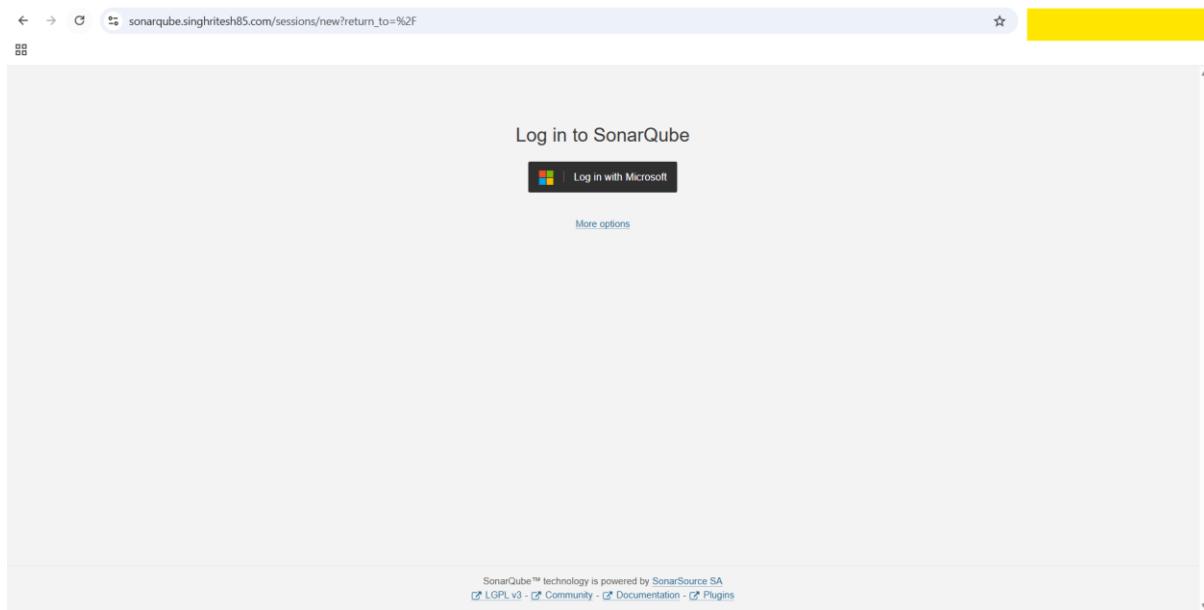
Screenshot 2: Configuration - General Settings - AAD Authentication (continued)

- Client Secret:** A field showing the Client key provided by Azure AD during application registration, with a **Change** button. Below it, a key value is listed: "Key: sonar.auth.aad.clientSecret.secured".
- Tenant ID:** A field showing the Azure AD Tenant ID, with a **Change** button. Below it, a key value is listed: "Key: sonar.auth.aad.tenantId".
- Allow users to sign-up:** A toggle switch is set to **Enabled**.

Screenshot 3: Configuration - General Settings - AAD Authentication (continued)

- Login generation strategy:** A dropdown menu is set to **Same as Azure AD login**. Below it, a key value is listed: "Key: sonar.auth.aad.loginStrategy".
- Directory Location:** A dropdown menu is set to **Azure AD (Global)**. Below it, a key value is listed: "Key: sonar.auth.aad.directoryLocation".
- Enable Client Credential Flow:** A checkbox is checked.

Then logout from SonarQube UI and login again as shown in the screenshot attached below.

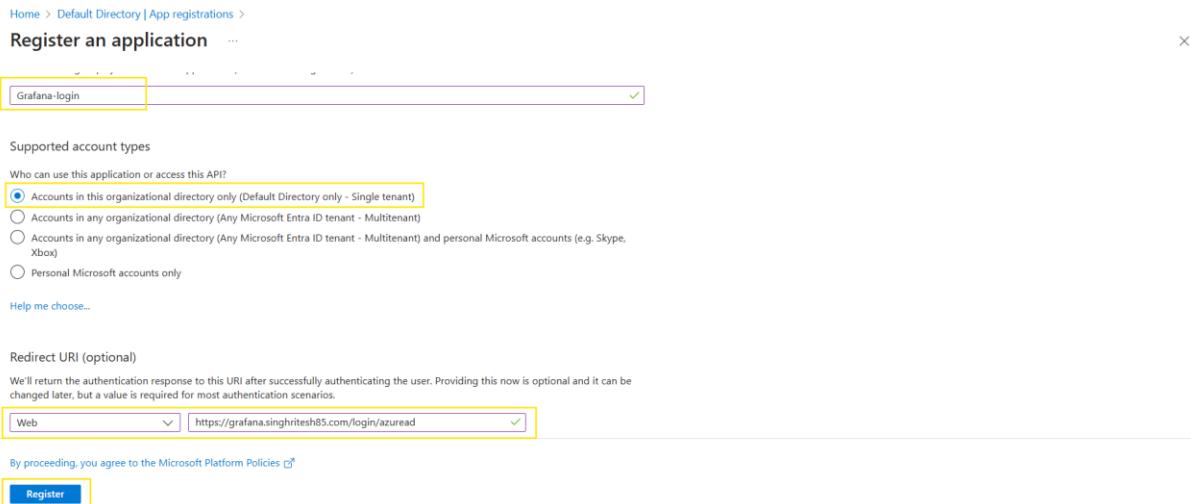


After the user will login once, you can change their permission as shown in screenshot attached below.

Group	Administer System	Administer	Execute Analysis	Create
sonar-administrators	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Quality Gates <input checked="" type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input checked="" type="checkbox"/> Projects
sonar-users	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Projects
Anyone DEPRECATED	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input type="checkbox"/> Projects
Administrator admin	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input type="checkbox"/> Projects
Ritesh [REDACTED] ritesh [REDACTED]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Quality Gates <input checked="" type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input checked="" type="checkbox"/> Projects
user1 user1 [REDACTED]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Quality Gates <input checked="" type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input checked="" type="checkbox"/> Projects

Integration of Azure Entra ID with Grafana

To integrate Azure Entra ID with Grafana I created an Azure Entra ID Service Principal (using APP Registration) as shown in the screenshot attached below. Go to **App Registration > New Registration** and configure a new Application as shown in the screenshot attached below.



Created the Secrets for Registered App Grafana-login in Azure Entra ID as shown in the screenshot attached below.

Home > Grafana-login

Grafana-login | Certificates & secrets

💡 Search Got feedback?

Overview Quickstart Integration assistant Diagnose and solve problems Manage Branding & properties Authentication Certificates & secrets Token configuration API permissions Expose an API App roles Owners Roles and administrators Manifest Support + Troubleshooting

Got a second to give us some feedback? →

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) Client secrets (1) Federated credentials (0)

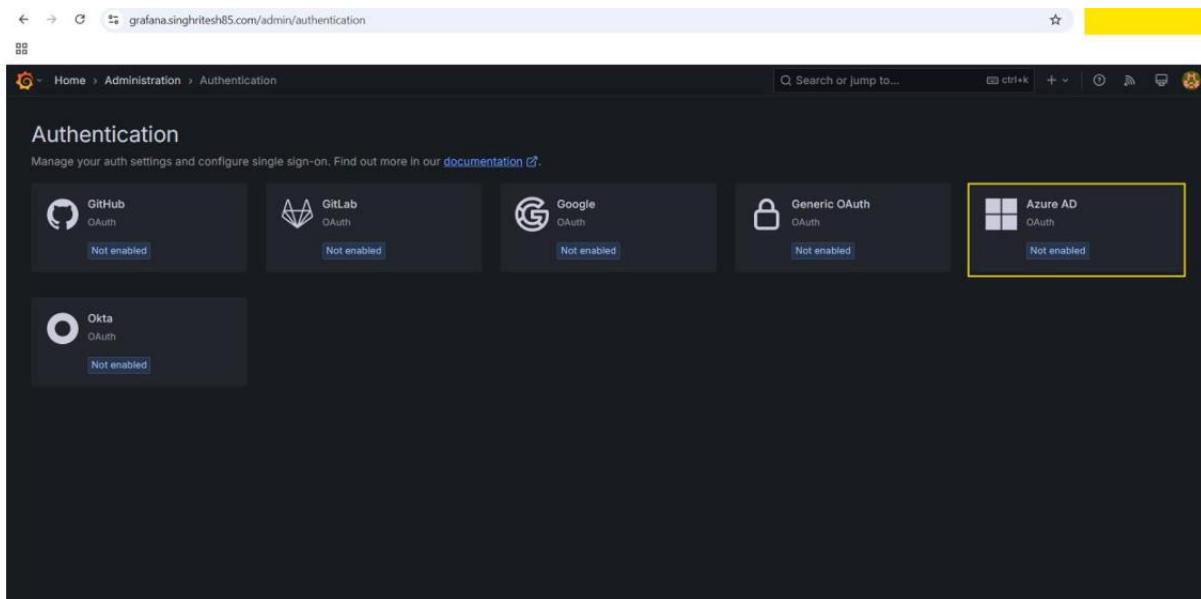
A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value	Secret ID
demo	01/2025	XXXXXXXXXX	XXXXXXXXXX

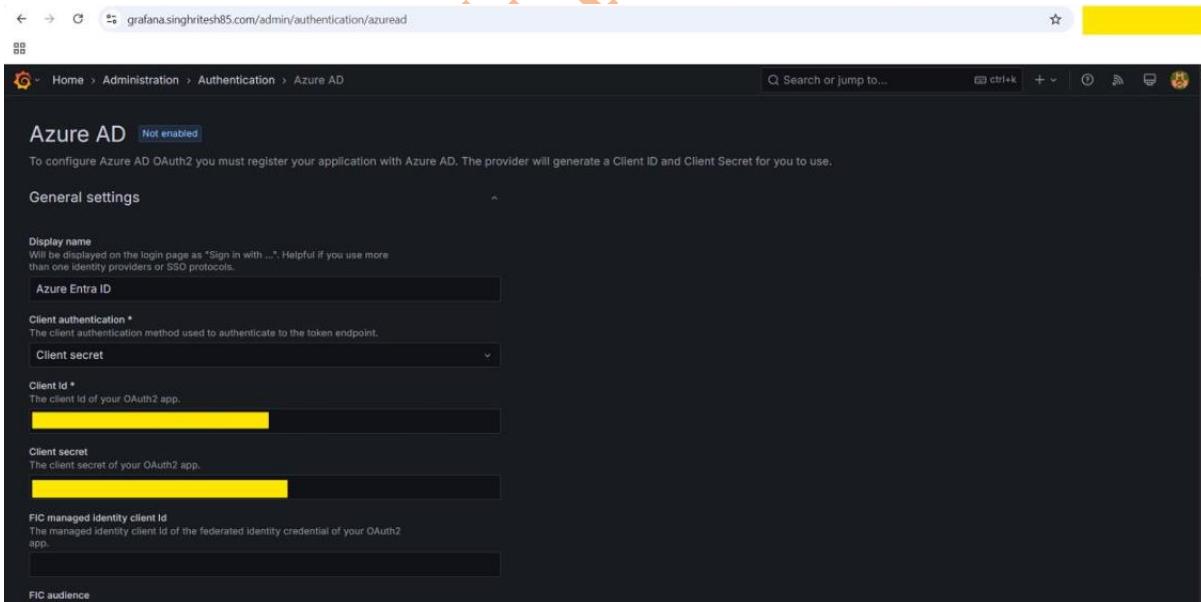
Client Secret

Now, login into the Grafana and update admin password then login into Grafana and Go to Grafana Home > Administration > Authentication and enable the Azure AD OAuth as shown in the screenshot attached below.



Provide the Client ID, Client Secret, Tenant ID and enable Allow Sign up and enable Skip organization role sync as shown in the screenshot attached below.

I enabled the **Skip organization role sync** otherwise Grafana will Sync the Azure Entra ID users with Main Org. Role as **Viewer** and Grafana Administrator cannot change it further but if I enabled **Skip organization role sync** then Grafana Administrator can change the viewer Role and can assign another Role as Editor or Admin. In this demonstration I created two users in Azure Entra ID user1 and user2. User1, I had provided as Administrator Role and user2 as default viewer access in Main Organisation (**Main Organisation always have Organisation ID 1**).



The image contains two screenshots of the Grafana configuration interface, specifically for setting up Azure AD OAuth authentication.

Screenshot 1: Main Configuration Page

- Scopes:** openid, email, profile
- Auth URL:** https://login.microsoftonline.com/[REDACTED]/oauth2/v2.0
- Token URL:** https://login.microsoftonline.com/[REDACTED]/oauth2/v2.0
- Allow sign up:** Enabled (radio button)
- Auto login:** Enabled (radio button)
- Sign out redirect URL:** [REDACTED]

User mapping

Extra security measures

Screenshot 2: Advanced Configuration Page

- Sign out redirect URL:** [REDACTED]

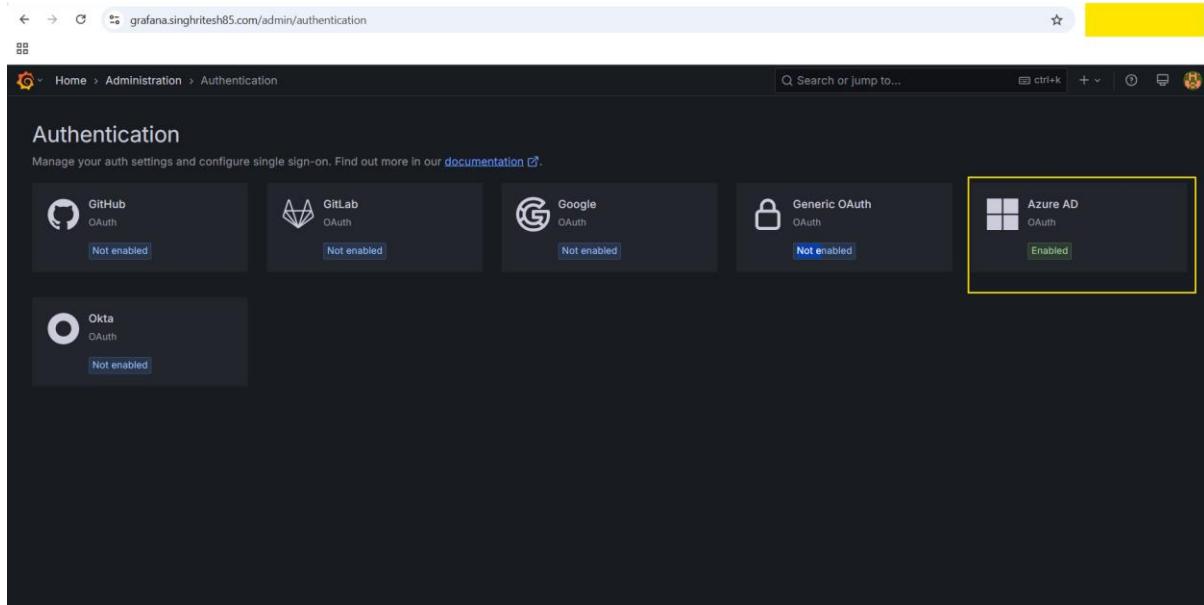
User mapping

- Role attribute strict mode:** Disabled (radio button)
- Organization mapping:** Enter mappings (my-team:1:Viewer,...) and press Enter to add
- Allow assign Grafana admin:** Disabled (radio button)
- Skip organization role sync:** Enabled (radio button)

Extra security measures

Action Buttons: Disable, Save, Discard

Now the Azure AD OAuth is enable as shown in the screenshot attached below.



Now go to the Grafana Server and open the file `/etc/grafana/grafana.ini` and edit `root_url` under the [server] then restart the `grafana-server` service as shown in the screenshot attached below.

```
[root@yellow ~]# vim /etc/grafana/grafana.ini

#####
[server]
# Protocol (http, https, h2, socket)
;protocol = http

# Minimum TLS version allowed. By default, this value is empty. Accepted values are: TLS1.2, TLS1.3. If nothing is set TLS1.2 would be taken
;min_tls_version = ""

# The ip address to bind to, empty will bind to all interfaces
;http_addr =

# The http port to use
;http_port = 3000

# The public facing domain name used to access grafana from a browser
;domain = localhost

# Redirect to correct domain if host header does not match domain
# Prevents DNS rebinding attacks
;enforce_domain = false

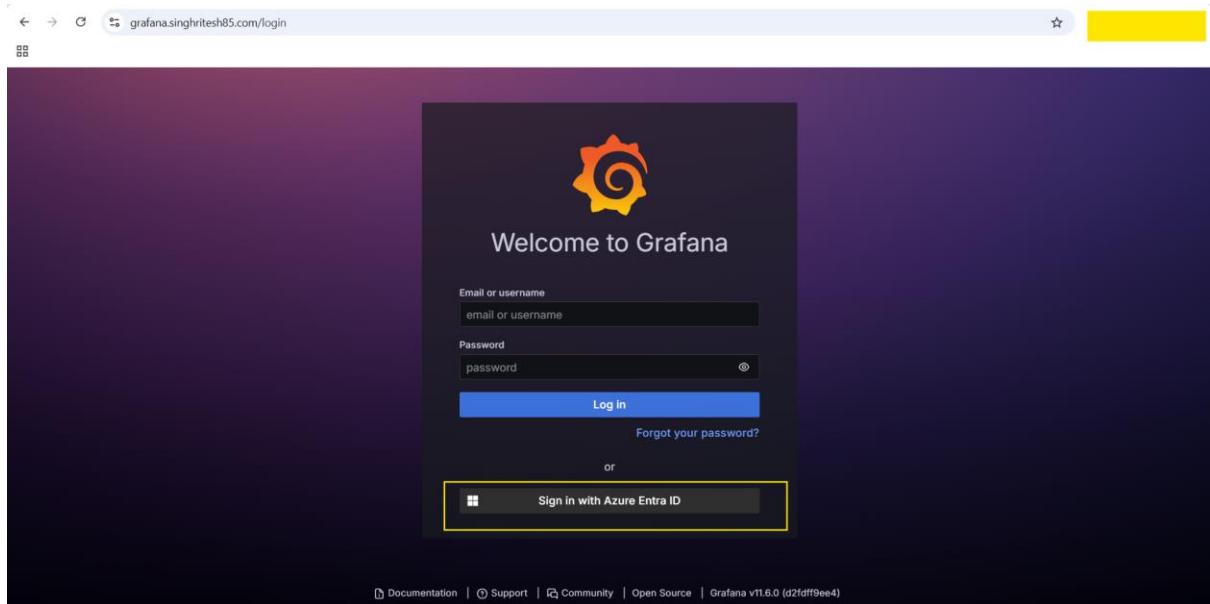
# The full public facing url you use in browser, used for redirects and emails
# If you use reverse proxy and sub path specify full url (with sub path)
root_url = "https://grafana.singhritesh85.com/"      #####(%(protocol)s://%(domain)s:%(http_port)s)

# Serve Grafana from subpath specified in `root_url` setting. By default it is set to `false` for compatibility reasons.
;serve_from_sub_path = false

# Log web requests
;router_logging = false

[root@yellow ~]# systemctl restart grafana-server.service
[root@yellow ~]# systemctl status grafana-server.service
● grafana-server.service - Grafana instance
  Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; vendor preset: disabled)
  Active: active (running) since Sun 2025 yellow UTC; 7s ago
    Docs: http://docs.grafana.org
```

Now, Grafana login dashboard will show the option of sign-in with Azure Entra ID as shown in the screenshot attached below. I logged-in to the Grafana dashboard with the Azure Entra ID user as shown in the screenshot attached below.



For the first time when a user logged-in, they had viewer access which Grafana Administrator can change as per the requirement as shown in the screenshot attached below.

Login	Email	Name	Last active	Origin
admin	admin@localhost		7 minutes	Edit
[REDACTED]	[REDACTED]	Ritesh [REDACTED]	2 minutes	Edit
user1@singhrithesh85.com	user1@singhrithesh85.com	user1	< 1 minute	Edit

user1@singhrithesh85.com
Manage settings for an individual user.

User information

Numerical identifier	3	Synced via AzureAD
Name	user1	Synced via AzureAD
Email	user1@singhrithesh85.com	Synced via AzureAD
Username	user1@singhrithesh85.com	Synced via AzureAD
Password	*****	Synced via AzureAD

Permissions

Grafana Admin	No	Change
---------------	----	---------------

Organizations

Main Org.	Viewer	Change role	Remove from organization
-----------	--------	--------------------	--------------------------

Add user to organization

I will provide Admin access to user1 as shown in the screenshot attached below.

user1@singhrithesh85.com
Manage settings for an individual user.

User information

Numerical identifier	3	Synced via AzureAD
Name	user1	Synced via AzureAD
Email	user1@singhrithesh85.com	Synced via AzureAD
Username	user1@singhrithesh85.com	Synced via AzureAD
Password	*****	Synced via AzureAD

Permissions

Grafana Admin	Yes	Change
---------------	------------	---------------

Organizations

Main Org.	Admin	Change role	Remove from organization
-----------	--------------	--------------------	--------------------------

Add user to organization

After that user1 will refresh Grafana UI and their Access will be reflected on the UI as well.

Integration of Azure Entra ID with Sonatype Nexus3 Repository

I had integrated Azure Entra ID with Sonatype Nexus3 Repository using the Secure LDAP authentication. To achieve this, I had to use the Microsoft Entra Domain Services (formerly known as Azure Active Directory Domain Services) which I had created earlier using the terraform script.

First this you should do here is go to Azure Microsoft Entra Domain Services > Properties and note down the LDAP external IP address and do the entry in Azure DNS Zone to create the Record Set of A Type as shown in the screenshot attached below.

Home > dexter-domainservices

dexter-domainservices | Properties

Microsoft Entra Domain Services

Search

Overview

Activity log

Access control (IAM)

Tags

Resource visualizer

Settings

Properties

Secure LDAP

Synchronization

Custom attributes

Replica sets

Trusts

Health

Notification settings

SKU

Security settings

DNS domain name

singhritesh85.com

Locations

East US

Virtual Networks/Subnets

East US/bankapp-vnet/bankapp-vnet/default

Network security groups

East US/domain-services-nsg

IP addresses

East US/192.168.1.5 192.168.1.4

Secure LDAP

Enabled

Secure LDAP external IP addresses

East US/ [REDACTED]

Synchronization

All

Admin group

AAD DC Administrators

Resource ID

RiteshS

Add record set

singhritesh85.com

Name

.singhritesh85.com

Type

A – IPv4 Address records

Alias record set ⓘ

No

TTL *

1

TTL unit

Hours

IP address

52. [REDACTED].151



0.0.0.0

Add

Cancel

Give feedback

I had updated the `/etc/resolv.conf` file present on **Nexus-Server** and used the Google's Public DNS Server as shown in the screenshot attached below.

```
[root@[REDACTED] ~]# cat /etc/resolv.conf
; generated by /usr/sbin/dhclient-script
search us-east-2.compute.internal
options timeout:2 attempts:5
nameserver 8.8.8.8 #10.10.0.2
```

Login into the Nexus and go to **Administration > Security > Realms** and make the LDAP Realm from Available to Active as shown in the screenshot attached below.

The screenshot shows the Sonatype Nexus Repository OSS 3.68.1-02 interface under the Administration > Security > Realms section. The 'Available' list contains several realms: Conan Bearer Token Realm, Default Role Realm, Docker Bearer Token Realm, npm Bearer Token Realm, NuGet API-Key Realm, and Rut Auth Realm. The 'Active' list contains two realms: Local Authenticating R... and LDAP Realm, with the LDAP Realm currently selected. A transfer operation is shown in progress, with 6 items available in the source list and 2 items transferred to the target list. The 'Save' button is highlighted with a yellow box.

Now go to the **Administration > Security > LDAP** and then **Create Connection**. While creating the connection add the certificate to the truststore as shown in the screenshot attached below.

Sonatype Nexus Repository OSS 3.68.1-02

Administration

- Repository
 - Repositories
 - Blob Stores
 - Proprietary Repositories
 - Content Selectors
 - Cleanup Policies
 - Routing Rules
- Security
 - Privileges
 - Roles
 - Users
 - Anonymous Access
- LDAP**
- Realms

LDAP / LDAPS

Certificate Details

Name: LDAPS

Subject: *.singhritesh85.com

Common name: *.singhritesh85.com

Organization:

Unit:

Use the Nexus Repository truststore: Use certificates stored in the Nexus Repository truststore to connect to external systems. [View certificate](#)

Search base DN:

LDAP server address: The LDAP server usually listens on port 389 (ldap://) or port 636 (ldaps://). **Idaps** **ldaps** **ldaps.singhritesh85.com** **636**

This field is required

Authentication method:

This field is required

Connection rules: Set timeout parameters and max connection attempts to avoid being blacklisted. **Wait:** 30 **seconds before timeout.** **Retry after:** 300 **seconds, max of:** 3 **failed attempts.**

Issued on:

Valid until:

Fingerprint:

This certificate was retrieved over an untrusted connection. Always verify the details before adding it.

Add certificate to truststore **Cancel**

Create LDAP Connection

Name: LDAPS

LDAP server address: The LDAP server usually listens on port 389 (ldap://) or port 636 (ldaps://). **Idaps** **ldaps** **ldaps.singhritesh85.com** **636**

Use the Nexus Repository truststore: Use certificates stored in the Nexus Repository truststore to connect to external systems. [View certificate](#)

Search base DN:

This field is required

Authentication method:

This field is required

Connection rules: Set timeout parameters and max connection attempts to avoid being blacklisted. **Wait:** 30 **seconds before timeout.** **Retry after:** 300 **seconds, max of:** 3 **failed attempts.**

Next **Cancel** **Verify connection**

Sonatype Nexus Repository OSS 3.68.1-02

Administration

- Repository
- Repositories
- Blob Stores
- Proprietary Repositories
- Content Selectors
- Cleanup Policies
- Routing Rules
- Security
- Privileges
- Roles
- Users
- Anonymous Access
- LDAP**
- Realms

LDAP / Create LDAP Connection

LDAP

LDAP server address:
The LDAP server usually listens on port 389 (ldap://) or port 636 (ldaps://)
: 636

Use the Nexus Repository truststore:
 Use certificate stored in the Nexus Repository truststore to connect to external systems [View certificate](#)

Search base DN:
LDAP location to be added to the connection URL (e.g. "dc=example,dc=com")

Authentication method:
Simple Authentication

Username or DN:
This must be a fully qualified username if simple authentication is used

Password:
The password to bind with

Connection rules:
Set timeout parameters and max connection attempts to avoid being blacklisted
Wait: 30 seconds before timeout. Retry after: 300 seconds, max of 3 failed attempts.

Next **Cancel** **Verify connection**

**Connection to LDAP server verified:
ldaps://ldaps.singhritesh85.com:636**

Sonatype Nexus Repository OSS 3.68.1-02

Administration

- Repository
- Repositories
- Blob Stores
- Proprietary Repositories
- Content Selectors
- Cleanup Policies
- Routing Rules
- Security
- Privileges
- Roles
- Users
- Anonymous Access
- LDAP**
- Realms

LDAP / Create LDAP Connection / Choose Users and Groups

Configuration template:
Select a template

User relative DN:
The relative DN where user objects are found (e.g. ou=people). This value will have the Search base DN value appended to form the full User search base DN

User subtree:
 Are users located in structures below the user base DN?

Object class:
LDAP class for user objects (e.g. inetOrgPerson)

User filter:
LDAP search filter to limit user search (e.g. "attribute=foo" or "(|(mail=@example.com)(uid=dom))")

User ID attribute:

Real name attribute:

Email attribute:

Password attribute:

The screenshot shows the 'Create LDAP Connection' dialog in the Sonatype Nexus Repository interface. The dialog contains the following configuration:

- User filter:** LDAPS search filter to limit user search (e.g. "attribute=foo" or "|(mail=*@example.com)(uid=dom*)")
- User ID attribute:** sAMAccountName
- Real name attribute:** cn
- Email attribute:** email
- Password attribute:** If this field is blank the user will be authenticated against a bind with the LDAP server.
- Map LDAP groups as roles:**
- Group type:** Dynamic Groups
- Group member of attribute:** Set this to the attribute used to store the attribute which holds groups DN in the user object. memberOf

At the bottom of the dialog, there are buttons for **Create**, **Cancel**, **Verify user mapping** (which is highlighted with a yellow box), and **Verify login**.

A green success message on the right side of the dialog box states: **LDAP server user mapping verified: ldaps://ldaps.singhritesh85.com:636**.

Now go to Administration > Users and Source: LDAP then you will see your Azure Entra ID Users will be listed here as shown in the screenshot attached below.

User ID	Realm	First name	Last name	Email	Status
[REDACTED]	LDAP	Ritesh	[REDACTED]	[REDACTED]	active >
user1	LDAP	user1			active >

Then I logged-in as user1 and the Nexus Administrator will provide the sufficient privileges to user1. In this project I provide user1 as Administrator privilege as shown in the screenshot attached below.

user1

Granted

- nx-admin

External roles:
AAD DC Administrators

Save **Discard**

Finally, I logged-in as user1 as shown in the screenshot attached below and found that user got the access as provided above.

The image contains two screenshots of the Sonatype Nexus Repository interface.

Top Screenshot: Shows the 'Welcome' screen of the Sonatype Nexus Repository. A 'Sign In' dialog box is open, prompting for a username ('user1') and password ('*****'). The background shows a 'Notification Center' with a message about upgrading the H2 database, a 'Release Notes' section, a 'Documentation' section, and a 'Community' section. A 'Latest Releases' box indicates a new release of Sonatype Nexus Repository 3.79.0.

Bottom Screenshot: Shows the 'Administration' screen under 'Repository'. The left sidebar lists 'Administration' categories: 'Repository' (Repositories, Blob Stores, Proprietary Repositories, Content Selectors, Cleanup Policies, Routing Rules), 'Security' (Privileges, Roles, Users, Anonymous Access, LDAP, Realms), and 'System' (Search components, Help, Sign in/Sign out). The main content area is titled 'Repository' and shows management options: Blob Stores, Cleanup Policies, Proprietary Repositories, Repositories, and Routing Rules.

For this project in Nexus, I created two repositories named as **maven-release** and **maven-snapshot** as shown in the screenshot attached below.

The screenshots illustrate the process of creating a new Maven repository in Sonatype Nexus Repository OSS 3.68.1-02.

Screenshot 1: Existing Repositories List

The left sidebar shows the navigation menu under "Administration". The "Repositories" section is selected, highlighted with a green bar. The main content area displays a list of existing repositories, including "apt (hosted)", "apt (proxy)", "bower (group)", "bower (hosted)", "bower (proxy)", "cocoapods (proxy)", "conan (proxy)", "conda (proxy)", "docker (group)", "docker (hosted)", "docker (proxy)", "gitlfs (hosted)", "go (group)", "go (proxy)", "helm (hosted)", "helm (proxy)", "maven2 (group)", "maven2 (hosted)" (which is highlighted with a yellow box), "maven2 (proxy)", and "npm (group)".

Screenshot 2: Create Repository Form - Basic Configuration

The "Name:" field is set to "maven-release" (highlighted with a yellow box). The "Online:" checkbox is checked. Under "Maven 2", the "Version policy:" dropdown is set to "Release" (highlighted with a yellow box) and the "Layout policy:" dropdown is set to "Strict" (highlighted with a yellow box). The "Content Disposition:" dropdown is set to "Inline".

Screenshot 3: Create Repository Form - Advanced Configuration

The "Deployment policy:" dropdown is set to "Disable redeploy" (highlighted with a yellow box). Under "Proprietary Components:", there is a checkbox for "Components in this repository count as proprietary for namespace conflict attacks (requires Sonatype Nexus Firewall)".

Common UI Elements:

- Left Sidebar:** Shows the navigation menu with sections like Administration, Repository, Security, and others.
- Top Bar:** Includes the Sonatype logo, version information (OSS 3.68.1-02), search bar, and user sign-in/out links.
- Bottom Bar:** Shows the URL (nexus.singhrithesh85.com/#admin/repository/repositories), browser navigation icons, and a yellow status bar.

The screenshots illustrate the process of creating a new repository in Sonatype Nexus Repository OSS 3.68.1-02.

Screenshot 1: Repositories List

The left sidebar shows the navigation menu under "Administration". The "Repositories" section is selected. The main area displays a table of existing repositories:

Name	Type	Format	Blob Store	Status	URL	Health check	Firewall Re...
maven-central	proxy	maven2	default	Online - Ready to Conn...	<input type="button" value="copy"/>	Analyze	<input type="button" value="..."/>
maven-public	group	maven2	default	Online	<input type="button" value="copy"/>	<input type="button" value="..."/>	<input type="button" value="..."/>
maven-release	hosted	maven2	default	Online	<input type="button" value="copy"/>	<input type="button" value="..."/>	<input type="button" value="..."/>
maven-releases	hosted	maven2	default	Online	<input type="button" value="copy"/>	<input type="button" value="..."/>	<input type="button" value="..."/>
maven-snapshots	hosted	maven2	default	Online	<input type="button" value="copy"/>	<input type="button" value="..."/>	<input type="button" value="..."/>
nuget-group	group	nuget	default	Online	<input type="button" value="copy"/>	<input type="button" value="..."/>	<input type="button" value="..."/>
nuget-hosted	hosted	nuget	default	Online	<input type="button" value="copy"/>	<input type="button" value="..."/>	<input type="button" value="..."/>
nuget.org-proxy	proxy	nuget	default	Online - Remote Availa...	<input type="button" value="copy"/>	Analyze	<input type="button" value="..."/>

Screenshot 2: Recipe Selection

The left sidebar shows the navigation menu under "Administration". The "Repositories" section is selected. The main area displays a list of available recipes:

- apt (hosted)
- apt (proxy)
- bower (group)
- bower (hosted)
- bower (proxy)
- cocoapods (proxy)
- conan (proxy)
- conda (proxy)
- docker (group)
- docker (hosted)
- docker (proxy)
- gitlfs (hosted)
- go (group)
- go (proxy)
- helm (hosted)
- helm (proxy)
- maven2 (group)
- maven2 (hosted)** (highlighted)
- maven2 (proxy)
- npm (group)

Screenshot 3: Create Repository Form

The left sidebar shows the navigation menu under "Administration". The "Repositories" section is selected. The main area displays a form for creating a new repository:

Name: maven-snapshot

Online: if checked, the repository accepts incoming requests

Maven 2

Version policy: Snapshot

Layout policy: Strict

Content Disposition: Inline

Storage

Blob store: default

Strict Content Type Validation: Validate that all content uploaded to this repository is of a MIME type appropriate for the repository format

Sonatype Nexus Repository OSS 3.68.1-02

Administration

- Repository
 - Repositories
 - Blob Stores
 - Proprietary Repositories
 - Content Selectors
 - Cleanup Policies
 - Routing Rules
- Security
 - Privileges
 - Roles
 - Users
 - Anonymous Access
 - LDAP
 - Realms

Repositories / Select Recipe / Create Repository: maven2 (hosted)

Deployment policy:
Controls if deployments of and updates to artifacts are allowed
 Disable redeploy

Proprietary Components:
 Components in this repository count as proprietary for namespace conflict attacks (requires Sonatype Nexus Firewall)

Cleanup

Cleanup Policies:
Components that match any of the Applied policies will be deleted

Available	Applied
Filter	

Create repository Cancel

Finally, the two repositories in Nexus named as **maven-release** and **maven-snapshot** had been created as shown in the screenshot attached below.

Sonatype Nexus Repository OSS 3.68.1-02

Administration

- Repository
 - Repositories
 - Blob Stores
 - Proprietary Repositories
 - Content Selectors
 - Cleanup Policies
 - Routing Rules
- Security
 - Privileges
 - Roles
 - Users
 - Anonymous Access
 - LDAP
 - Realms

Repositories Manage repositories

Create repository

Name ↑	Type	Format	Blob Store	Status	URL	Health check	Firewall Re...
maven-central	proxy	maven2	default	Online - Ready to Conn...	<input checked="" type="button"/> copy	<input type="button"/> Analyze	<input type="button"/>
maven-public	group	maven2	default	Online	<input checked="" type="button"/> copy	<input type="button"/>	<input type="button"/>
maven-release	hosted	maven2	default	Online	<input checked="" type="button"/> copy	<input type="button"/>	<input type="button"/>
maven-releases	hosted	maven2	default	Online	<input checked="" type="button"/> copy	<input type="button"/>	<input type="button"/>
maven-snapshot	hosted	maven2	default	Online	<input checked="" type="button"/> copy	<input type="button"/>	<input type="button"/>
maven-snapshots	hosted	maven2	default	Online	<input checked="" type="button"/> copy	<input type="button"/>	<input type="button"/>
nuget-group	group	nuget	default	Online	<input checked="" type="button"/> copy	<input type="button"/>	<input type="button"/>
nuget-hosted	hosted	nuget	default	Online	<input checked="" type="button"/> copy	<input type="button"/>	<input type="button"/>
nuget.org-proxy	proxy	nuget	default	Online - Remote Availa...	<input checked="" type="button"/> copy	<input type="button"/> Analyze	<input type="button"/>

Installation of Nginx Ingress Controller in EKS Cluster

To install Nginx ingress controller in EKS Cluster use the Command as provided below and use your own SSL Certificate ARN instead of arn:aws:acm:us-east-2:02XXXXXXXXXX6:certificate/XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX.

```
kubectl create ns ingress-nginx
```

```
helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
```

helm repo update

```
helm install ingress-nginx ingress-nginx/ingress-nginx -n ingress-nginx --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-ssl-cert"=arn:aws:acm:us-east-2:02XXXXXXXXXX6:certificate/XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-connection-idle-timeout"="60" --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-cross-zone-load-balancing-enabled"="true" --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-type"="elb" --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-backend-protocol"="http" --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-ssl-ports"="https" --set controller.service.targetPorts.https=http --set-string controller.config.use-forwarded-headers="true"
```

```
[root@yellow ~]# kubectl create ns ingress-nginx
namespace/ingress-nginx created

[root@yellow ~]# helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
"ingress-nginx" has been added to your repositories
[root@yellow ~]# helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "ingress-nginx" chart repository
Update Complete. Happy Helming!
```



```
[root@yellow ~]# helm install ingress-nginx ingress-nginx/ingress-nginx -n ingress-nginx --set controller.service.annotations."service\\.beta\\.kubernetes\\.io/aws-load-balancer-ssl-cert"=arn:aws:acm:us-east-2:02:123456789012:certificate/12345678901234567890123456789012 --set controller.service.annotations."service\\.beta\\.kubernetes\\.io/aws-load-balancer-connection-idle-timeout"=60 --set controller.service.annotations."service\\.beta\\.kubernetes\\.io/aws-load-balancer-cross-zone-load-balancing-enabled"=true --set controller.service.annotations."service\\.beta\\.kubernetes\\.io/aws-load-balancer-backend-protocol"=http --set controller.service.annotations."service\\.beta\\.kubernetes\\.io/aws-load-balancer-ssl-ports"=https --set controller.service.targetPorts.https=http --set-string controller.config.use-forwarded-headers=true
```



```
[root@yellow ~]# kubectl get pods -n ingress-nginx
NAME                      READY   STATUS    RESTARTS   AGE
ingress-nginx-controller   1/1     Running   0          98s
[root@yellow ~]# kubectl get svc -n ingress-nginx
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)
               AGE
ingress-nginx-controller   LoadBalancer  172.17.0.49   a.us-east-2.elb.amazonaws.com  80:30009/TCP,443
ingress-nginx-controller-admission ClusterIP  172.17.0.196  <none>        443/TCP
```

Installation of ArgoCD in EKS Cluster

In this project for EKS Cluster deployment I used ArgoCD and the ArgoCD CLI, I installed the ArgoCD as shown in the screenshot attached below.

```
kubectl create namespace argocd
```

```
kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-  
cd/stable/manifests/install.yaml
```

```
[root@yellow ~]# kubectl get nodes  
NAME           STATUS   ROLES      AGE   VERSION  
ip-10-10-1-1.us-east-2.compute.internal   Ready    <none>    10m    v1.30.4-eks-a  
ip-10-10-1-2.us-east-2.compute.internal   Ready    <none>    10m    v1.30.4-eks-a  
[root@yellow ~]# kubectl create namespace argocd  
namespace/argocd created  
[root@yellow ~]# kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install.yaml
```

The ingress rule for ArgoCD is as shown in the screenshot attached below.

```

apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: minimal-ingress
  namespace: argocd
  annotations:
    kubernetes.io/ingress.class: nginx
    nginx.ingress.kubernetes.io/backend-protocol: "HTTPS" #### You can use this option for this
    particular case for ArgoCD but not for all
    # nginx.ingress.kubernetes.io/ssl-redirect: "false"
spec:
  ingressClassName: nginx
  rules:
    - host: argocd.singhrithesh85.com
      http:
        paths:
          - path: /
            pathType: Prefix
        backend:
          service:
            name: argocd-server #### Provide your service Name
            port:
              number: 80 ###### Provide your service port for this particular example you can also choose
443

```

```
[root@yellow ~]# cat argocd-ingress-rule.yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: minimal-ingress
  namespace: argocd
  annotations:
    kubernetes.io/ingress.class: nginx
    nginx.ingress.kubernetes.io/backend-protocol: "HTTPS"    ### You can use this option for this particular case for ArgoCD but not for all
    #   nginx.ingress.kubernetes.io/ssl-redirect: "false"
spec:
  ingressClassName: nginx
  rules:
  - host: argocd.singhritesh85.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: argocd-server    ### Provide your service Name
            port:
              number: 80    ### Provide your service port for this particular example you can also choose 443
[root@yellow ~]# kubectl apply -f argocd-ingress-rule.yaml
ingress.networking.k8s.io/minimal-ingress created
[root@yellow ~]# kubectl get ing -A
NAMESPACE     NAME           CLASS      HOSTS          ADDRESS
argocd        minimal-ingress   nginx     argocd.singhritesh85.com  ayellow.us-east-2.elb.amazonaws.com  80      8s
```

The entry for DNS Name corresponding to HOST **argocd.singhritesh85.com** in Azure DNS Zone to create the record set is as shown in the screenshot attached below.

NAME	Type	TTL
argocd	CNAME	1

I generated the password for ArgoCD is as shown in the screenshot attached below.

kubectl -n argocd get secret argocd-initial-admin-secret -o jsonpath='{.data.password}' | base64 -d

```
[root@yellow ~]# kubectl -n argocd get secret argocd-initial-admin-secret -o jsonpath='{.data.password}' | base64 -d
e[yellow]#
```

After login into the ArgoCD for the first time I updated the ArgoCD Password as shown in the screenshot attached below.

The image displays two screenshots of the Argo CD web interface.

Top Screenshot (Applications Page):

- URL: argocd.singhritesh85.com/applications
- Header: Argo v2.14.6+fe2a6e9
- Left Sidebar:
 - Applications
 - Settings
 - User Info** (highlighted)
 - Documentation
- Top Bar Buttons: + NEW APP, SYNC APPS, REFRESH APPS, Search applications...
- Section: APPLICATIONS SUMMARY
- Middle Content: A large circular icon with three stacked cards, indicating "No applications available to you just yet". Below it is the text "Create new application to start managing resources in your cluster" and a "CREATE APPLICATION" button.
- Right Sidebar: Log out

Bottom Screenshot (User Info Page):

- URL: argocd.singhritesh85.com/user-info
- Header: Argo v2.14.6+fe2a6e9
- Left Sidebar:
 - Applications
 - Settings
 - User Info** (highlighted)
 - Documentation
- Top Bar Buttons: UPDATE PASSWORD (highlighted), USER INFO, Log out
- Section: User Info
- Middle Content: A box containing "Username: admin" and "Issuer: argocd".

The image contains two screenshots of the Argo UI. The top screenshot shows a modal titled 'Update account password' with fields for 'Current Password', 'New Password', and 'Confirm New Password'. The bottom screenshot shows the 'User Info' page with a success message: 'Your password has been successfully updated.'

Installation of ArgoCD CLI using the commands as shown in the screenshot attached below.

```
curl -sSL -o argocd-linux-amd64 https://github.com/argoproj/argo-cd/releases/latest/download/argocd-linux-amd64
```

```
sudo install -m 555 argocd-linux-amd64 /usr/local/bin/argocd
```

```
rm argocd-linux-amd64
```

```
[root@REDACTED ~]# curl -sSL -o argocd-linux-amd64 https://github.com/argoproj/argo-cd/releases/latest/download/argocd-linux-amd64
[root@REDACTED ~]# sudo install -m 555 argocd-linux-amd64 /usr/local/bin/argocd
[root@REDACTED ~]# rm argocd-linux-amd64
rm: remove regular file 'argocd-linux-amd64'? yes
```

Configuration of Email to Send notification on Group Email-ID using Jenkins and Grafana

To configure Gmail to send notification to group Email ID I should have App Password for my Gmail account as shown in the screenshot attached below.

Go to your **Gmail Account > Manage your Google Account > Security** and then search for **app password** and click on **App Passwords** as shown in the screenshot attached below.

The screenshot shows the Google Account interface under the 'Security' tab. A search bar at the top right contains the text 'app password'. Below it, a sidebar lists various security-related options like 'Password Manager', 'Web & App Activity', and 'Sign in with app passwords'. The 'App passwords' option is highlighted with a yellow box. To the right, there's a 'Recent security activity' section stating 'No security activity or alerts in the last 28 days' and a 'How you sign in to Google' section with a note about keeping information up-to-date.

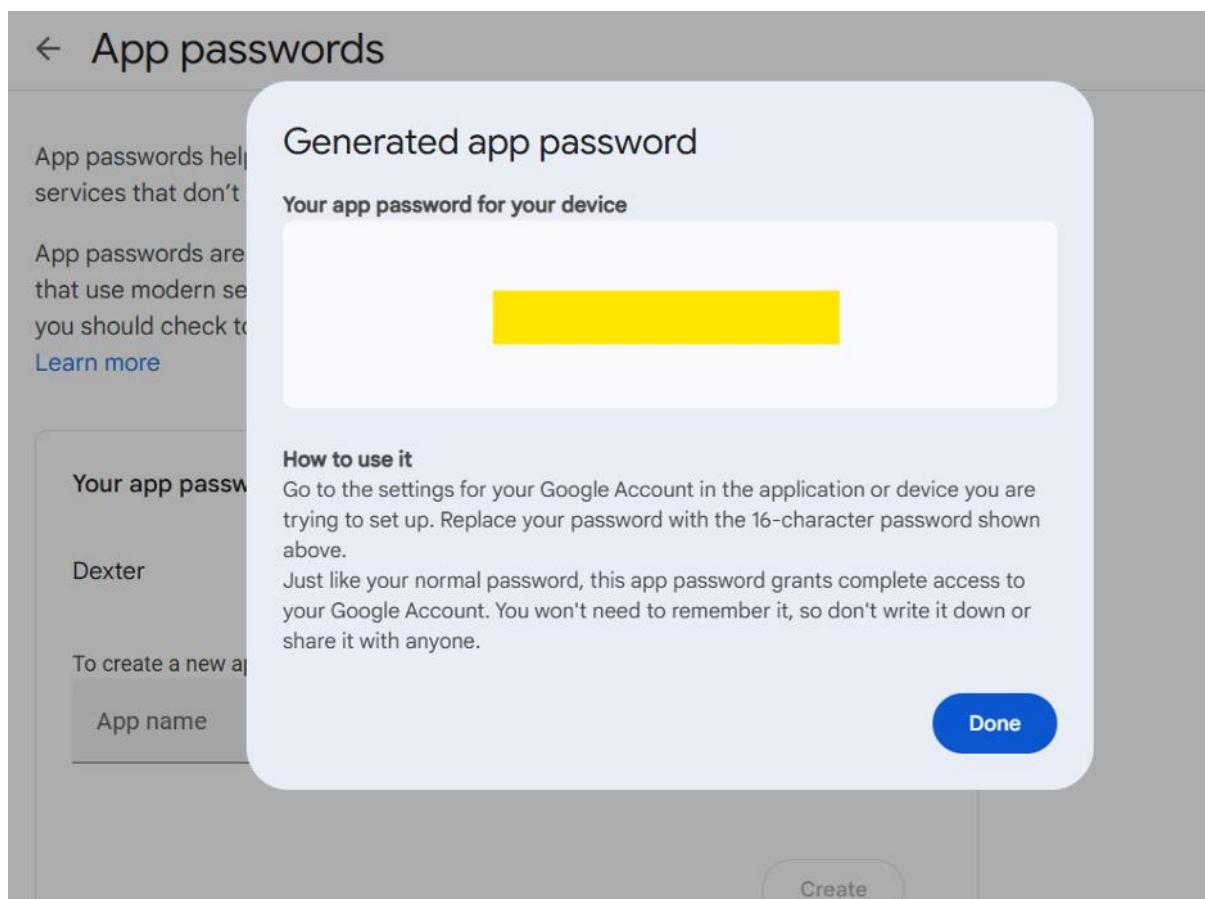
← App passwords

App passwords help you sign into your Google Account on older apps and services that don't support modern security standards.

App passwords are less secure than using up-to-date apps and services that use modern security standards. Before you create an app password, you should check to see if your app needs this in order to sign in.

[Learn more](#)

The screenshot shows a form for creating an app password. It asks for an 'App name' which is currently 'Dexter'. A large yellow box highlights the 'Create' button at the bottom right of the form area.

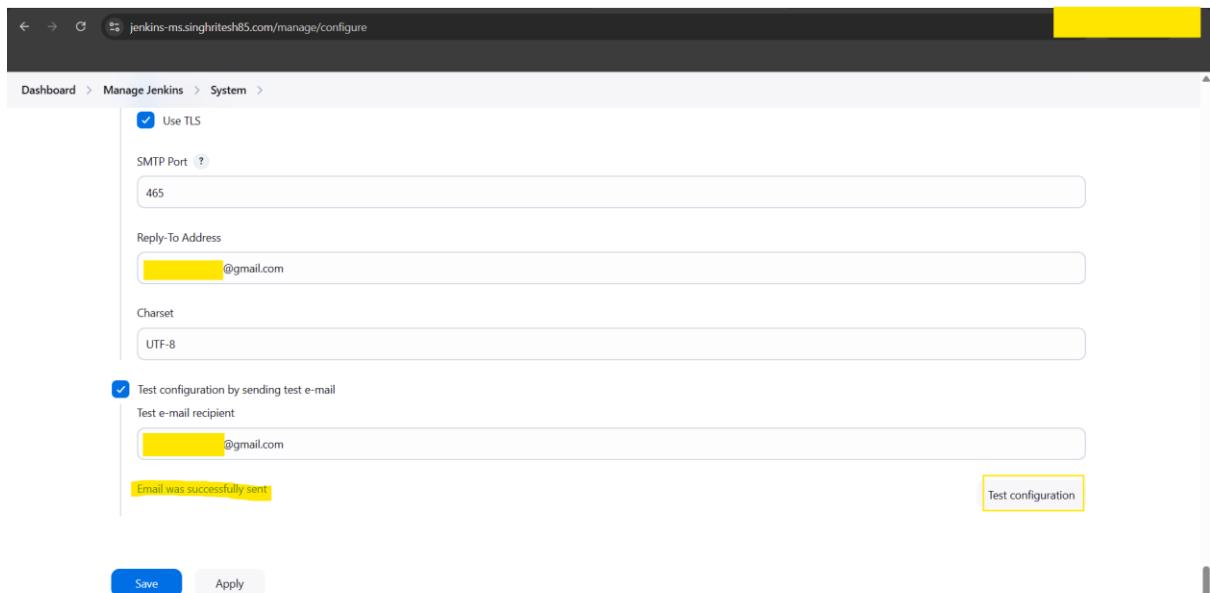


I had deleted this App Password after completion of this project, this App password does not exist anymore. You can use your own Gmail Account's App Passwords.

It is Possible to Send notification to group Email ID using Jenkins and Grafana through Amazon Simple Email Services (Amazon SES). To know more you can refer the project present in my GitHub Repo <https://github.com/singhritesh85/DevOps-Project-2tier-WebApp-Deployment.git>. However, for this project I used Gmail App Passwords as explained above.

Now, configuration of email to Send notification on Group Email-ID using Jenkins and Grafana is as discussed below.

The screenshot shows the Jenkins 'Manage Jenkins > System' configuration page. Under the 'E-mail Notification' section, the 'SMTP server' is set to 'smtp.gmail.com'. The 'Default user e-mail suffix' is '@gmail.com'. In the 'Advanced' section, the 'Use SMTP Authentication' checkbox is checked, and the 'User Name' is filled with a redacted email address ending in '@gmail.com'. A warning message states: 'For security when using authentication it is recommended to enable either TLS or SSL'. The 'Password' field is also redacted. Both 'Use SSL' and 'Use TLS' checkboxes are checked. At the bottom are 'Save' and 'Apply' buttons.



To configure Alerts in Grafana, first I created contact points with the Email ID and changed smtp settings in the configuration file **/etc/grafana/grafana.ini** of Grafana as shown in the screenshot attached below.

```
#####
[smtp]
enabled = true
host = smtp.gmail.com:587
user = redacted@gmail.com
# If the password contains # or ; you have to wrap it with triple quotes. Ex """#password;"""
password = redacted
;cert_file =
;key_file =
skip_verify = true
from_address = redacted@gmail.com
from_name = Grafana Alert for BankApp
# EHLO identity in SMTP dialog (defaults to instance_name)
;ehlo_identity = dashboard.example.com
# SMTP startTLS policy (defaults to 'OpportunisticStartTLS')
;startTLS_policy = NoStartTLS
# Enable trace propagation in e-mail headers, using the 'traceparent', 'tracestate' and (optionally) 'baggage' fields (defaults to false)
;enable_tracing = false
```

Then restart the **grafana-server** service as shown in the screenshot attached below.

```
[root@redacted ~]# systemctl restart grafana-server.service
[root@redacted ~]# systemctl status grafana-server.service
● grafana-server.service - Grafana instance
  Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; vendor preset: disabled)
  Active: active (running) since [REDACTED] 2025-01-12 UTC; 9s ago
    Docs: http://docs.grafana.org
```

Creation of Alerts in Grafana I will discuss later here I will discuss first integration of Jenkins with SonarQube and Sonatype Nexus3 Repository then will create the Jenkins Job and deploy the Bank Application. To do so I used the Jenkinsfile as shown in the screenshot attached below. This Jenkinsfile is also available in my GitHub Repo <https://github.com/singhritesh85/DevOps-Project-BankApplication-BlueGreen-Deployment-MultiCloud.git>.

Integration of Jenkins with SonarQube

To Integrate Jenkins with SonarQube I need a Security Token in SonarQube which I created as shown in the screenshot attached below.

Screenshots of the SonarQube 'Tokens' page showing the generation of a security token.

Top Screenshot: Generating a Global Analysis Token for 'SonarQube' that never expires. The 'Generate' button is highlighted.

Name	Type	Expires in
SonarQube	Global Analysis Token	No expiration

Bottom Screenshot: After generating the token, a success message appears: 'New token "SonarQube" has been created. Make sure you copy it now, you won't be able to see it again!'. The 'Copy' button is highlighted.

Name	Type	Project	Last use	Created	Expiration
SonarQube	Global		Never	2025	Revoke

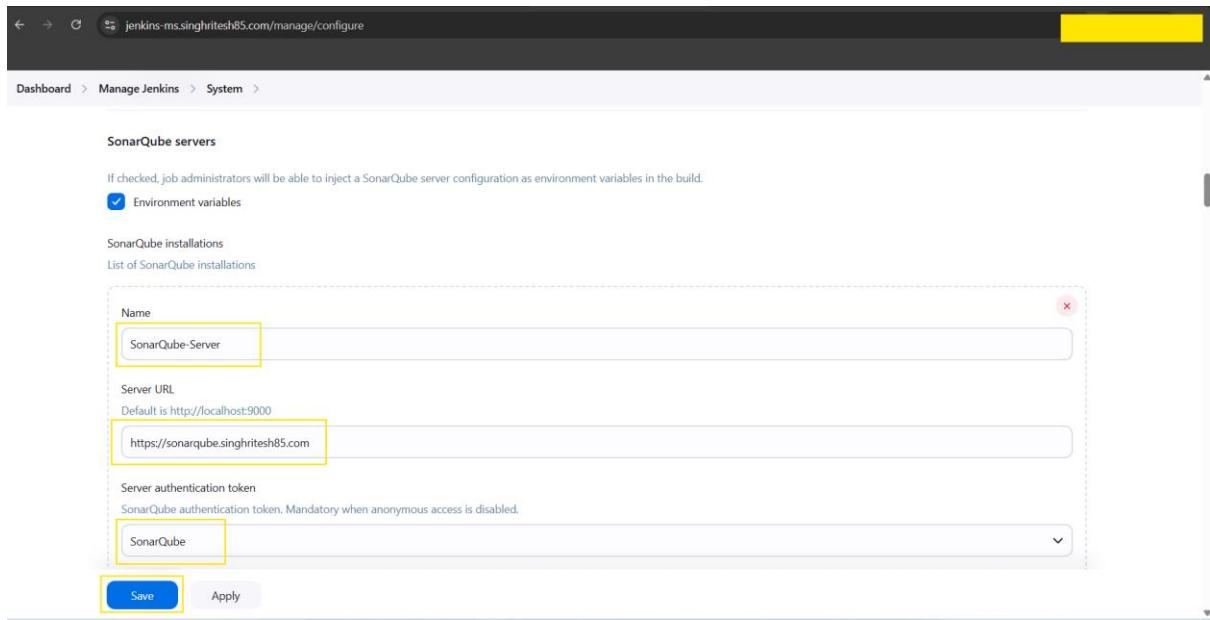
Now go to Jenkins **Manage Jenkins > System > Plugins > Available Plugins** and install the **SonarQube Scanner** Plugin as shown in the screenshot attached below.

The screenshot shows the Jenkins Plugin Manager interface. In the search bar at the top right, 'sonarQube Scanner' is typed. Below the search bar, there is a button labeled 'Install' with a yellow border. On the left sidebar, under the 'Available plugins' section, the 'SonarQube Scanner' plugin is listed with a checked checkbox. The main panel displays the plugin's details: Name: SonarQube Scanner 2.18, Category: External Site/Tool Integrations, Subcategory: Build Reports, Description: This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality, Released: 2 mo 8 days ago, and Version: 2.18. At the bottom right of the main panel, it says REST API Jenkins 2.492.3.

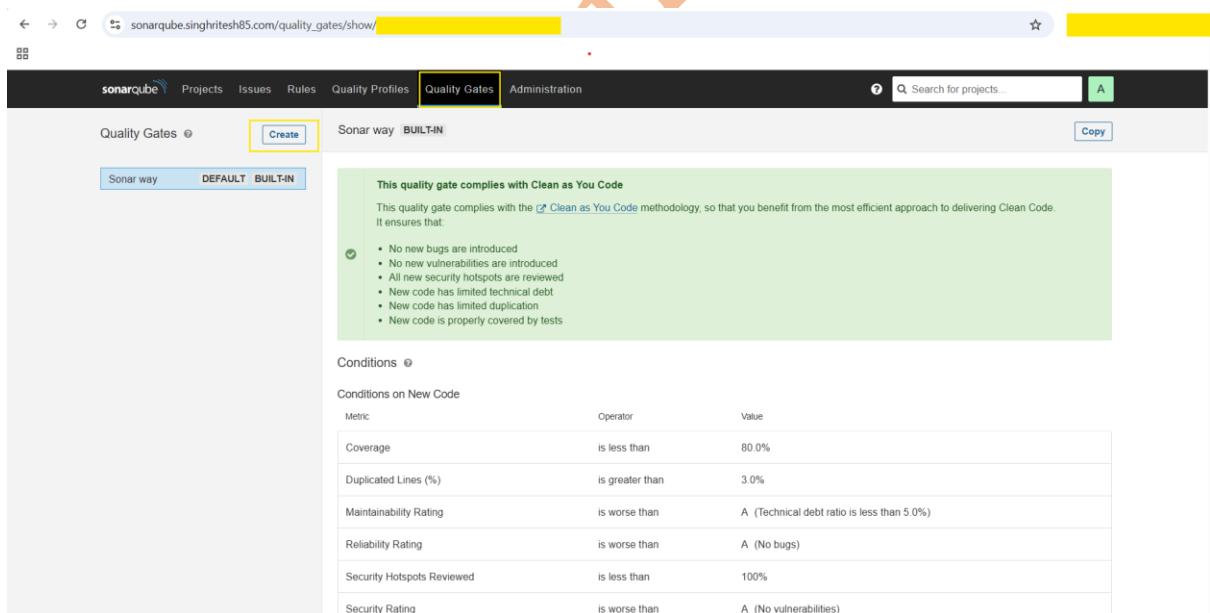
Do not restart the Jenkins after installation of **SonarQube-Scanner** Plugin. Then create a Jenkins Credential of kind Secret text with the SonarQube Secret Token which I created earlier as shown in the screenshot attached below.

The screenshot shows the Jenkins 'New credentials' creation page. Under the 'Kind' dropdown, 'Secret text' is selected. In the 'Scope' dropdown, 'Global (Jenkins, nodes, items, all child items, etc)' is chosen. The 'Secret' field contains a redacted value. The 'ID' field is set to 'SonarQube'. The 'Description' field is also set to 'SonarQube'. At the bottom left, there is a blue 'Create' button with a yellow border.

Now go to Jenkins **Manage Jenkins > System** and search for SonarQube and do the configuration as shown in the screenshot attached below.



In SonarQube the **Quality Gate** is the predefined condition and threshold which decides the minimum Quality Standards of the code. During Jenkins Pipeline I had tested whether the Code passed the Quality Gate or not. If it does not pass the Quality Gate then Jenkins Job will fail there itself and do not move for further stages. Below screenshot shows how I created the Quality Gate in SonarQube for this Project.



https://sonarqube.singhritesh85.com/quality_gates/show/

The screenshot shows the SonarQube interface for creating a new Quality Gate. The 'Create' button is highlighted. The 'Name' field contains 'mederma'. The 'Save' button is highlighted.

https://sonarqube.singhritesh85.com/quality_gates/show/

The screenshot shows the SonarQube interface displaying the newly created Quality Gate 'mederma'. The 'Conditions' section is expanded, showing conditions on New Code. The 'Unlock editing' button is highlighted.

https://sonarqube.singhritesh85.com/quality_gates/show/

The screenshot shows the SonarQube interface with the Quality Gate 'mederma' in edit mode. Edit icons (pencil and trash) are visible next to each condition in the 'Conditions on New Code' table.

Projects With Without All Search

Permissions
Users with the global "Administer Quality Gates" permission and those listed below can manage this Quality Gate.

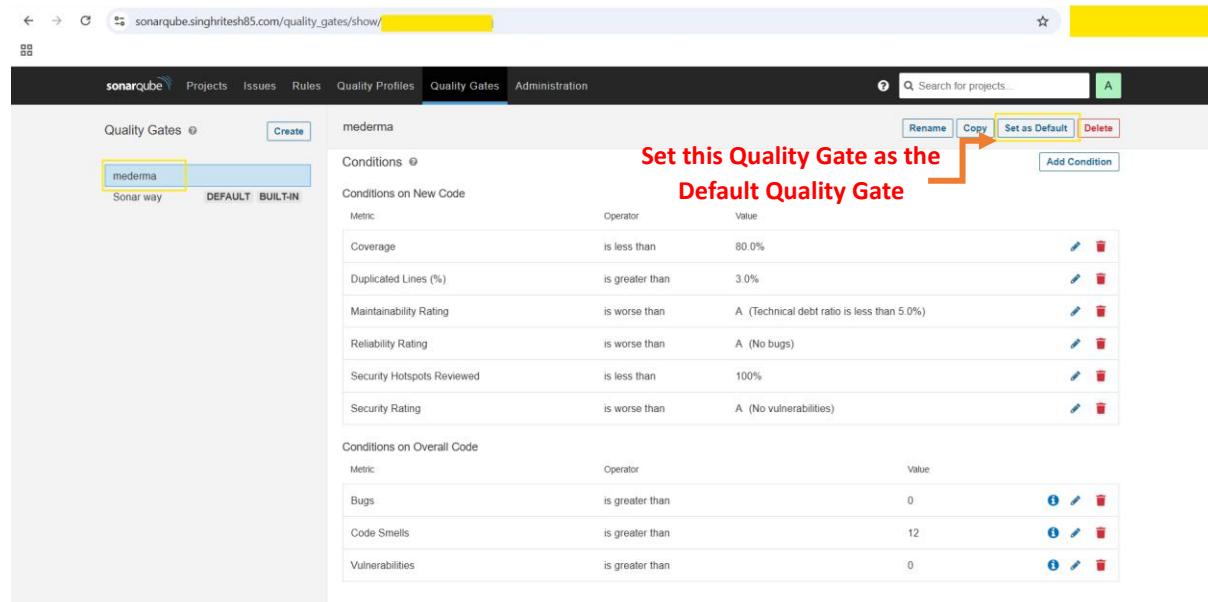
[Grant permissions to a user or a group](#)

The screenshots show the SonarQube Quality Gates configuration interface. In each screenshot, a modal dialog is open for "Add Condition".

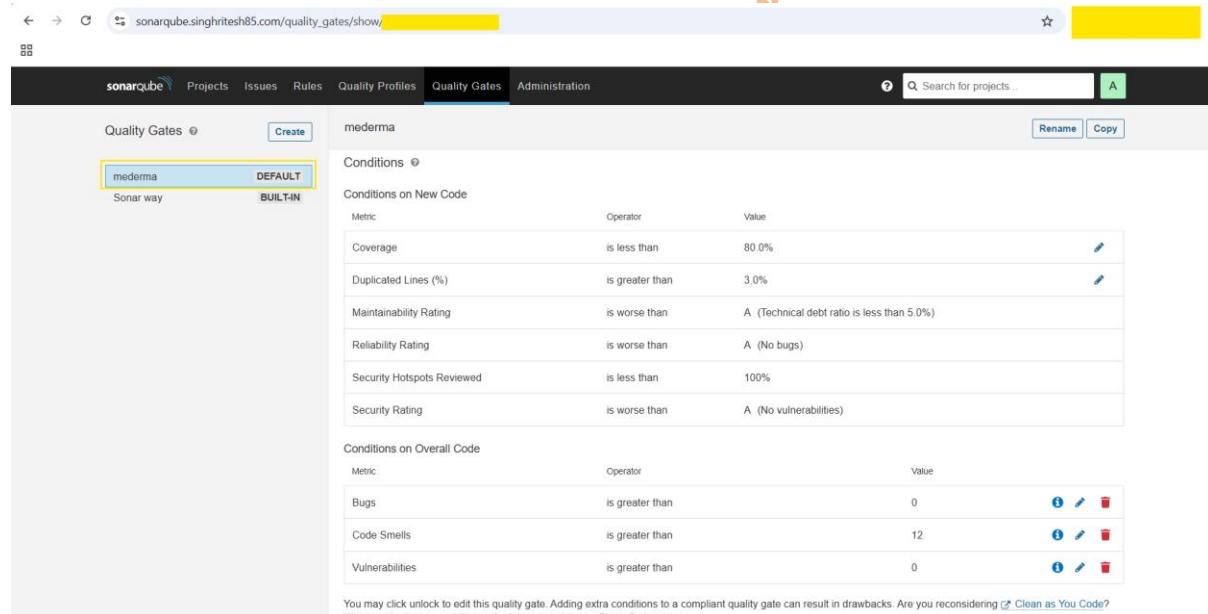
- Screenshot 1 (Top):** The "Quality Gate fails when" dropdown is set to "Bugs". The "Operator" is "is greater than" and the "Value" is "0".
- Screenshot 2 (Middle):** The "Quality Gate fails when" dropdown is set to "Code Smells". The "Operator" is "is greater than" and the "Value" is "12".
- Screenshot 3 (Bottom):** The "Quality Gate fails when" dropdown is set to "Vulnerabilities". The "Operator" is "is greater than" and the "Value" is "0".

Below the modal, the main Quality Gates page shows a table of rules. The first rule is "mederma" (Sonar way, DEFAULT, BUILT-IN) with the condition "Bugs > 0". The second rule is "debt ratio is less than 5.0%" (Sonar way, DEFAULT, BUILT-IN) with the condition "debt ratio < 5.0%". The third rule is "A (No vulnerabilities)" (Sonar way, DEFAULT, BUILT-IN) with the condition "Security Rating is worse than A".

Finally, the predefine condition/threshold for Quality Gate of the SonarQube is as shown in the screenshot attached below.



The screenshot shows the SonarQube interface for managing Quality Gates. In the top navigation bar, the 'Quality Gates' tab is selected. On the left, there's a sidebar with 'Quality Gates' and a 'Create' button. Below it, a list shows 'mederma' (Sonar way) as the current quality gate, which is highlighted with a yellow box. To the right of this list are buttons for 'Rename', 'Copy', 'Set as Default' (which has a red arrow pointing to it), and 'Delete'. A search bar at the top right says 'Search for projects...' with a magnifying glass icon. Below the sidebar, the main content area is titled 'Conditions' and contains two sections: 'Conditions on New Code' and 'Conditions on Overall Code', each with several rows of metrics, operators, and values.



This screenshot is identical to the one above, showing the same SonarQube interface for managing Quality Gates. The 'mederma' quality gate is again highlighted with a yellow box. The 'Set as Default' button is also highlighted with a red arrow. The rest of the interface, including the 'Conditions' sections and the bottom note about unlocking the quality gate, remains the same.

I had created the credential named as **github-cred** in Jenkins for Authenticating GitHub with Jenkins as shown in the screenshot attached below.

The screenshot shows the Jenkins 'Global credentials (unrestricted)' configuration page. The 'Kind' dropdown is set to 'Username with password'. The 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc.)'. The 'Username' field contains a redacted value, and the 'Password' field also contains a redacted value. The 'ID' field is set to 'github-cred'. The 'Description' field contains the text 'github-cred'. A blue 'Create' button at the bottom left is highlighted with a yellow border.

Create a Jenkins Secret to integrate Jenkins Slave Node with Jenkins Master as shown in the screenshot attached below.

The screenshot shows the Jenkins 'Global credentials (unrestricted)' configuration page. The 'Kind' dropdown is set to 'Username with password'. The 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc.)'. The 'Username' field contains a redacted value, and the 'Password' field also contains a redacted value. The 'ID' field is set to 'jenkins-cred'. The 'Description' field contains the text 'jenkins-cred'. A blue 'Create' button at the bottom left is highlighted with a yellow border.

I had integrated a Slave node in Jenkins with the name of **Slave-1**. To do so go to **Manage Jenkins > Nodes > New Node** as shown in the screenshot attached below.

New node

Node name: Slave-1

Type: Permanent Agent

Create

REST API Jenkins 2.492.3

Name: Slave-1

Description: This is a Slave Node.

Plain text Preview

Number of executors: 2

Remote root directory: /home/jenkins

Save

Labels: Slave-1

Usage: Use this node as much as possible

Launch method: Launch agents via SSH

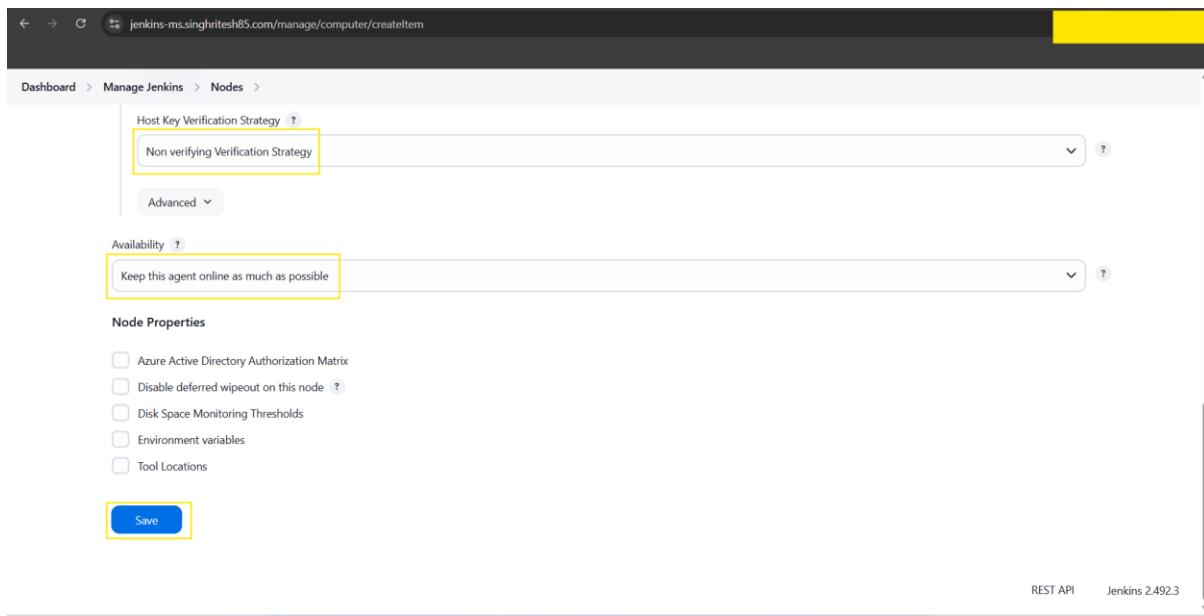
Host: Private IP Address of Jenkins Slave

Credentials

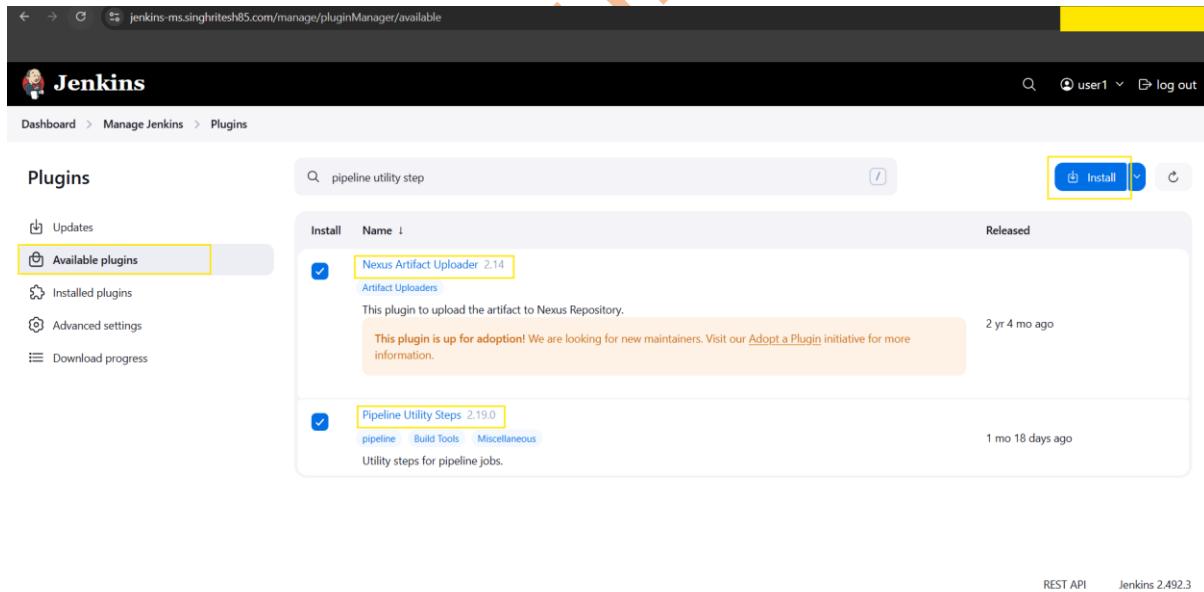
+ Add

Host Key Verification Strategy

Save



Finally, the Slave-1 is in Online mode. To **integrate Jenkins with Sonatype Nexus3** and to use it in **Jenkins Pipeline** I installed the **Nexus Artifact Uploader** and **Pipeline Utility Step** Plugin which is as shown in the screenshot attached below. **Do not restart the Jenkins** after installation of these two plugins.



I had created a credential in Jenkins for Sonatype Nexus3 as shown in the screenshot attached below.

The screenshot shows the Jenkins 'New credentials' configuration page. The 'Kind' dropdown is set to 'Username with password'. The 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc.)'. The 'Username' field contains a redacted value, and the 'Password' field also contains a redacted value. A checkbox labeled 'Treat username as secret' is unchecked. The 'ID' field is set to 'nexus'. At the bottom, there is a blue 'Create' button.

I had provided restricted access to the deployment user **jenkins** using Service Account, Role and Role Binding as shown in the screenshot attached below. The deployment user had all the accesses in the namespaces **bankapp** and **mysql** but does not have access for the entire EKS cluster. That means deployment user **jenkins** access was restricted to the namespaces **bankapp** and **mysql** in the EKS Cluster.

Ritesh Kumar Singh

```
[root@... ~]# cat sa-role-rolebinding.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-bankapp
  namespace: bankapp
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: bankapp
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: bankapp
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: bankapp
  kind: ServiceAccount
  name: jenkins-bankapp
[root@... ~]# kubectl apply -f sa-role-rolebinding.yaml
namespace/bankapp created
serviceaccount/jenkins-bankapp created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding created
```

```
cat sa-role-rolebinding.yaml

apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-bankapp
  namespace: bankapp
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: bankapp
rules:
- apiGroups: ["*"]
  resources: ["*"]
  verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: bankapp
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: bankapp
  kind: ServiceAccount
  name: jenkins-bankapp
```

Created Kubernetes Secrets Which Token was utilized in the kubeconfig file (which was shared with the deployment user jenkins) as shown in the screenshot attached below.

```
[root@yellow ~]# cat secrets.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname
  namespace: bankapp
  annotations:
    kubernetes.io/service-account.name: jenkins-bankapp
[root@yellow ~]# kubectl apply -f secrets.yaml
secret/mysecretname created
[root@yellow ~]# kubectl get secrets -n bankapp
NAME          TYPE           DATA   AGE
mysecretname  kubernetes.io/service-account-token  3      9s
```

cat secrets.yaml

```
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname
  namespace: bankapp
  annotations:
    kubernetes.io/service-account.name: jenkins-bankapp
```

```
[root@yellow ~]# kubectl describe secrets mysecretname -n bankapp
Name:         mysecretname
Namespace:    bankapp
Labels:       <none>
Annotations: kubernetes.io/service-account.name: jenkins-bankapp
              kubernetes.io/service-account.uid: 8
Type:        kubernetes.io/service-account-token

Data
====
ca.crt:     1107 bytes
namespace:  7 bytes
token:      [REDACTED]
```

```
[root@]# cat sa-role-rolebinding-mysql.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-mysql
  namespace: mysql
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: mysql
rules:
- apiGroups: ["*"]
  resources: ["*"]
  verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: mysql
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: mysql
  kind: ServiceAccount
  name: jenkins-mysql

[root@ ~]# kubectl apply -f sa-role-rolebinding-mysql.yaml
namespace/mysql created
serviceaccount/jenkins-mysql created
role.rbac.authorization.k8s.io/user-role created
```

```
cat sa-role-rolebinding-mysql.yaml

apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-mysql
  namespace: mysql
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: mysql
rules:
- apiGroups: ["*"]
  resources: ["*"]
  verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: mysql
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: mysql
  kind: ServiceAccount
  name: jenkins-mysql
```

Created Kubernetes Secrets Which Token was utilized in the kubeconfig file (which was shared with the deployment user jenkins) as shown in the screenshot attached below.

```
[root@... ~]# cat secrets-mysql.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-mysql
  namespace: mysql
  annotations:
    kubernetes.io/service-account.name: jenkins-mysql
[root@... ~]# kubectl apply -f secrets-mysql.yaml
secret/mysecretname-mysql created
[root@... ~]# kubectl get secrets -n mysql
NAME          TYPE           DATA   AGE
mysecretname-mysql  kubernetes.io/service-account-token  3      13s
[root@... ~]# kubectl get secrets -n mysql
NAME          TYPE           DATA   AGE
mysecretname-mysql  kubernetes.io/service-account-token  3      13s
[root@... ~]# kubectl describe secrets mysecretname-mysql -n mysql
Name:         mysecretname-mysql
Namespace:   mysql
Labels:      <none>
Annotations: kubernetes.io/service-account.name: jenkins-mysql
              kubernetes.io/service-account.uid: c
Type:        kubernetes.io/service-account-token
Data
====

token: [REDACTED]
ca.crt: 1107 bytes
namespace: 5 bytes
```

Below is the screenshot of kubeconfig file which I shared with the deployment user **jenkins**. The deployment user **jenkins** will create a directory named as **.kube** and will keep the kubeconfig file at this path as shown in the screenshot attached. The 600 permissions had been provided to the file **.kube/config** using the command **chmod 600 ~/.kube/config**.

```
[jenkins@... ~]$ chmod 600 .kube/config
[jenkins@... ~]$ cat .kube/config
apiVersion: v1
clusters:
- cluster:
  certificate-authority-data: [REDACTED]

server: https://9[REDACTED].us-east-2.eks.amazonaws.com
name: arn:aws:eks:us-east-2:02[REDACTED]:cluster/eks-demo-cluster-dev
contexts:
- context:
  cluster: arn:aws:eks:us-east-2:02[REDACTED]:cluster/eks-demo-cluster-dev
  user: jenkins
  name: mederma
  current-context: mederma
kind: Config
preferences: {}
users:
- name: jenkins
  user:
    token: [REDACTED]
```

After getting the kubeconfig file the deployment user **jenkins** checked its access in EKS Cluster and found that they have only the access inside the **bankapp** and **mysql** namespaces and not for the entire EKS Cluster as shown in the screenshot attached below.

```
[jenkins@[REDACTED] ~]$ kubectl get nodes
Error from server (Forbidden): nodes is forbidden: User "system:serviceaccount:bankapp:jenkins-bankapp" cannot list resource "nodes" in API group "" at the cluster scope
[jenkins@[REDACTED] ~]$ kubectl get pods -n mysql --context=mysql
No resources found in mysql namespace.
[jenkins@[REDACTED] ~]$ kubectl get pods -n bankapp --context=bankapp
No resources found in bankapp namespace.
```

I had created two Jenkins Job one for Bank Application and another for MySQL 8 Pod deployment as shown in the screenshot attached below.

Ritesh Kumar Singh

Jenkinsfile-MySQL

```

pipeline{
    agent{
        node{
            label "Slave-1"
            customWorkspace "/home/jenkins/mysql"
        }
    }
    environment{
        JAVA_HOME="/usr/lib/jvm/java-17-amazon-corretto.x86_64"
        PATH="$PATH:$JAVA_HOME/bin:/opt/apache-maven/bin:/opt/node-v16.0.0/bin:/usr/local/bin"
    }
    stages{
        stage("MySQL-Deployment"){
            steps{
                //MySQL
                sh 'argocd login argocd.singhritesh85.com --username admin --password Admin@123 --skip-test-tls --grpc-web'
                sh 'argocd app create mysql --project default --repo https://github.com/singhritesh85/helm-repo-for-bitnami.git --path ./bitnami/mysql --dest-namespace mysql --sync-option CreateNamespace=true --dest-server https://kubernetes.default.svc --helm-set secondary.replicaCount=1 --helm-set primary.persistence.enabled=true --helm-set primary.persistence.size=1Gi --helm-set architecture=replication --helm-set secondary.persistence.enabled=true --helm-set secondary.persistence.size=1Gi --helm-set primary.service.type=ClusterIP --helm-set auth.rootPassword=Dexter@123 --helm-set auth.database=bankappdb --upsert'
                sh 'argocd app sync mysql'
            }
        }
        post {
            always {
                mail bcc: "", body: "A Jenkins Job with Job Name ${env.JOB_NAME} has been executed", cc: "", from: "", replyTo: "", subject: "Jenkins Job ${env.JOB_NAME} has been executed", to: 'abc@gmail.com'
            }
            success {

```

```
    mail bcc: "", body: "A Jenkins Job with Job Name ${env.JOB_NAME} and Build  
Number=${env.BUILD_NUMBER} has been executed Successfully, Please Open the URL  
${env.BUILD_URL} and click on Console Output to see the Log. The Result of execution is  
${currentBuild.currentResult}", cc: "", from: "", replyTo: "", subject: "Jenkins Job ${env.JOB_NAME} has  
been Sucessfully Executed", to: 'abc@gmail.com'  
}  
  
failure {  
  
    mail bcc: "", body: "A Jenkins Job with Job Name ${env.JOB_NAME} and Build  
Number=${env.BUILD_NUMBER} has been Failed, Please Open the URL ${env.BUILD_URL} and click  
on Console Output to see the Log. The Result of execution is ${currentBuild.currentResult}", cc: "",  
from: "", replyTo: "", subject: "Jenkins Job ${env.JOB_NAME} has been Failed", to: 'abc@gmail.com'  
}  
  
}
```

Ritesh Kumar Singh

Jenkinsfile-BankApp-BlueGreen-Deployment

```

pipeline{
    agent{
        node{
            label "Slave-1"
            customWorkspace "/home/jenkins/bankapp"
        }
    }
    environment{
        JAVA_HOME="/usr/lib/jvm/java-17-amazon-corretto.x86_64"
        PATH="$PATH:$JAVA_HOME/bin:/opt/apache-maven/bin:/opt/node-v16.0.0/bin:/usr/local/bin"
    }
    parameters {
        string(name: 'COMMIT_ID', defaultValue: "", description: 'Provide the Commit ID')
        string(name: 'REPO_NAME', defaultValue: "", description: 'Provide the ECR Repository Name for Application Image')
        string(name: 'TAG_NAME', defaultValue: "", description: 'Provide the TAG Name')
    }
    stages{
        stage("Clone-Code"){
            steps{
                cleanWs()
                checkout scmGit(branches: [[name: '${COMMIT_ID}']], extensions: [], userRemoteConfigs: [[credentialsId: 'github-cred', url: 'https://github.com/singhritesh85/Bank-App.git']])

            }
        }
        stage("SonarQube-Analysis"){
            steps{
                withSonarQubeEnv('SonarQube-Server') {
                    sh 'mvn clean install sonar:sonar'
                }
            }
        }
    }
}

```

```

}

stage("Quality Gate") {
    steps {
        timeout(time: 1, unit: 'HOURS') {
            waitForQualityGate abortPipeline: true
        }
    }
}

stage("Nexus-Artifact Upload"){
    steps{
        script{
            def mavenPom = readMavenPom file: 'pom.xml'

            def nexusRepoName = mavenPom.version.endsWith("SNAPSHOT") ? "maven-snapshot" :
            "maven-release"

            nexusArtifactUploader artifacts: [[artifactId: 'bankapp', classifier: "", file: "target/bankapp-
            ${mavenPom.version}.jar", type: 'jar']], credentialsId: 'nexus', groupId: 'com.example', nexusUrl:
            'nexus.singhritesh85.com', nexusVersion: 'nexus3', protocol: 'https', repository:
            "${nexusRepoName}", version: "${mavenPom.version}"
        }
    }
}

stage("Docker Build"){
    steps{
        sh 'docker system prune -f --all'
        sh 'docker build -t myimage:1.01 -f Dockerfile-Project-1 .'
        sh 'docker tag myimage:1.01 ${REPO_NAME}:${TAG_NAME}'
        sh 'trivy image --exit-code 0 --severity MEDIUM,HIGH ${REPO_NAME}:${TAG_NAME}'
        sh 'trivy image --exit-code 1 --severity CRITICAL ${REPO_NAME}:${TAG_NAME}'
        sh 'aws ecr get-login-password --region us-east-2 | docker login --username AWS --
        password-stdin 027330342406.dkr.ecr.us-east-2.amazonaws.com'
        sh 'docker push ${REPO_NAME}:${TAG_NAME}'
    }
}

```

```

stage("Deployment"){
    steps{
        sh 'argocd login argocd.singhritesh85.com --username admin --password Admin@123 --skip-test-tls --grpc-web'

        sh 'argocd app create bankapp --project default --repo https://github.com/singhritesh85/helm-repo-for-Blue-Green-Deployment.git --path ./folo --dest-namespace bankapp --sync-option CreateNamespace=true --dest-server https://kubernetes.default.svc --helm-set service.port=80 --helm-set image.repository=${REPO_NAME} --helm-set image.tag=${TAG_NAME} --helm-set replicaCount=1 --upsert'

        sh 'argocd app sync bankapp'
    }
}
}

post {
    always {
        mail bcc: "", body: "A Jenkins Job with Job Name ${env.JOB_NAME} has been executed", cc: "", from: "", replyTo: "", subject: "Jenkins Job ${env.JOB_NAME} has been executed", to: 'abc@gmail.com'
    }
    success {
        mail bcc: "", body: "A Jenkins Job with Job Name ${env.JOB_NAME} and Build Number=${env.BUILD_NUMBER} has been executed Successfully, Please Open the URL ${env.BUILD_URL} and click on Console Output to see the Log. The Result of execution is ${currentBuild.currentResult}", cc: "", from: "", replyTo: "", subject: "Jenkins Job ${env.JOB_NAME} has been Sucessfully Executed", to: 'abc@gmail.com'
    }
    failure {
        mail bcc: "", body: "A Jenkins Job with Job Name ${env.JOB_NAME} and Build Number=${env.BUILD_NUMBER} has been Failed, Please Open the URL ${env.BUILD_URL} and click on Console Output to see the Log. The Result of execution is ${currentBuild.currentResult}", cc: "", from: "", replyTo: "", subject: "Jenkins Job ${env.JOB_NAME} has been Failed", to: 'abc@gmail.com'
    }
}
}

```

The Argo Rollout Controller and Argo Rollout Kubectl Plugins are needed for Blue-Green Deployment using Argo Rollout. Which I installed as discussed below.

Install Argo Rollout Controller using the commands as written below.

```
kubectl create namespace argo-rollouts
```

```
kubectl apply -n argo-rollouts -f https://github.com/argoproj/argo-rollouts/releases/latest/download/install.yaml
```

```
[root@... ~]# kubectl create namespace argo-rollouts
```

```
[root@... ~]# kubectl apply -n argo-rollouts -f https://github.com/argoproj/argo-rollouts/releases/latest/download/install.yaml
```

Install Argo Rollouts Kubectl plugin

```
curl -LO https://github.com/argoproj/argo-rollouts/releases/latest/download/kubectl-argo-rollouts-linux-amd64
```

```
chmod +x ./kubectl-argo-rollouts-linux-amd64
```

```
sudo mv ./kubectl-argo-rollouts-linux-amd64 /usr/local/bin/kubectl-argo-rollouts
```

```
[root@... ~]# curl -LO https://github.com/argoproj/argo-rollouts/releases/latest/download/kubectl-argo-rollouts-linux-amd64
% Total    % Received % Xferd  Average Speed   Time   Time     Current
          Dload  Upload   Total Spent   Left Speed
0       0      0      0      0      0      0 --:--:-- --:--:-- --:--:--   0
0       0      0      0      0      0      0 --:--:-- --:--:-- --:--:--   0
100  123M  100  123M      0      0  138M      0 --:--:-- --:--:-- --:--:-- 266M
[root@... ~]# chmod +x ./kubectl-argo-rollouts-linux-amd64
[root@... ~]# sudo mv ./kubectl-argo-rollouts-linux-amd64 /usr/local/bin/kubectl-argo-rollouts
```

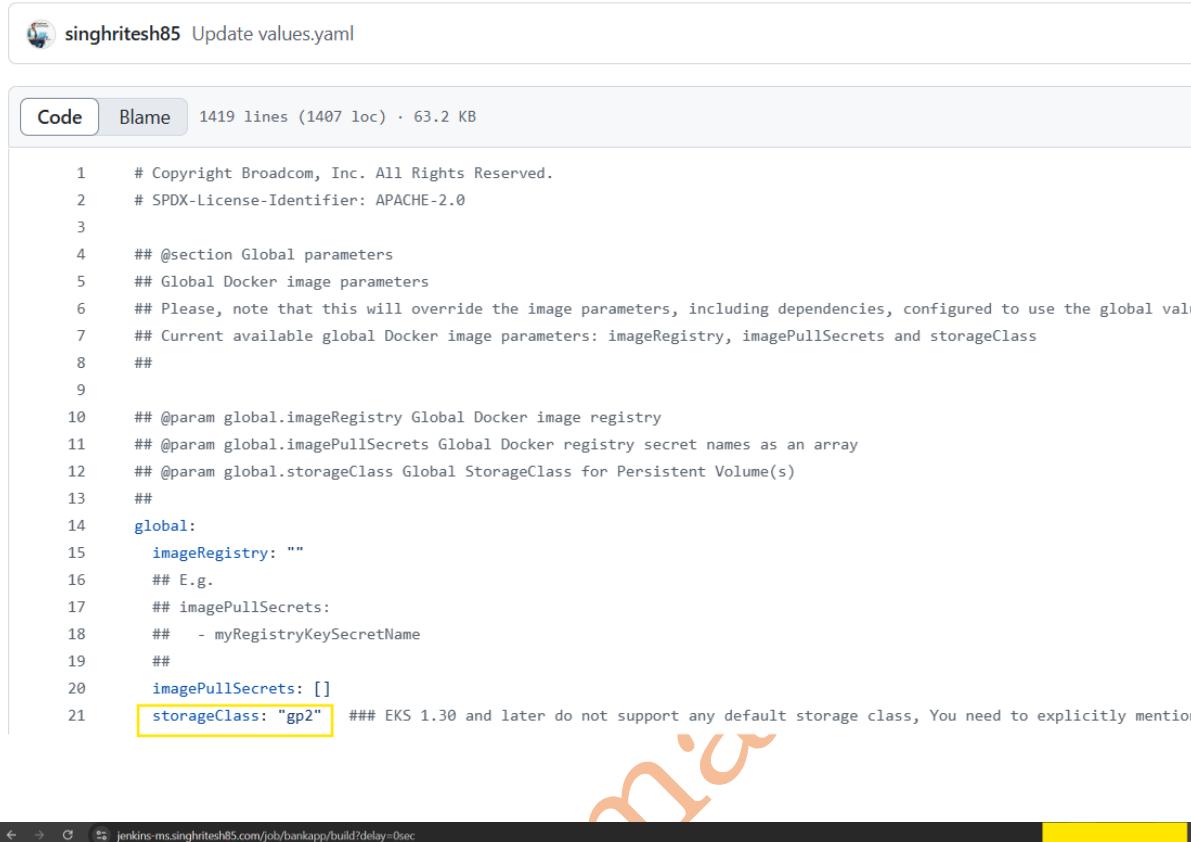
Before Running the Jenkins Job, I changed the Nameserver on Jenkins Slave node as shown in the screenshot attached below. Because for this project the DHCP Option which I am using is the default DHCP Option Set.

```
[root@... ~]# cat /etc/resolv.conf
; generated by /usr/sbin/dhclient-script
search us-east-2.compute.internal
options timeout:2 attempts:5
nameserver 8.8.8.8 #10.10.0.2
```

As mentioned earlier I had created two Jenkins Job **mysql** and **bankapp**. The **mysql** Jenkins Job do not require any parameter for its execution while **bankapp** Jenkins Job required the string parameters for its execution as shown in the screenshot attached below.

Remember for execution of the **mysql** Jenkins Job I used the helm chart present in my GitHub Repo <https://github.com/singhritesh85/helm-repo-for-bitnami.git> at the path **bitnami/mysql**. In this project I am using EKS Cluster 1.30 and EKS Cluster 1.30 and later do not support any default storage class while for earlier version it was **gp2**. So, for EKS Cluster 1.30 and later versions you need to explicitly mention the storage class which you want to use. In this project I used the storage class **gp2** which I mentioned in the **values.yaml** file of the helm chart as shown in the screenshot attached below.

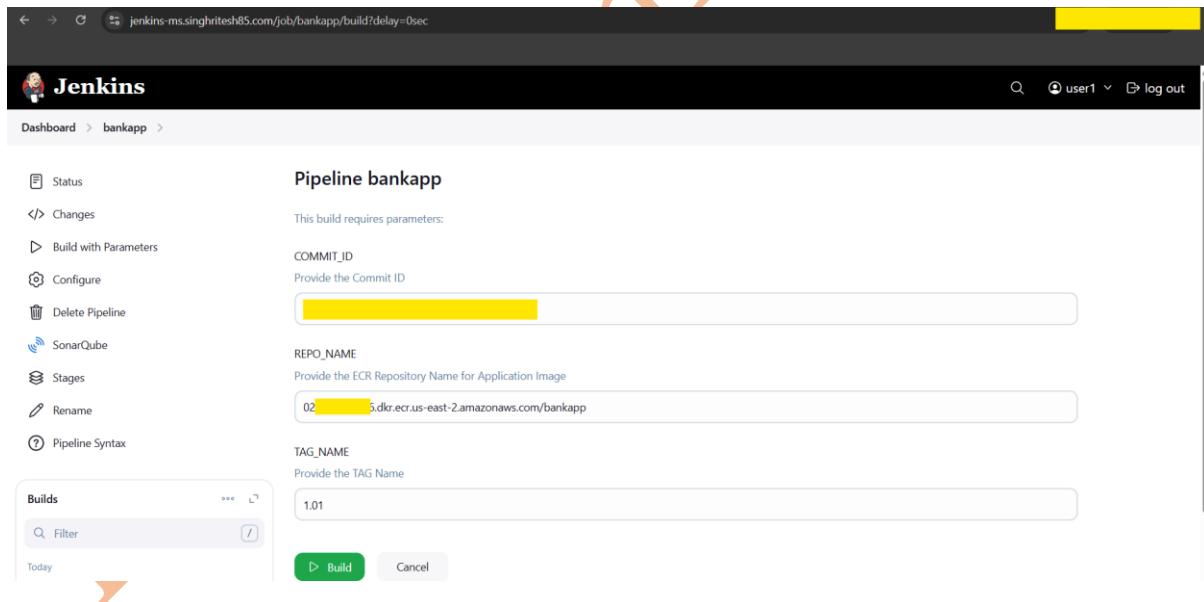
[helm-repo-for-bitnami / bitnami / mysql / values.yaml](#)



```

1  # Copyright Broadcom, Inc. All Rights Reserved.
2  # SPDX-License-Identifier: APACHE-2.0
3
4  ## @section Global parameters
5  ## Global Docker image parameters
6  ## Please, note that this will override the image parameters, including dependencies, configured to use the global value
7  ## Current available global Docker image parameters: imageRegistry, imagePullSecrets and storageClass
8  ##
9
10 ## @param global.imageRegistry Global Docker image registry
11 ## @param global.imagePullSecrets Global Docker registry secret names as an array
12 ## @param global.storageClass Global StorageClass for Persistent Volume(s)
13 ##
14 global:
15   imageRegistry: ""
16   ## E.g.
17   ## imagePullSecrets:
18   ##   - myRegistryKeySecretName
19   ##
20   imagePullSecrets: []
21   storageClass: "gp2"    ### EKS 1.30 and later do not support any default storage class, You need to explicitly mention

```



Pipeline bankapp

This build requires parameters:

- COMMIT_ID**: Provide the Commit ID (input field)
- REPO_NAME**: Provide the ECR Repository Name for Application Image (input field)
- TAG_NAME**: Provide the TAG Name (input field)

Builds

Build Cancel

The screenshot for the two Jenkins Job after its execution is as shown below.

S	W	Name ↑	Last Success	Last Failure	Last Duration	
✓	☀️	mysql	16 min #2	N/A	2 min 35 sec	▶
✓	☁️	bankapp	9 min 14 sec #3	12 min #1	1 min 52 sec	▶

After the successful deployment of the two Jenkins Job **bankapp** and **mysql** I checked the pods and was able to see the Running pods as shown in the screenshot attached below. As I configured Email Notification for Running of Jenkins Job and its Successful completion and found that an Email was sent to the Group Email ID configured as shown in the screenshot attached below.

```
[jenkins@[REDACTED] ~]$ kubectl get pods -n mysql --context=mysql --watch
NAME          READY   STATUS    RESTARTS   AGE
mysql-primary-0  1/1     Running   0          4m10s
mysql-secondary-0  1/1     Running   0          4m10s
[jenkins@[REDACTED] ~]$ kubectl get pods -n bankapp --context=bankapp --watch
NAME          READY   STATUS    RESTARTS   AGE
bankapp-folo-7  1/1     Running   0          117s

[jenkins@[REDACTED] ~]$ kubectl get pvc -n mysql --context=mysql
NAME        STATUS  VOLUME                                     CAPACITY  ACCESS MODES  STORAGECLASS  VOLUME ATTRIBUTESCLASS  AGE
data-mysql-primary-0  Bound  pvc-9                               1Gi       RWO          gp2           <unset>      <unset>          117s
data-mysql-secondary-0  Bound  pvc-1                               1Gi       RWO          gp2           <unset>      <unset>
```

Jenkins Job mysql has been executed Inbox ×



[REDACTED]@gmail.com

to me ▾

A Jenkins Job with Job Name mysql has been executed

Reply

Forward



Jenkins Job mysql has been Sucessfully Executed Inbox ×



[REDACTED]@gmail.com

to me ▾

A Jenkins Job with Job Name mysql and Build Number=6 has been executed Successfully. Please Open the URL <https://jenkins-ms.singhritesh85.com/job/mysql/6/> and click on Console Output to see the Log. The Result of execution is SUCCESS

Reply

Forward



Jenkins Job bankapp has been executed Inbox ×



[REDACTED]@gmail.com

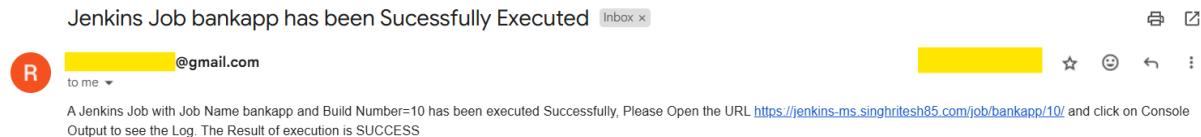
to me ▾

A Jenkins Job with Job Name bankapp has been executed

Reply

Forward





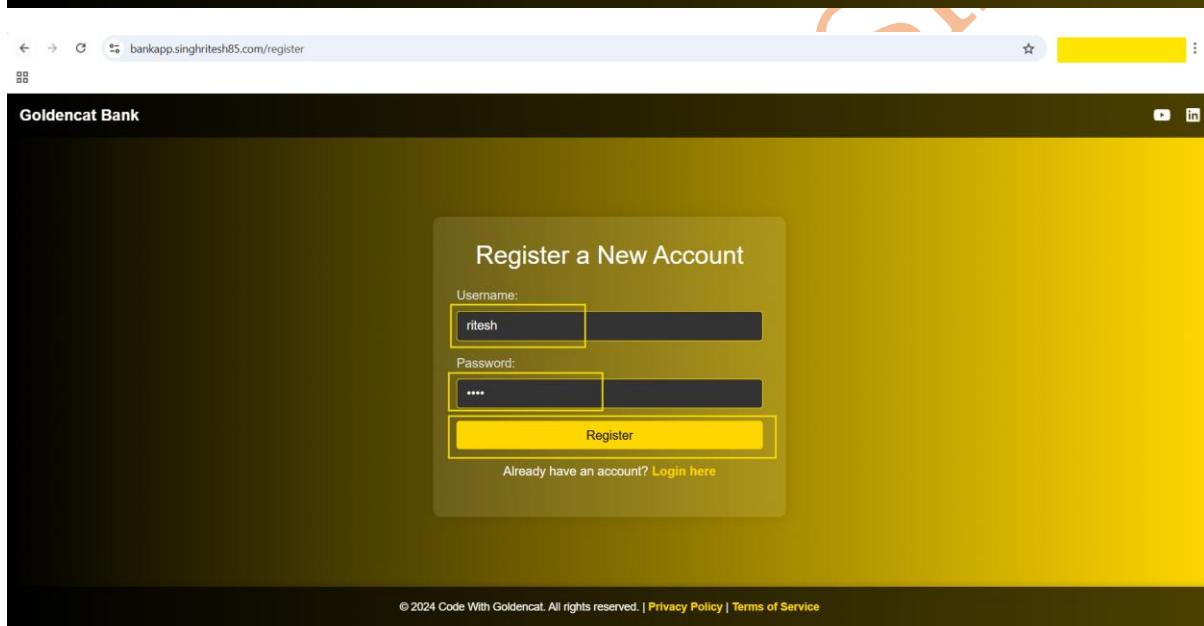
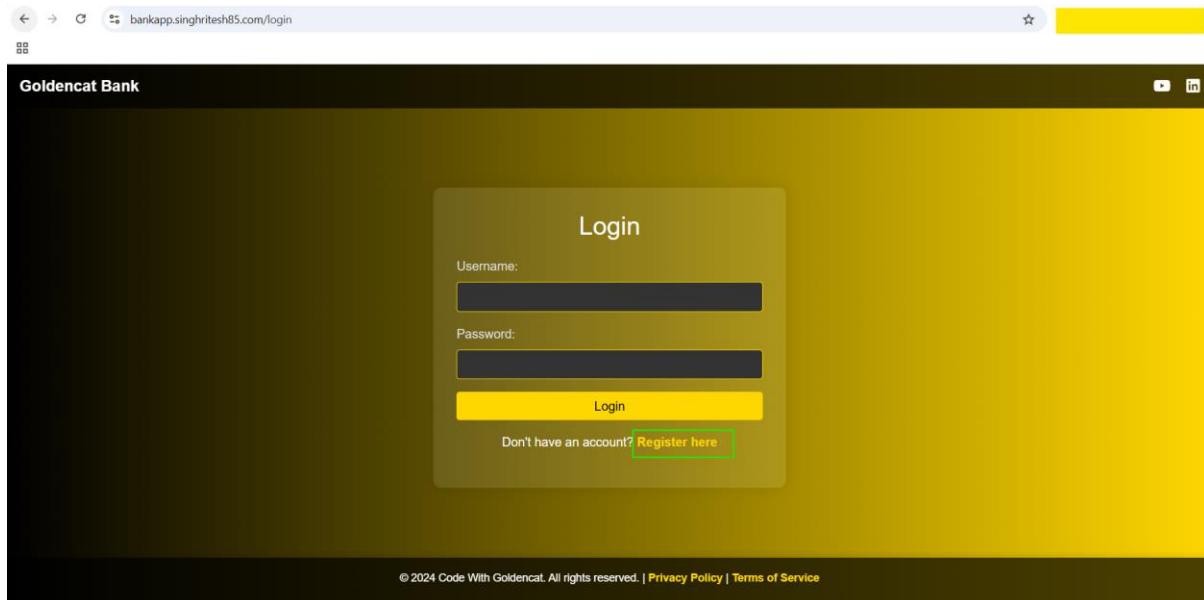
I created the ingress rule for the bank application as shown in the screenshot attached below.

```
[jenkins@yellow ~]$ cat ingress-rule.yaml
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: bankapp-ingress
  namespace: bankapp
  annotations:
    kubernetes.io/ingress.class: nginx
spec:
  ingressClassName: nginx
  rules:
  - host: bankapp.singhritesh85.com
    http:
      paths:
        - path: /
          backend:
            service:
              name: bankapp-folo-active
              port:
                number: 80
            pathType: Prefix
[jenkins@yellow ~]$ kubectl apply -f ingress-rule.yaml
ingress.networking.k8s.io/bankapp-ingress created
[jenkins@yellow ~]$ kubectl get ing -n bankapp --context=bankapp
NAME      CLASS   HOSTS           ADDRESS   PORTS   AGE
bankapp-ingress   nginx   bankapp.singhritesh85.com   80       13s
[jenkins@yellow ~]$ kubectl get ing -n bankapp --context=bankapp --watch
NAME      CLASS   HOSTS           ADDRESS   PORTS   AGE
bankapp-ingress   nginx   bankapp.singhritesh85.com   80       17s
bankapp-ingress   nginx   bankapp.singhritesh85.com   a [REDACTED] 0.us-east-2.elb.amazonaws.com   80       50s
```

I did the entry for DNS Name of the LoadBalancer in Azure DNS Zone to create the Record Set for the HOSTS bankapp.singhritesh85.com as shown in the screenshot attached below.

Name	Type	TTL	Value
@	NS	172800	[REDACTED]
@	SOA	3600	[REDACTED]
[REDACTED]	CNAME	3600	[REDACTED]

Finally, I was able to access the Bank Application as shown in the screenshot attached below.



Now I tried to login with the created user and found that I was able to login successfully as shown in the screenshot attached below.

The screenshots illustrate the user registration and login process for the Goldencat Bank application.

Registration Page:

- URL: bankapp.singhrites85.com/register
- Form Fields:
 - Username: ritesh
 - Password: (redacted)
- Buttons: Register, Already have an account? [Login here](#)

Dashboard Page:

- URL: bankapp.singhrites85.com/dashboard
- Welcome Message: Welcome, ritesh
- Current Balance: \$0.00
- Account Details:
 - Account Number: 1
 - Account Type: Savings
- Buttons: Deposit, Withdraw, Transfer Money
- Navigation: Dashboard, Transactions, Logout

Now I checked the MySQL 8 Pod entry and found that same user was existing in the MySQL 8 database as shown in the screenshot attached below.

```
[jenkins@yellow ~]$ kubectl get pods -n mysql --context=mysql
NAME        READY   STATUS    RESTARTS   AGE
mysql-primary-0  1/1     Running   0          18m
mysql-secondary-0 1/1     Running   0          18m
[jenkins@yellow ~]$ kubectl exec -it mysql-primary-0 -n mysql bash --context=mysql
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND] instead.
Defaulted container "mysql" out of: mysql, preserve-logs-symlinks (init)
I have no name!@mysql-primary-0:/# mysql -h localhost -u root --password
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 244
Server version: 8.4.0 Source distribution

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| bankappdb |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql> use bankappdb;
Database changed
mysql> show tables;
+-----+
| Tables_in_bankappdb |
+-----+
| account |
| transaction |
+-----+
2 rows in set (0.01 sec)

mysql> select * from account;
+-----+-----+-----+-----+
| id | balance | password | username |
+-----+-----+-----+-----+
| 1  | 0.00  | yellow      | ritesh   |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

The screenshot for SonarQube UI, Sonatype Nexus3 Repository and ArgoCD is as shown in the screenshot attached below.

<http://argocd.singhritesh85.com/applications?showFavorites=false&proj=&sync=&autoSync=&health=&namespace=&cluster=&labels=>

<http://nexus.singhritesh85.com/#browse/browse:maven-snapshot>

<http://sonarqube.singhritesh85.com/projects>

bankapp main

Overview Issues Security Hotspots Measures Code Activity

QUALITY GATE STATUS

Passed
All conditions passed.

You added extra conditions to the "Clean as You Code" quality gate, which is something we do not recommend. Review your [quality gate](#).

[Learn why](#)

MEASURES

New Code	Overall Code
2025 Started 23 minutes ago	

Bugs	Vulnerabilities	Security Hotspots	Code Smells
0	0	1	11

0.0% Reviewed

2h 34min Debt

Project Information

Description Banking Web Application

No tags

Lines of Code (Main branch)
471 xs

Quality Gate used
(Default) mederrma

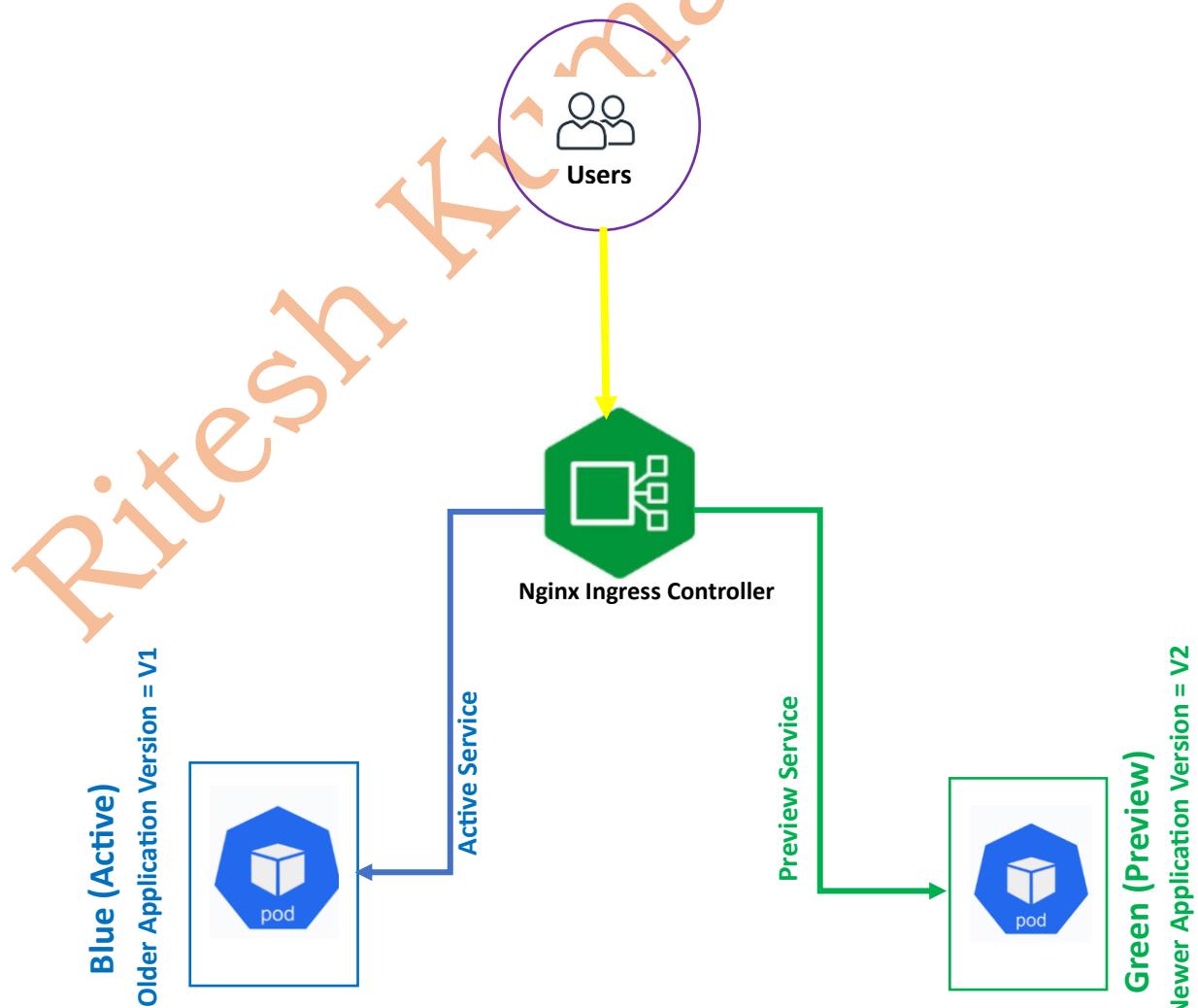
Quality Profiles used
(Java) Sonar way
(XML) Sonar way

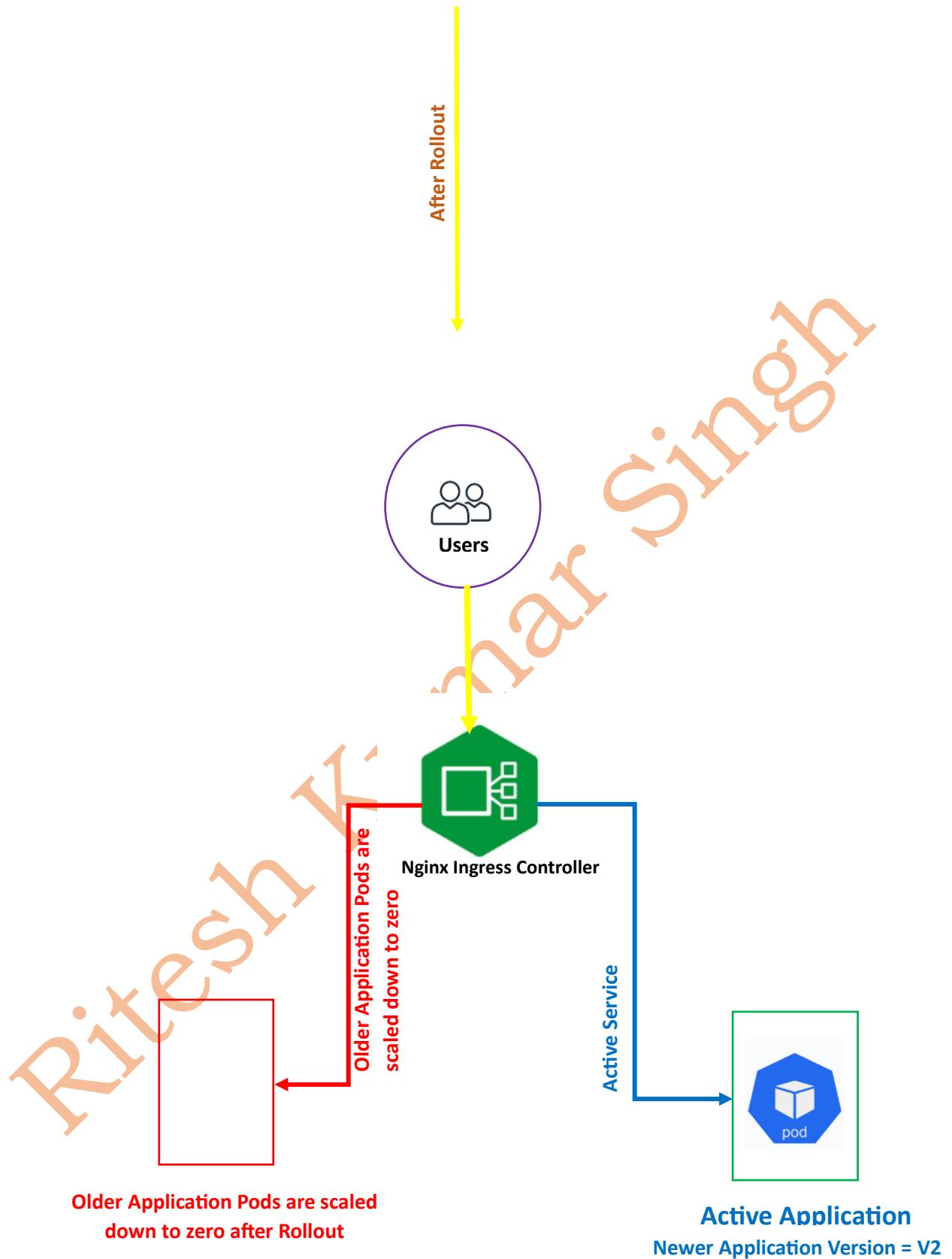
Project Key com.example.bankapp [Copy](#)

[Get project badges](#)

[Set notifications](#)

For a New Release Developer did some change in the Code as per the new Code in Bank App Dashboard instead of Goldencat it will display GoldenDuck. To deploy the new Release of the code I used Blue-Green deployment. In Blue-Green deployment the Blue (Active) represents the Previous version and Green (Preview) represents the newer version of the Application.

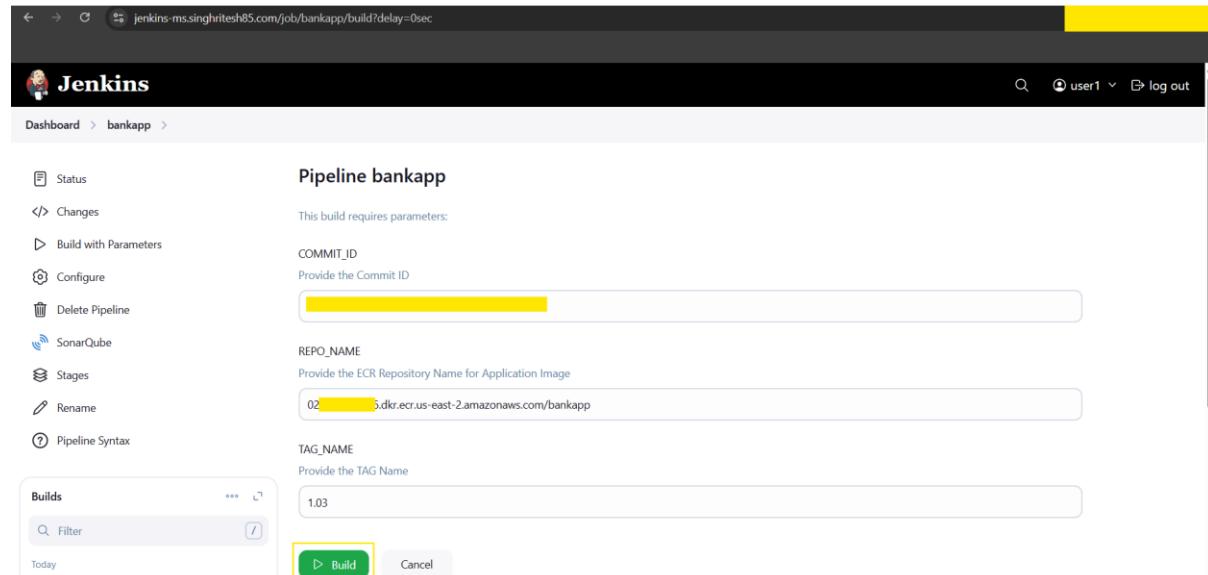




To deploy the newer version of the code I ran the Jenkins Job **bankapp** and changed the Service Type from ClusterIP to LoadBalancer then tested that the Application was working properly or not. Finally,

after testing the behaviour of Bank Application new Release I changed the service type from LoadBalancer to ClusterIP and Promoted the Rollout as discussed below.

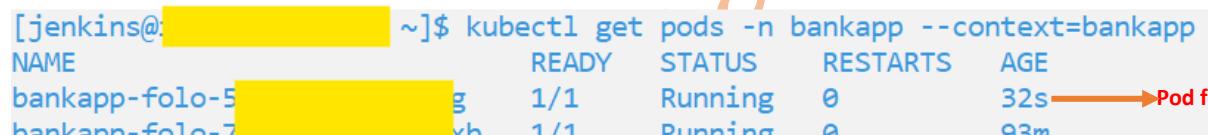
The string parameters which I used to run the Jenkins Job **bankapp** is as shown below.



Jenkins Pipeline bankapp configuration:

- COMMIT_ID**: Provide the Commit ID (redacted)
- REPO_NAME**: Provide the ECR Repository Name for Application Image (02...5.dkr.ecr.us-east-2.amazonaws.com/bankapp)
- TAG_NAME**: Provide the TAG Name (1.03)

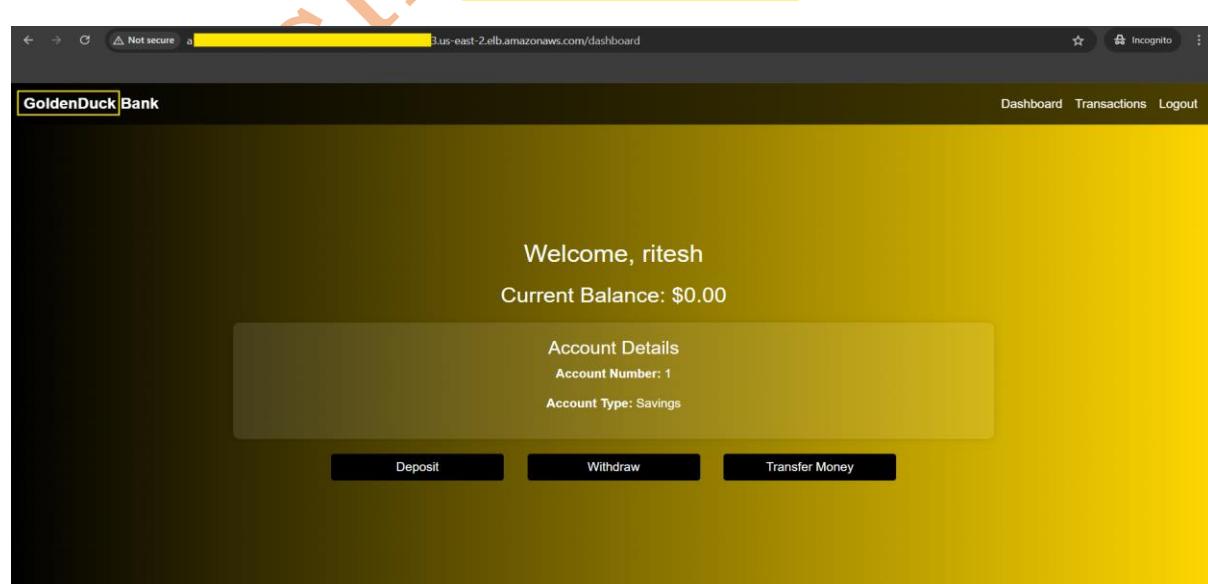
Build button highlighted with a yellow box.



```
[jenkins@... ~]$ kubectl get pods -n bankapp --context=bankapp
NAME           READY   STATUS    RESTARTS   AGE
bankapp-folo-5   1/1     Running   0          32s → Pod for newer release
bankapp-folo-7   1/1     Running   0          93m
```




```
[jenkins@... ~]$ kubectl get svc -n bankapp --context=bankapp
NAME        TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
bankapp-folo-active   ClusterIP  172.17.0.22   <none>       80/TCP   95m
bankapp-folo-preview  ClusterIP  172.17.0.174  <none>       80/TCP   95m
[jenkins@... ~]$ kubectl edit svc bankapp-folo-preview -n bankapp --context=bankapp
service/bankapp-folo-preview edited
[jenkins@... ~]$ kubectl get svc -n bankapp --context=bankapp
NAME        TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
bankapp-folo-active   ClusterIP  172.17.0.22   <none>
bankapp-folo-preview  LoadBalancer 172.17.0.174  a...8.us-east-2.elb.amazonaws.com  80:30782/TCP   96m
```

Welcome, ritesh
Current Balance: \$0.00

Account Details
Account Number: 1
Account Type: Savings

Deposit Withdraw Transfer Money

After testing the newer version of Application (Green) I found it was working properly. Finally promoted the rollout to make the changes as Active which is shown in the screenshot attached below.

kubectl get ro -n <namespace>

kubectl argo rollouts get rollout <rollout-name> -n <namespace>

kubectl argo rollouts promote <rollout-name> -n <namespace>

```
[jenkins@[REDACTED] ~]$ kubectl config get-contexts
CURRENT   NAME    CLUSTER                                     AUTHINFO          NAMESPACE
*   bankapp   arn:aws:eks:us-east-2:02   6:cluster/eks-demo-cluster-dev   jenkins-bankapp
      mysql    arn:aws:eks:us-east-2:02   6:cluster/eks-demo-cluster-dev   jenkins-mysql

[jenkins@[REDACTED] ~]$ kubectl get ro -n bankapp
NAME      DESIRED   CURRENT   UP-TO-DATE   AVAILABLE   AGE
bankapp-folo  1         2         1           1          113m

[jenkins@[REDACTED] ~]$ kubectl argo rollouts get rollout bankapp-folo -n bankapp
Name:          bankapp-folo
Namespace:     bankapp
Status:        || Paused
Message:       BlueGreenPause
Strategy:      BlueGreen
Images:        02 [REDACTED] 6.dkr.ecr.us-east-2.amazonaws.com/bankapp:1.02 (stable, active)
                02 [REDACTED] 6.dkr.ecr.us-east-2.amazonaws.com/bankapp:1.05 (preview)
Replicas:
  Desired:    1
  Current:    2
  Updated:    1
  Ready:      1
  Available:  1

NAME                           KIND   STATUS   AGE   INFO
bankapp-folo                   Rollout  || Paused  113m
└─# revision:3
   └─bankapp-folo-6[REDACTED]8   ReplicaSet  ✓Healthy  10m  preview
      └─bankapp-folo-6[REDACTED]8-w[REDACTED]x   Pod      ✓Running  10m  ready:1/1
   └─# revision:2
      └─bankapp-folo-5[REDACTED]8   ReplicaSet  • ScaledDown  20m
   └─# revision:1
      └─bankapp-folo-7[REDACTED]c   ReplicaSet  ✓Healthy  113m  stable,active
         └─bankapp-folo-7[REDACTED]c-g[REDACTED]b   Pod      ✓Running  113m  ready:1/1

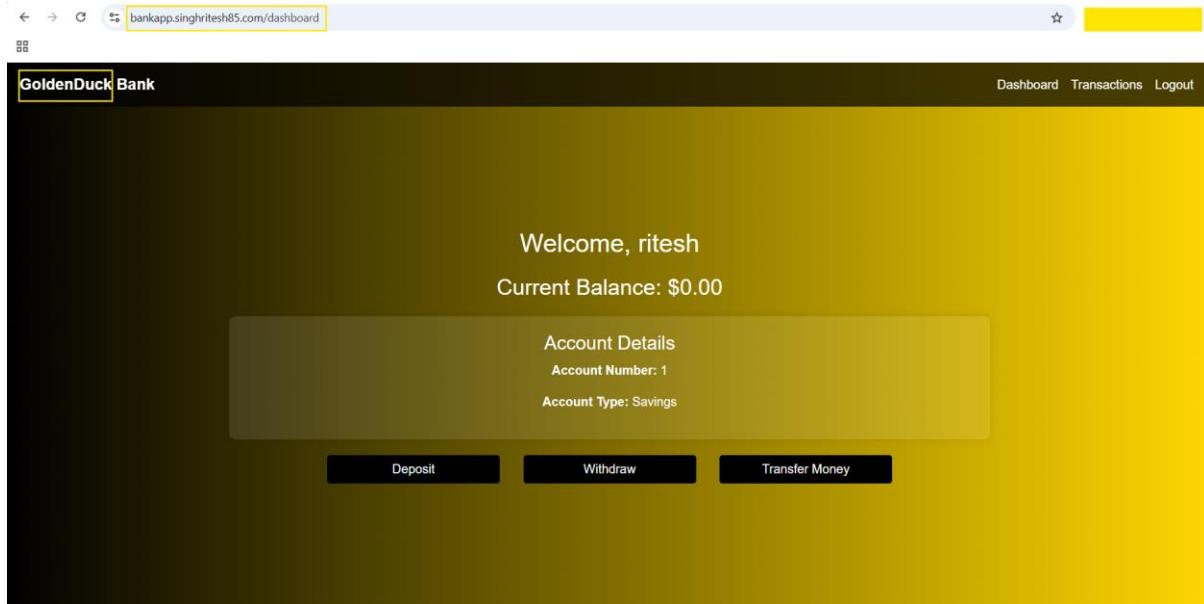
[jenkins@[REDACTED] ~]$ kubectl argo rollouts promote bankapp-folo -n bankapp
rollout 'bankapp-folo' promoted

[jenkins@[REDACTED] ~]$ kubectl get all -n bankapp
NAME          READY   STATUS    RESTARTS   AGE
pod/bankapp-folo-[REDACTED]  1/1     Running   0          [REDACTED]

NAME              TYPE   CLUSTER-IP      EXTERNAL-IP   PORT(S)   AGE
service/bankapp-folo-active  ClusterIP  172.[REDACTED]  <none>      80/TCP   [REDACTED]
service/bankapp-folo-preview  ClusterIP  172.[REDACTED]  <none>      80/TCP   [REDACTED]

NAME          DESIRED   CURRENT   READY   AGE
replicaset.apps/bankapp-folo-[REDACTED]  1         1         1
replicaset.apps/bankapp-folo-[REDACTED]  0         0         0
replicaset.apps/bankapp-folo-[REDACTED]  0         0         0
```

Finally, the newer release came out which was ready for others to use as shown in the screenshot attached below.



The SonarQube UI after this release (considering this as a minor release) is as shown in the screenshot attached below.

For this minor release I changed the version from pom.xml

Installation of node-exporter and promtail had been done using the helm chart in the EKS Cluster as written below.

```
helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
kubectl create ns node-exporter
helm install my-prometheus-node-exporter prometheus-community/prometheus-node-exporter --version 4.37.1 --set service.type=LoadBalancer -n node-exporter
```

```
[root@yellow ~]# helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
"prometheus-community" has been added to your repositories
[root@yellow ~]# kubectl create ns node-exporter
namespace/node-exporter created
[root@yellow ~]# helm install my-prometheus-node-exporter prometheus-community/prometheus-node-exporter --version 4.37.1 --set service.type=LoadBalancer -n node-exporter
NAME: my-prometheus-node-exporter
LAST DEPLOYED: [REDACTED] Apr [REDACTED] 2025
NAMESPACE: node-exporter
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
1. Get the application URL by running these commands:
  NOTE: It may take a few minutes for the LoadBalancer IP to be available.
  You can watch the status of by running 'kubectl get svc -w my-prometheus-node-exporter'
  export SERVICE_IP=$(kubectl get svc --namespace node-exporter my-prometheus-node-exporter -o jsonpath='{.status.loadBalancer.ingress[0].ip}')
  echo http://$SERVICE_IP:9100
[root@yellow ~]#
```

Below screenshot shows the Kubernetes Service which was created for node-exporter using the helm chart.

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
my-prometheus-node-exporter	LoadBalancer	172.31.63.a	5.us-east-2.elb.amazonaws.com	9100:31063/TCP	2m55s

The updated prometheus configuration file **/etc/prometheus/prometheus.yml** is as shown in the screenshot attached below.

```
- job_name: "EKS"
  static_configs:
    - targets: ["a.us-east-2.elb.amazonaws.com:9100"]
```

Then restarted the prometheus service as shown in the screenshot attached below.

```
[root@yellow ~]# systemctl restart prometheus.service
[root@yellow ~]# systemctl status prometheus.service
● prometheus.service - Prometheus
  Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor preset: disabled)
  Active: active (running) since [REDACTED] 2025-04-11 10:45:44 +0000 UTC; 9s ago
    Main PID: 2998 (prometheus)
```

I installed the promtail using the helm chart as written below. First, I cloned the helm chart present in GitHub Repo.

git clone https://github.com/singhritesh85/helm-chart-promtail.git

After cloning helm chart from GitHub, I updated the values.yaml file of promtail helm chart with Loki Servers Private IP Addresses as shown in the screenshot attached below.

kubectl create ns promtail && helm upgrade --install promtail ./helm-chart-promtail -f ./helm-chart-promtail/values.yaml -n promtail

kubectl get pods -n promtail --watch

```

# -- The log level of the Promtail server
# Must be reference in `config.file` to configure `server.log_level`
# See default config in `values.yaml`
logLevel: info

# -- The log format of the Promtail server
# Must be reference in `config.file` to configure `server.log_format`
# Valid formats: `logfmt, json`
# See default config in `values.yaml`
logFormat: logfmt

# -- The port of the Promtail server
# Must be reference in `config.file` to configure `server.http_listen_port`
# See default config in `values.yaml`
serverPort: 3101

# -- The config of clients of the Promtail server
# Must be reference in `config.file` to configure `clients`
# @default -- See `values.yaml`
clients:
  - url: http://10.165.31.10:3100/loki/api/v1/push
  - url: http://10.14.31.10:3100/loki/api/v1/push
  - url: http://10.195.31.10:3100/loki/api/v1/push

# -- Configures where Promtail will save it's positions file, to resume reading after restarts.
# Must be referenced in `config.file` to configure `positions`
positions:
  filename: /run/promtail/positions.yaml

# -- The config to enable tracing
enableTracing: false

# -- A section of reusable snippets that can be reference in `config.file`.
# Custom snippets may be added in order to reduce redundancy.

```

```

[root@10.165.31.10 ~]# git clone https://github.com/singhritesh85/helm-chart-promtail.git
[root@10.165.31.10 ~]# kubectl create ns promtail && helm upgrade --install promtail ./helm-chart-promtail -f ./helm-chart-promtail/values.yaml -n promtail
[root@10.165.31.10 ~]# kubectl get pods -n promtail --watch
NAME        READY   STATUS    RESTARTS   AGE
promtail-cr   1/1     Running   0          98s
promtail-xd   1/1     Running   0          98s

```

Monitoring Using Prometheus and Grafana and Log Aggregation using Loki

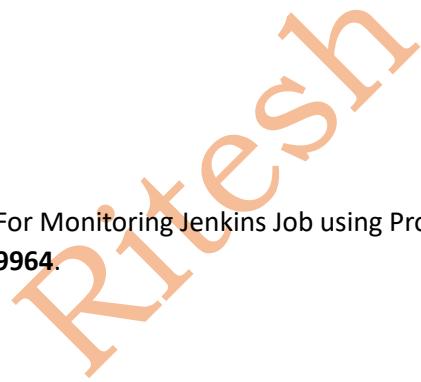
For Monitoring Tool I had used Prometheus and Grafana. To monitor SonarQube I had used SonarQube-Prometheus-Exporter which was installed using terraform at the path `/opt/sonarqube/extensions/plugins`. It was downloaded from the link <https://github.com/dmeiners88/sonarqube-prometheus-exporter/releases/download/v1.0.0-SNAPSHOT-2018-07-04/sonar-prometheus-exporter-1.0.0-SNAPSHOT.jar>. These steps had been covered in the terraform `user_data_sonarqube.sh`. It is basically a bootstrap script for SonarQube Server. For Monitoring Jenkins, you need to install the plugin **Prometheus metrics** and then restart Jenkins, these steps already been discussed at the starting. The configuration for prometheus had already been done in the terraform. I had taken sonarqube username and password as **admin** and

Admin123 respectively, you can choose as of your own choice and update the terraform script accordingly (Prometheus needs username and password to extract the metrics from SonarQube). I had provided the terraform script with this GitHub Repository. I had already integrated Prometheus and Loki as a data source for Grafana which was already discussed above.

Here I checked the prometheus console and I found all the Targets was UP as shown in the screenshot attached below.

The screenshot shows the Prometheus Targets page with four sections:

- BlacboxExporter-Server (1/1 up)**: Shows one endpoint at `http://10.4.67.9100/metrics` in the `UP` state, last scraped 24.533s ago, with a scrape duration of 38.264ms. Labels include `instance="10.4.67.9100"` and `job="BlacboxExporter-Server"`.
- EKS (1/1 up)**: Shows one endpoint at `http://a[REDACTED].us-east-2.elb.amazonaws.com:9100/metrics` in the `UP` state, last scraped 26.261s ago, with a scrape duration of 26.531ms. Labels include `instance="a[REDACTED].us-east-2.elb.amazonaws.com:9100"` and `job="EKS"`.
- Grafana-Server (1/1 up)**: Shows one endpoint at `http://10.4.151.9100/metrics` in the `UP` state, last scraped 18.125s ago, with a scrape duration of 22.267ms. Labels include `instance="10.4.151.9100"` and `job="Grafana-Server"`.
- Jenkins-Job (1/1 up)**: Shows one endpoint at `http://10.4.46.8080/prometheus` in the `UP` state, last scraped 19.307s ago, with a scrape duration of 12.612ms. Labels include `instance="10.4.46.8080"` and `job="Jenkins-Job"`.



Prometheus Alerts Graph Status ▾ Help

Loki-Server-2 (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.4.244:9100/metrics	UP	instance="10.10.4.244:9100" job="Loki-Server-2"	24.906s ago	48.090ms	

Loki-Server-3 (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.4.87:9100/metrics	UP	instance="10.10.4.87:9100" job="Loki-Server-3"	24.92s ago	19.303ms	

Nexus-Server (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.4.51:9100/metrics	UP	instance="10.10.4.51:9100" job="Nexus-Server"	27.261s ago	27.284ms	

Prometheus-Server (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9100/metrics	UP	instance="localhost:9100" job="Prometheus-Server"	18.969s ago	38.597ms	

Battery saver ... ×
Battery saver is on
Consider plugging in your device.

Prometheus Alerts Graph Status ▾ Help

SonarQube-Server (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.4.104:9100/metrics	UP	instance="10.10.4.104:9100" job="SonarQube-Server"	23.746s ago	21.465ms	

blackbox (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.4.67:9115/probe module="http_2xx_example" target="https://bankapp.singhriteshs85.com"	UP	instance="https://bankapp.singhriteshs85.com" job="blackbox"	27.694s ago	24.761ms	

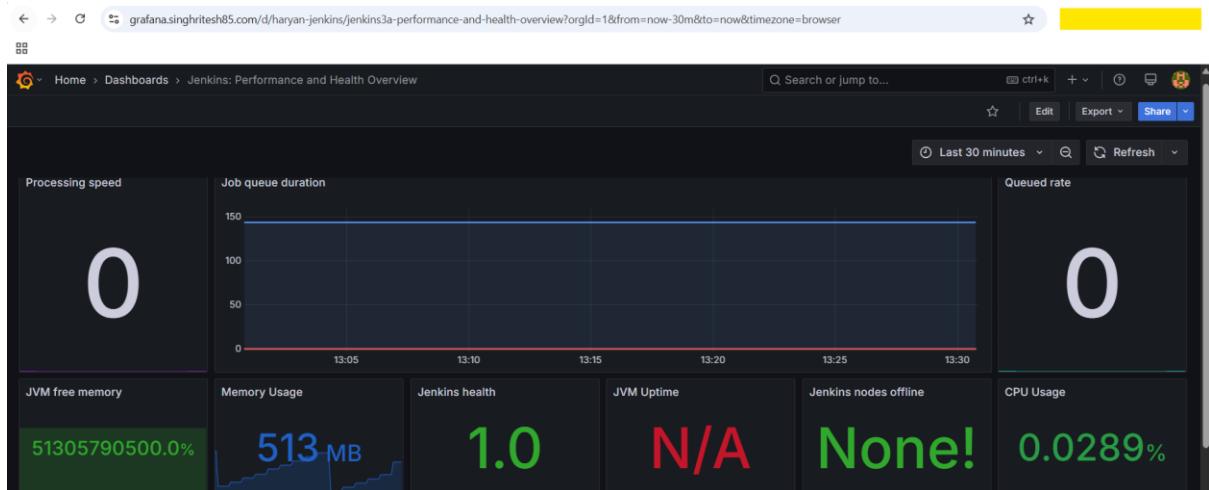
prometheus (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	19.93s ago	9.034ms	

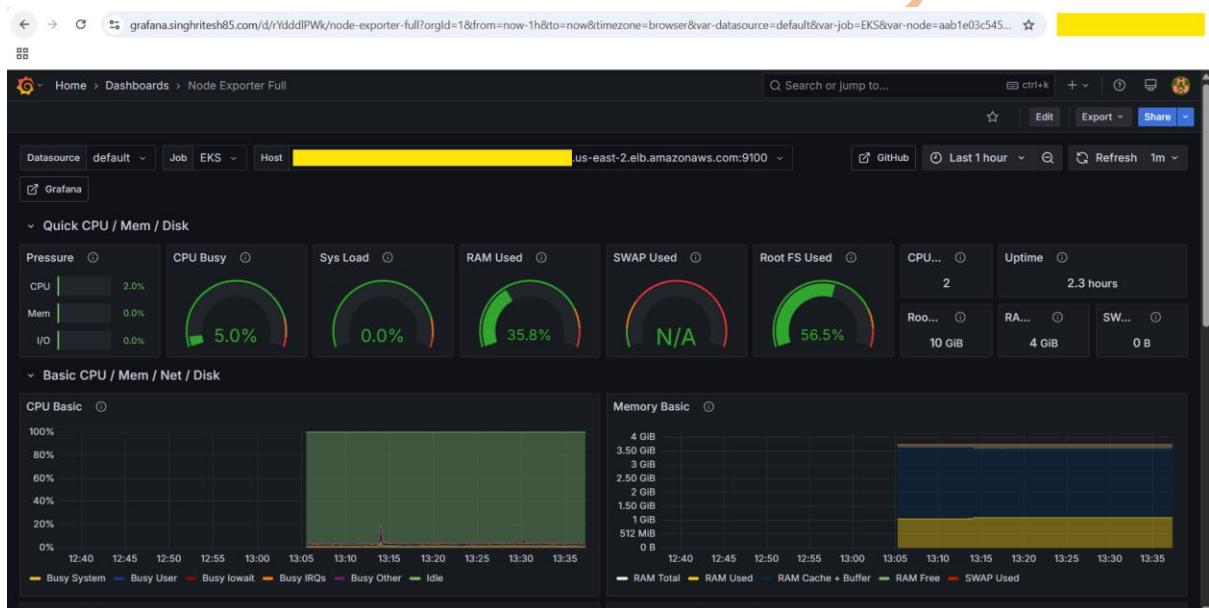
sonarqube (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.4.104:9000/api/prometheus/metric	UP	instance="10.10.4.104:9000" job="sonarqube"	21.884s ago	235.911ms	

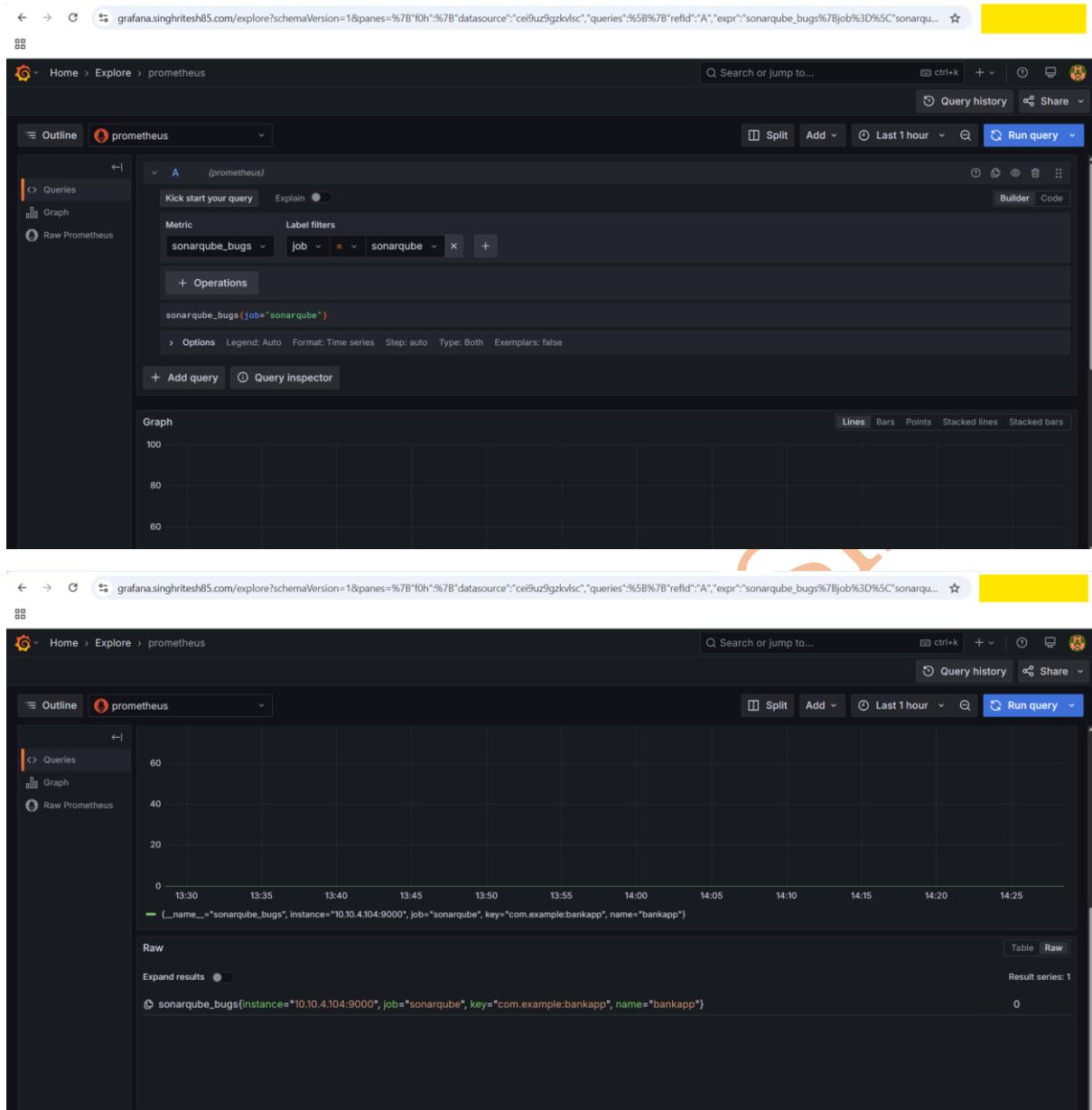
For Monitoring Jenkins Job using Prometheus I created the Grafana Dashboard using the Grafana ID 9964.



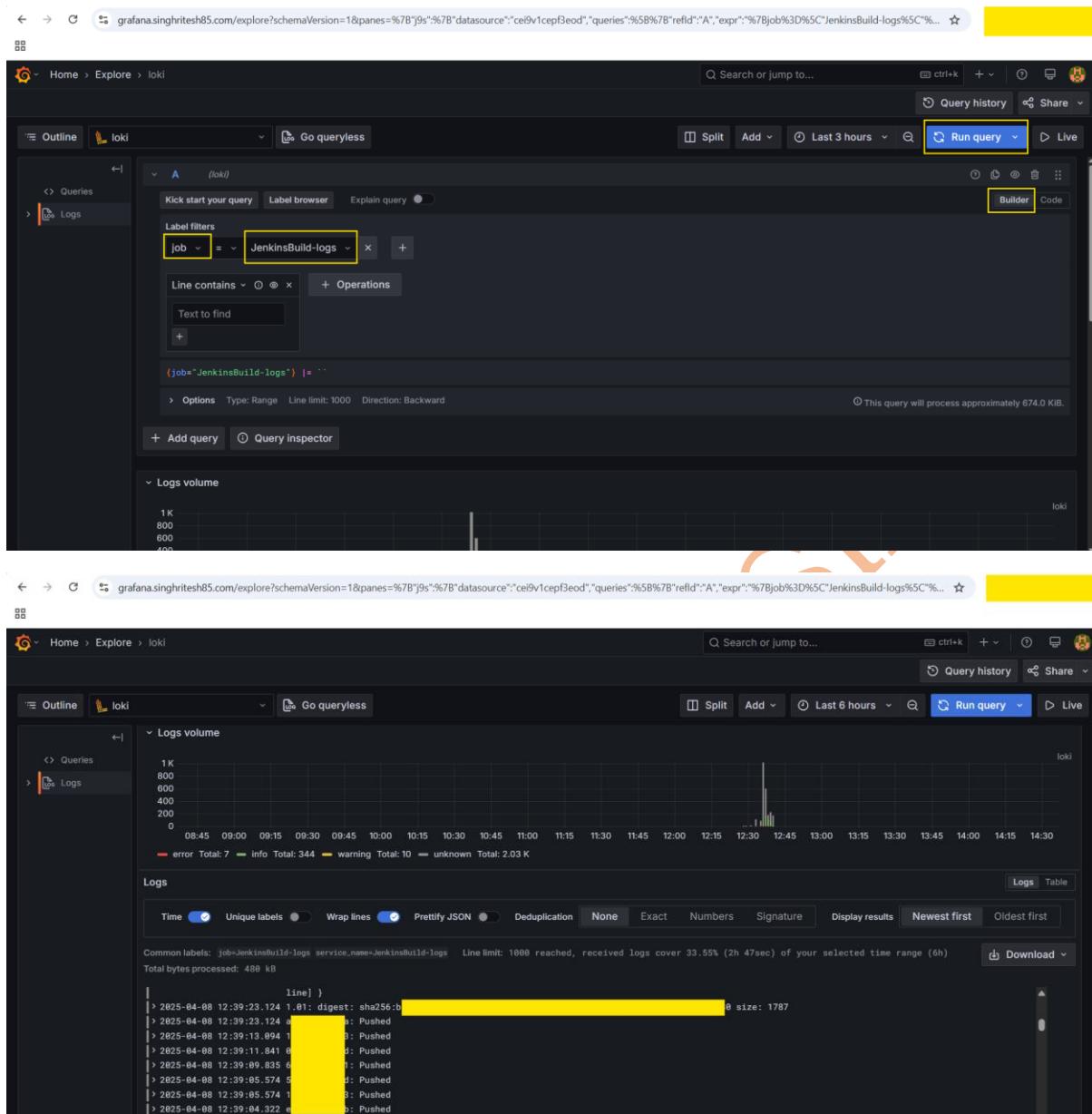
For Monitoring all the Servers and EKS Cluster health using the Node Exporter I used Grafana ID **1860**.



Grafana Metrics for SonarQube I started exploring as shown in the screenshot attached below.



Logs using Loki through Grafana I started exploring as shown in the screenshot attached below.



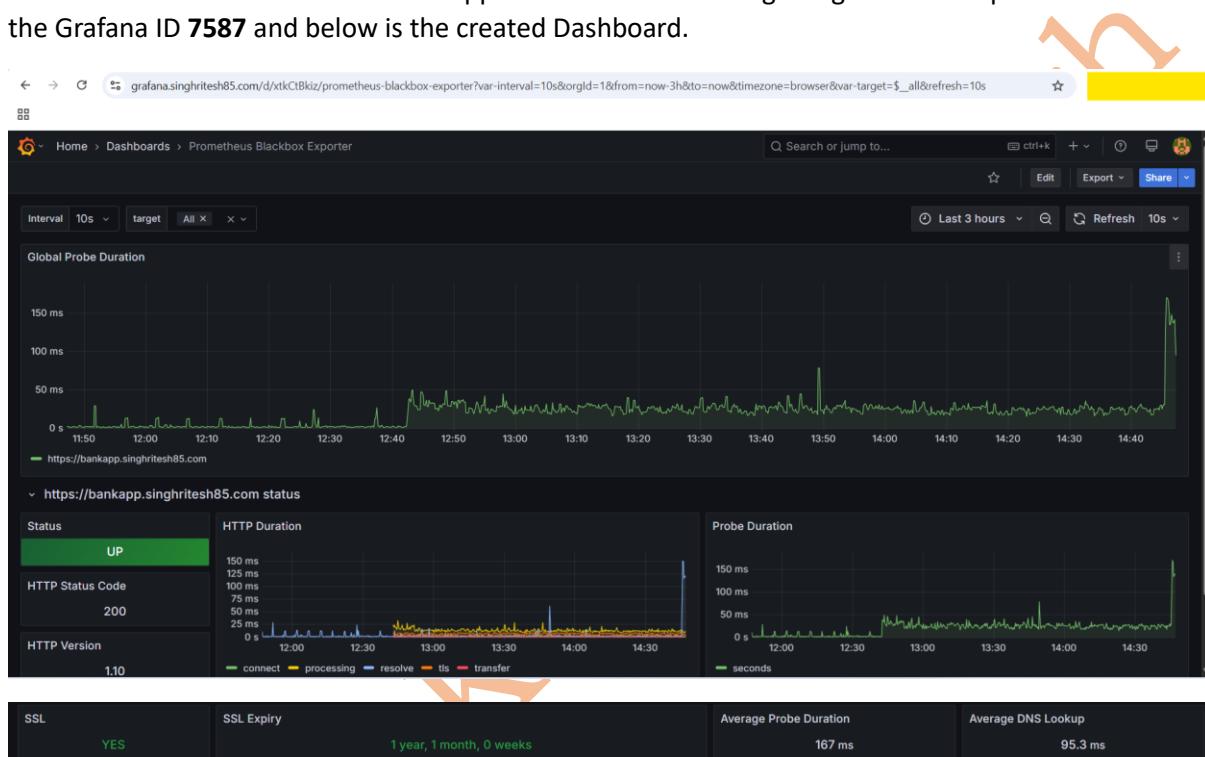
To achieve synthetic monitoring using Prometheus Blackbox Exporter I updated the `/etc/resolv.conf` file for Blackbox Exporter Server as shown in the screenshot attached below. I had used Google's Public DNS Server which is shown in the attached screenshot below.

```
[root@REDACTED ~]# cat /etc/resolv.conf
; generated by /usr/sbin/dhclient-script
search us-east-2.compute.internal
options timeout:2 attempts:5
nameserver 8.8.8.8      #10.10.0.2
```

Finally, I was able to perform the synthetics monitoring on the Bankapp Application URL as shown in the screenshot attached below. Application URL <https://bankapp.singhritesh85.com> had been monitored using blackbox exporter.

I had installed Blackbox Exporter on a different server and not on the Prometheus Server. The **module name** is monitor_website.yml present of the blackbox exporter server at the path (/opt/blackbox_exporter_linux_amd64/monitor_website.yml). Prometheus blackbox operator is used for endpoint monitoring (Synthetic Monitoring) across the protocol http, https, TCP and ICMP. In this project I am monitoring the Application URL <https://bankapp.singhritesh85.com> with the help of Prometheus Blackbox-Exporter. Prometheus blackbox exporter will send the metrics to Prometheus. For this project Prometheus acts as a DataSource for Grafana and send metrics to Grafana which we can see with the help of Charts and Graphs.

To create the Grafana Dashboard for Application URL Monitoring using blackbox exporter I had used the Grafana ID **7587** and below is the created Dashboard.



Configuration of Alerts in Grafana

To configure Alerts in Grafana, first I created **contact points** with the Email ID and changed smtp settings in the configuration file /etc/grafana/grafana.ini of Grafana which I already discussed above. Here I had configured the contact points in Grafana UI as shown in the screenshot attached below.

grafana.singhriteshs85.com/?orgId=1&from=now-6h&to=now&timezone=browser

Need help? Documentation Tutorials Community Public Slack

Remove this panel

TUTORIAL DATA SOURCE AND DASHBOARDS
Grafana fundamentals

Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the "Data source" and "Dashboards" steps to the right.

COMPLETE Add your first data source

COMPLETE Create your first dashboard

Learn how in the docs [\[link\]](#) Learn how in the docs [\[link\]](#)

Latest from the blog

Apr 07 A privacy-first, data-driven approach to optimize the user experience: Introducing Geolocation Insights in Frontend Observability

grafana.singhriteshs85.com/alerting/notifications/receivers/new

Contact points

Choose how to notify your contact points when an alert instance fires

Create contact point

Name * mederma

Integration Email

Addresses You can enter multiple email addresses using a ";", "\n" or "," separator mederma@gmail.com

> Optional Email settings

> Notification settings

+ Add contact point integration

Save contact point Cancel

grafana.singhriteshs85.com/alerting/notifications/receivers/new

Contact points

Choose how to notify your contact points when an alert instance fires

Create contact point

Name * mederma

Integration Email

Addresses You can enter multiple email addresses using a ";", "\n" or "," separator mederma@gmail.com

> Optional Email settings

> Notification settings

+ Add contact point integration

Save contact point Cancel

Test contact point

Notification message Predefined Custom

You will send a test notification that uses a predefined alert. If you have defined a custom template or message, for better results switch to custom notification message, from above.

Send test notification

Test alert sent.

The Default Notification Policy had been changed as shown in the screenshot attached below.

Configure Alert Rule as shown in the screenshot attached below.

To create New Alerts first click on + sign

grafana.singhritesh85.com/alerting/new

New alert rule

- 1. Enter alert rule name**
Enter a name to identify your alert rule.
Name
zago-medema
- 2. Define query and alert condition**
Define query and alert condition [Need help?](#)

A prometheus Options 10 minutes Set as alert condition

Kick start your query Explain Advanced options

Metric process_cpu_seconds_total Label filters Instance = us-east-2.elb.amazonaws.com:9100

+ Operations hint: add rate

process.cpu.seconds.total{instance="us-east-2.elb.amazonaws.com:9100"} us-east-2.elb.amazonaws.com:9100

> Options Legend: Auto Format: Time series Step: auto Type: Instant
- 3. Add folder and labels**
Organize your alert rule with a folder and set of labels. [Need help?](#)

Folder Select a folder to store your rule in.

Creating new folder...

Labels Add labels to your rule for searching, silencing, or routing to a notification policy. [Need help?](#)
No labels selected + Add labels
- 4. Set evaluation behavior**
Define how the alert rule is evaluated. [Need help?](#)
Select a folder before setting evaluation group and interval
Select an evaluation group... or + New evaluation group
Pending period Period during which the threshold condition must be met to trigger an alert. Selecting "None" triggers the alert immediately once the condition is met.

3. Add folder and labels
Organize your alert rule with a folder and set of labels. [Need help?](#)

Folder
Select a folder to store your rule in.

CPU time or + New folder

Labels
Add labels to your rule for searching, silencing, or routing to a notification policy. [Need help?](#)

No labels selected + Add labels

4. Set evaluation behavior
Define how the alert rule is evaluated. [Need help?](#)

Evaluation group and interval
Select an evaluation group... or + New evaluation group

Pending period
Period during which the threshold condition must be met to trigger an alert. Selecting "None" triggers the alert immediately once the condition is met.

1m
None 1m 2m 3m 4m 5m

> Configure no data and error handling

New evaluation group

Create a new evaluation group to use for this alert rule.

Evaluation group name
A group evaluates all its rules over the same evaluation interval.
CPU time

Evaluation interval
How often all rules in the group are evaluated.
1m
10s 30s 1m 5m 10m 15m 30m 1h

Cancel Create

4. Set evaluation behavior
Define how the alert rule is evaluated. [Need help?](#)

Evaluation group and interval
CPU time or + New evaluation group

All rules in the selected group are evaluated every 1m.

Pending period
Period during which the threshold condition must be met to trigger an alert. Selecting "None" triggers the alert immediately once the condition is met.

1m
None 1m 2m 3m 4m 5m

> Configure no data and error handling

5. Configure notifications
Select who should receive a notification when an alert rule fires.

Recipient
Notifications for firing alerts are routed to a selected contact point. [Need help?](#)

Alertmanager: grafana

Contact point

5. Configure notifications
Select who should receive a notification when an alert rule fires.

Recipient
Notifications for firing alerts are routed to a selected contact point. [Need help?](#)

Alertmanager: grafana

Contact point: mederma

Email: [REDACTED]@gmail.com

Muting, grouping and timings (optional) ▾

6. Configure notification message
Add more context to your alert notifications. [Need help?](#)

Summary (optional)
Short summary of what happened and why.

Enter a summary...

If the Alert Rule is in firing state after condition crosses the threshold condition, then Grafana console screenshot will be showing the same as shown in the screenshot attached below.

Alert rules
Rules that determine whether an alert will fire

Search by data sources All data sources Dashboard State Rule type

Health Contact point

Ok	No Data	Error	Choose
----	---------	-------	--------

Search Q Search View as Grouped List State

1 rule **1 firing**

Grafana-managed

CPU time > CPU time

State	Name	Health	Summary	Next evaluation	Actions
Firing	for 35s	zago-mederma	ok	in a few seconds	More ▾

Data source-managed

No rules found.

An Email was sent to the Email ID as shown in the screenshot attached below.

[FIRING:1] zago-medderma CPU time ([REDACTED].us-east-2.elb.amazonaws.com:9100 EKS) [Inbox](#)

R Grafana Alert for BankApp <[REDACTED]@gmail.com>
to me ▾ 3:44PM (1 minute ago) [☆](#) [😊](#) [🖨️](#) [🔗](#) [⋮](#)



📁 CPU time > zago-medderma

🔥 1 firing instances

	Firing	zago-medderma	View alert
Values	A=16.03 B=16.03 C=1		
Labels	alertname zago-medderma grafana_fold er instance [REDACTED].us-east-2.elb.amazonaws.com:9100 job EKS		

Source Code: - <https://github.com/singhritesh85/Bank-App.git>

GitHub Repo: - <https://github.com/singhritesh85/DevOps-Project-BankApplication-BlueGreen-Deployment-MultiCloud.git>

Helm Chart: - <https://github.com/singhritesh85/helm-repo-for-Blue-Green-Deployment.git>

<https://github.com/singhritesh85/helm-repo-for-bitnami.git>

Terraform Script: - <https://github.com/singhritesh85/DevOps-Project-BankApplication-BlueGreen-Deployment-MultiCloud.git>

Reference: - <https://github.com/Goldencat98/Bank-App.git>

Ritesh Kumar Singh