

DevOps-Project-BankApp-CICD-Monitoring-LogAggregartion-Multicloud-Multibranch-Pipeline-with-Webhook (AWSandAzure)



By Ritesh Kumar Singh

Email Address: - riteshkumarsingh9559@gmail.com

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

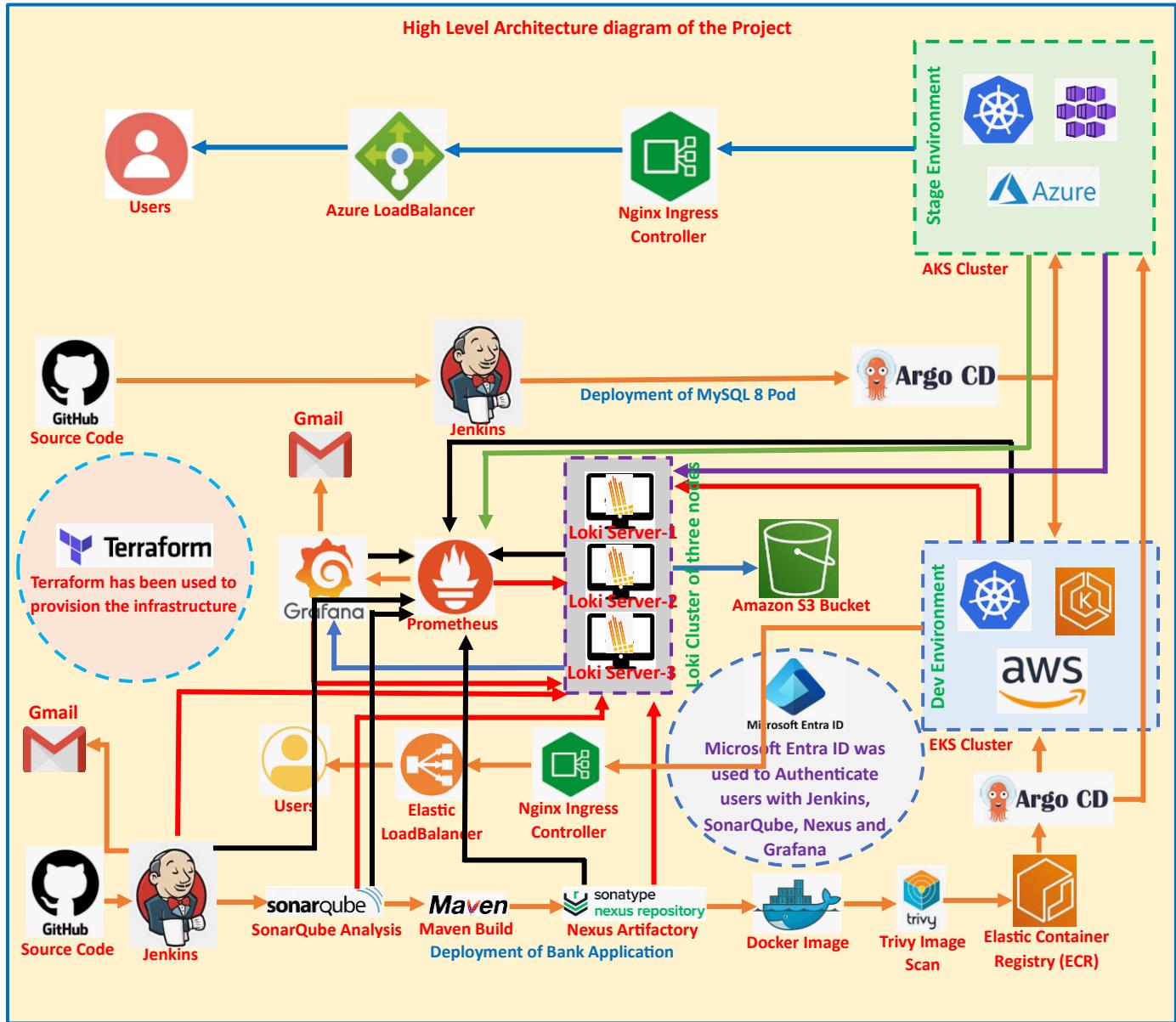
GitHub: - <https://github.com/singhritesh85>



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

DevOps-Project-BankApp-CI-CD-Monitoring-LogAggregation-Multibranch-Pipeline-with-Webhook-Multicloud

Module-1: [Non-Production Environment Dev and Stage]



In the above diagram the black arrow (→) indicated the metrics exported by node exporter to the prometheus and red arrow (→) indicates logs extracted by the promtail to loki.

This DevOps Project deals with creation of Infrastructure using Terraform and setup of CI/CD 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 and AKS 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-multibranch.git>. 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 and AKS 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.

In this Project I had created multibranch Jenkins pipeline for non-production environment (dev and stage environment). The Dev Environment is available on EKS Cluster and Stage environment is available on AKS Cluster as shown in the diagram drawn above.

For this project the production environment is completely different and which was present of AKS Cluster. For deployment to production environment, I used Azure DevOps CI/CD which I explained in **module-2**.

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.

Domain	Status	Renewal status	Type	CNAME name
*.singhritesh85.com	Success	-	CNAME	f.singhritesh85.m.

The screenshot shows the Azure DNS Recordsets page for the domain `singhritesh85.com`. On the left, there's a sidebar with options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Settings, DNS Management, and Recordsets (which is selected). The main area displays a table of records:

Name	Type
@	NS
@	SOA
<code>f[REDACTED]3</code>	CNAME

A message at the top states: "A record set is a collection of records in a zone that have the same name. They all share the same type and TTL values. All records in a record set must have the same type. If you don't see what you're looking for, try loading more." Below the table is a "Metadata" section with fields for Name, Type, Alias record set, TTL, TTL unit, and Alias.

Then I wait for around 20-25 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`

The screenshot shows the Azure Custom Domain settings page for `singhritesh85.com`. It includes a note: "To use `singhritesh85.com` with your Microsoft Entra tenant, create a new TXT record with your domain name registrar using the info below." Below this are fields for Record type (set to TXT), Alias or host name (@), Destination or points to address (`[REDACTED]`), and TTL (set to 3600). A tooltip "Copied" appears over the destination field. At the bottom, there's a "Verify" button.

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

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

Add Delete Get 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

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**.

I installed terraform on Alma Linux2 Azure VM and State file was kept in the S3 Bucket and state lock had been achieved using the AWS DynamoDB. Before running the terraform script I ran the shell script present in the directory **terraform-differentbranch-multibranch-multicloud** as shown in the screenshot attached below. This shell script will install the aws cli, kubectl and helm.

```
[root@[REDACTED] terraform-multi-kubernetes-cluster-multicloud]# ./initial-setup.sh
```

Then I logged-out and login again ran the command aws configure as shown in the screenshot attached below. As it is an important step to Authenticate and Authorize the user before creating resources using terraform in the AWS Account.

```
[root@[REDACTED] terraform-multi-kubernetes-cluster-multicloud]# cd main/
[root@[REDACTED] main]# aws configure
AWS Access Key ID [*****]: [REDACTED]
AWS Secret Access Key [*****]: [REDACTED]
Default region name [REDACTED]: [REDACTED]
Default output format [REDACTED]: [REDACTED]
[root@[REDACTED] main]#
```

Then I installed Azure CLI and authenticated and authorize the user as shown in the screenshot attached below.

```
[root@[REDACTED] main]# yum install -y https://packages.microsoft.com/config/rhel/8/packages-microsoft-prod.rpm
[root@[REDACTED] main]# yum install azure-cli -y
[root@[REDACTED] main]# az login
To sign in, use a web browser to open the page [REDACTED] and enter the code [REDACTED] to authenticate.
```

Then you can run the below commands

terraform init -----> initializes a working directory containing configuration files and installs plugins for required providers.

terraform validate -----> verify that terraform configuration file is correct or not

terraform plan -----> Check which resources are going to be created.

Then you can run the command **terraform apply -auto-approve** -----> Finally, Create the resources.

```
module.eks_cluster.azurerm_active_directory_domain_service.entra_ds: Still creating... [1h13m40s elapsed]
module.eks_cluster.azurerm_active_directory_domain_service.entra_ds: Still creating... [1h13m50s elapsed]
module.eks_cluster.azurerm_active_directory_domain_service.entra_ds: Creation complete after 1h13m56s [id=/subscriptions/[REDACTED]/resourceGroups/aks-ds-rg/providers/Microsoft.AAD/domainServices/dexter-domainservices/initialReplicaSetId/[REDACTED]]
Apply complete! Resources: 2 added, 5 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]6.dkr.ecr.us-east-2.amazonaws.com/bankapp"
}
```

After creation of the resources a kubeconfig file was generated, I did below change in the kubeconfig file as shown in the screenshot attached below.

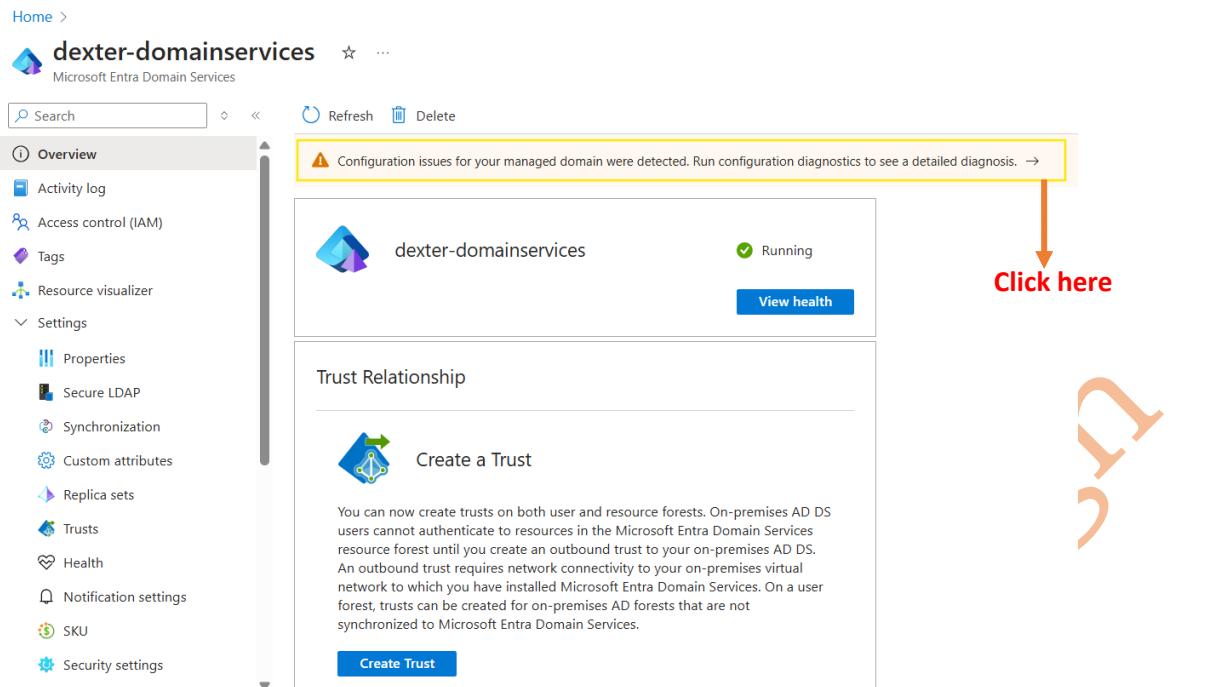
```
server: https://aks-cluster-dns-[REDACTED].eastus-[REDACTED].io:443
name: aks-cluster
- cluster:
  - certificate-authority-data: [REDACTED]

server: https://[REDACTED].us-east-2.eks.amazonaws.com
name: eks-demo-cluster-dev
contexts:
- context:
  - cluster: aks-cluster
    user: clusterUser_aks-rg_eks-cluster
    name: aks-cluster
  - context:
    - cluster: eks-demo-cluster-dev
      user: arn:aws:eks:us-east-2:02-[REDACTED]:cluster/eks-demo-cluster-dev
      name: eks-demo-cluster-dev
    current-context: aks-cluster
    kind: Config
    preferences: {}
  users:
  - name: clusterUser_aks-rg_aks-cluster
    user:
      client-certificate-data: [REDACTED]

@@@
"~/.kube/config" 42L, 11795C
20,28 19%
```

It is my suggestion before any change in the kubeconfig file take a backup of the original kubeconfig file.

This terraform script took total 1 hour 15 minutes in all to get 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.



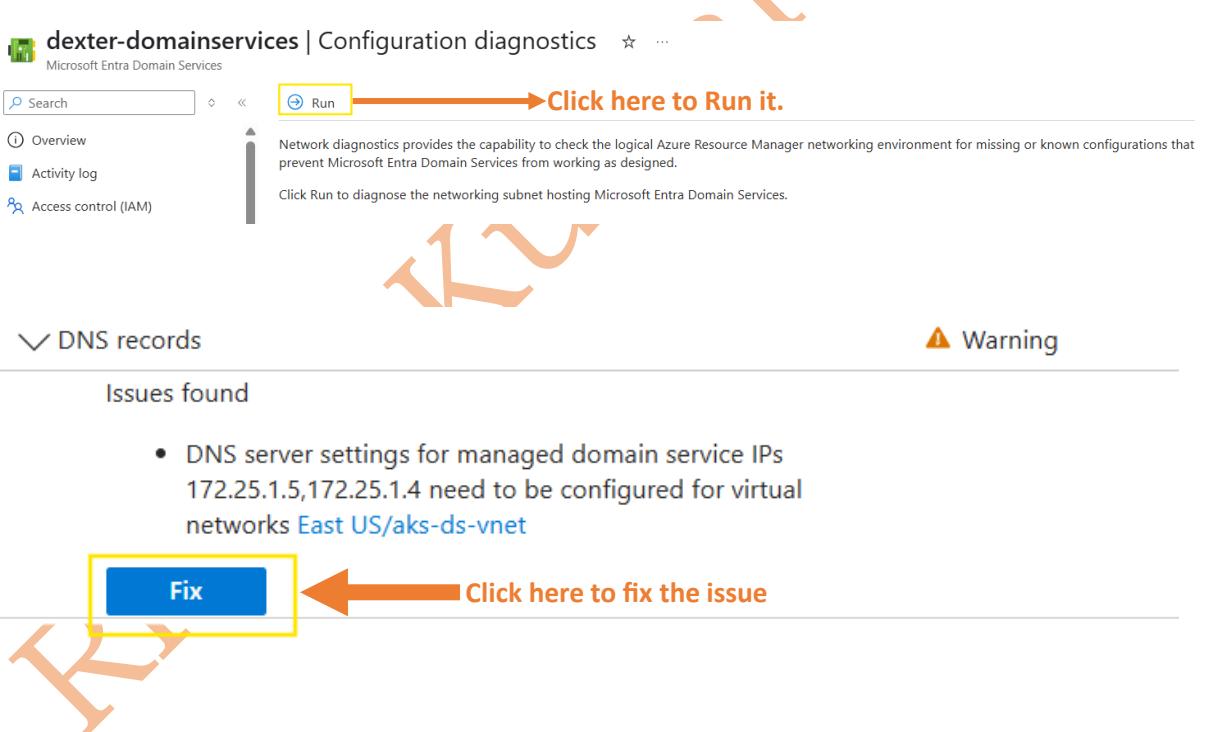
The screenshot shows the Microsoft Entra Domain Services Overview page for the domain "dexter-domainservices". A yellow box highlights a warning message: "Configuration issues for your managed domain were detected. Run configuration diagnostics to see a detailed diagnosis." An orange arrow points from this message to a red button labeled "Click here".

Trust Relationship

Create a Trust

You can now create trusts on both user and resource forests. On-premises AD DS users cannot authenticate to resources in the Microsoft Entra Domain Services resource forest until you create an outbound trust to your on-premises AD DS. An outbound trust requires network connectivity to your on-premises virtual network to which you have installed Microsoft Entra Domain Services. On a user forest, trusts can be created for on-premises AD forests that are not synchronized to Microsoft Entra Domain Services.

Create Trust



The screenshot shows the Configuration diagnostics page for the domain "dexter-domainservices". A yellow box highlights the "Run" button. An orange arrow points from this button to a red button labeled "Click here to Run it." Another orange arrow points from the "Run" button to another red button labeled "Click here to fix the issue".

DNS records

Issues found

- DNS server settings for managed domain service IPs 172.25.1.5, 172.25.1.4 need to be configured for virtual networks [East US/aks-ds-vnet](#)

Fix

Warning

DNS records

X

Issues found

- DNS server settings for managed domain service IPs 172.25.1.5,172.25.1.4 need to be configured for virtual networks East US/aks-ds-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 172.25.1.5,172.25.1.4 on East US/aks-ds-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.

A screenshot of the 'dexter-domainservices | Properties' page for Microsoft Entra Domain Services. The left sidebar shows navigation options like Overview, Activity log, Access control (IAM), Tags, Resource visualizer, and Settings. Under Settings, the 'Properties' tab is selected. The main pane displays various properties: DNS domain name (singhritesh85.com), Locations (East US), Virtual Networks/Subnets (East US/aks-ds-vnet/aks-ds-vnet/default), Network security groups (East US/domain-services-nsg), IP addresses (East US/172.25.1.5 172.25.1.4), Secure LDAP (Enabled), Secure LDAP external IP addresses (East US/52. [REDACTED].244), Synchronization (All), Admin group (AAD DC Administrators), and Health (All). A yellow box highlights the IP address entry 'East US/172.25.1.5 172.25.1.4'.

aks-ds-vnet | DNS servers

DNS servers

- Default (Azure-provided)
- Custom

IP Address

172.25.1.5
172.25.1.4

Add DNS server

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

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

Microsoft Entra Domain Services has multiple security settings that can be used to harden the domain service. When choosing to enable or disable a security setting, it is important to first understand the impact on the workloads using the domain service. Learn more about Microsoft Entra Domain Services 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

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 image consists of three vertically stacked screenshots from Microsoft's account management interface.

Screenshot 1: Apps dashboard (myapps.microsoft.com)

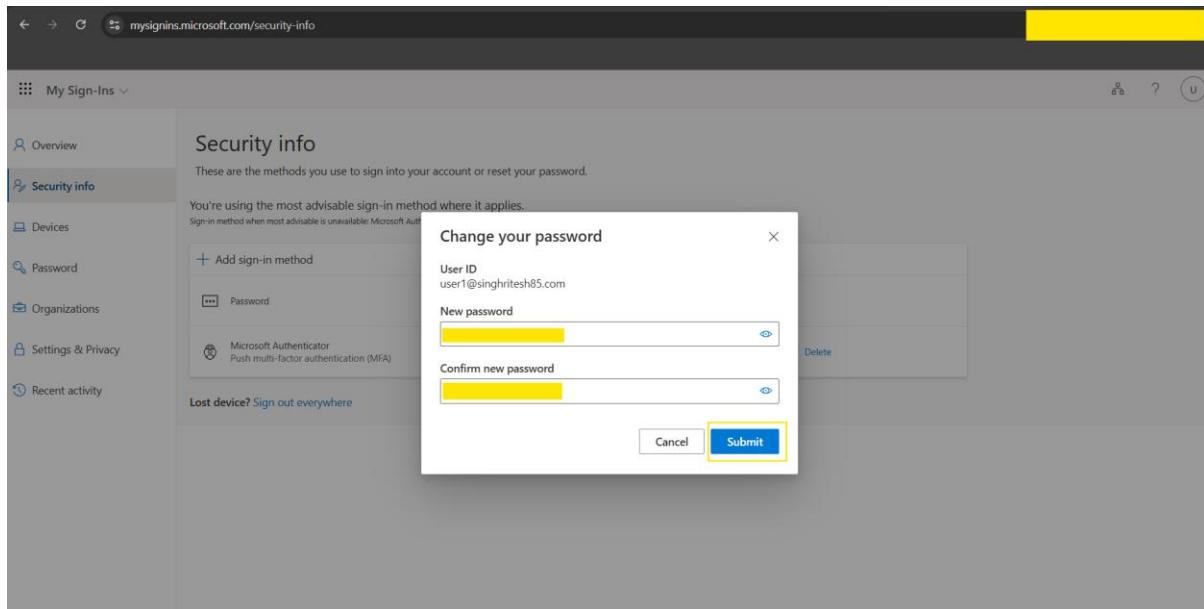
This screenshot shows the "Apps dashboard" with a navigation bar at the top. The "My Apps" section is selected. A red arrow points to the "user1" profile icon in the top right corner, which includes options like "View account" and "Switch organization".

Screenshot 2: My Account (myaccount.microsoft.com/?ref=MeControl)

This screenshot shows the "My Account" page. The left sidebar has "Security info" highlighted with a red arrow. The main area shows sections for "Security info", "Devices", "Organizations", and "Password".

Screenshot 3: Security info (mysignins.microsoft.com/security-info)

This screenshot shows the "Security info" page. The left sidebar has "Security info" selected. The main area displays sign-in methods: "Password" and "Microsoft Authenticator". A red arrow points to the "Change" button next to the "Last updated" field for the password entry.



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.

The screenshot shows the Azure portal interface. At the top, there's a search bar and a 'Create load balancer' button. Below it, a table lists 'Load balancers (5)' with columns for Name, DNS name, State, VPC ID, Availability Zones, Type, and Date create. The entries are: jenkins-ms (DNS name: jenkins-ms-[REDACTED].us-ea..., State: Active, VPC ID: vpc-[REDACTED], Availability Zones: 3 Availability Zones, Type: application, Date create: 2025); Grafana (DNS name: Grafana-[REDACTED].us-east-..., State: Active, VPC ID: vpc-[REDACTED], Availability Zones: 3 Availability Zones, Type: application, Date create: 2025); SonarQube (DNS name: SonarQube-[REDACTED].us..., State: Active, VPC ID: vpc-[REDACTED], Availability Zones: 3 Availability Zones, Type: application, Date create: 2025); and Loki (DNS name: Loki-[REDACTED].us-east-2..., State: Active, VPC ID: vpc-[REDACTED], Availability Zones: 3 Availability Zones, Type: application, Date create: 2025).

Below this, a 'singhritesh85.com | Recordssets' blade is open. It shows a list of record sets: '@' (Type: NS) and '@' (Type: SOA). To the right, a 'Add record set' dialog box is open for 'sonarqube'. The 'Name' field is 'sonarqube' and the 'Type' field is 'CNAME – Link your subdomain to another record'. The 'Alias' field contains 'SonarQube-[REDACTED].us-east-2.elb.amazonaws.com'. The 'Add' button is highlighted with a yellow box.

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.

The screenshot shows the AWS Route 53 console under the 'singhritesh85.com' DNS zone. The left sidebar shows navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Settings, DNS Management, and Records. The 'Records' tab is selected. A search bar at the top left is empty. The main area displays five CNAME records:

Name	Type	TTL	Value
f . ███████████	CNAME	3600	███████████
grafana	CNAME	3600	Grafana- ███████████ .us-east-2.elb.amazonaws.com
jenkins-ms	CNAME	3600	jenkins-ms- ███████████ .us-east-2.elb.amazonaws.com
loki	CNAME	3600	Loki- ███████████ .us-east-2.elb.amazonaws.com
nexus	CNAME	3600	Nexus-ALB- ███████████ .us-east-2.elb.amazonaws.com
sonarqube	CNAME	3600	SonarQube- ███████████ .us-east-2.elb.amazonaws.com

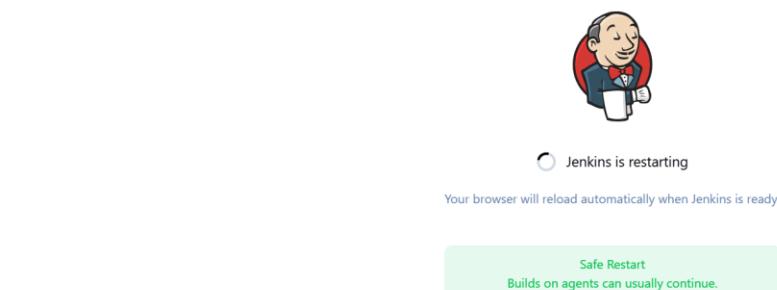
It is possible to monitor Jenkins Job using Prometheus and Grafana and for that I installed **Prometheus Metrics** plugin in Jenkins, after its installation I restarted the Jenkins as shown in the screenshot attached below.

The screenshot shows the Jenkins management interface under the 'Manage Jenkins' section and then 'Plugins'. The left sidebar has links for Updates, Available plugins (which is selected), Installed plugins, Advanced settings, and Download progress. The main area shows the 'Available plugins' search results for 'prometheus metrics'. A single result is listed:

Install	Name	Released
<input checked="" type="checkbox"/>	Prometheus metrics (███████████)	2 mo 14 days ago

Below the table, it says: 'Jenkins Prometheus Plugin expose an endpoint (default /prometheus) with metrics where a Prometheus Server can scrape.' At the bottom right, there are links for 'REST API' and 'Jenkins 2.504.1'.

The screenshot shows the Jenkins plugin manager interface. The left sidebar has links for Updates, Available plugins, Installed plugins, Advanced settings, and Download progress. The main area lists installed plugins with status indicators: 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.), and Loading plugin extensions (Success). Below the list 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 are links for REST API and Jenkins 2.504.1.



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 screenshot shows the Grafana interface for configuring a Prometheus data source. The URL in the address bar is `grafana.singhrithesh85.com/connections/datasources/edit/aekuqlud83vuob`. The top navigation bar includes Home, Connections, Data sources, and prometheus. The main content area has tabs for Settings (selected), Dashboards, and a preview section. The Settings tab shows the data source name is "prometheus". The Connection section contains a "Prometheus server URL" input field with the value "http://10.10.4.235:9090". The Authentication section includes options for Incremental querying (beta) and Disable recording rules (beta). The Other section includes fields for Custom query parameters, HTTP method (set to POST), and Use series endpoint. A success message at the bottom states "Successfully queried the Prometheus API." with a green checkmark icon. At the bottom right are "Delete" and "Save & test" buttons.

The image contains two screenshots of the Grafana interface, both titled "Connections > Data sources > loki".

Top Screenshot (Settings Tab):

- URL: http://Loki-[REDACTED]us-east-2.elb.amazonaws.com
- Authentication method: No Authentication

Bottom Screenshot (Alerting Tab):

- Alerting: Manage alert rules for the Loki data source.
- Queries: Maximum lines: 1000
- Derived fields: + Add
- Message: Data source successfully connected.
- Buttons: Delete, Save & test (highlighted with a yellow box)

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.

The screenshot shows the Jenkins Plugin Manager interface. In the search bar at the top right, 'microsoft entra id' is typed. Below the search bar, there is a button labeled 'Install' with a yellow border. The main area displays a list of available plugins. One plugin, 'Microsoft Entra ID (previously Azure AD)', is highlighted with a yellow border. This plugin is categorized under 'Security' and 'Authentication and User Management'. It has a status of 'Released' and was last updated '1 mo 9 days ago'. At the bottom right of the screen, it says 'REST API Jenkins 2.504.1'.

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.

The screenshot shows the 'App registrations' section of the Azure Portal. A new application named 'jenkins-login' is being registered. The 'Supported account types' section is visible, showing that accounts in the organizational directory only are selected. The 'Redirect URI (optional)' field contains 'https://jenkins-ms.singhritesh85.com/securityRealm/finishLogin'. The 'Register' button is highlighted with a yellow border.

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 'Security' configuration page. Under 'Authentication', there is a checkbox for 'Disable "Keep me signed in"'. Below that is a dropdown menu for 'Security Realm' which has 'Azure Active Directory' selected. There are fields for 'Client ID' and 'Client Secret'. Under 'Authentication Type', 'Client Secret' is selected. At the bottom 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 an Azure AD registered application. Under 'Implicit grant and hybrid flows', there are two checkboxes: 'Access tokens (used for implicit flows)' and 'ID tokens (used for implicit and hybrid flows)', with the latter being checked. Below this is a section for 'Supported account types' with radio buttons for 'Accounts in this organizational directory only (Default Directory only - Single tenant)' and 'Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)'. At the bottom are 'Save' and 'Discard' buttons, with 'Save' highlighted by a yellow box.

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

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](#)

Add a permission Grant admin consent for Default Directory

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

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 about permissions and consent](#)

Add a permission Grant admin consent for Default Directory

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			Credentials			Agent			Job			Run			View			SCM			Metrics		
	Administrator	Read	Create	Delete	Update	View	Build	Configure	Connect	Create	Delete	Discover	Move	Read	Reply	Update	Workspace	Configure	Create	Delete	Read	Tag	View	ThreadDump
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 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"/>
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 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"/>
	Azure User/group to add																							
<input type="text" value="Search for a name"/>																								
<input type="button" value="Add"/>																								

Markup Formatter

Markup Formatter ?

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: Name (user1), Email (user1@singhritesh85.com), Type (Member), and Object ID (redacted). Below this, there is a card for the 'AAD DC Administrators' group with details: Name (AAD DC Administrators), Type (Security), and Object ID (redacted). 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' (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) and Jenkins features (Overall, Credentials, Agent, Job, Run, View, SCM, Metrics). The 'user1' row has checkboxes checked for most features, indicating elevated privileges. The 'Save' button at the bottom is 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

Build Queue
No builds in the queue.

Build Executor Status
0/2

Add description

REST API Jenkins 2.504.1

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	Homepage	Issue Tracker	Licensed under	Developed by	Action
Ansible Lint EXTERNAL ANALYZERS	2.5.1	Support for SonarQube 9.2 ...			Apache License, Version 2.0		Install
Apigee EXTERNAL ANALYZERS	3.0.2	Support for SQ 9.7+ ...			Apache License, Version 2.0		Install
Azure Active Directory (AAD) Authentication Plug-in for SonarQube INTEGRATION	1.3.2	Updates commons-text to fix CVE-2022-42889. ...			The MIT License (MIT)	Crédit Mutuel Arkéa	Install
CVS INTEGRATION	1.1.1	Fix classnotfound error ...			GNU GPL, Version 3		Install
Checkstyle EXTERNAL ANALYSERS	10.23.0	Upgrade to Checkstyle 10.23.0 ...			Apache License, Version 2.0		Install
Chinese Pack LOCALIZATION	9.9	Support SonarQube 9.9 ...			GNU GPL, Version 3	Mossile	Install
Clover COVERAGE	4.1	Makes the plugin compatible with SQ 7.9 ...			GNU GPL, Version 3		Install
Codehawk Java EXTERNAL ANALYSERS	1.6	Added new rules ...					Install

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

SonarQube Marketplace screenshot showing various plugins available for installation. Plugins listed include:

- Apigee EXTERNAL ANALYZERS**: Adds XML rules test Apigee proxies. 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.
- Checkstyle EXTERNAL ANALYSERS**: Provide Checkstyle rules for Java projects. Version 10.23.0, Upgrade to Checkstyle 10.23.0.
- Chinese Pack LOCALIZATION**: SonarQube Chinese Pack. Version 9.9, Support SonarQube 9.9.
- Clover COVERAGE**: Provides the ability to compute coverage with Clover. Version 4.1, Makes the plugin compatible with SQ 7.9.
- Codehawk Java EXTERNAL ANALYSERS**: Analyze Java code smell. Version 1.6, Added new rules.

Each plugin entry includes a brief description, version number, license information, and an "Install" button.

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

SonarQube General Settings screenshot showing the "Server base URL" configuration. The URL `https://sonarqube.singhritesh85.com/` is entered into the input field, which is highlighted with a yellow box. The "Save" button is visible at the bottom right of the dialog.

The left sidebar shows other settings categories like Analysis Scope, Authentication, Azure Active Directory, DevOps Platform Integrations, External Analyzers, Issues, and Languages. The "General" category is currently selected and highlighted with a yellow box.

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 [\[?\]](#)

Register

The screenshot shows the Azure App Registrations overview page. In the 'Essentials' section, the 'Display name' is 'SonarQube'. Below it, the 'Client ID' and 'Tenant ID' fields are highlighted with red arrows pointing to them. The 'Client ID' field contains a long string of characters, and the 'Tenant ID' field also contains a long string of characters. Other visible details include 'Object ID', 'Directory (tenant) ID', 'Client credentials' (0 certificate_1 secret), 'Redirect URIs' (1 web_0 spa_0 public client), 'Application ID URI' (Add an Application ID URI), and 'Managed application in ...' (SonarQube). A sidebar on the left lists various management options like Quickstart, Integration assistant, Diagnose and solve problems, and Authentication. A large orange watermark 'Ritesh' is diagonally across the page.

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.

sonarqube.singhritesh85.com/admin/settings?category=aad

The screenshot shows the SonarQube Administration General Settings page. The left sidebar lists various settings categories. The main panel displays the '1. General' section under 'Azure Active Directory'. It includes a note about enabling AAD authentication, a 'Enabled' toggle switch (set to 'Enabled'), and fields for 'Client ID' and 'Client Secret'. The 'Client ID' field contains 'Key: sonar.auth.aad.clientId.secured'. The 'Client Secret' field contains 'Key: sonar.auth.aad.clientSecret.secured'. Below these, there's a 'Tenant ID' field with 'Key: sonar.auth.aad.tenantId.secured'. At the bottom, there's a 'Allow users to sign-up' toggle switch (set to 'Enabled').

sonarqube.singhritesh85.com/admin/settings?category=aad

This screenshot is identical to the one above, showing the 'General' settings for Azure Active Directory. The 'Client Secret' field is highlighted with a yellow box.

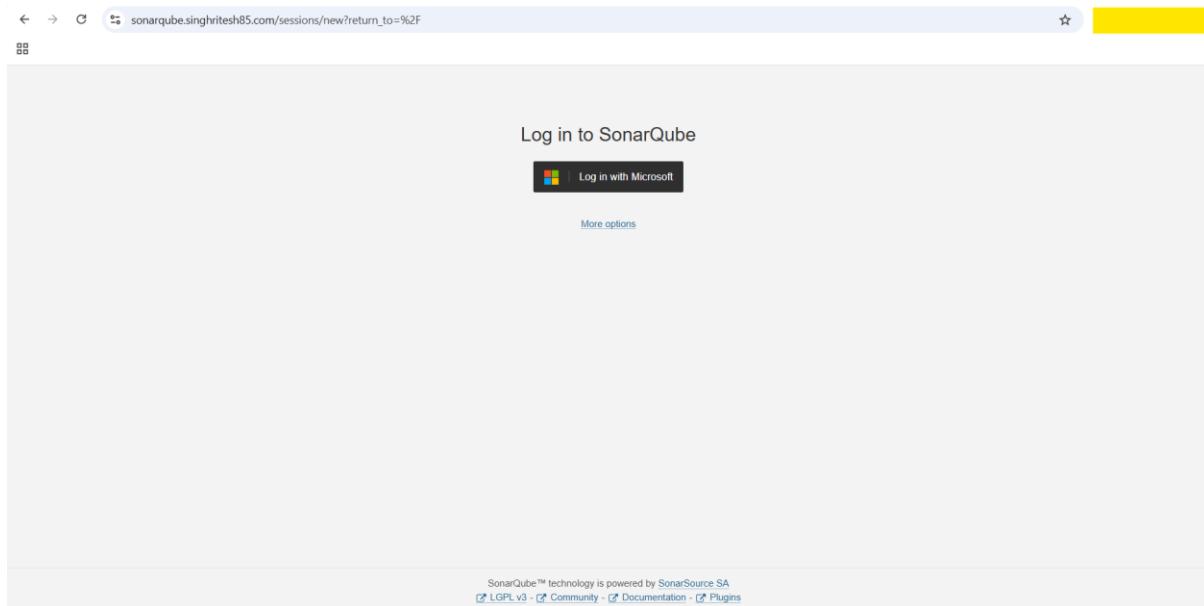
sonarqube.singhritesh85.com/admin/settings?category=aad

This screenshot is identical to the previous ones, showing the 'General' settings for Azure Active Directory. The 'Client Secret' field is highlighted with a yellow box.

sonarqube.singhritesh85.com/admin/settings?category=aad

The screenshot shows the 'General' settings for Azure Active Directory. The 'Login generation strategy' section is highlighted with a yellow box. It contains a dropdown menu set to 'Same as Azure AD login' and a note that 'Logins will be set the following way'. Below it, there's a 'Reset' button and a 'Default: Unique' note. The 'Directory Location' section is also highlighted with a yellow box, showing a dropdown menu set to 'Azure AD (Global)' with '(default)' below it. The 'Enable Client Credential Flow' section at the bottom is partially visible.

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.

All	Users	Groups	Search for users or groups...	Administrator System	Administrator	Execute Analysis	Create
				<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-administrators System administrators		<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Projects
		sonar-users Every authenticated user automatically belongs to this group		<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Projects
		user1 user1 [REDACTED] user1@singhrithesh85.com		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Quality Gates <input checked="" type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input checked="" type="checkbox"/> Projects
		Anyone DEPRECATED Anybody who browses the application belongs to this group. If authentication is not enforced, assigned permissions also apply to non-authenticated users.		<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]		<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.

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](#) ↗

Register

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? →

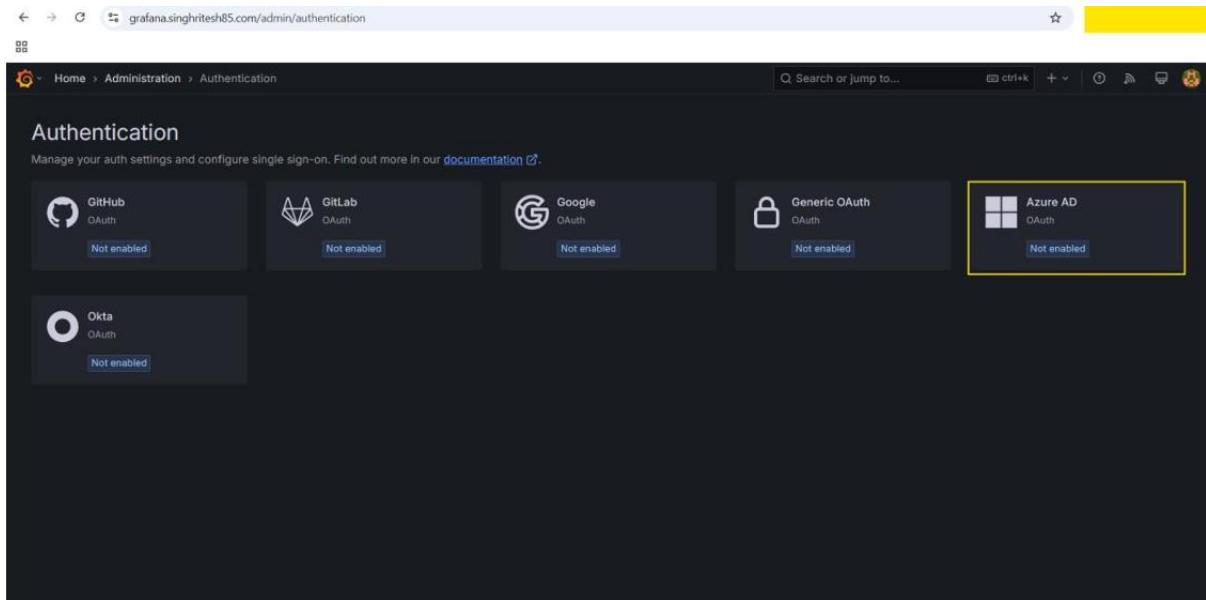
Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

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

Certificates (0)	<u>Client secrets (1)</u>	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 <table border="1"> <thead> <tr> <th>Description</th> <th>Expires</th> <th>Value</th> <th>Secret ID</th> </tr> </thead> <tbody> <tr> <td>demo</td> <td>12/2025</td> <td>XXXXXXXXXX</td> <td>█</td> </tr> </tbody> </table>			Description	Expires	Value	Secret ID	demo	12/2025	XXXXXXXXXX	█
Description	Expires	Value	Secret ID							
demo	12/2025	XXXXXXXXXX	█							

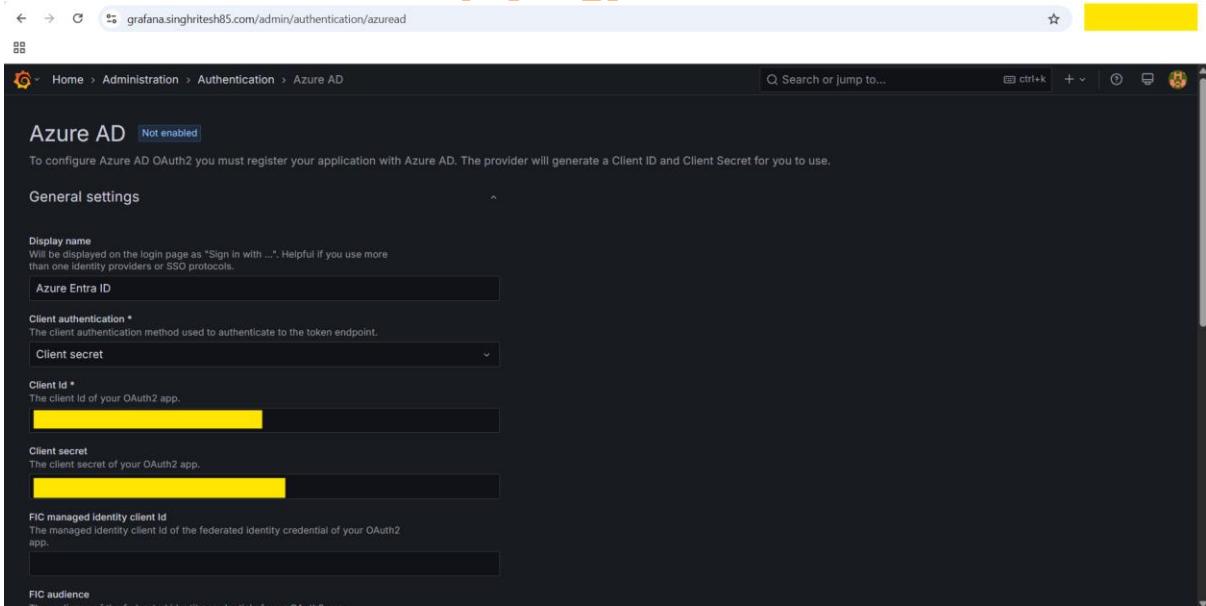
█ Client Secret

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**).



Auth URL *
The authorization endpoint of your OAuth2 provider.
`https://login.microsoftonline.com/[REDACTED]/oauth2/v2.0`

Token URL *
The token endpoint of your OAuth2 provider.
`https://login.microsoftonline.com/[REDACTED]/oauth2/v2.0`

Allow sign up
If not enabled, only existing Grafana users can log in using OAuth.

Auto login
Log in automatically, skipping the login screen.

Sign out redirect URL
The URL to redirect the user to after signing out from Grafana.

User mapping

Role attribute strict mode
If enabled, denies user login if the Grafana role cannot be extracted using Role attribute path.

Organization mapping
List of <GroupId><OrgIdOrName><Role>* mappings.
Enter mappings (my-team:1:Viewer...) and press Enter to add

Allow assign Grafana admin
If enabled, it will automatically sync the Grafana server administrator role.

Skip organization role sync
Prevent synchronizing users' organization roles from your IdP.

Extra security measures

Save and enable **Save** **Discard**

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

Manage your auth settings and configure single sign-on. Find out more in our [documentation](#).

GitHub OAuth Not enabled	GitLab OAuth Not enabled	Google OAuth Not enabled	Generic OAuth Not enabled
Okta OAuth Not enabled	Azure AD OAuth Enabled		

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

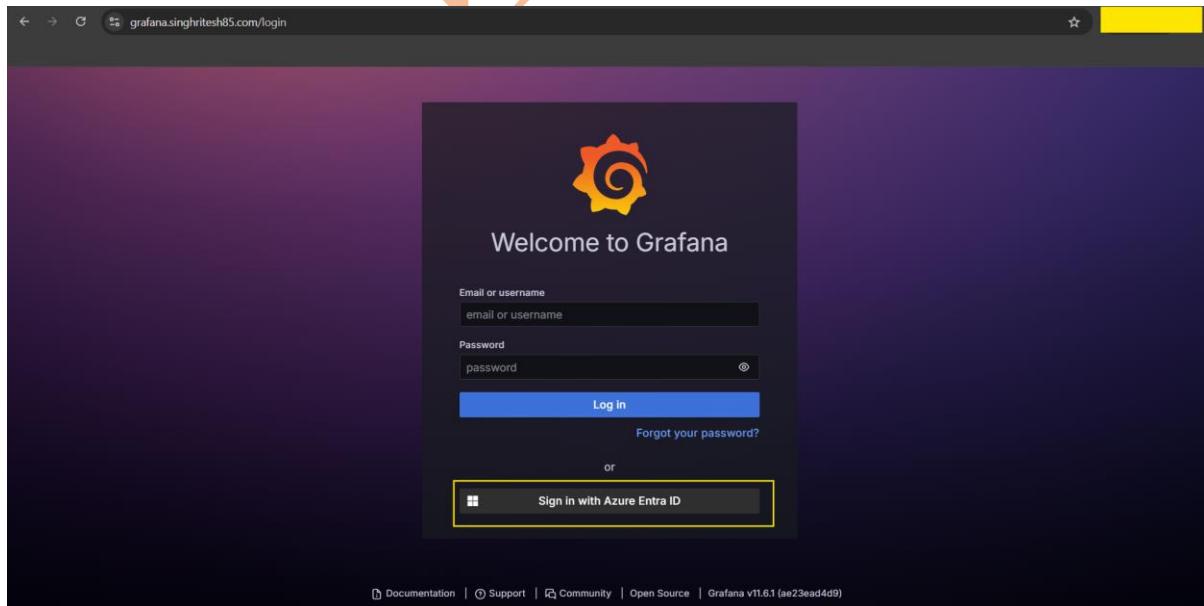
# The full public facing url you use in browser, used for redirects and emails
# If you use reverse proxy and sub path specify full url (with sub path)
[root_url = https://grafana.singhrites85.com]      #%(protocol)s://%(domain)s:%(http_port)s

# 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 2025-01-28 10:45:12 UTC; 7s ago
    Docs: http://docs.grafana.org
   Main PID: 4864 (grafana)
      Tasks: 1 (since Sun 2025-01-28 10:45:12 UTC)
```

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.

The top screenshot shows the 'Users' page in Grafana's Admin section. It lists three users: 'admin' (Login: admin, Email: admin@localhost, Name: Ritesh, Last active: 4 minutes, Origin: AzureAD), 'Ritesh' (Login: [redacted], Email: [redacted], Name: Ritesh, Last active: 1 minute, Origin: AzureAD), and 'user1@singhrithesh85.com' (Login: user1@singhrithesh85.com, Email: user1@singhrithesh85.com, Name: user1, Last active: < 1 minute, Origin: AzureAD). The bottom screenshot shows the detailed view for 'user1@singhrithesh85.com'. It includes sections for User information (Numerical identifier: 3, Name: user1, Email: user1@singhrithesh85.com, Username: user1@singhrithesh85.com, Password: *****), Permissions (Grafana Admin: Yes), and Organizations (Main Org: Admin). Buttons for 'Delete user' and 'Disable user' are visible.

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.

dexter-domainservices | Properties star ...

Microsoft Entra Domain Services

Search Search

- Activity log
- Access control (IAM)
- Tags
- Resource visualizer
- Settings

Properties

- Secure LDAP
- Synchronization
- Custom attributes
- Replica sets
- Trusts
- Health
- Notification settings
- SKU
- Security settings
- Locks

DNS domain name
singhritesh85.com

Locations
East US

Virtual Networks/Subnets
[East US/aks-ds-vnet/aks-ds-vnet/default](#)

Network security groups
[East US/domain-services-nsg](#)

IP addresses
East US/172.25.1.5 172.25.1.4

Secure LDAP
Enabled

Secure LDAP external IP addresses
East US/52.████.244

Synchronization
All

Admin group
[AAD DC Administrators](#)

Add record set

singhritesh85.com

Name .singhritesh85.com

Type

Alias record set

TTL *

TTL unit

IP address

Delete

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@... ~]# 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 two instances of the Sonatype Nexus Repository interface. The top instance displays the 'Realms' page under the 'Administration > Security' menu. It shows two lists: 'Available' and 'Active'. In the 'Available' list, there are several realms including 'Conan Bearer Token Realm', 'Default Role Realm', 'Docker Bearer Token Realm', 'npm Bearer Token Realm', 'NuGet API-Key Realm', and 'Rut Auth Realm'. The 'LDAP Realm' is highlighted with a yellow box and an orange arrow pointing to it. In the 'Active' list, only the 'Local Authenticating R...' realm is listed. The bottom instance shows the result of the transfer. The 'Available' list now has 6 items available, and the 'Active' list now has 2 items transferred, with the 'LDAP Realm' included. The 'Save' button at the bottom right is highlighted with a yellow box and an orange arrow.

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.

The screenshot displays two consecutive steps in the process of creating an LDAP connection in Sonatype Nexus Repository.

Top Window: Certificate Details

- Name:** LDAPS-Connection
- LDAP server address:** 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:** dc=singhritesh85,dc=com
- Authentication method:** Select an 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.
- Certificate:** Issued on: [redacted], Valid until: [redacted], Fingerprint: [redacted]
- A note at the bottom states: "This certificate was retrieved over an untrusted connection. Always verify the details before adding it."
- Add certificate to truststore** button (highlighted in yellow)

Bottom Window: Create LDAP Connection

- Name:** LDAPS-Connection
- LDAP server address:** 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:** dc=singhritesh85,dc=com
- Authentication method:** Select an 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** and **Cancel** buttons
- Verify connection** button (highlighted in yellow)
- A green success message: "SSL Certificate created: *singhritesh85.com"

The screenshot shows two consecutive steps in the 'Create LDAP Connection' wizard:

Step 1: Create LDAP Connection

- LDAP server address:** Idaps://ldaps.singhritesh85.com:636
- Use the Nexus Repository truststore:** Checked, with a link to 'View certificate'.
- Search base DN:** dc=singhritesh85,dc=com
- Authentication method:** Simple Authentication
- Username or DN:** user@singhritesh85.com
- Password:** [REDACTED]
- Connection rules:** Wait: 30 seconds before timeout. Retry after: 300 seconds, max of 3 failed attempts.

Step 2: Choose Users and Groups

- Configuration template:** Select a template
- User relative DN:** OU=AADDUsers
- User subtree:** Checked: Are users located in structures below the user base DN?
- Object class:** user
- 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:**

Both steps show a green success message at the top right: 'Connection to LDAP server verified: Idaps://ldaps.singhritesh85.com:636'.

The screenshot shows the 'Create LDAP Connection' configuration page in the Sonatype Nexus Repository interface. The 'Group type' dropdown is highlighted with a yellow box and set to 'Dynamic Groups'. A green success message box in the top right corner displays the text: '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.

Screenshot of the Sonatype Nexus Repository Administration interface. The URL is nexus.singhritesh85.com/. The left sidebar under 'Administration' has 'Users' selected. The main area shows a table titled 'Manage users' with the following data:

User ID	Realm	First name	Last name	Email	Status
[REDACTED]	LDAP	Ritesh	[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.

Screenshot of the Sonatype Nexus Repository Administration interface showing the 'user1' user profile. The URL is nexus.singhritesh85.com/#admin/security/users/user1. The left sidebar is 'Administration' with 'Users' selected. The right panel shows the 'user1' profile with the 'Roles' tab selected. The 'Granted' section contains the role 'nx-admin'.

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

The screenshot shows the Sonatype Nexus Repository OSS 3.68.1-02 interface. The top navigation bar includes links for 'Release Notes', 'Documentation', and 'Community'. A 'Notification Center' modal is displayed, prompting the user to sign in with 'user1' and a password. The main content area features a 'New Formats Supported' section and a 'Latest Releases' summary for April 10, 2019.

The screenshot shows the 'Administration' section of the Nexus interface, specifically the 'Repository' administration page. The left sidebar lists various repository-related options. The main content area contains links and icons for managing blob stores, cleanup policies, proprietary repositories, standard repositories, and content selectors.

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

The screenshot shows the 'Administration' section of the Nexus interface, specifically the 'Repositories' administration page. The left sidebar shows 'Repositories' selected. The main content area lists various repository recipes, including apt, bower, cocoapods, conan, conda, docker, glibfs, go, helm, maven2, and npm. Two specific entries, 'maven2 (hosted)' and 'maven2 (proxy)', are highlighted with a yellow box.

<nexus.singhritesh85.com/#admin/repository/repositories>

Sonatype Nexus Repository OSS 3.68.1-02 Administration Repository Repositories Search components

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

Name: A unique identifier for this repository **maven-release**

Online: If checked, the repository accepts incoming requests

Maven 2

Version policy: What type of artifacts does this repository store? **Release**

Layout policy: Validate that all paths are Maven artifact or metadata paths **Strict**

Content Disposition: Add Content-Disposition header as 'Attachment' to disable some content from being inline in a browser. **Inline**

Storage

Blob store: Blob store used to store repository contents **default**

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

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

<nexus.singhritesh85.com/#admin/repository/repositories>

Sonatype Nexus Repository OSS 3.68.1-02 Administration Repository Repositories Search components

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 type="button" value="copy"/>	Analyze	<input type="button" value="analyze"/>
maven-public	group	maven2	default	Online	<input type="button" value="copy"/>	<input type="button" value="analyze"/>	<input type="button" value="analyze"/>
maven-release	hosted	maven2	default	Online	<input type="button" value="copy"/>	<input type="button" value="analyze"/>	<input type="button" value="analyze"/>
maven-releases	hosted	maven2	default	Online	<input type="button" value="copy"/>	<input type="button" value="analyze"/>	<input type="button" value="analyze"/>
maven-snapshots	hosted	maven2	default	Online	<input type="button" value="copy"/>	<input type="button" value="analyze"/>	<input type="button" value="analyze"/>
nuget-group	group	nuget	default	Online	<input type="button" value="copy"/>	<input type="button" value="analyze"/>	<input type="button" value="analyze"/>
nuget-hosted	hosted	nuget	default	Online	<input type="button" value="copy"/>	<input type="button" value="analyze"/>	<input type="button" value="analyze"/>
nuget.org-proxy	proxy	nuget	default	Online - Remote Availab...	<input type="button" value="copy"/>	Analyze	<input type="button" value="analyze"/>

Filter

The screenshot shows two instances of the Sonatype Nexus Repository interface. The top instance displays a list of repositories under the 'Repositories' section, with 'maven2 (hosted)' highlighted. The bottom instance shows a 'Create Repository' form for 'maven2 (hosted)'. The 'Name' field is set to 'maven-snapshot', and the 'Version policy' is set to 'Snapshot'. Other fields include 'Online' (checked), 'Layout policy' (Strict), 'Content Disposition' (Inline), and 'Storage' (Blob store: default). A large watermark reading 'Ritesh' is diagonally across the bottom left of the interface.

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...		Analyze	
maven-public	group	maven2	default	Online			
maven-release	hosted	maven2	default	Online			
maven-releases	hosted	maven2	default	Online			
maven-snapshot	hosted	maven2	default	Online			
maven-snapshots	hosted	maven2	default	Online			
nuget-group	group	nuget	default	Online			
nuget-hosted	hosted	nuget	default	Online			
nuget.org-proxy	proxy	nuget	default	Online - Remote Availa...		Analyze	

Installation of Nginx Ingress Controller in EKS Cluster and AKS 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-2:02XXXXXXXXX6:certificate/XXXXXXXXXX-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-2:02XXXXXXXXX6:certificate/XXXXXXX-XXXX-XXXX-XXXX-
XXXXXXXXXX --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@REDACTED ~]# kubectl config use-context eks-demo-cluster-dev
Switched to context "eks-demo-cluster-dev".
[root@REDACTED ~]# kubectl create ns ingress-nginx
namespace/ingress-nginx created
[root@REDACTED ~]# helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
"ingress-nginx" has been added to your repositories
[root@REDACTED ~]# 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 Helm-ing!⚡
[root@REDACTED ~]# 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-2:02REDACTED:certificate/REDACTED" --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-protocol"="http" --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-type"="elb" --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@REDACTED ~]# kubectl get all -n ingress-nginx
NAME                                         READY   STATUS    RESTARTS   AGE
pod/ingress-nginx-controller-REDACTED        1/1     Running   0          2m45s

NAME                           TYPE      CLUSTER-IP       EXTERNAL-IP   PORT(S)
service/ingress-nginx-controller   LoadBalancer   10.REDACTED.196   REDACTED   80:32339/TCP
service/ingress-nginx-controller-admission   ClusterIP   10.REDACTED.16   <none>      443/TCP

NAME                               READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/ingress-nginx-controller   1/1     1           1           2m45s

NAME                         DESIRED   CURRENT   READY   AGE
replicaset.apps/ingress-nginx-controller-REDACTED  1         1         1         2m45s
```

To install Nginx Ingress controller in AKS Cluster use the commands as written below.

```
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
```

```
helm upgrade ingress-nginx ingress-nginx/ingress-nginx --namespace ingress-nginx --set controller.service.annotations."service\\.beta\\.kubernetes\\.io/azure-load-balancer-health-probe-request-path"/=/healthz --set controller.service.externalTrafficPolicy=Local
```

```
[root@yellow ~]# kubectl config use-context aks-cluster
Switched to context "aks-cluster".
[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" already exists with the same configuration, skipping
[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

[root@yellow ~]# helm upgrade ingress-nginx ingress-nginx/ingress-nginx --namespace ingress-nginx --set controller.service.annotations."service\\.beta\\.kubernetes\\.io/azure-load-balancer-health-probe-request-path"/=/healthz --set controller.service.externalTrafficPolicy=Local

[root@yellow ~]# kubectl get all -n ingress-nginx
NAME                                         READY   STATUS    RESTARTS   AGE
pod/ingress-nginx-controller-[REDACTED]      1/1     Running   0          97s
NAME                                         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/ingress-nginx-controller             LoadBalancer   10.[REDACTED].145  20.[REDACTED].52  80:30130/TCP,443:32694/TCP  98s
service/ingress-nginx-controller-admission   ClusterIP    10.[REDACTED].81    <none>        443/TCP   98s
NAME                                         READY   UP-TO-DATE  AVAILABLE   AGE
deployment.apps/ingress-nginx-controller     1/1     1           1           98s
NAME                                         DESIRED  CURRENT   READY   AGE
replicaset.apps/ingress-nginx-controller-[REDACTED]  1        1        1       98s
```

Installation of ArgoCD in EKS Cluster

In this project I used ArgoCD and the ArgoCD CLI for Deployment, 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 config use-context eks-demo-cluster-dev
Switched to context "eks-demo-cluster-dev".
[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.

```
[root@yellow ~]# kubectl get ing -n argocd --context=eks-demo-cluster-dev --watch
NAME      CLASS   HOSTS          ADDRESS                                     PORTS   AGE
minimal-ingress  nginx  argocd.singhritesh85.com [REDACTED].us-east-2.elb.amazonaws.com  80      20s
```

```
[root@XXXXXXXXXX ~]# 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.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
```

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.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
```

The entry for DNS Name corresponding to HOST **argocd.singhrithesh85.com** in Azure DNS Zone to create the record set is as shown in the screenshot attached below.

Home > singhritesh85.com

singhritesh85.com | Recordsets

DNS zone

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

A record set is a collection of records in a zone that have the same name and are the same type. You can search for records that have been loaded on this page. If you don't see what you're looking for, you can try scrolling to allow more records to load. Learn more

Fetched 9 record set(s).

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

Add or remove favorites by pressing Ctrl+Shift+F

Add Cancel Give feedback

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@REDACTED ~]# kubectl -n argocd get secret argocd-initial-admin-secret -o jsonpath='{.data.password}' | base64 -d
VT1-tqWnrrL3zGmz[root@REDACTED ~]#
[root@REDACTED ~]#
```

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

argocd.singhritesh85.com/applications

Applications

+ NEW APP SYNC APPS REFRESH APPS Search applications...

APPLICATIONS SUMMARY

No applications available to you just yet

Create new application to start managing resources in your cluster

CREATE APPLICATION

The image consists of three vertically stacked screenshots of the Argo UI, version v2.14.6+fe2a6e9.

Screenshot 1: Shows the "User Info" page. The URL is argo.cd.singhritesh85.com/user-info. The page displays "User Info" and "USER INFO" sections. It shows "Username: admin" and "Issuer: argo.cd". A yellow box highlights the "UPDATE PASSWORD" button.

Screenshot 2: Shows the "Update account password" dialog. The URL is argo.cd.singhritesh85.com/user-info?changePassword=true. The dialog has "SAVE NEW PASSWORD" and "CANCEL" buttons. It contains fields for "Current Password", "New Password", and "Confirm New Password", all of which are redacted with yellow bars.

Screenshot 3: Shows the "User Info" page again. The URL is argo.cd.singhritesh85.com/user-info?changePassword=false. It displays "User Info" and "USER INFO" sections with "Username: admin" and "Issuer: argo.cd". A green success message box at the bottom right says "Your password has been successfully updated." with a checkmark icon.

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
```

The ArgoCD CLI was installed on Jenkins Slave Node.

```
[jenkins@██████████ ~] $ curl -sSL -o argocd-linux-amd64 https://github.com/argoproj/argo-cd/releases/latest/download/argocd-linux-amd64
[jenkins@██████████ ~] $ sudo install -m 555 argocd-linux-amd64 /usr/local/bin/argocd
[jenkins@██████████ ~] $ rm argocd-linux-amd64
```

I had added the aks-cluster to the ArgoCD using the command **argocd cluster add aks-cluster** as shown in the screenshot attached below.

```
[root@██████████ ~] # argocd login argocd.singhritesh85.com --username admin --password Admin@123 --skip-test-tls --grpc-web
'admin:login' logged in successfully
Context 'argocd.singhritesh85.com' updated
[root@██████████ ~] # argocd cluster add aks-cluster
WARNING: This will create a service account 'argocd-manager' on the cluster referenced by context `aks-cluster` with full cluster level privileges. Do you want to continue [y/N]? y
INFO[0001] ServiceAccount "argocd-manager" already exists in namespace "kube-system"
INFO[0001] ClusterRole "argocd-manager-role" updated
INFO[0002] ClusterRoleBinding "argocd-manager-role-binding" updated
Cluster 'https://aks-cluster-dns-██████████.hcp.eastus.azureaks.io:443' added
```

In ArgoCD UI I went to **Settings > Clusters** and you will find the two clusters one is default cluster (EKS Cluster) and another is aks-cluster as shown in the screenshot attached below.

NAME	URL	VERSION	CONNECTION STATUS
aks-cluster	https://aks-cluster-dns-47fffk.hcp.eastus.azureaks.io:443		Unknown
in-cluster	https://kubernetes.default.svc		Unknown

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 account settings: Home, Personal info, Data & privacy, **Security**, People & sharing, Payments & subscriptions, and About. The 'Security' item is selected and highlighted with a yellow box. A dropdown menu titled 'Google Account results' also has 'App passwords' highlighted with a yellow box. To the right of the search bar, there's a green shield icon with a checkmark and a small diagram of a computer screen with a password field.

← 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.

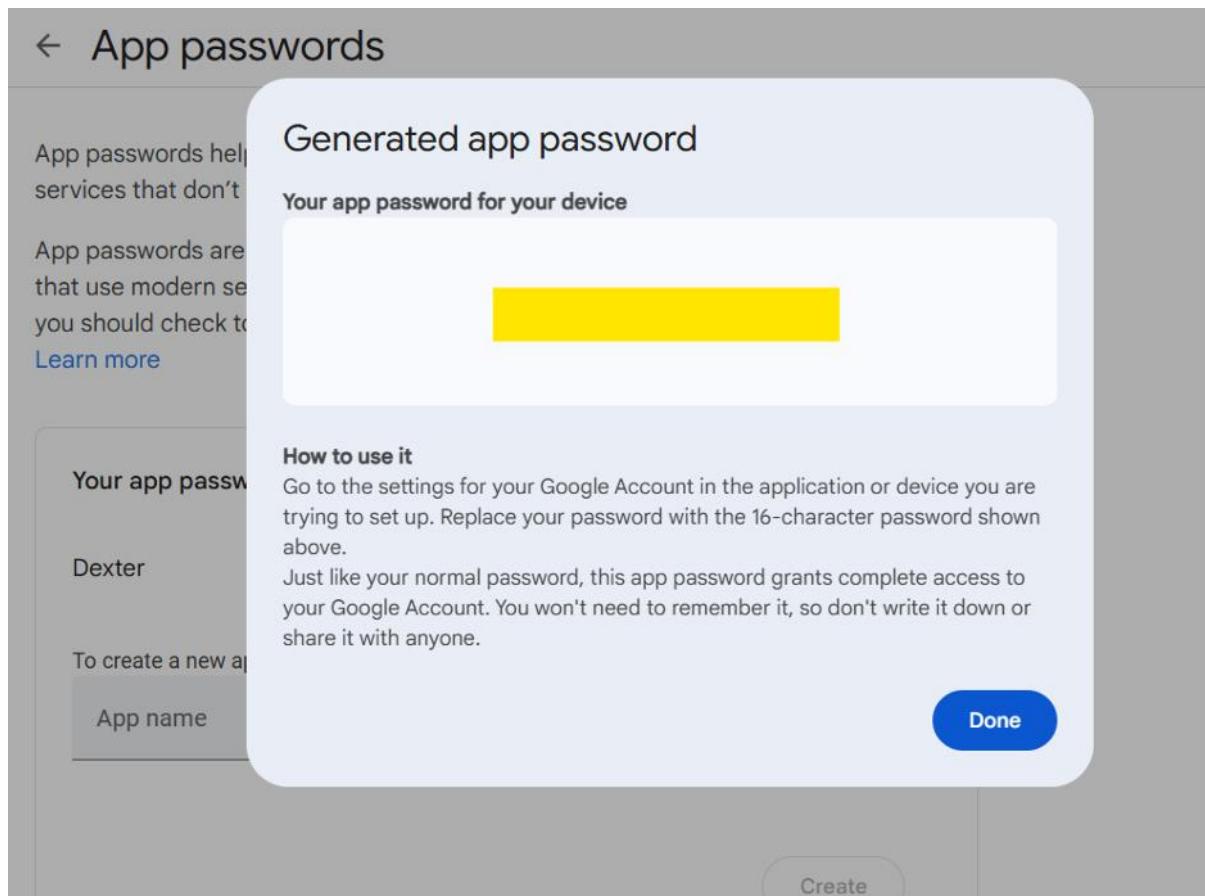
[Learn more](#)

You don't have any app passwords.

To create a new app specific password, type a name for it below...

App name
Dexter

Create



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 screenshots show the Jenkins 'Manage Jenkins > System' configuration page. The top part is the 'E-mail Notification' section, where you can set the SMTP server to 'smtp.gmail.com' and the default user email suffix to '@gmail.com'. Under 'Advanced', there are fields for 'User Name' and 'Password' with a note about security, and checkboxes for 'Use SSL' and 'Use TLS'. The bottom part is the 'Test configuration by sending test e-mail' section, which includes a recipient field, a 'Test configuration' button, and a message indicating the email was successfully sent.

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 / Emailing #####
[smtp]
enabled = true
host = smtp.gmail.com:587
user =
# If the password contains # or ; you have to wrap it with triple quotes. Ex """#password;"""
password =
;cert_file =
;key_file =
skip_verify = true
from_address =
from_name = Multibranch Pipeline Grafana Alert for Bankapp
# EHLO identity in SMTP dialog (defaults to instance_name)
;jehlo_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@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 [REDACTED] 2025-01-15 10:29:43 +0530
     Docs: http://docs.grafana.org
 Main PID: 6543 (grafana)
```

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-Multibranch-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.

The image contains three screenshots of the SonarQube web interface, specifically the 'Security' section under 'Account'.

- Screenshot 1:** Shows the 'Tokens' page. A new token named "SonarQube" is being generated. The 'Type' is set to "Global Analysis Token" and "Expires in" is set to "No expiration". A "Generate" button is highlighted.
- Screenshot 2:** Shows the same 'Tokens' page after the token has been generated. The token "SonarQube" is listed with a "Copy" button next to it, which is highlighted.
- Screenshot 3:** Shows the 'Enter a new password' page, which is part of the token creation or configuration process. It includes fields for 'Old Password' and 'New password'.

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 checkmark next to it. To the right of the plugin listing, there is a 'Released' timestamp: '3 mo 6 days ago'. At the bottom right of the page, it says 'REST API Jenkins 2.504.1'.

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 screen. 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 also contains '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.

SonarQube servers

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

Environment variables

SonarQube installations

List of SonarQube installations

Name	<input type="text" value="SonarQube-Server"/>	X
Server URL	Default is http://localhost:9000	
	<input type="text" value="https://sonarqube.singhritesh85.com"/>	
Server authentication token	SonarQube authentication token. Mandatory when anonymous access is disabled.	
	<input type="text" value="sonarqube"/>	
+ Add		
Save		Apply

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.

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration

Quality Gates [Create](#)

Sonar way BUILT-IN

This quality gate complies with Clean as You Code

This quality gate complies with the [Clean as You Code](#) methodology, so that you benefit from the most efficient approach to delivering Clean Code. It ensures that:

- No new bugs are introduced
- No new vulnerabilities are introduced
- All new security hotspots are reviewed
- New code has limited technical debt
- New code has limited duplication
- New code is properly covered by tests

Conditions

Conditions on New Code

Metric	Operator	Value
Coverage	is less than	80.0%
Duplicated Lines (%)	is greater than	3.0%
Maintainability Rating	is worse than	A (Technical debt ratio is less than 5.0%)
Reliability Rating	is worse than	A (No bugs)
Security Hotspots Reviewed	is less than	100%
Security Rating	is worse than	A (No vulnerabilities)

The screenshot shows two consecutive pages from the SonarQube interface related to Quality Gates.

Page 1: Create Quality Gate

- The URL is sonarqube.singhritesh85.com/quality_gates/show/AZaZqJUY-XhVUDQM0yfb.
- The page title is "Sonar way BUILT-IN".
- A modal window titled "Create Quality Gate" is open, showing the following details:
 - Name ***: therema (highlighted with a yellow box)
 - Description**: This quality gate ensures that the code is clean and follows best practices.
 - Conditions**: All fields marked with * are required.
 - Save** and **Cancel** buttons.
- Conditions on New Code** table:

Metric	Operator	Value
Coverage	is less than	80.0%
Duplicated Lines (%)	is greater than	3.0%
Maintainability Rating	is worse than	A (Technical debt ratio is less than 5.0%)
Reliability Rating	is worse than	A (No bugs)
Security Hotspots Reviewed	is less than	100%
Security Rating	is worse than	A (No vulnerabilities)

Page 2: Quality Gate Details

- The URL is sonarqube.singhritesh85.com/quality_gates/show/ (with the ID redacted).
- The page title is "Quality Gates".
- The Quality Gate is named "therema".
- Conditions** table:

Metric	Operator	Value	Actions
Coverage	is less than	80.0%	
Duplicated Lines (%)	is greater than	3.0%	
Maintainability Rating	is worse than	A (Technical debt ratio is less than 5.0%)	
Reliability Rating	is worse than	A (No bugs)	
Security Hotspots Reviewed	is less than	100%	
Security Rating	is worse than	A (No vulnerabilities)	
- A note at the bottom states: "You may click unlock to edit this quality gate. Adding extra conditions to a compliant quality gate can result in drawbacks. Are you reconsidering [Clean as You Code?](#) We strongly recommend this methodology to achieve a Clean Code status."
- Unlock editing** button (highlighted with a yellow box).

sonarqube.singhritesh85.com/quality_gates/show/AZacOGMqMmNEC3p7QZA1

Quality Gates Projects Issues Rules Quality Profiles Quality Gates Administration

therema

Sonar way DEFAULT BUILT-IN therema

Conditions on New Code

Metric	Operator	Value
Coverage	is less than	80.0%
Duplicated Lines (%)	is greater than	3.0%
Maintainability Rating	is worse than	A (Technical debt ratio is less than 5.0%)
Reliability Rating	is worse than	A (No bugs)
Security Hotspots Reviewed	is less than	100%
Security Rating	is worse than	A (No vulnerabilities)

Add Condition

sonarqube.singhritesh85.com/quality_gates/show/AZacOGMqMmNEC3p7QZA1

Quality Gates Projects Issues Rules Quality Profiles Quality Gates Administration

therema

Conditions on New Code

Metric	Operator	Value
Coverage	is greater than	0
Duplicated Lines (%)		
Maintainability Rating		
Reliability Rating		
Security Hotspots Reviewed		
Security Rating		

Add Condition

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 illustrate the configuration of a Quality Gate in SonarQube. The 'Add Condition' dialog is open, showing the following settings:

- On New Code:** Unchecked.
- On Overall Code:** Checked.
- Quality Gate fails when:** Set to "Code Smells".
- Operator:** "is greater than".
- Value:**
 - In the first screenshot, the value is 12.
 - In the second screenshot, the value is 0.

The background of the interface lists other predefined conditions for various metrics such as Coverage, Duplication, Maintainability, Reliability, and Security.

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

therema

Conditions

Conditions on New Code

Metric	Operator	Value
Coverage	is less than	80.0%
Duplicated Lines (%)	is greater than	3.0%
Maintainability Rating	is worse than	A (Technical debt ratio is less than 5.0%)
Reliability Rating	is worse than	A (No bugs)
Security Hotspots Reviewed	is less than	100%
Security Rating	is worse than	A (No vulnerabilities)

Conditions on Overall Code

Metric	Operator	Value
Bugs	is greater than	0
Code Smells	is greater than	12
Vulnerabilities	is greater than	0

mederma

Conditions

Conditions on New Code

Metric	Operator	Value
Coverage	is less than	80.0%
Duplicated Lines (%)	is greater than	3.0%
Maintainability Rating	is worse than	A (Technical debt ratio is less than 5.0%)
Reliability Rating	is worse than	A (No bugs)
Security Hotspots Reviewed	is less than	100%
Security Rating	is worse than	A (No vulnerabilities)

Conditions on Overall Code

Metric	Operator	Value
Bugs	is greater than	0
Code Smells	is greater than	12
Vulnerabilities	is greater than	0

You may click unlock to edit this quality gate. Adding extra conditions to a compliant quality gate can result in drawbacks. Are you reconsidering [Clean as You Code?](#)
We strongly recommend this methodology to achieve a Clean Code static.

Ritesh

SonarQube Quality Gates configuration for the 'therema' project. The 'Conditions on New Code' section contains the following rules:

Metric	Operator	Value
Coverage	is less than	80.0%
Duplicated Lines (%)	is greater than	3.0%
Maintainability Rating	is worse than	A (Technical debt ratio is less than 5.0%)
Reliability Rating	is worse than	A (No bugs)
Security Hotspots Reviewed	is less than	100%
Security Rating	is worse than	A (No vulnerabilities)

The 'Conditions on Overall Code' section contains the following rules:

Metric	Operator	Value
Bugs	is greater than	0
Code Smells	is greater than	12
Vulnerabilities	is greater than	0

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

Jenkins Global credentials creation page for a 'Username with password' credential. The fields filled are:

- Kind:** Username with password
- Scope:** Global (Jenkins, nodes, items, all child items, etc)
- Username:** [REDACTED]
- Treat username as secret:**
- Password:** [REDACTED]
- ID:** github-cred
- Description:** github-cred

The 'Create' button is highlighted with a yellow box.

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

The screenshot shows the Jenkins 'New credentials' interface. 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'. A blue 'Create' button is at the bottom left.

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.

The screenshot shows the Jenkins 'New node' interface. The 'Node name' field is set to 'Slave-1'. The 'Type' section has a radio button selected for 'Permanent Agent', with a descriptive text explaining it adds a plain, permanent agent to Jenkins. A blue 'Create' button is at the bottom left.

REST API Jenkins 2.492.3



The screenshot shows two consecutive steps of creating a Jenkins slave node.

Step 1: Basic Configuration

- Name:** Slave-1
- Description:** This is a Slave Node.
- Number of executors:** 2
- Remote root directory:** /home/jenkins

Step 2: Advanced Configuration

- Labels:** Slave-1
- Usage:** Use this node as much as possible
- Launch method:** Launch agents via SSH
- Host:** [REDACTED] (highlighted with an orange arrow)
- Credentials:** [REDACTED]
- Host Key Verification Strategy:** [REDACTED]

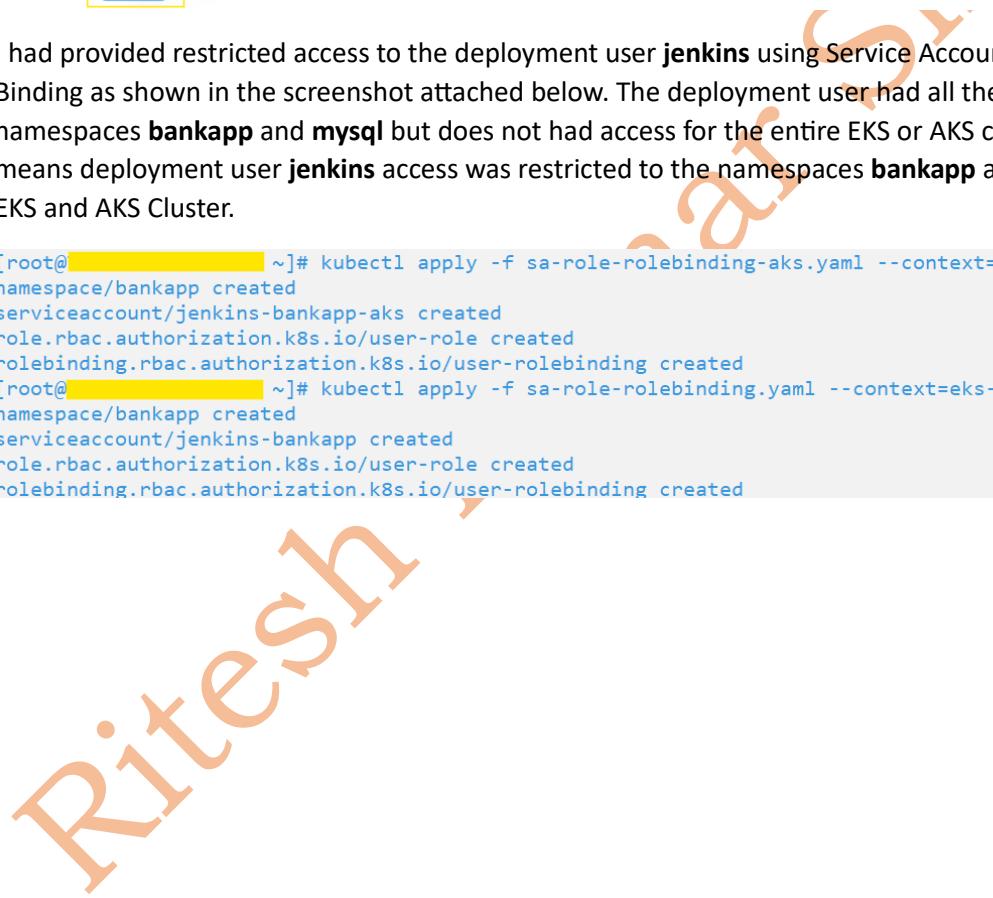
A blue "Save" button is visible at the bottom of both forms.

The screenshot shows the Jenkins Node Configuration page. It includes sections for Host Key Verification Strategy (Non verifying Verification Strategy), Availability (Keep this agent online as much as possible), and Node Properties (checkboxes for Azure Active Directory Authorization Matrix, Disable deferred wipeout on this node, Disk Space Monitoring Thresholds, Environment variables, and Tool Locations). A blue 'Save' button is at the bottom.

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.

The screenshot shows the Jenkins Plugin Manager page. The search bar contains 'pipeline utility step'. Two plugins are listed: 'Nexus Artifact Uploader' (version 2.14, Released 2 yr 5 mo ago) and 'Pipeline Utility Steps' (version 2.19.0, Released 2 mo 15 days ago). Both have a checked checkbox next to them, indicating they are selected for installation. A blue 'Install' button is visible.

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



A screenshot of the Jenkins 'New credentials' configuration page. The URL in the address bar is `jenkins-ms.singhrithesh85.com/manage/credentials/store/system/domain/_/newCredentials`. The page shows fields for creating a new credential of type 'Username with password'. 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 yellowed-out value. The 'Treat username as secret' checkbox is unchecked. The 'Password' field contains a yellowed-out value. The 'ID' field contains the value 'nexus'. A blue 'Create' button is at the bottom.

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 or AKS cluster. That means deployment user **jenkins** access was restricted to the namespaces **bankapp** and **mysql** in the EKS and AKS Cluster.

```
[root@yellow ~]# kubectl apply -f sa-role-rolebinding-aks.yaml --context=aks-cluster
namespace/bankapp created
serviceaccount/jenkins-bankapp-aks created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding created
[root@yellow ~]# kubectl apply -f sa-role-rolebinding.yaml --context=eks-demo-cluster-dev
namespace/bankapp created
serviceaccount/jenkins-bankapp created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding created
```

```
[root@redacted ~]# 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@]# cat sa-role-rolebinding-aks.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-bankapp-aks
  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-aks
```

K

```
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
```

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

apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-bankapp-aks
  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-aks
```

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.

```
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
```

```
cat secrets-aks.yaml

apiVersion: v1

kind: Secret

type: kubernetes.io/service-account-token

metadata:

name: mysecretname-aks

namespace: bankapp

annotations:

kubernetes.io/service-account.name: jenkins-bankapp-aks
```

```
[root@yellow ~]# kubectl apply -f secrets-aks.yaml --context=aks-cluster
secret/mysecretname-aks unchanged
[root@yellow ~]# kubectl apply -f secrets-aks.yaml --context=eks-demo-cluster-dev
secret/mysecretname-aks created

[root@yellow ~]# kubectl describe secrets mysecretname-aks -n bankapp --context=aks-cluster
Name:         mysecretname-aks
Namespace:    bankapp
Labels:       <none>
Annotations: kubernetes.io/service-account.name: jenkins-bankapp-aks
              kubernetes.io/service-account.uid: yellow

Type:  kubernetes.io/service-account-token

Data
====

token:  e

ca.crt:  1765 bytes
namespace:  2 bytes...
```

```
[root@REDACTED ~]# kubectl describe secrets mysecretname -n bankapp --context=eks-demo-cluster-dev
Name:      mysecretname
Namespace: bankapp
Labels:    <none>
Annotations: kubernetes.io/service-account.name: jenkins-bankapp
             kubernetes.io/service-account.uid: REDACTED
Type:     kubernetes.io/service-account-token

Data
====

token:     e
REDACTED

ca.crt:   1107 bytes
namespace: 7 bytes
```

```
[root@REDACTED ~]# 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@REDACTED ~]# cat secrets-aks.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-aks
  namespace: bankapp
  annotations:
    kubernetes.io/service-account.name: jenkins-bankapp-aks
```

Ritesh V

```
[root@XXXXXXXXXX ~]# 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@yellow ~]# cat sa-role-rolebinding-mysql-aks.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-mysql-aks
  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-aks
```

```
[root@yellow ~]# kubectl apply -f sa-role-rolebinding-mysql-aks.yaml --context=aks-cluster
namespace/mysql created
serviceaccount/jenkins-mysql-aks created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding created
[root@yellow ~]# kubectl apply -f sa-role-rolebinding-mysql.yaml --context=eks-demo-cluster-dev
namespace/mysql created
serviceaccount/jenkins-mysql created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding 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
```

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

apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-mysql-aks
  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-aks
```

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.

```
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  
  
cat secrets-mysql-aks.yaml  
apiVersion: v1  
kind: Secret  
type: kubernetes.io/service-account-token  
metadata:  
  name: mysecretname-mysql-aks  
  namespace: mysql  
  annotations:  
    kubernetes.io/service-account.name: jenkins-mysql-aks
```

```
[root@yellow ~]# kubectl apply -f secrets-mysql-aks.yaml --context=aks-cluster  
secret/mysecretname-mysql-aks created  
[root@yellow ~]# kubectl apply -f secrets-mysql.yaml --context=eks-demo-cluster-dev  
secret/mysecretname-mysql created
```

```
[root@[REDACTED] ~]# 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@[REDACTED] ~]# cat secrets-mysql-aks.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-mysql-aks
  namespace: mysql
  annotations:
    kubernetes.io/service-account.name: jenkins-mysql-aks
```

```
[root@[REDACTED] ~]# kubectl describe secrets mysecretname-mysql-aks -n mysql --context=aks-cluster
Name:      mysecretname-mysql-aks
Namespace: mysql
Labels:    <none>
Annotations:  kubernetes.io/service-account.name: jenkins-mysql-aks
              kubernetes.io/service-account.uid: [REDACTED]

Type:  kubernetes.io/service-account-token

Data
====

ca.crt:  1765 bytes
namespace: 5 bytes
token:   e
[REDACTED]
```

```
[root@[REDACTED] ~]# kubectl describe secrets mysecretname-mysql -n mysql --context=eks-demo-cluster-dev
Name:      mysecretname-mysql
Namespace: mysql
Labels:    <none>
Annotations:  kubernetes.io/service-account.name: jenkins-mysql
              kubernetes.io/service-account.uid: [REDACTED]

Type:  kubernetes.io/service-account-token

Data
====

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

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**.

```
cat ~/.kube/config

apiVersion: v1

clusters:

- cluster:

  certificate-authority-data:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXX

  server: https://aks-cluster-dns-XXXXXXX.hcp.eastus.azmk8s.io:443

  name: aks-cluster

- cluster:

  certificate-authority-data:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXX

  server: https://XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX.gr7.us-east-2.eks.amazonaws.com

  name: eks-demo-cluster-dev

contexts:

- context:

  cluster: aks-cluster

  user: jenkins-aks

  name: aks-cluster-bankapp

- context:

  cluster: aks-cluster

  user: jenkins-mysql-aks

  name: aks-cluster-mysql

- context:

  cluster: eks-demo-cluster-dev

  user: jenkins

  name: eks-demo-cluster-dev-bankapp

- context:

  cluster: eks-demo-cluster-dev

  user: jenkins-mysql

  name: eks-demo-cluster-dev-mysql

current-context: eks-demo-cluster-dev-bankapp

kind: Config
```

```

preferences: {}

users:

- name: jenkins

  user:

    token:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXX

- name: jenkins-mysql

  user:

    token:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXX

- name: jenkins-aks

  user:

    token:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXX

- name: jenkins-mysql-aks

  user:

    token:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXX

```

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

```

[jenkins@XXXXXXXXX ~]$ kubectl config get-contexts
CURRENT  NAME          CLUSTER      AUTHINFO        NAMESPACE
*        aks-cluster-bankapp   aks-cluster   jenkins-aks
          aks-cluster-mysql   aks-cluster   jenkins-mysql-aks
          eks-demo-cluster-dev-bankapp   eks-demo-cluster-dev   jenkins
          eks-demo-cluster-dev-mysql   eks-demo-cluster-dev   jenkins-mysql
[jenkins@XXXXXXXXX ~]$ kubectl get nodes --context=aks-cluster-bankapp
Error from server (Forbidden): nodes is forbidden: User "system:serviceaccount:bankapp:jenkins-bankapp-aks" cannot list resource "nodes" in API group "" at the cluster scope
[jenkins@XXXXXXXXX ~]$ kubectl get nodes --context=aks-cluster-mysql
Error from server (Forbidden): nodes is forbidden: User "system:serviceaccount:mysql:jenkins-mysql-aks" cannot list resource "nodes" in API group "" at the cluster scope
[jenkins@XXXXXXXXX ~]$ kubectl get nodes --context=eks-demo-cluster-dev-bankapp
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@XXXXXXXXX ~]$ kubectl get nodes --context=eks-demo-cluster-dev-mysql
Error from server (Forbidden): nodes is forbidden: User "system:serviceaccount:mysql:jenkins-mysql" cannot list resource "nodes" in API group "" at the cluster scope
[jenkins@XXXXXXXXX ~]$ kubectl get all -n bankapp --context=aks-cluster-bankapp
No resources found in bankapp namespace.
[jenkins@XXXXXXXXX ~]$ kubectl get all -n mysql --context=aks-cluster-mysql
No resources found in mysql namespace.
[jenkins@XXXXXXXXX ~]$ kubectl get all -n bankapp --context=eks-demo-cluster-dev-bankapp
No resources found in bankapp namespace.
[jenkins@XXXXXXXXX ~]$ kubectl get all -n mysql --context=eks-demo-cluster-dev-mysql
No resources found in mysql namespace.

```

Creation of multibranch pipeline in Jenkins

Before creation of multibranch pipeline I created Jenkins Credentails to store the MYSQL_DATABASE name and MYSQL_ROOT_PASSWORD which I will be utilized in the Jenkinsfile. As it is not a good practice to hardcode the password or sensitive information in the Jenkinsfile So I had created the Jenkins Credentials. Below screenshot shows how I had created the Jenkins Credentails of kind secret text.

The screenshots show the Jenkins 'New credentials' creation interface. Both screenshots have the 'Kind' dropdown set to 'Secret text'. The 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc)'. The 'Secret' field contains '.....'. The 'ID' field is highlighted in both screenshots and contains 'mysql_database' in the first and 'mysql_root_password' in the second. The 'Description' field is also highlighted in both screenshots and contains 'mysql_database' in the first and 'mysql_root_password' in the second. The 'Create' button at the bottom is highlighted in both screenshots.

I had created the Jenkins credentials for ArgoCD Password as shown in the screenshot attached below which I had utilized in the Jenkinsfile.

The screenshot shows the Jenkins 'New credentials' configuration page. A 'Secret text' credential is being created with the following details:

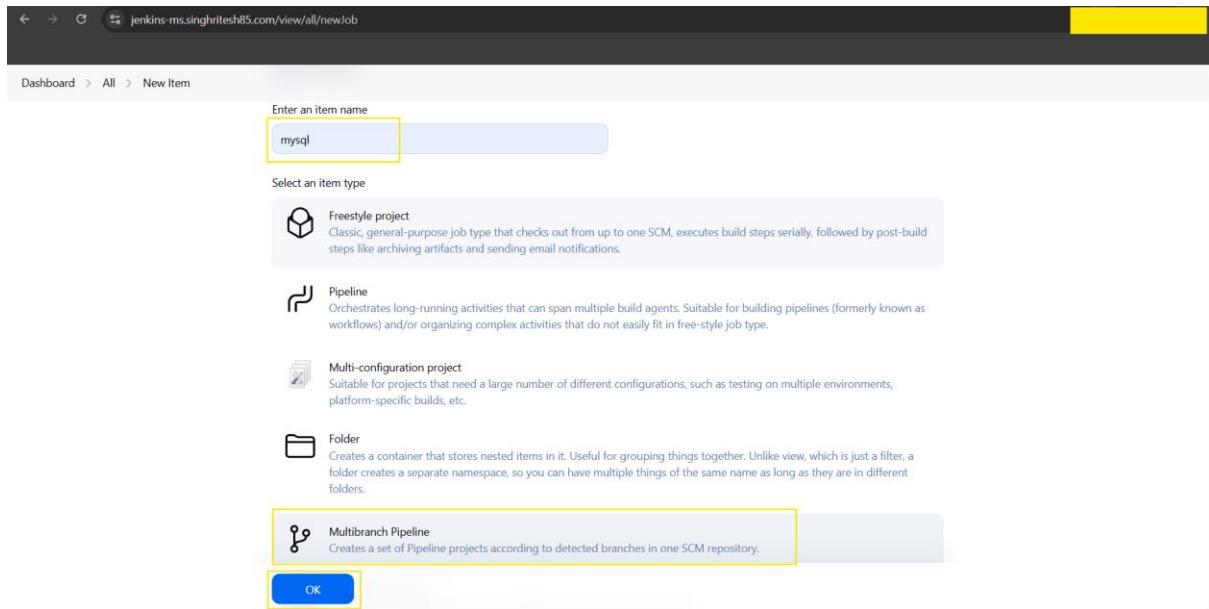
- Kind:** Secret text (highlighted with a yellow box)
- Scope:** Global (Jenkins, nodes, items, all child items, etc)
- Secret:** ARGOCD_PASSWORD (highlighted with a yellow box)
- ID:** ARGOCD_PASSWORD (highlighted with a yellow box)
- Description:** ARGOCD_PASSWORD (highlighted with a yellow box)
- Create:** A blue 'Create' button.

Then Created a multi branch pipeline in Jenkins for dev and stage branch as shown in the screenshot attached below.

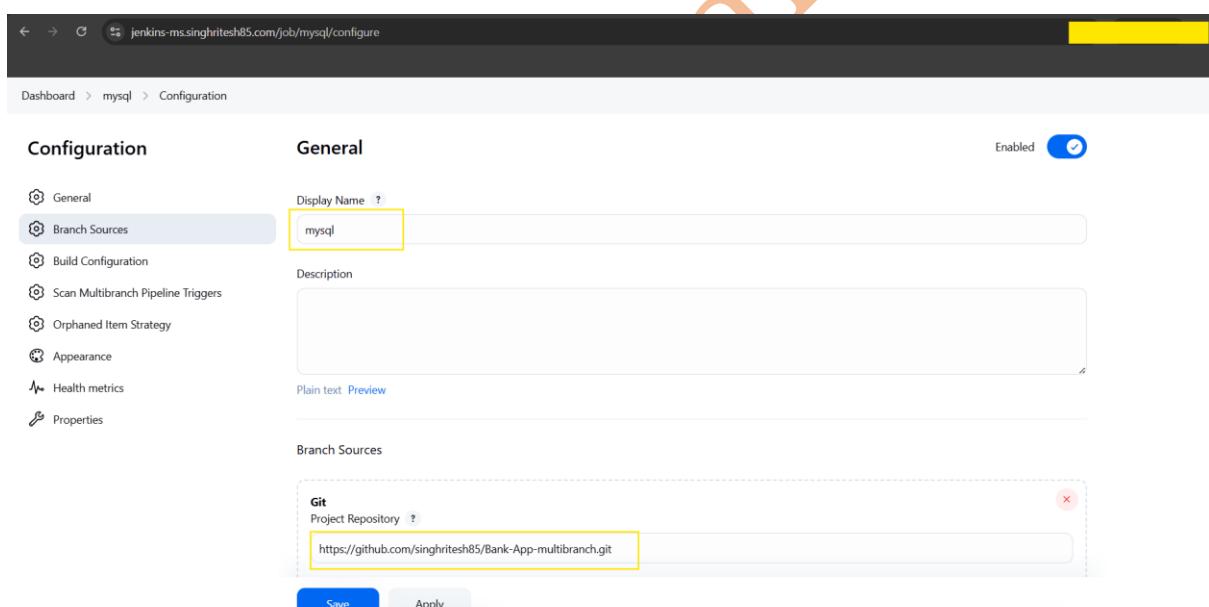
The screenshot shows the Jenkins dashboard. Key elements include:

- The Jenkins logo at the top left.
- A navigation bar with links: Dashboard, Manage Jenkins, Credentials, System, and Global credentials (unrestricted).
- A search bar and user dropdown.
- A 'Welcome to Jenkins!' message: "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." Below it is a "Start building your software project" button.
- A sidebar with links: + New Item, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, and My Views.
- Widgets:
 - Build Queue: "No builds in the queue."
 - Build Executor Status:

Node	Status
Built-in Node	0/2
Slave-1	0/2
- Bottom right: REST API and Jenkins 2.504.1.



Then first, I created the multibranch pipeline for MySQL Pods as shown in the screenshot attached below.



The screenshot shows the Jenkins MySQL job configuration page. In the left sidebar, 'Branch Sources' is selected. The main area displays 'Behaviors' settings. Under 'Discover branches', there is a single entry: 'dev stage'. Under 'Filter by name (with wildcards)', there is an 'Include' entry: 'dev stage'. There is also an 'Exclude' section which is currently empty. At the bottom are 'Save' and 'Apply' buttons.

The screenshot shows the Jenkins MySQL job configuration page. In the left sidebar, 'Build Configuration' is selected. The main area displays 'Property strategy' set to 'All branches get the same properties'. Below it, 'Add property' and 'Add source' buttons are visible. Under 'Build Configuration', 'Mode' is set to 'by Jenkinsfile' and 'Script Path' is set to 'mysql/Jenkinsfile'. At the bottom are 'Save' and 'Apply' buttons.

Before creating the pipeline for bankapp you need to create a kubernetes secrets to pull docker image from private ECR (Elastic Container Registry) Repository on AKS (Azure Kubernetes Services).

```
kubectl create secret docker-registry regcred --docker-server=02XXXXXXXXXX6.dkr.ecr.us-east-2.amazonaws.com --docker-username=AWS --docker-password=$(aws ecr get-login-password) --namespace=bankapp --context=aks-cluster
```

```
[root@REDACTED ~]# kubectl create secret docker-registry regcred --docker-server=02REDACTED.dkr.ecr.us-east-2.amazonaws.com --docker-username=AWS --docker-password=$(aws ecr get-login-password) --namespace=bankapp --context=aks-cluster
secret/regcred created
```

To create a Webhook you need to install the plugin **multibranch scan webhook trigger** as shown in the screenshot attached below.

The screenshot shows two Jenkins web pages. The top page is the 'Available Plugins' section where the 'Multibranch Scan Webhook Trigger' plugin is being installed. The bottom page is the configuration screen for a 'bankapp' job, showing its general settings and a Git branch source configuration.

Top Page: Available Plugins

- URL: jenkins-ms.singhritesh85.com/manage/pluginManager/available
- Plugin: Multibranch Scan Webhook Trigger (1.0.11)
- Description: Trigger that can receive any HTTP request and trigger a multibranch job scan when token matched.
- Status: Released, 1 yr 4 mo ago
- Action: Install (button highlighted with yellow box)

Bottom Page: Configuration - bankapp

- URL: jenkins-ms.singhritesh85.com/job/bankapp/configure
- Section: General
- Display Name: bankapp
- Enabled: Yes
- Branch Sources: Git (Project Repository)
- Buttons: Save, Apply

The screenshot shows the Jenkins job configuration page for 'bankapp'. The left sidebar has 'Branch Sources' selected. The main area shows the 'Branch Sources' section with a 'Git' repository set to 'https://github.com/singhritesh85/Bank-App-multibranch.git' and a credential named '***** (github-cred)'. Below this are 'Behaviors' sections: 'Discover branches' and 'Filter by name (with wildcards)' which includes 'Include' and 'dev stage'.

The screenshot shows the Jenkins job configuration page for 'bankapp'. The left sidebar has 'Build Configuration' selected. The main area shows the 'Build Configuration' section with 'Property strategy' set to 'All branches get the same properties'. Below this is the 'Mode' section, which is set to 'by Jenkinsfile' and has 'Script Path' set to 'Jenkinsfile'.

The screenshot shows the Jenkins job configuration page for 'bankapp'. Under 'Scan Multibranch Pipeline Triggers', the 'Scan by webhook' checkbox is checked, and the 'Trigger token' field contains 'dexter'. A red arrow points from the text 'To create Webhook for multibranch pipeline add this line to the GitHub Settings as shown in the screenshot attached below.' to this configuration section.

To create Webhook for multibranch pipeline add this line to the GitHub Settings as shown in the screenshot attached below.

The screenshot shows the GitHub repository settings for 'Bank-App-multibranch'. In the 'Webhooks' section, a new webhook is being configured with the payload URL 'https://jenkins-ms.singhritesh85.com/multibranch-webhook-trigger/invoke?token=dexter', content type 'application/x-www-form-urlencoded', and SSL verification set to 'Enable SSL verification'. The 'Just the push event' option is selected. A red arrow points from the 'Webhooks' section in the Jenkins screenshot to this GitHub configuration.

The screenshot shows the GitHub repository settings after the webhook has been added. The newly configured webhook is listed under 'Webhooks', with the URL 'https://jenkins-ms.singhritesh85.com/multibranch-webhook-trigger/invoke?token=dexter' and the event type '(push)'. A red checkmark icon is next to the URL, indicating successful delivery.

In this project for the demonstration purpose, I had configured webhook however to run the Jenkins Job **bankapp/dev** and **bankapp/stage** I need to run those Jenkins Job manually because I was

supposed to pass the string parameter REPO_NAME and TAG_NAME to run those Jenkins Jobs. It was mandatory to pass the parameters to run bankapp Jenkins multibranch pipeline to dev and stage pipeline as shown in the screenshot attached below.

The image contains three vertically stacked screenshots of the Jenkins web interface, each showing a different stage of a multibranch pipeline named "bankapp".

- Top Screenshot (dev):** Shows the "Status" tab selected. The "Build with Parameters" button is highlighted with a yellow border. The "SonarQube Quality Gate" section shows "bankapp" status as "Passed" with "server-side processing: Success". The "Builds" table lists six builds from #6 to #1, with build #6 being the most recent and successful. A large orange "high" watermark is overlaid on the right side of the page.
- Middle Screenshot (stage):** Shows the "Status" tab selected. The "Build with Parameters" button is highlighted with a yellow border. The "Permalinks" section shows a list of builds from #6 to #1. The "Builds" table lists six builds from #6 to #1, with build #6 being the most recent and successful.
- Bottom Screenshot (build configuration):** Shows the "Pipeline stage" configuration page. The "Build with Parameters" button is highlighted with a yellow border. It requires two parameters: "REPO_NAME" (with a placeholder "Provide the ECR Repository Name for Application Image") and "TAG_NAME" (with a placeholder "Provide the TAG Name"). The "Builds" table lists six builds from #6 to #1, with build #6 being the most recent and successful.

S	W	Name ↓	Last Success	Last Failure	Last Duration	
⌚	☀️	bankapp	3 min 55 sec log	N/A	0.42 sec	▶
⌚	☀️	mysql	23 min log	N/A	0.38 sec	▶

GitHub Webhook, trigger I had applied to only multibranch Jenkins Job bankapp but not to the MySQL Jenkin pods.

```
[root@[REDACTED] ~]# kubectl get pods -n mysql --context=aks-cluster --watch
NAME        READY   STATUS    RESTARTS   AGE
mysql-primary-0  1/1     Running   0          95m
mysql-secondary-0 1/1     Running   1 (93m ago)  95m
^C[root@[REDACTED] ~]#
[root@Terraform-Server ~]# kubectl get pods -n bankapp --context=eks-demo-cluster-dev --watch
NAME        READY   STATUS    RESTARTS   AGE
bankapp-folo-[REDACTED] 1/1     Running   0          75m
^C[root@[REDACTED] ~]#
[root@[REDACTED] ~]# kubectl get pods -n mysql --context=aks-cluster --watch
NAME        READY   STATUS    RESTARTS   AGE
mysql-primary-0  1/1     Running   0          96m
mysql-secondary-0 1/1     Running   1 (94m ago)  96m
^C[root@[REDACTED] ~]#
[root@[REDACTED] ~]# kubectl get pods -n bankapp --context=eks-demo-cluster-dev --watch
NAME        READY   STATUS    RESTARTS   AGE
bankapp-folo-[REDACTED] 1/1     Running   0          76m
^C[root@[REDACTED] ~]#
```

To access the Bank Application, I created the ingress rule using the yaml manifests file as shown in the screenshot attached below.

```
[root@[REDACTED] ~]# cat ingress-rule-dev.yaml
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: bankapp-ingress-dev
  namespace: bankapp
  annotations:
    kubernetes.io/ingress.class: nginx
spec:
  ingressClassName: nginx
  rules:
  - host: bankapp-dev.singhritesh85.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: bankapp-folo
            port:
              number: 80
[root@[REDACTED] ~]# kubectl apply -f ingress-rule-dev.yaml --context=eks-demo-cluster-dev
```

```
[root@XXXXXXXXXX ~]# cat ingress-rule-stage.yaml
# kubectl create secret tls ingress-tls --key mykey.key --cert STAR_singhritesh85_com.crt --namespace bankapp --context=aks-cluster
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: bankapp-ingress-stage
  namespace: bankapp
  annotations:
    kubernetes.io/ingress.class: nginx
spec:
  ingressClassName: nginx
  tls:
  - hosts:
    - bankapp-stage.singhritesh85.com
    secretName: ingress-tls
  rules:
  - host: bankapp-stage.singhritesh85.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: bankapp-folo
            port:
              number: 80
[root@XXXXXXXXXX ~]# kubectl apply -f ingress-rule-stage.yaml --context=aks-cluster
ingress.networking.k8s.io/bankapp-ingress-stage created
```

NAMESPACE	NAME	CLASS	HOSTS	ADDRESS	PORTS	AGE
bankapp	bankapp-ingress-stage	nginx	bankapp-stage.singhritesh85.com	20. XXXXXX .52	80	80s
[root@ XXXXXXXXXX ~]	# kubectl get ing -A --context=eks-demo-cluster-dev					
NAMESPACE	NAME	CLASS	HOSTS	ADDRESS	PORTS	AGE
argocd	minimal-ingress	nginx	argocd.singhritesh85.com	a. XXXXXX 5.us-east-2.elb.amazonaws.com	80	7h4
bankapp	bankapp-ingress-dev	nginx	bankapp-dev.singhritesh85.com	a. XXXXXX 6.us-east-2.elb.amazonaws.com	80	117s

I did the entry for HOST with LoadBalancer External IP/DNS Name in the Azure DNS Zone to create the Record Set of A Type and CNAME Type respectively for Stage and Dev Environment as shown in the screenshot attached below.

Add record set

singhritesh85.com

Name

.singhritesh85.com

Type



Alias record set



TTL *

TTL unit



IP address



Give feedback

Add record set

singhritesh85.com

Name

.singhritesh85.com

Type

CNAME – Link your subdomain to another record

Alias record set ⓘ

TTL *

TTL unit

Hours

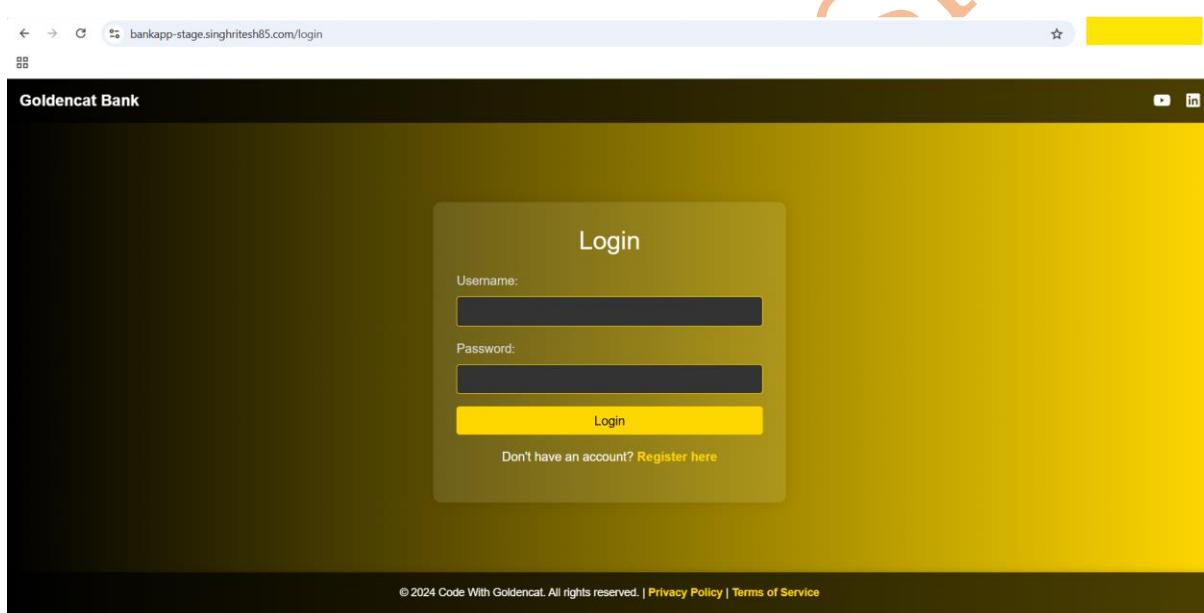
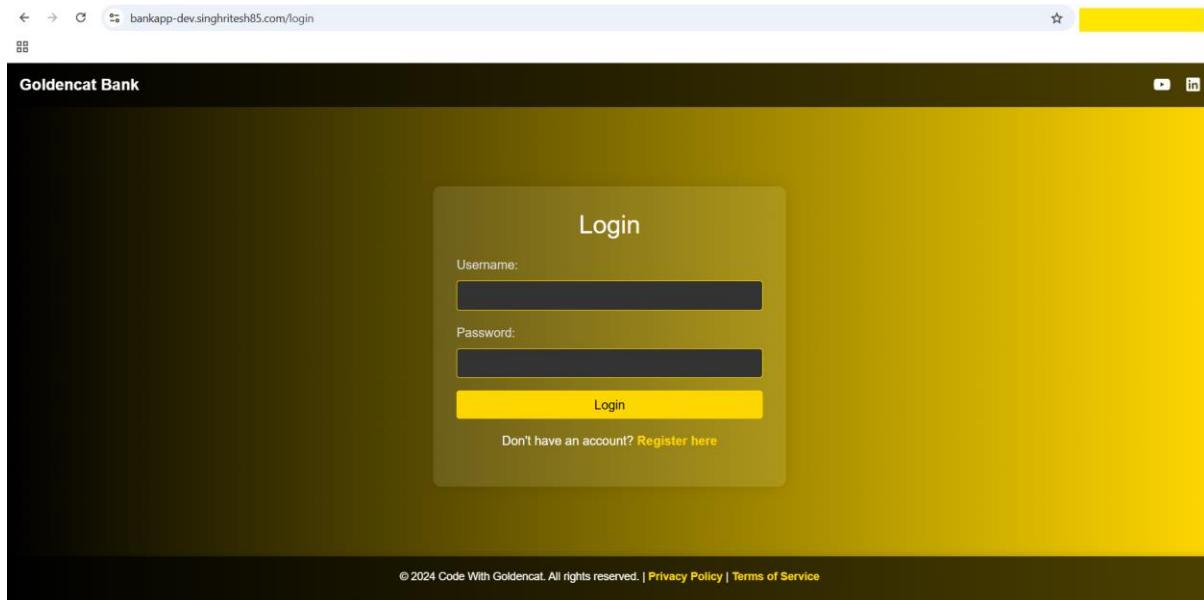
Alias

Add

Cancel

Give feedback

Finally, the Bank Application was accessible through the URL as shown in the screenshot attached below.



The SonarQube, Nexus Artifactory and Trivy Image Scan had been done in the non-production environment as shown in the Jenkinsfile present in the GitHub Repo <https://github.com/singhritesh85/Bank-App-multibranch.git> at **dev** and **stage** branch. The screenshot of **SonarQube**, **Nexus Artifactory** and **ArgoCD** after running the two CICD pipeline successfully is as shown in the screenshot attached below.

The image displays three separate screenshots of the SonarQube web application interface.

Screenshot 1: Project Analysis Overview

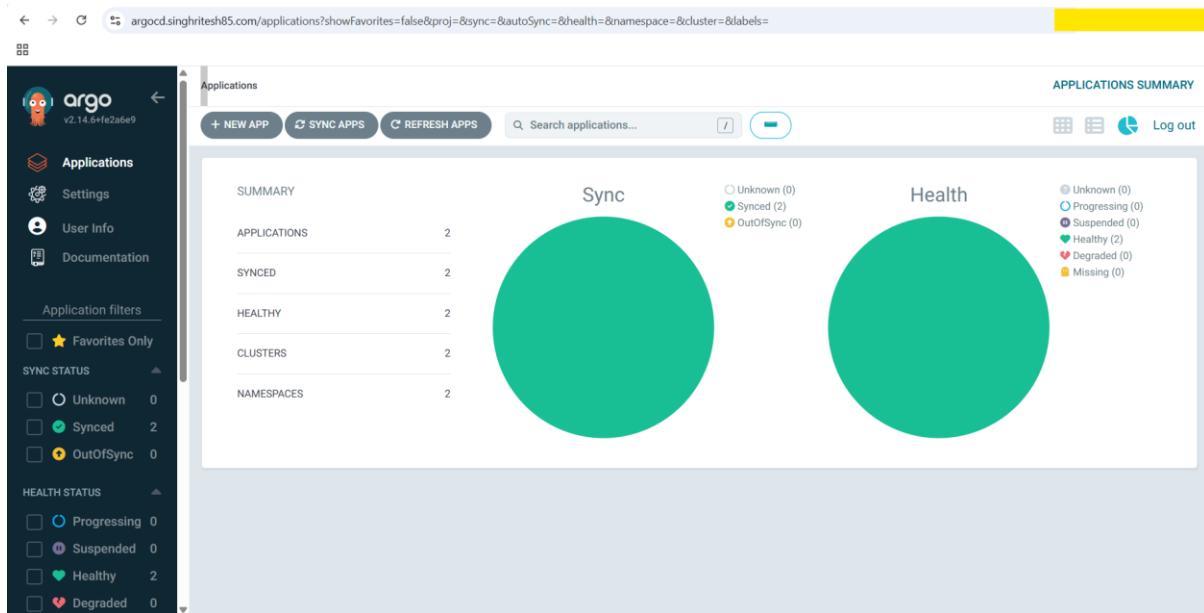
- Left Sidebar:** Shows filters for Quality Gate (Passed: 1, Failed: 0), Reliability (Bugs: A rating 1, B rating 0, C rating 0, D rating 0, E rating 0), and Security (Vulnerabilities: A rating 1, B rating 0, C rating 0, D rating 0, E rating 0).
- Top Bar:** Includes a search bar ("Search by project name or key"), "Create Project" button, and navigation tabs.
- Project Summary:** Shows 1 project(s) named "bankapp" with a "Passed" status. Metrics include Bugs (0 A), Vulnerabilities (0 A), Hotspots Reviewed (0.0% E), Code Smells (11 A), Coverage (0.0% O), Duplications (0.0% G), and Lines (471 X3 Java, XML).
- Bottom Footer:** Mentions SonarQube™ technology is powered by SonarSource SA, Community Edition - v9.9.5 (build 90363), and links to LGPL v3, Community, Documentation, Plugins, and Web API.

Screenshot 2: Maven Snapshot File Browser

- Left Sidebar:** Shows "Browse" selected in the navigation menu.
- Top Bar:** Shows "Browse / maven-snapshot".
- Content Area:** An "HTML View" tree structure showing the directory structure: com > example > bankapp > 0.0.1-SNAPSHOT > 0.0.1-20250505.093412-1 > maven-metadata.xml, maven-metadata.xml.md5, and maven-metadata.xml.sha1.

Screenshot 3: Maven Snapshot Stage File Browser

- Left Sidebar:** Shows "Browse" selected in the navigation menu.
- Top Bar:** Shows "Browse / maven-snapshot-stage".
- Content Area:** An "HTML View" tree structure showing the directory structure: com > example > bankapp > 0.0.1-SNAPSHOT > 0.0.1-20250505.094f33-1 > maven-metadata.xml, maven-metadata.xml.md5, and maven-metadata.xml.sha1.



After Execution of Jenkins Job, I received the below notification on the group email Id which I configure for Jenkins.

Jenkins Job mysql/dev has been executed Inbox x

R [REDACTED]@gmail.com
to me ▾

A Jenkins Job with Job Name mysql/dev has been executed

Reply Forward

Jenkins Job mysql/dev has been Sucessfully Executed Inbox x

R [REDACTED]@gmail.com
to me ▾

[REDACTED] (2 minutes ago)

A Jenkins Job with Job Name mysql/dev and Build Number=5 has been executed Successfully. Please Open the URL <https://jenkins-ms.singhritesh85.com/job/mysql/dev/5/> and click on Console Output to see the Log. The Result of execution is SUCCESS

Reply Forward

Jenkins Job mysql/stage has been executed Inbox x

R [REDACTED]@gmail.com
to me ▾

A Jenkins Job with Job Name mysql/stage has been executed

Reply Forward

Jenkins Job mysql/stage has been Sucessfully Executed Inbox x



[\[REDACTED\]@gmail.com](#)

to me ▾

(3 minutes ago)

A Jenkins Job with Job Name mysql/stage and Build Number=4 has been executed Successfully, Please Open the URL <https://jenkins-ms.singhritesh85.com/job/mysql/job/stage/4/> and click on Console Output to see the Log. The Result of execution is SUCCESS

Reply

Forward



Jenkins Job bankapp/stage has been executed Inbox x



[\[REDACTED\]@gmail.com](#)

to me ▾

A Jenkins Job with Job Name bankapp/stage has been executed

Reply

Forward



Jenkins Job bankapp/stage has been Failed Inbox x



[\[REDACTED\]@gmail.com](#)

to me ▾

(6 minutes ago)

A Jenkins Job with Job Name bankapp/stage and Build Number=5 has been Failed, Please Open the URL <https://jenkins-ms.singhritesh85.com/job/bankapp/job/stage/5/> and click on Console Output to see the Log. The Result of execution is FAILURE

Reply

Forward



Jenkins Job bankapp/dev has been executed



[\[REDACTED\]@gmail.com](#)

to me ▾

A Jenkins Job with Job Name bankapp/dev has been executed

Reply

Forward



Jenkins Job bankapp/dev has been Sucessfully Executed Inbox x



[\[REDACTED\]@gmail.com](#)

to me ▾

(2 minutes ago)

A Jenkins Job with Job Name bankapp/dev and Build Number=9 has been executed Successfully, Please Open the URL <https://jenkins-ms.singhritesh85.com/job/bankapp/job/dev/9/> and click on Console Output to see the Log. The Result of execution is SUCCESS

Reply

Forward



Installation of node-exporter and promtail had been done using the helm chart in the EKS and AKS 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@REDACTED ~]# kubectl config use-context eks-demo-cluster-dev
Switched to context "eks-demo-cluster-dev".
[root@REDACTED ~]# helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
"prometheus-community" has been added to your repositories
[root@REDACTED ~]# kubectl create ns node-exporter
namespace/node-exporter created
[root@REDACTED ~]# helm install my-prometheus-node-exporter prometheus-community/prometheus-node-exporter --version 4.37.1 --set service.type=LoadBalancer -n node-exporter

[root@REDACTED ~]# kubectl config use-context aks-cluster
Switched to context "aks-cluster".
[root@REDACTED ~]# helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
"prometheus-community" already exists with the same configuration, skipping
[root@REDACTED ~]# kubectl create ns node-exporter
namespace/node-exporter created
[root@REDACTED ~]# helm install my-prometheus-node-exporter prometheus-community/prometheus-node-exporter --version 4.37.1 --set service.type=LoadBalancer -n node-exporter
```

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

```
[root@REDACTED ~]# kubectl get svc -n node-exporter --context=aks-cluster
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)        AGE
my-prometheus-node-exporter   LoadBalancer   10.REDACTED.141   135.REDACTED.160   9100:31055/TCP   3m3s
[root@REDACTED ~]# kubectl get svc -n node-exporter --context=eks-demo-cluster-dev
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)        AGE
my-prometheus-node-exporter   LoadBalancer   10.REDACTED.137   aREDACTED.us-east-2.elb.amazonaws.com   9100:32322/TCP   5m13s
```

Then I had updated the prometheus configuration file `/etc/prometheus/prometheus.yml` as shown in the screenshot attached below.

```
- job_name: "EKS"
  static_configs:
    - targets: ["aREDACTED.us-east-2.elb.amazonaws.com:9100"]
- job_name: "AKS"
  static_configs:
    - targets: ["135.REDACTED.160:9100"]

[root@REDACTED ~]# systemctl restart prometheus.service
[root@REDACTED ~]# systemctl status prometheus.service
● prometheus.service - Prometheus
  Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor preset: disabled)
  Active: active (running) since Mon 2025-05-05 12:05:04 UTC; 7s ago
```

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@192.168.1.11 ~]# git clone https://github.com/singhrithesh85/helm-chart-promtail.git
[root@192.168.1.11 ~]# kubectl create ns promtail && helm upgrade --install promtail ./helm-chart-promtail -f ./helm-chart-promtail/values.yaml -n promtail
1
namespace/promtail created
Release "promtail" does not exist. Installing it now.
NAME: promtail
LAST DEPLOYED: [REDACTED] 2025
NAMESPACE: promtail
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
*****
Welcome to Grafana Promtail
Chart version: 6.16.6
Promtail version: 3.0.0
*****
Verify the application is working by running these commands:
* kubectl --namespace promtail port-forward daemonset/promtail 3101
* curl http://127.0.0.1:3101/metrics
[root@Terraform-Server ~]# kubectl config get-contexts
CURRENT   NAME          CLUSTER          AUTHINFO          NAMESPACE
*         aks-cluster    aks-cluster      clusterUser_aks_rg_aks-cluster
*         eks-demo-cluster-dev  eks-demo-cluster-dev  arn:aws:eks:us-east-2:02[REDACTED]:cluster/eks-demo-cluster-dev

```

```
[root@yellow ~]# kubectl config use-context eks-demo-cluster-dev
Switched to context "eks-demo-cluster-dev".
[root@yellow ~]# kubectl create ns promtail && helm upgrade --install promtail ./helm-chart-promtail -f ./helm-chart-promtail/values.yaml -n promtail
1
namespace/promtail created
Release "promtail" does not exist. Installing it now.
NAME: promtail
LAST DEPLOYED: [REDACTED] 2025
NAMESPACE: promtail
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
*****
Welcome to Grafana Promtail
Chart version: 6.16.6
Promtail version: 3.0.0
*****
Verify the application is working by running these commands:
* kubectl --namespace promtail port-forward daemonset/promtail 3101
* curl http://127.0.0.1:3101/metrics

[root@yellow ~]# kubectl get pods -n promtail --context=aks-cluster
NAME      READY   STATUS    RESTARTS   AGE
promtail-[REDACTED] 1/1     Running   0          3m3s
promtail-[REDACTED] 1/1     Running   0          3m3s
[root@yellow ~]# kubectl get pods -n promtail --context=eks-demo-cluster-dev
NAME      READY   STATUS    RESTARTS   AGE
promtail-[REDACTED] 1/1     Running   0          79s
promtail-[REDACTED] 1/1     Running   0          79s
```

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.

AKS (1/1 up)					
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://135.***.9100/metrics	UP	instance="135.***.160.9100" job="AKS"	1m 1s ago	74.555ms	

BlacboxExporter-Server (1/1 up)					
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.4.98:9100/metrics	UP	instance="10.10.4.98:9100" job="BlacboxExporter-Server"	57.591s ago	12.688ms	

EKS (1/1 up)					
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://a.***.us-east-2.elb.amazonaws.com:9100	UP	instance="a.***.us-east-2.elb.amazonaws.com:9100" job="EKS"	51.432s ago	27.097ms	
amazonaws.com:9100/metrics	5				

Prometheus Alerts Graph Status Help

Grafana-Server (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.120:9100/metrics	UP	instance="10.10.120:9100" job="Grafana-Server"	50.609s ago	14.817ms	

Jenkins-Job (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.233.8080/prometheus	UP	instance="10.10.233.8080" job="Jenkins-Job"	56.930s ago	11.050ms	

Jenkins-Master (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.233.9100/metrics	UP	instance="10.10.233.9100" job="Jenkins-Master"	1m 0s ago	49.437ms	

Jenkins-Slave (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.15.9100/metrics	UP	instance="10.10.15.9100" job="Jenkins-Slave"	59.257s ago	15.488ms	

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

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.242.9100/metrics	UP	instance="10.10.242.9100" job="Loki-Server-1"	50.696s ago	15.097ms	

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

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.153.9100/metrics	UP	instance="10.10.153.9100" job="Loki-Server-2"	48.455s ago	12.057ms	

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

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.170.9100/metrics	UP	instance="10.10.170.9100" job="Loki-Server-3"	56.396s ago	13.456ms	

Nexus-Server (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.111.9100/metrics	UP	instance="10.10.111.9100" job="Nexus-Server"	53.686s ago	15.417ms	

Prometheus Alerts Graph Status Help

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"	7.67s ago	12.870ms	

SonarQube-Server (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.93.9100/metrics	UP	instance="10.10.93.9100" job="SonarQube-Server"	11.882s ago	15.020ms	

blackbox (2/2 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.98.9115/probe module="http_zxx_example" target="https://bankapp-dev.singhritesh85.com"	UP	instance="https://bankapp-dev.singhritesh85.com" job="blackbox"	4.42s ago	23.051ms	
http://10.10.4.98.9115/probe module="http_zxx_example" target="https://bankapp-stage.singhritesh85.com"	UP	instance="https://bankapp-stage.singhritesh85.com" job="blackbox"	13.989s ago	110.996ms	

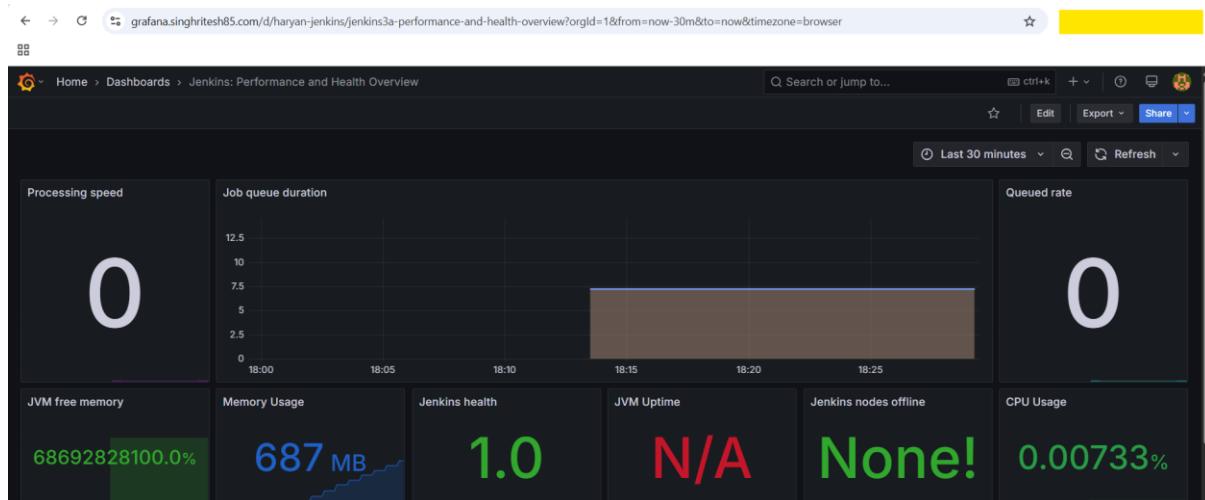
prometheus (1/1 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	13.644s ago	6.975ms	

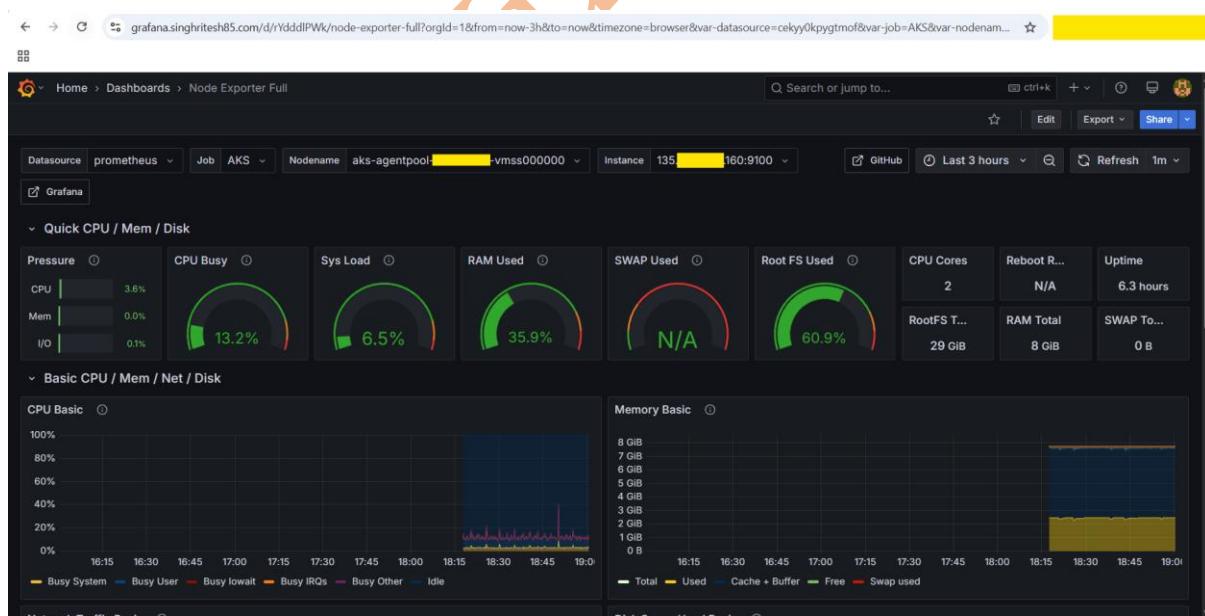
sonarqube (1/1 up)

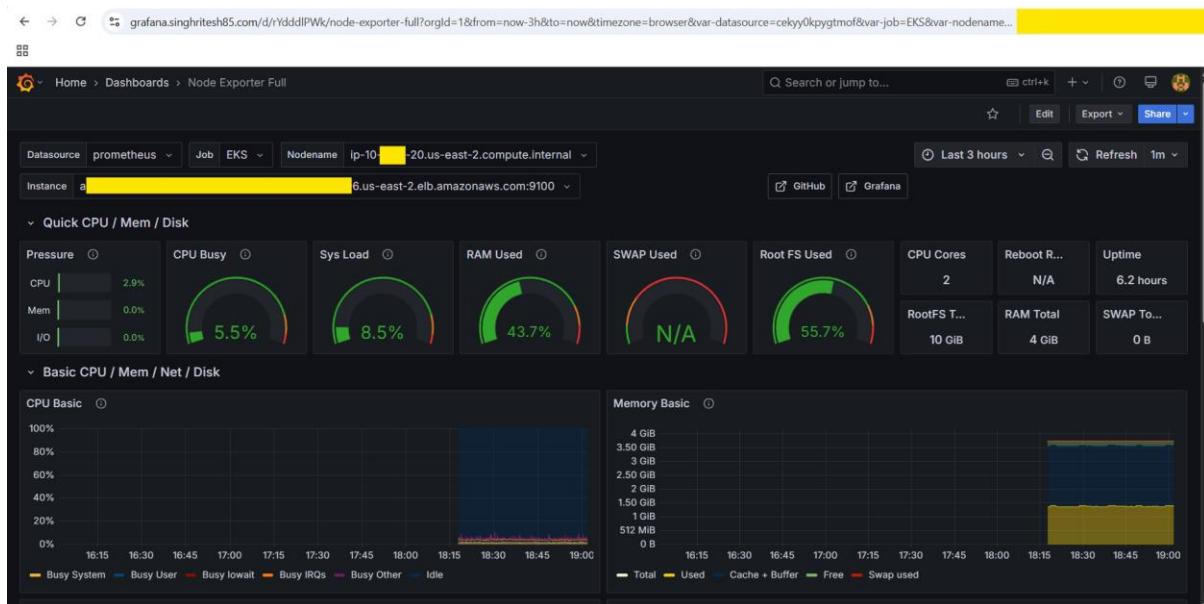
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.10.93.9000/api/prometheus/metrics	UP	instance="10.10.93.9000" job="sonarqube"	5.755s ago	163.547ms	

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

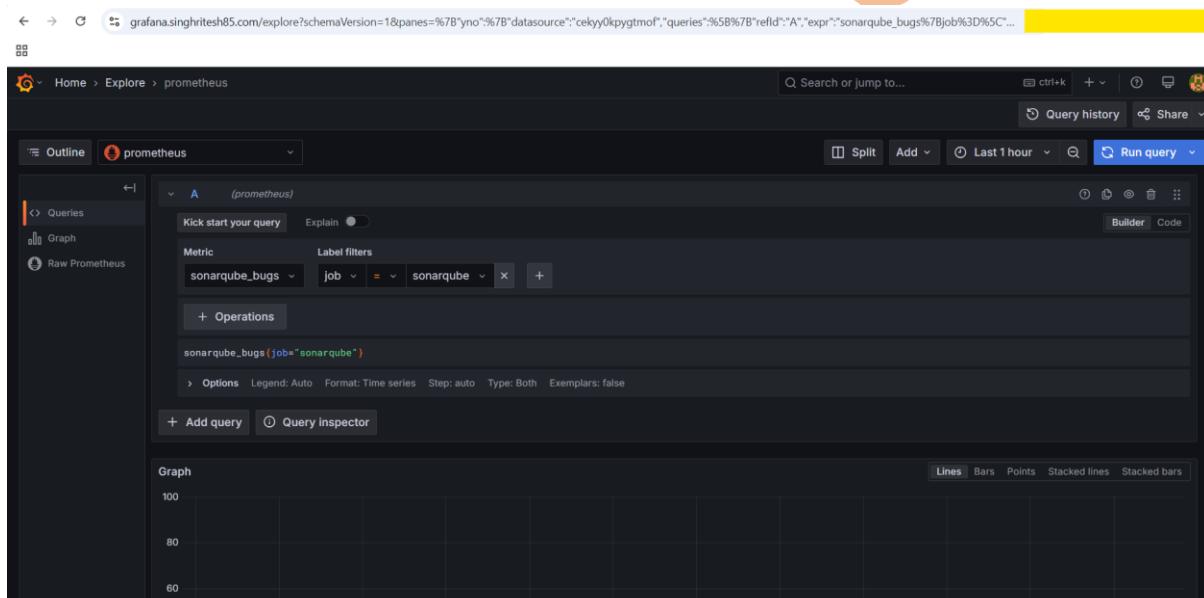


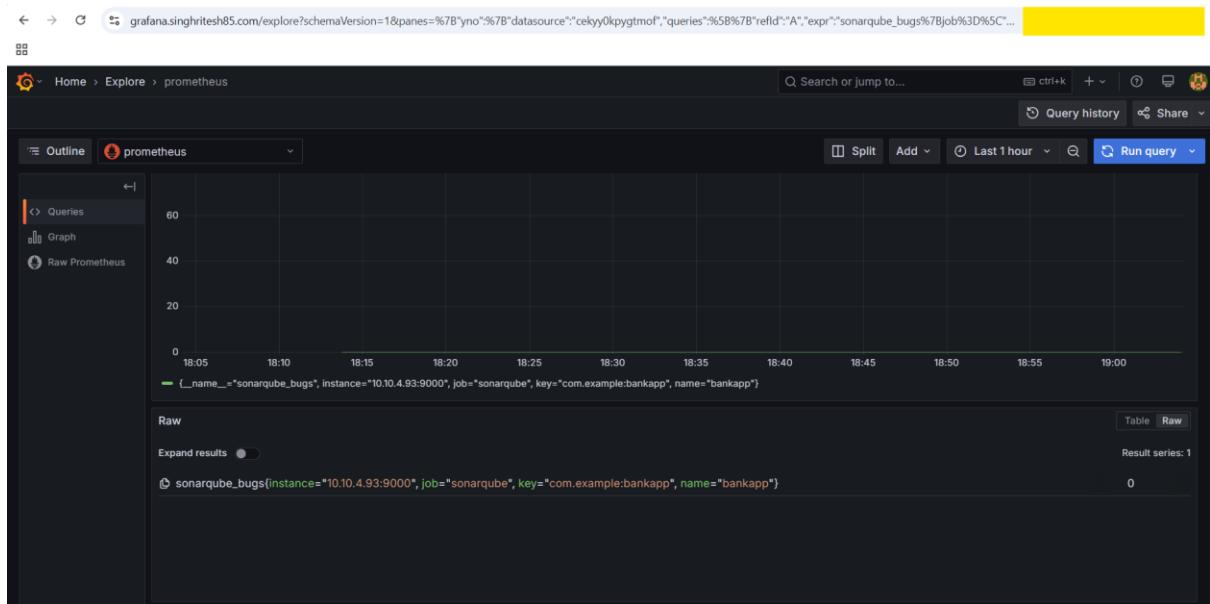
For Monitoring all the Servers and EKS and AKS Cluster health using the Node Exporter I used Grafana ID **1860**.



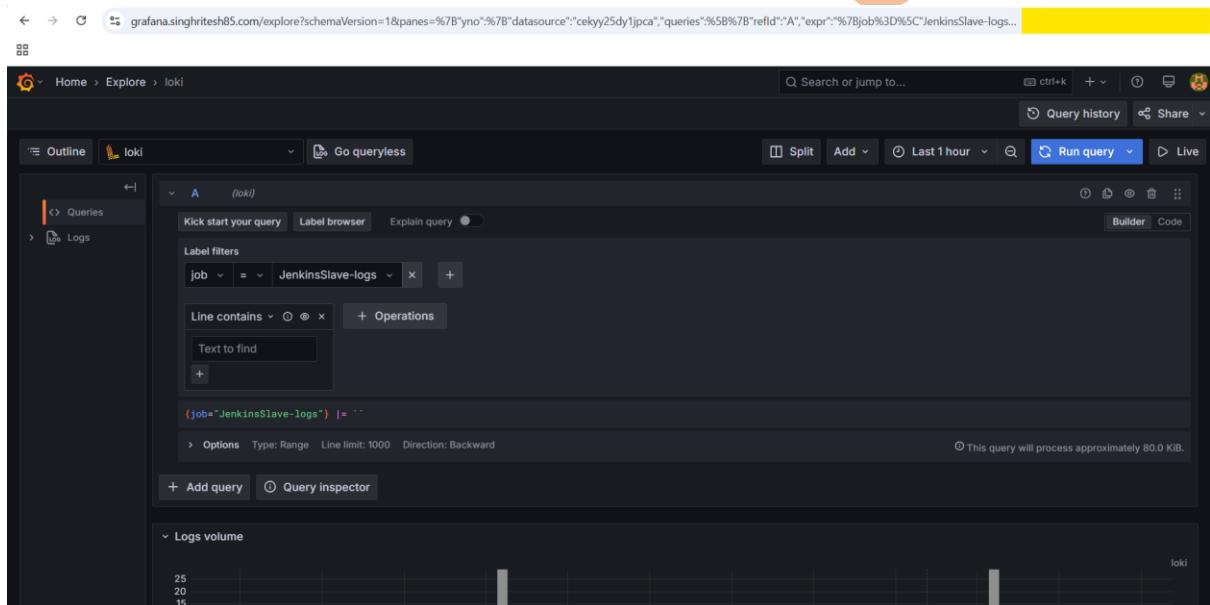


Grafana Metrics for SonarQube I started exploring as shown in the screenshot attached below.





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



```

Line limit: 1000 reached, received logs cover 21.29% (zh 33min 19sec) of your selected time range (12h). Total bytes processed: 761 kB

| 2025-05-05 16:34:42.65 May 5 11:04:42 ip-10-10-4-15 log4j:cve-2021-44228-hotpatch: [log4j-hotpatch] Hotpatch application returned e
|> 2025-05-05 16:34:42.657 May 5 11:04:42 ip-10-10-4-15 systemd: Removed slice User Slice of root.
|> 2025-05-05 16:34:41.986 May 5 11:04:41 ip-10-10-4-15 systemd: Started Session c1@0 of user root.
|> 2025-05-05 16:34:41.986 May 5 11:04:41 ip-10-10-4-15 systemd: Created slice User Slice of root.
|> 2025-05-05 16:34:41.986 May 5 11:04:41 ip-10-10-4-15 root: Using Java 17 hotpatch
|> 2025-05-05 16:34:41.986 May 5 11:04:41 ip-10-10-4-15 log4j:cve-2021-44228-hotpatch: [log4j-hotpatch] Using Java 17 hotpatch
|> 2025-05-05 16:34:41.986 May 5 11:04:41 ip-10-10-4-15 systemd: Started Session c1@0 of user jenkins.
|> 2025-05-05 16:34:41.986 May 5 11:04:41 ip-10-10-4-15 JVM(openjdk) of (openjdk version '17.0.15' 2025-04-15 LTS) with major version 17
|> 2025-05-05 16:34:41.986 May 5 11:04:41 ip-10-10-4-15 log4j:cve-2021-44228-hotpatch: [log4j-hotpatch] Identified JVM(openjdk) of (openjdk version '17.0.15' 2025-04-15
LTS) with major version 17
|> 2025-05-05 16:34:41.986 May 5 11:04:41 ip-10-10-4-15 root: JVM version is openjdk version "17.0.15" 2025-04-15 LTS
|> 2025-05-05 16:34:41.986 May 5 11:04:41 ip-10-10-4-15 log4j:cve-2021-44228-hotpatch: [log4j-hotpatch] JVM version is openjdk version "17.0.15" 2025-04-15 LTS
|> 2025-05-05 16:34:41.986 May 5 11:04:41 ip-10-10-4-15 systemd: Removed slice User Slice of root.
|> 2025-05-05 16:34:40.984 May 5 11:04:40 ip-10-10-4-15 systemd: Started Session c1@0 of user root.
|> 2025-05-05 16:34:40.984 May 5 11:04:40 ip-10-10-4-15 systemd: Created slice User Slice of root.
|> 2025-05-05 16:34:40.984 May 5 11:04:40 ip-10-10-4-15 root: Found JVM for pid 9134 at /usr/lib/jvm/java-17-amazon-corretto.x86_64/bin/java
|> 2025-05-05 16:34:40.984 May 5 11:04:40 ip-10-10-4-15 log4j:cve-2021-44228-hotpatch: [log4j-hotpatch] Found JVM for pid 9134 at /usr/lib/jvm/java-17-amazon-corretto.x
86_64/bin/java
|> 2025-05-05 16:34:40.984 May 5 11:04:40 ip-10-10-4-15 root: Found JVM running with effective UID of 1002
|> 2025-05-05 16:34:40.984 May 5 11:04:40 ip-10-10-4-15 log4j:cve-2021-44228-hotpatch: [log4j-hotpatch] Found JVM running with effective UID of 1002
|> 2025-05-05 16:34:38.988 May 5 11:04:38 ip-10-10-4-15 root: Attempting to patch 9134
|> 2025-05-05 16:34:38.988 May 5 11:04:38 ip-10-10-4-15 log4j:cve-2021-44228-hotpatch: [log4j-hotpatch] Attempting to patch 9134
|> 2025-05-05 16:34:38.988 May 5 11:04:38 ip-10-10-4-15 root: Found JVMs with pids [9134 978]
|> 2025-05-05 16:34:38.988 May 5 11:04:38 ip-10-10-4-15 log4j:cve-2021-44228-hotpatch: [log4j-hotpatch] Found JVMs with pids [9134 978]
|> 2025-05-05 16:34:38.988 May 5 11:04:38 ip-10-10-4-15 root: Starting up now...
|> 2025-05-05 16:34:38.988 May 5 11:04:38 ip-10-10-4-15 log4j:cve-2021-44228-hotpatch: [log4j-hotpatch] Starting up now...

```

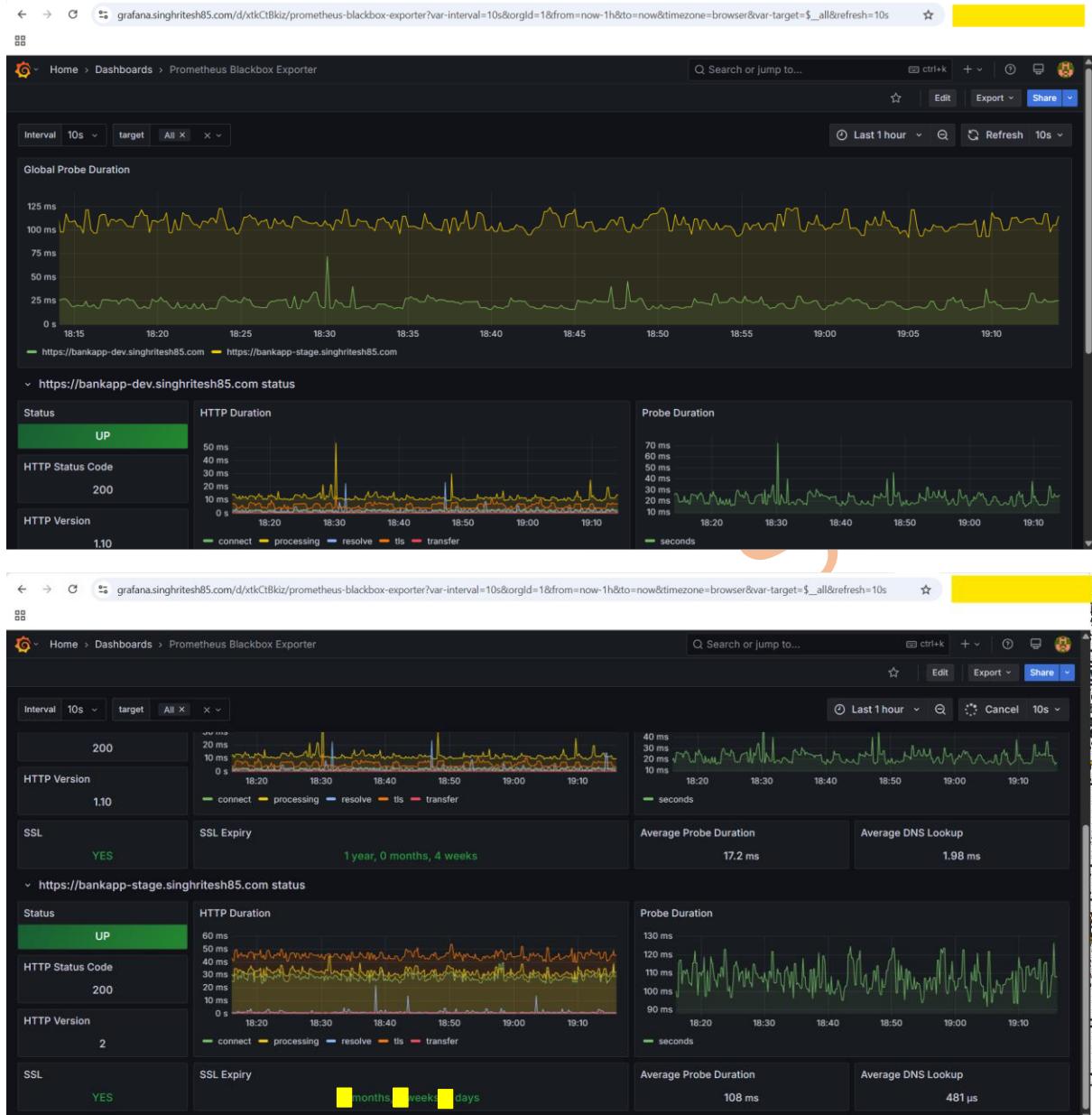
To achieve synthetic monitoring using Prometheus Blackbox Exporter I updated the `/etc/resolv.conf` file for Blackbox Exporter Server as shown in the screenshot attached below. I had used Google's Public DNS Server which is shown in the attached screenshot below.

```
[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-dev.singhrites85.com> and <https://bankapp-stage.singhrites85.com> had been monitored using blackbox exporter.

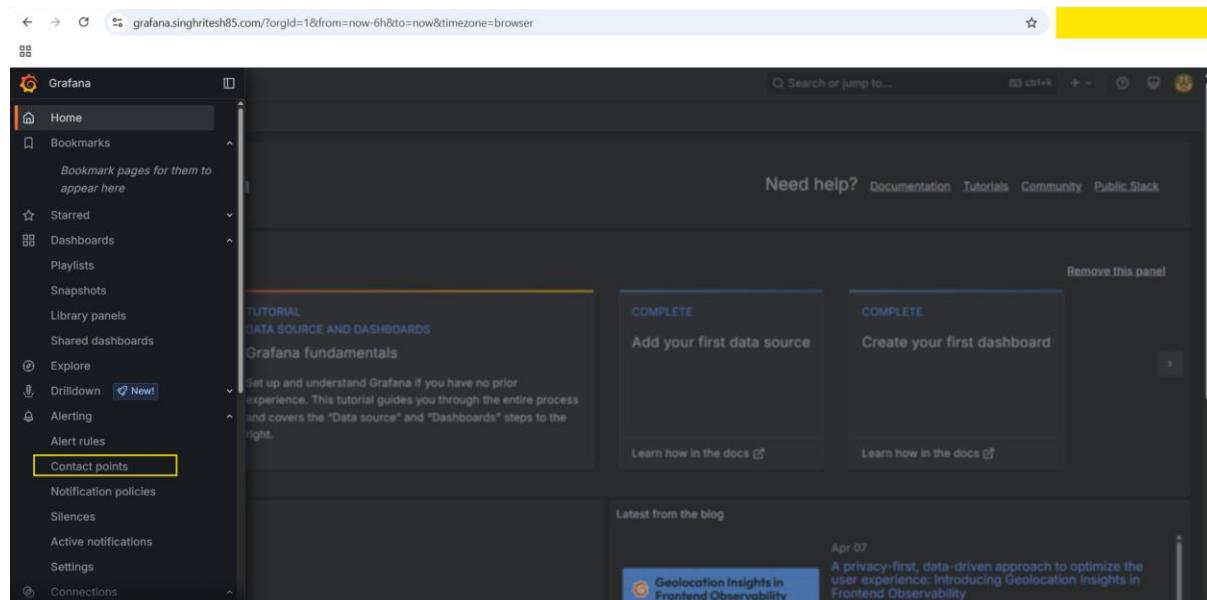
I had installed Blackbox Exporter on a different server and not on the Prometheus Server. The **module name** is `monitor_website.yml` present of the blackbox exporter server at the path `(/opt/blackbox_exporter_linux_amd64/monitor_website.yml)`. Prometheus blackbox operator is used for endpoint monitoring (Synthetic Monitoring) across the protocol http, https, TCP and ICMP. In this project I am monitoring the Application URL <https://bankapp-dev.singhrites85.com> and <https://bankapp-stage.singhrites85.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.



The screenshot shows the Grafana homepage with the 'Contact points' section highlighted in yellow. The left sidebar includes options like Home, Bookmarks, Starred, Dashboards, Playlists, Snapshots, Library panels, Shared dashboards, Explore, Drilldown, Alerting, Alert rules, Contact points (which is selected), Notification policies, Silences, Active notifications, Settings, and Connections.

Contact points

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

Create contact point

Name * mederma

Integration Email

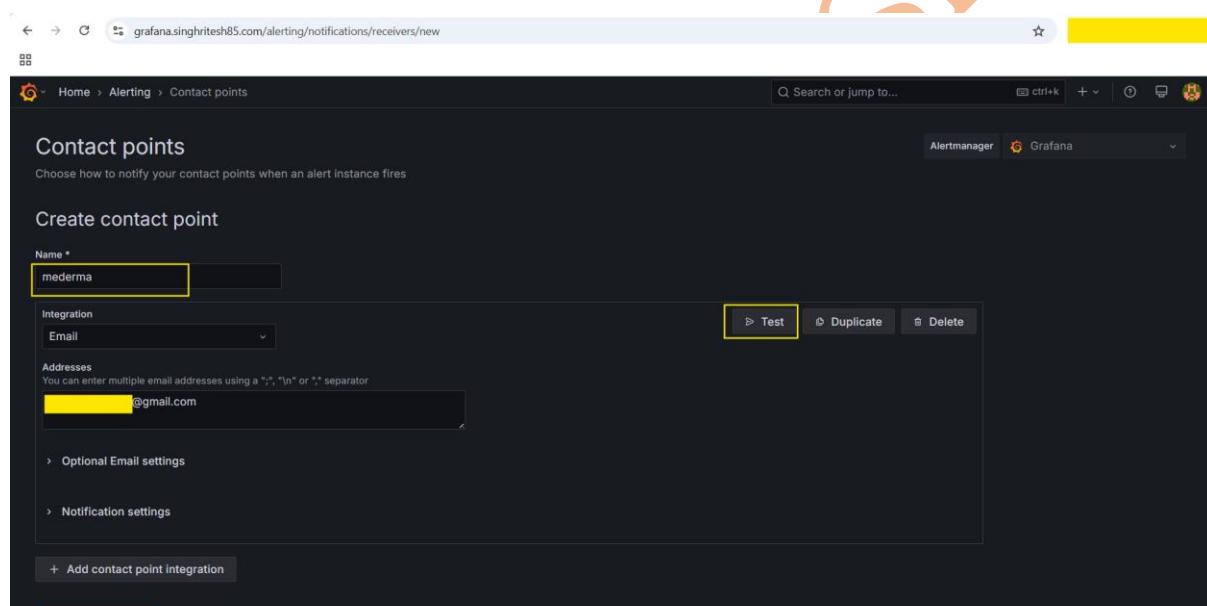
Addresses You can enter multiple email addresses using a ";", "\n" or "," separator mederma@gmail.com

Optional Email settings

Notification settings

+ Add contact point integration

Save contact point Cancel



The screenshot shows the 'Create contact point' form with the 'Test' button highlighted in yellow. The 'Test' button is located at the bottom right of the main form area.

Contact points

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

Create contact point

Name * mederma

Integration Email

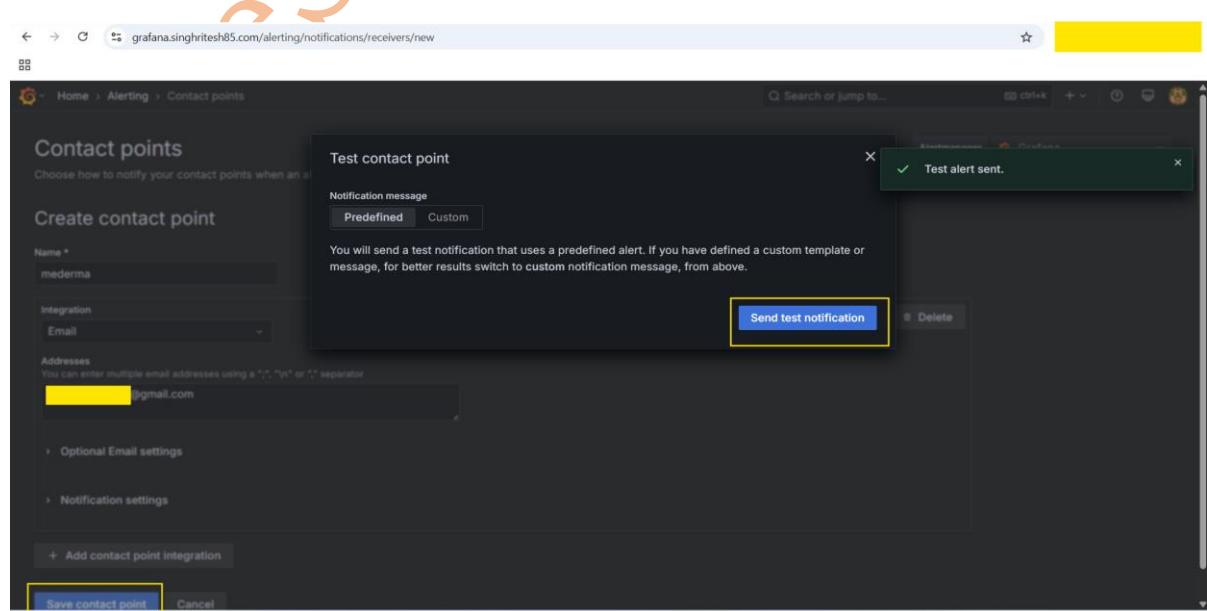
Addresses You can enter multiple email addresses using a ";", "\n" or "," separator mederma@gmail.com

Optional Email settings

Notification settings

+ Add contact point integration

Save contact point Cancel



The screenshot shows the 'Test contact point' dialog with the 'Send test notification' button highlighted in yellow. A success message 'Test alert sent.' is displayed in a green toast notification.

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

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?](#) Advanced options

A prometheus Options 10 minutes Set as alert condition
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"}
Options Legend: Auto Format: Time series Step: auto Type: Instant

Rule type
Select where the alert rule will be managed. [Need help?](#)
Grafana-managed Data source-managed
Based on the selected data sources this alert rule will be Grafana-managed.

Expressions
Manipulate data returned from queries with math and other operations.

C Threshold Alert condition
Takes one or more time series returned from a query or an expression and checks if any of the series match the threshold condition.
Input B IS ABOVE 0.7 Custom recovery threshold
B Reduce Set "B" as alert condition
Takes one or more time series returned from a query or an expression and turns each series into a single number.
Input A Function Last Mode Strict

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...

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 behaviorDefine 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

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.

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...

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.

Advanced options

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 [View or create contact points](#)

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 [O](#) Dashboard State Rule type

All data sources Select dashboard Firing Normal Pending Alert Recording

Health Contact point

Ok No Data Error Choose

Search [Q Search](#) View as Grouped List State

1 rule **1 firing** [Export rules](#)

Grafana-managed

CPU time > CPU time [1 firing](#) | [1m](#) | [%](#) [↻](#)

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

Data source-managed

No rules found. [+ New recording rule](#)

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

[FIRING:1] zago-medderma CPU time (a [REDACTED] 6.us-east-2.elb.amazonaws.com:9100 EKS) [Inbox](#)

R Multibranch Pipeline Grafana Alert for Bankapp <[REDACTED]@gmail.com> to me [View alert](#)

Grafana

CPU time > zago-medderma

1 firing instances

Firing	zago-medderma	View alert
Values	A=9.29 B=9.29 C=1	
Labels	alertname zago-medderma	

Grafana

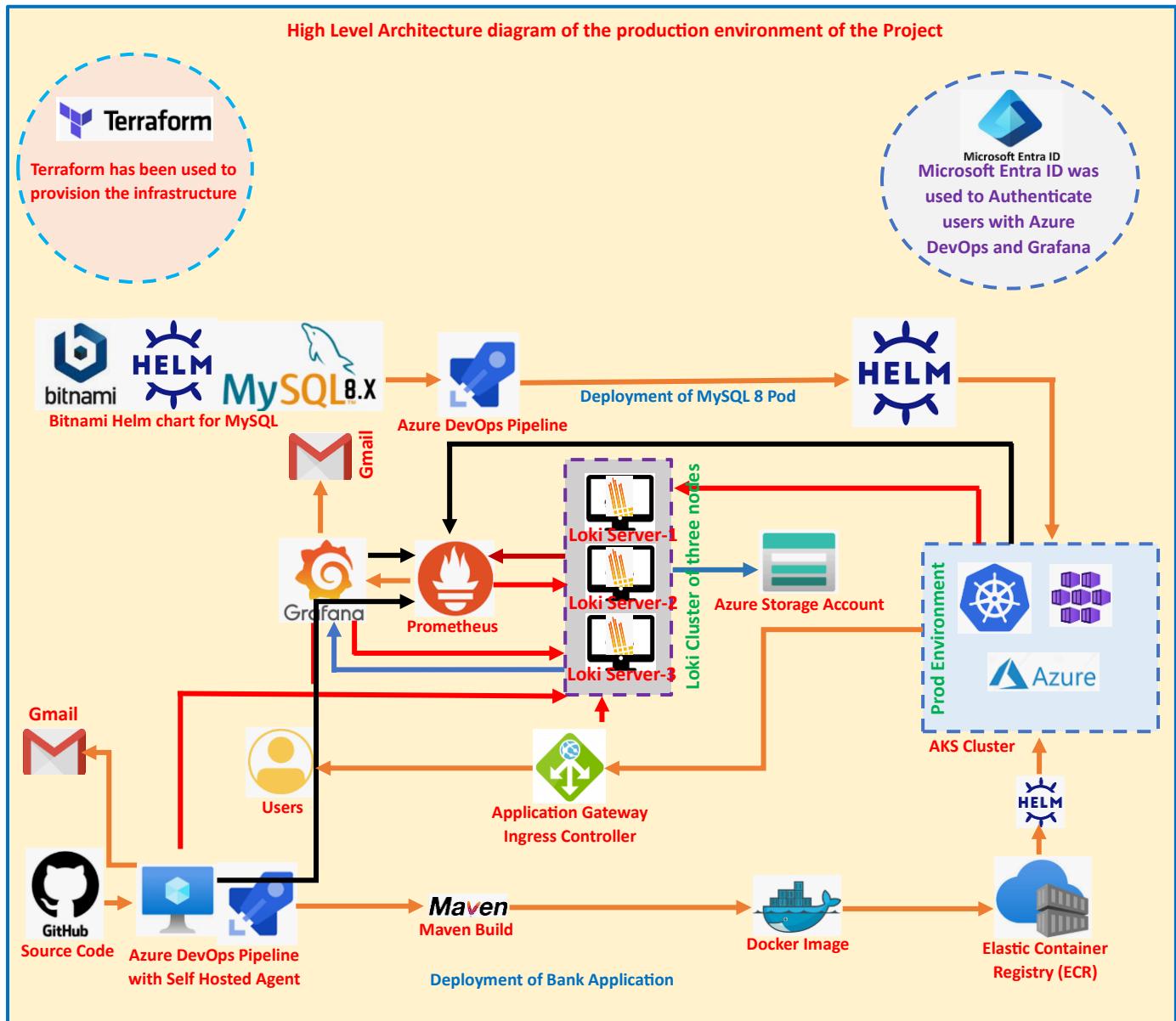
CPU time > zago-medderma

1 firing instances

Firing	zago-medderma	View alert
Values	A=9.29 B=9.29 C=1	
Labels	alertname zago-medderma grafana_fold er CPU time instance a [REDACTED] 6.us-east-2.elb.amazonaws.com:9100 job EKS	

In this project to refrain from higher cloud cost for **dev** and **stage** environment I had used the same ECR (Elastic Container Registry) but different tag name for the Docker Images. You can use different ECR for your project in **dev** and **stage** environment.

Module-2: [Production Environment]



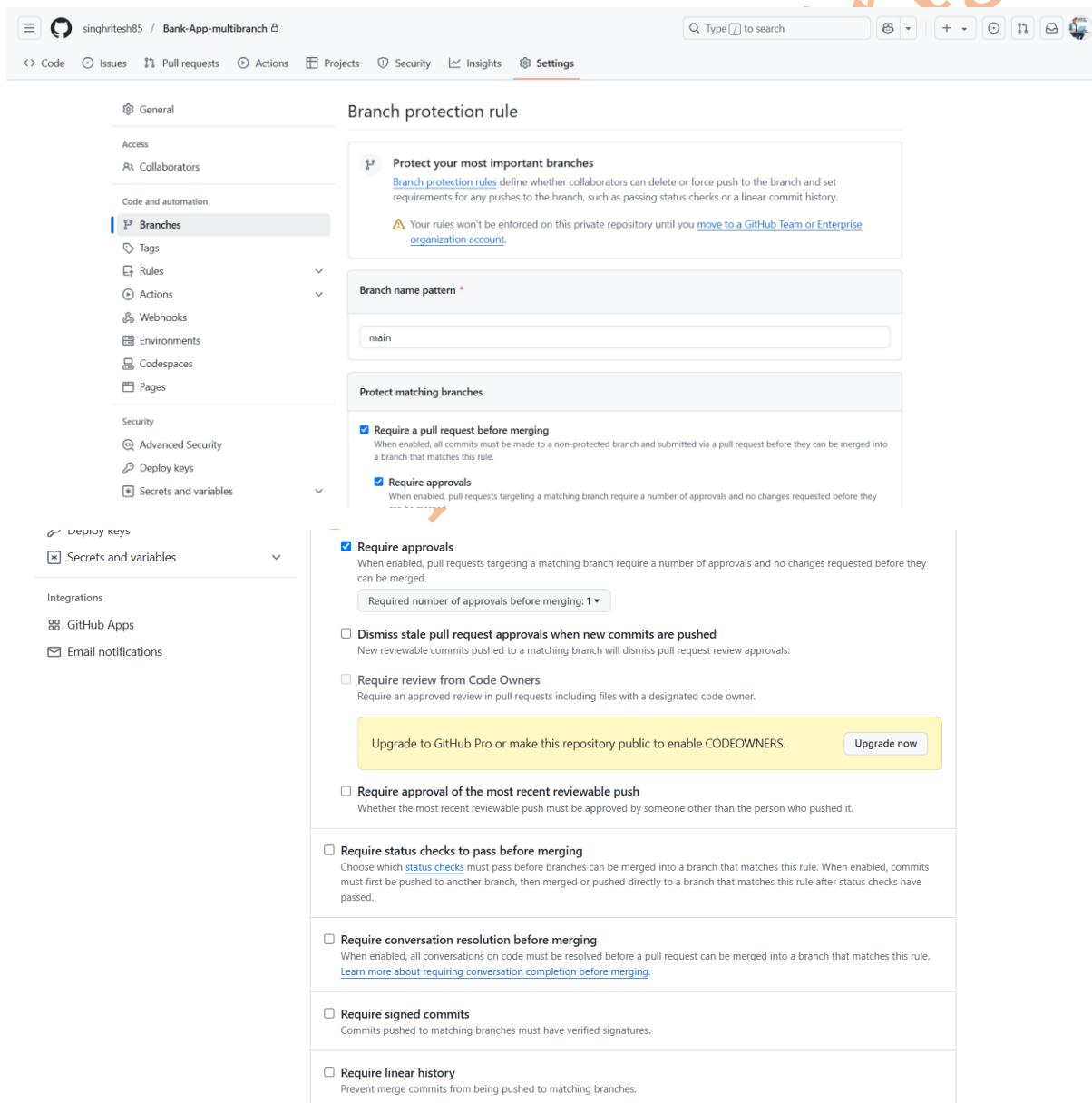
In the above diagram the black arrow (\longrightarrow) indicated the metrics exported by node exporter to the prometheus and red arrow (\longrightarrow) indicates logs extracted by the promtail to loki.

GitHub Branching Strategies

In some of the organisation there are three environments **dev**, **stage** and **prod** and there may be one more branch **pre-prod** which depends on the project and the organisation. Before proceeding further, I discuss here first **GitHub Branch Protection**. In Real time projects Developers are allowed to push their code in dev environment and if it becomes successful after proper testing of the deployed code in dev branch then dev branch will be merged to the stage branch using Pull Request (PR). To approve the pull request of stage environment I would suggest you can keep two approvers, you can keep two approvers as your team-mates. Developers will create a **feature branch** from the **dev branch** then after doing their coding part in **feature branch** they will merge their code into **dev**

branch. For dev branch Pull Request (PR) you can keep one approver who can be your team-mate. Finally, to merge code into the **main branch** from **stage branch** at least three approvals required, the three approvals required including the dignitary member of the organisation because the code merged in this branch will go live. You can create the webhook in Jenkins Pipeline or Azure DevOps Pipeline, so that whenever the code will be merged into the **dev/stage** or **main branch** then it will be deployed automatically. However, I will suggest do not apply webhook in production environment and provide its access to the DevOps Team and DevOps Team should run the appropriate Jenkins Job with required parameters after getting the approval from the required dignitary member of your organisation on **Jira ticket**.

In this project for demonstration purpose, I had applied the GitHub Branch Protection rule in only main branch as shown in the screenshot attached below. No one can push directly to the main branch developer can only merge the stage branch to the main branch after required approval as shown in the screenshot attached below.



The screenshot shows the GitHub repository settings for 'Bank-App-multibranch'. The 'Branches' tab is selected under the 'Code' section. On the right, the 'Branch protection rule' section is displayed. It shows a 'Branch name pattern' of 'main'. Under 'Protect matching branches', two rules are enabled: 'Require a pull request before merging' and 'Require approvals'. The 'Require approvals' rule is set to require 1 approval. Other optional rules like 'Dismiss stale pull request approvals when new commits are pushed' and 'Require review from Code Owners' are disabled. A yellow callout highlights the 'Require approvals' section. At the bottom, there's a note about enabling CODEOWNERS and an 'Upgrade now' button.

Require linear history
Prevent merge commits from being pushed to matching branches.

Require deployments to succeed before merging
Choose which environments must be successfully deployed to before branches can be merged into a branch that matches this rule.

Lock branch
Branch is read-only. Users cannot push to the branch.

Do not allow bypassing the above settings
The above settings will apply to administrators and custom roles with the "bypass branch protections" permission.

Rules applied to everyone including administrators

Allow force pushes
Permit force pushes for all users with push access.

Allow deletions
Allow users with push access to delete matching branches.

Create

Branch protection rules

Add rule

main

Currently applies to 1 branch

Edit

Delete

For this project to refrain from higher cloud bills I used the Instance Type of t3.medium and Standard_B2s. However, in Organisation you can select the Instance type of m4.4xlarge/t3.2xlarge and vm_size of Standard_DS3_v2/Standard_DS4_v2/Standard_DS5_v2 or some other Instance Type/VM Size depending on your project requirement.

I installed terraform on Alma Linux2 Azure VM and the State file was kept in and state lock had been achieved using the Azure Storage Account container. Before running the terraform script I ran the shell script present in the directory **terraform-bankapp-prod-azure** as shown in the screenshot attached below. This shell script will install the aws cli, kubectl and helm.

```
[root@REDACTED terraform-multi-kubernetes-cluster-multicloud]# ./initial-setup.sh
```

Then I logged-out and login again. Then I installed Azure CLI and authenticated and authorize the user as shown in the screenshot attached below.

```
[root@REDACTED main]# yum install -y https://packages.microsoft.com/config/rhel/8/packages-microsoft-prod.rpm
[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://login.microsoftonline.com/b2c932f2-527b-4093-9eef-8cde774723c8 and enter the code REDACTED to authenticate.
```

Then you can run the below commands

terraform init -----> initializes a working directory containing configuration files and installs plugins for required providers.

terraform validate -----> verify that terraform configuration file is correct or not

terraform plan -----> Check which resources are going to be created.

Then you can run the command **terraform apply -auto-approve** -----> Finally, Create the resources.

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

```
module.aks.null_resource.kubectl: Creation complete after 3s [id=...]
Apply complete! Resources: 84 added, 0 changed, 0 destroyed.

Outputs:

acr_azure_loki_vm_private_ip = {
  "acr_login_server" = "bankappcontainer24registry.azurecr.io"
  "azure_loki_vm" = [
    "...",
    "...",
    "...",
  ]
}
```

This terraform script took total 20-25 minutes in all to get executed successfully as shown in the screenshot attached above. After that All the resources in Azure had been created. As the created AKS Cluster was Private AKS Cluster (API Server was private) and it was created in the different VNet as compared with the Terraform-Server. If you want to access AKS Cluster from the Terraform-Server using kubectl command then you need to create the VNet Peering between the Terraform-Server VNet and AKS Cluster VNet. Then you also need to create the Virtual Network Link in Private DNS Zone for Private AKS Cluster as shown in the screenshot attached below.

[Home](#) > [Virtual networks](#) > [bankapp-vnet | Peerings](#) >

Add peering

...

bankapp-vnet

Virtual network peering enables you to seamlessly connect two or more virtual networks in Azure. This will allow resources in either virtual network to directly connect and communicate with resources in the peered virtual network.

Remote virtual network summary

Peering link name *	<input type="text" value="peer-2"/>
Virtual network deployment model ⓘ	<input checked="" type="radio"/> Resource manager <input type="radio"/> Classic
I know my resource ID ⓘ	<input type="checkbox"/>
Subscription *	<input type="text" value="Pay-As-You-Go"/>
Virtual network *	<input type="text" value="Terraform-Server-vnet (ritesh)"/>

Remote virtual network peering settings

Allow 'Terraform-Server-vnet' to access	<input checked="" type="checkbox"/>
Add Cancel	

Home > Virtual networks > bankapp-vnet | Peerings >

Add peering ...

bankapp-vnet
Server

Local virtual network summary

Peering link name *

peer-1

Local virtual network peering settings

Allow 'bankapp-vnet' to access 'Terraform-Server-vnet'

Allow 'bankapp-vnet' to receive forwarded traffic from 'Terraform-Server-vnet'

Allow gateway or route server in 'bankapp-vnet' to forward traffic to 'Terraform-Server-vnet'

Enable 'bankapp-vnet' to use 'Terraform-Server-vnet's remote gateway or route server

Add

Cancel

bankapp-vnet | Peerings

Virtual network

	Name	Peering sync status	Peering state	Remote virtual network name	Virt...	Cross-tenant
	peer-1	<input checked="" type="checkbox"/> Fully Synchronized	<input checked="" type="checkbox"/> Connected	Terraform-Server-vnet	Disabled	No

Terraform-Server-vnet | Peerings

Virtual network

	Name	Peering sync status	Peering state	Remote virtual network name	Virt...	Cross-tenant
	peer-2	<input checked="" type="checkbox"/> Fully Synchronized	<input checked="" type="checkbox"/> Connected	bankapp-vnet	Disabled	No

Home > Resource groups > bankapp-rg > .privatelink.eastus.azmk8s.io

Overview

- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Resource visualizer
- Locks
- Properties
- Records
- Virtual Network Links**

Essentials

Resource group (move) :	bankapp-rg	Records	: 2
Location	Global	Virtual Network Links	: 1
Subscription (move)	Pay-As-You-Go	Virtual Network Links Wi...	: 0
Subscription ID	5		
Tags (edit)	Add tags		

Monitoring Capabilities Tutorials Tools + SDKs Recommendations (0)

Azure Monitor Tools

- Get alerted to issues Create alerts to monitor resource
- Get started with Log Analytics Leverage the value of the data in
- Monitor at scale with Insights Get visibility into the resource's

Home > Resource groups > bankapp-rg > .privatelink.eastus.azmk8s.io | Virtual Network Links

Virtual Network Links

+ Add Refresh Delete Give feedback

Link Name	Link Status	Virtual Network	Auto-Registration	Fallback to Internet
bankapp-cluster-dns	Completed	bankapp-vnet	Disabled	Disabled

Add Virtual Network Link

Link name *

Virtual network details

Only virtual networks with Resource Manager deployment model are supported for linking with Private DNS zones. Virtual networks with Classic deployment model are not supported.

I know the resource ID of virtual network

Subscription *

Virtual Network *

Configuration

Enable auto registration Enable fallback to internet

Create **Cancel** Give feedback

The screenshot shows the Azure portal interface for a private DNS zone. On the left, the navigation pane includes options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Settings (Locks, Properties), DNS Management, Recordsets, and Virtual Network Links (which is selected). The main content area displays a table of virtual network links:

Link Name	Link Status	Virtual Network	Auto-Registration	Fallback to Internet
bankapp-cluster-dns	Completed	bankapp-vnet	Disabled	Disabled
dexter	Completed	Terraform-Server-vnet	Disabled	Disabled

Below this, there are two terminal windows showing the output of the `kubectl get nodes` command:

```
[root@Terraform-Server ~]# kubectl get nodes
NAME                               STATUS   ROLES      AGE   VERSION
aks-agentpool-[REDACTED]-vmss000000   Ready    <none>   52m   v1.30.0
aks-userpool-[REDACTED]-vmss000000   Ready    <none>   37m   v1.30.0
```

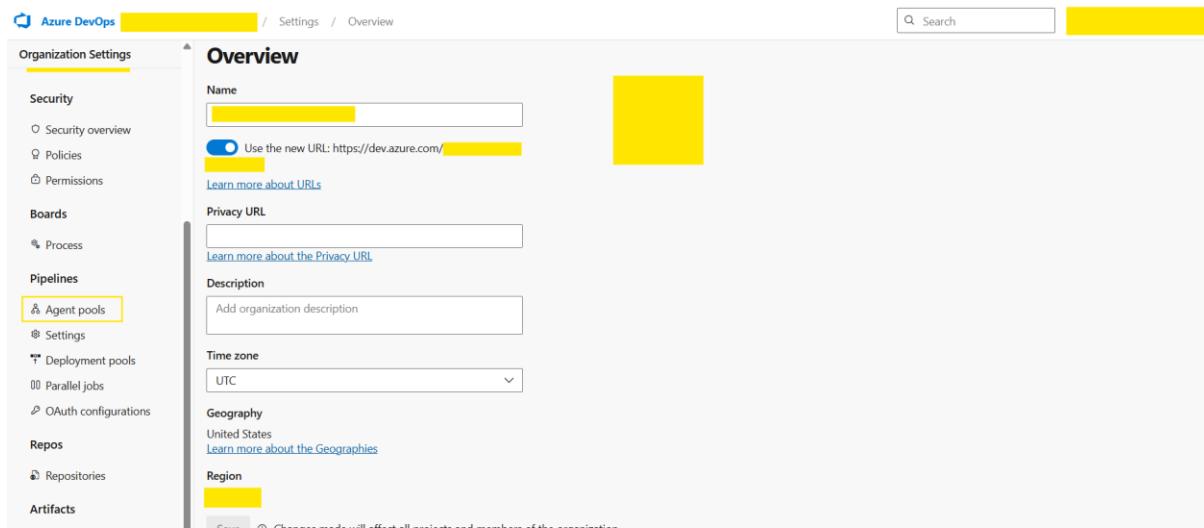


```
[root@Terraform-Server ~]# kubectl get nodes --kubeconfig=/root/.kube/config
NAME                               STATUS   ROLES      AGE   VERSION
aks-agentpool-[REDACTED]-vmss000000   Ready    <none>   55m   v1.30.0
aks-userpool-[REDACTED]-vmss000000   Ready    <none>   40m   v1.30.0
```

In production for CI/CD pipeline I used Azure DevOps Pipeline. For Azure DevOps Pipeline I had used Self-hosted-Agent and followed the below procedure to install it.

To install the self-hosted agent for Azure DevOps pipeline first I opened the Azure DevOps UI and went to Organisations Settings > Agent Pools > Add Pool.

The screenshot shows the Azure DevOps organization settings page. At the top, there are tabs for New organization, Projects (selected), My work items, and My pull requests. A search bar and a 'New project' button are also present. Below the tabs, a list of projects is shown, with 'my-demo-project' highlighted. At the bottom left, there is a link to 'Organization settings'.



The screenshot shows the 'Organization Settings' page in Azure DevOps. The 'Agent pools' section is highlighted with a yellow box. It lists two existing agent pools: 'Azure Pipelines' and 'Default'. A 'Save' button and a note about changes affecting all projects are visible at the bottom.

Agent pools

Name	Queued jobs	Running jobs
Azure Pipelines		
Default		

Add agent pool

Agent pools are shared across an organization.

Managed DevOps Pool
Reduce the effort spent in maintaining custom agents by creating a Microsoft managed pool of scalable agents. [Learn more](#).

Self-hosted
Create a pool of custom agents hosted on your own infrastructure for maximum control and flexibility. [Learn more](#).

Azure virtual machine scale set
Create a pool of custom agents based on an Azure Virtual machine scale set hosted in your own Azure subscription. [View configuration instructions](#).

Name:

Description (optional):

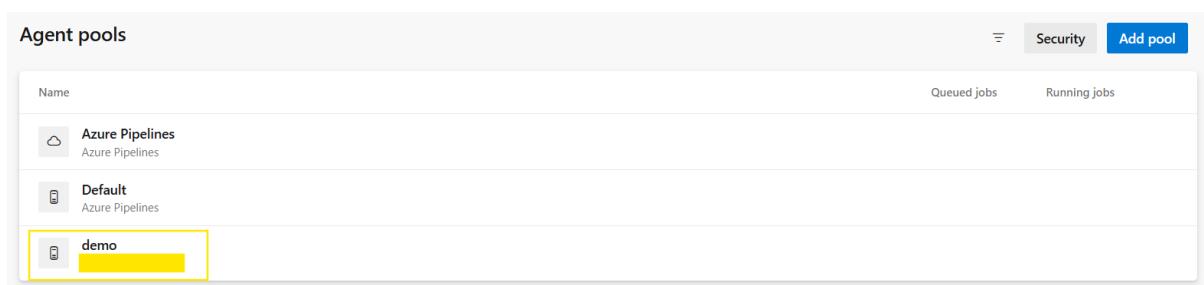
Markdown supported.

Pipeline permissions:

Auto-provision this agent pool in all projects

Create

New Agent Pool had been added as shown in the scrrenshot attached below.



The screenshot shows the 'Agent pools' page after adding a new pool. The 'demo' pool is now listed in the table along with 'Azure Pipelines' and 'Default'.

Agent pools

Name	Queued jobs	Running jobs
Azure Pipelines		
Default		
demo		

Then added the agent as shown in the screenshot attached below.

The screenshot shows the 'Agents' section of the Azure DevOps interface. At the top, there's a navigation bar with 'Jobs', 'Agents', 'Details', 'Security', 'Settings', 'Maintenance History', and 'Analytics'. On the right, there are buttons for 'Update all agents' and 'New agent' (which is highlighted with a yellow border). Below the navigation, there's a small illustration of a person standing on a beach with a dog, looking through a telescope at the sky. The main message is 'No jobs have run on this agent pool' with a link to 'Run a pipeline on this agent pool to see more details'.

For Azure DevOps Pipeline I had used Self-hosted-Agent and followed the below procedure to install it.

```
[root@devopsagent-vm ~]# cd /opt && mkdir myagent && cd myagent
[root@devopsagent-vm myagent]# wget https://vstsagentpackage.azureedge.net/agent/[REDACTED]/vsts-agent-linux-x64-[REDACTED].tar.gz
[root@devopsagent-vm myagent]# tar -xvf vsts-agent-linux-x64-[REDACTED].tar.gz
[root@devopsagent-vm myagent]# rm -f vsts-agent-linux-x64-[REDACTED].tar.gz
[root@devopsagent-vm myagent]# ./bin/installdependencies.sh
[demo@devopsagent-vm myagent]$ sudo chown -R demo:demo /opt/myagent/
[demo@devopsagent-vm myagent]$ ./config.sh

AZURE PIPELINES
agent [REDACTED] [REDACTED]

>> End User License Agreements:
Building sources from a TFVC repository requires accepting the Team Explorer Everywhere End User License Agreement. This step is not required for building sources from Git repositories.

A copy of the Team Explorer Everywhere license agreement can be found at:
/opt/myagent/license.html

Enter (Y/N) Accept the Team Explorer Everywhere license agreement now? (press enter for N) > Y
>> Connect:
Enter server URL > [REDACTED]
Enter authentication type (press enter for PAT) >
Enter personal access token > [REDACTED]
Connecting to server ...
>> Register Agent:
Enter agent pool (press enter for default) > demo
Enter agent name (press enter for devopsagent-vm) > demo
Scanning for tool capabilities.
Connecting to the server.
Successfully added the agent
Testing agent connection.
Enter work folder (press enter for _work) >
2025 [REDACTED] Settings Saved.

[demo@devopsagent-vm myagent]$ ./env.sh
[demo@devopsagent-vm myagent]$
[demo@devopsagent-vm myagent]$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/bin:/opt/sonar-scanner/bin:/opt/apache-maven/bin:/opt/node-v16.0.0/bin:/opt/dependency-check/bin:/usr/local/bin

[demo@devopsagent-vm myagent]$ sudo ./svc.sh install
[demo@devopsagent-vm myagent]$ sudo ./svc.sh start
```

Now the Agent came in online state as shown in the screenshot attached below.

The screenshot shows the 'Agents' tab in the Azure DevOps interface. A single agent named 'demo' is listed. The 'Current status' column shows 'Idle', the 'Agent version' column shows a yellow bar, and the 'Enabled' column has a toggle switch set to 'On'. The 'demo' row is highlighted with a yellow background.

The Azure DevOps account was connected to the Azure Entra ID as shown in the screenshot attached below.

The screenshot shows the 'Organization Settings' page in the Microsoft Entra interface. It displays a message stating 'Your organization is connected to the **Default Directory** directory.' Below this, it shows the 'Default Directory' logo and the 'Tenant Id: [REDACTED]'. There is a 'Disconnect directory' button and a link to 'other frequently asked questions'. At the bottom, there is a 'Download' button for 'Azure DevOps organizations connected to **Default Directory** directory.'

Then I went into the **Azure DevOps project > Project Settings** and created the **Service connections** for GitHub Account, Docker Registry as shown in the screenshot attached blow.

The screenshot shows the 'Project settings' page in the Azure DevOps interface. On the left, there is a sidebar with links like Overview, Summary, Dashboards, Wiki, Boards, Repos, Pipelines, Test Plans, and Artifacts. The 'Project settings' link at the bottom of the sidebar is highlighted with a yellow box. The main content area shows the 'About this project' section with a 'Help others to get on board!' placeholder and a 'Add Project Description' button. To the right, there is a cartoon illustration of a person running on clouds.

The screenshot shows the 'Service connections' page in the Azure DevOps interface. The left sidebar has 'Service connections' selected under 'Project Settings'. The main area shows a list of service connections, with one entry highlighted.

It is possible to Import the GitHub Repo to the Azure DevOps Repo as shown in the screenshot attached below.

For Private GitHub Repo It is necessary to provide the Username and Password/Personal Access Token (PAT) of your GitHub Account

Repository "dexter" not found

It is possible the repository did exist at one point, but your administrator renamed or deleted it. Please ensure that the repository exists and that you have access.

Import a Git repository

Repository type: Git

Clone URL *: https://github.com/singhritesh85/Bank-App-multibranch.git

Requires Authentication:

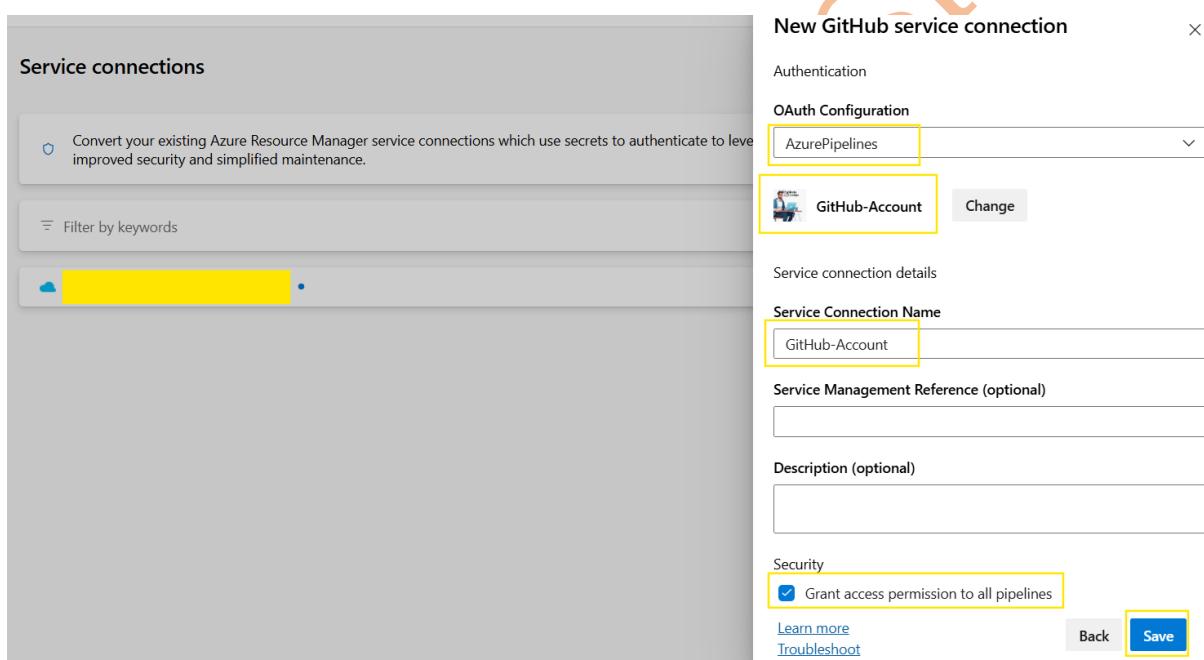
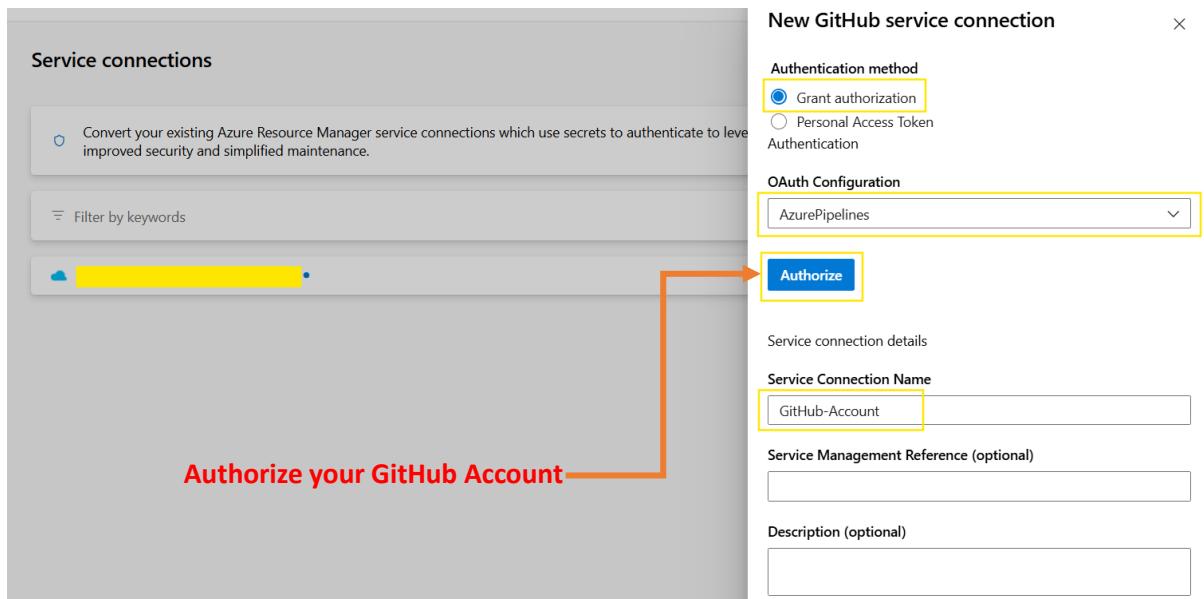
Username: [REDACTED]

Password / PAT *: [REDACTED]

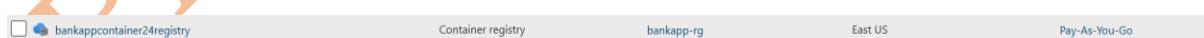
Name *: Bank-App-multibranch

Cancel Import

However, for this project I connected GitHub Account with Azure DevOps using the Service Connection as shown in the screenshot attached below.



The Azure Container Registry which I had created for this project is as shown in the screenshot attached below.



Then I had created the Service Connection for Docker Registry as shown in the screenshot attached below.

Container registries

bankappcontainer24registry | Access keys

Registry name: bankappcontainer24registry

Login server: bankappcontainer24registry.azurecr.io

Admin user:

- Name:** bankappcontainer24registry
- Password:** (Copied)
- Password2:** (Show)

User:

- Name:** password
- Password:** (Copied)
- Password2:** (Show)

New Docker Registry service connection

Registry type: Others (selected)

Docker Registry: https://bankappcontainer24registry.azurecr.io

Docker ID: bankappcontainer24registry

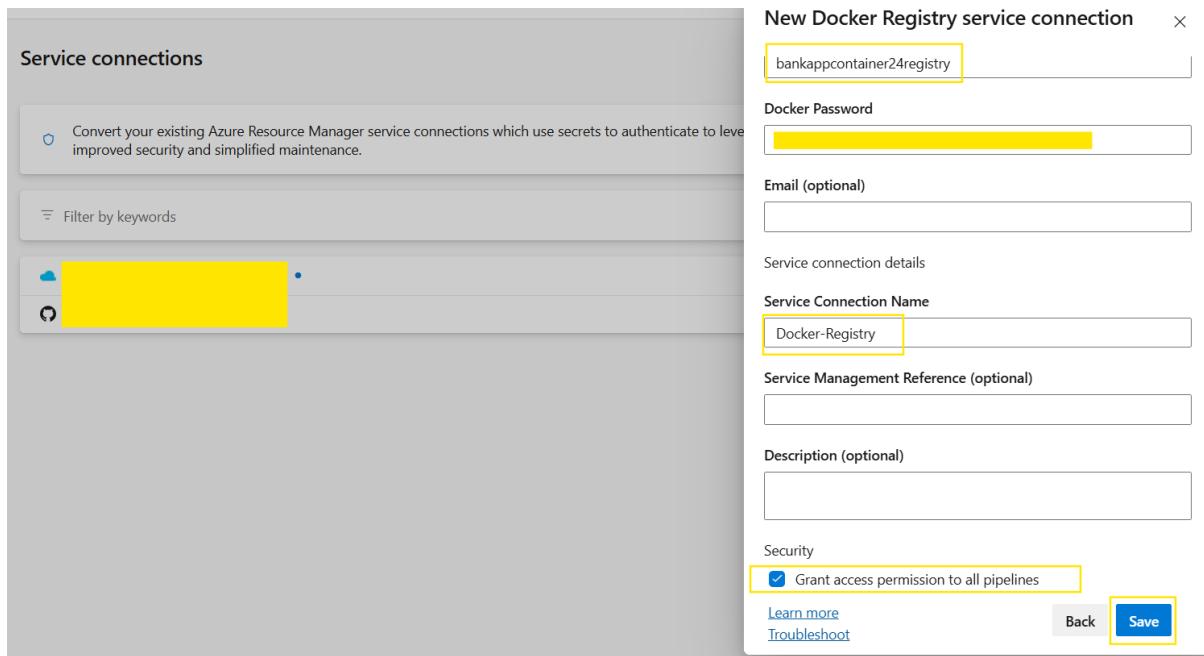
Docker Password: (Redacted)

Email (optional): (Redacted)

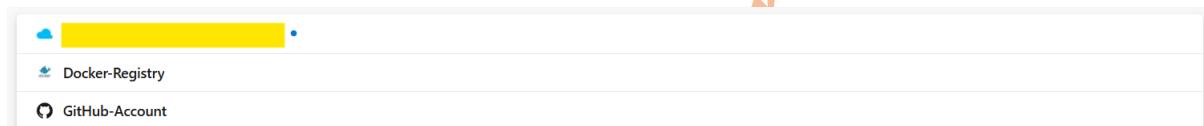
Service connection details:

Service Connection Name: Docker-Registry

Description (optional): (Redacted)

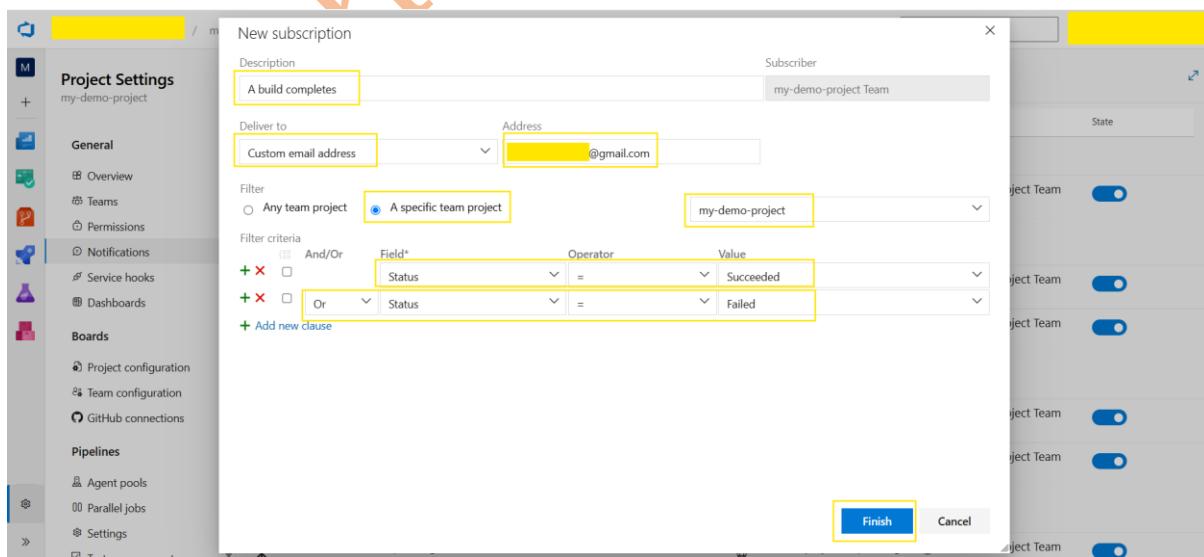


Finally the service connections had been created as shown in the screenshot attached below.

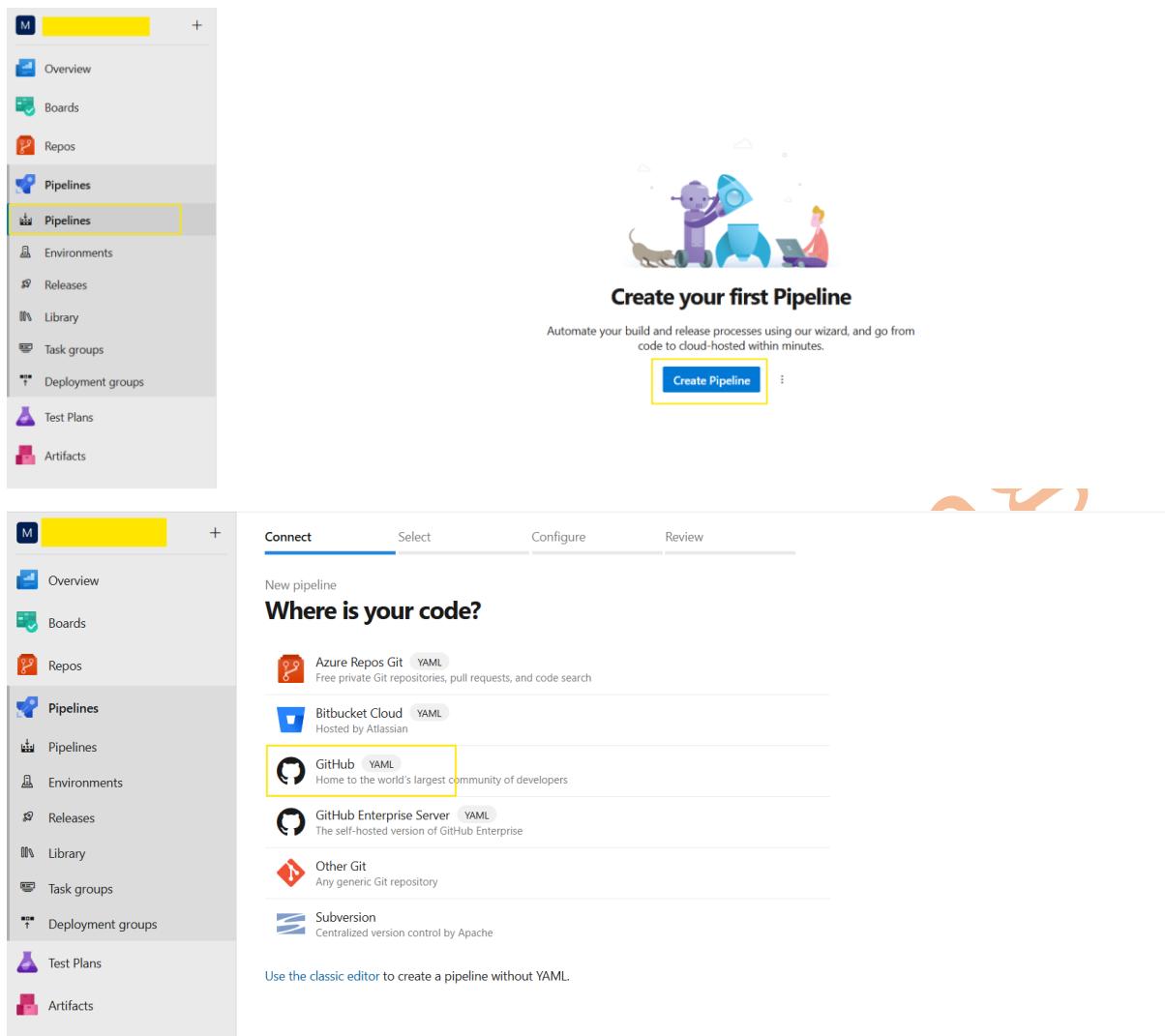


Configuration of Email notification for sending email to group Email Id regarding Success or failure of the Azure DevOps Pipeline

The email notification for sending email for successful or failed pipeline execution to the group email Id had been configured. To configure email notification first go to **Azure DevOps Organisation > Project > Project Settings > Notifications** and do the further configuration as per the screenshot attached below for your reference.



Then I created the Azure DevOps CI/CD pipeline as shown in the screenshot attached below.



Create your first Pipeline

Automate your build and release processes using our wizard, and go from code to cloud-hosted within minutes.

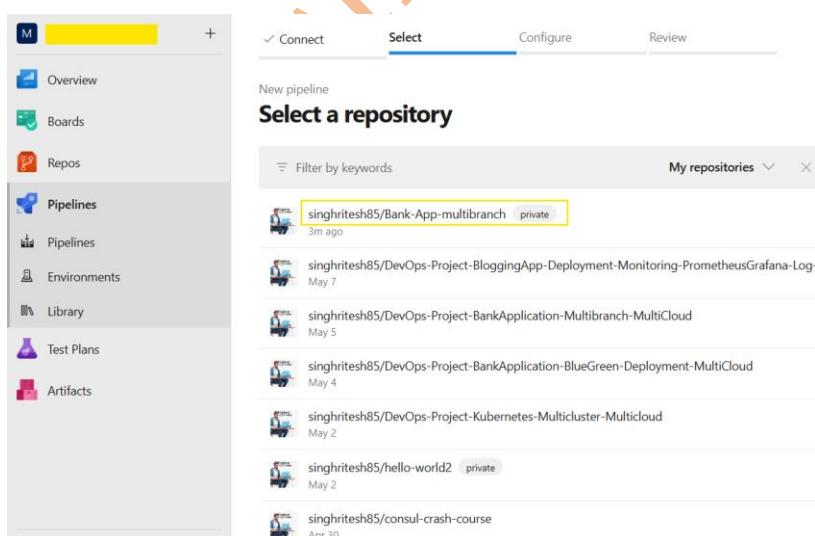
Create Pipeline

Where is your code?

- Azure Repos Git - YAML
- Bitbucket Cloud - YAML
- GitHub - YAML**
- GitHub Enterprise Server - YAML
- Other Git
- Subversion

Use the [classic editor](#) to create a pipeline without YAML.

At this stage it will ask to authorize your GitHub Account and after authorizing your GitHub Account select the desired GitHub Repo as shown in the screenshot attached below.

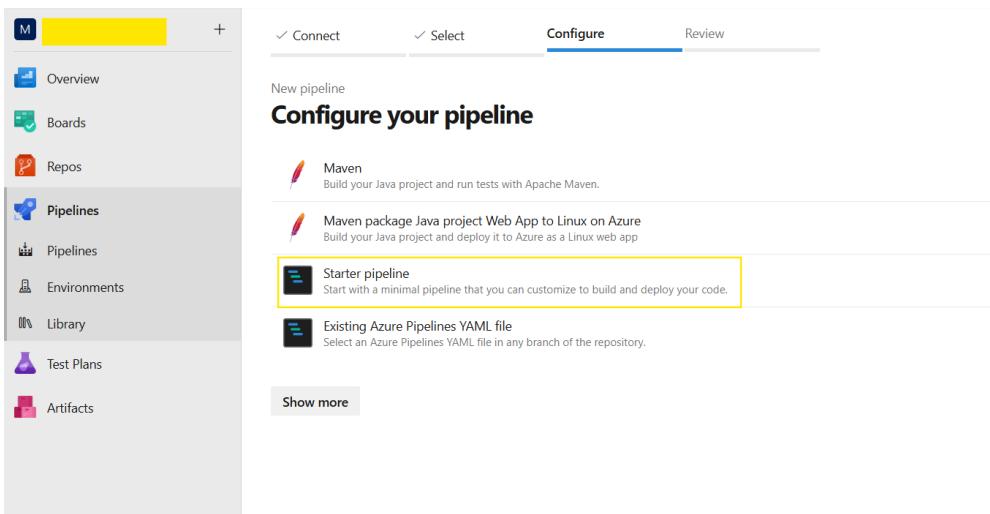


Select a repository

Filter by keywords

My repositories

- singhritesh85/Bank-App-multibranch private 3m ago
- singhritesh85/DevOps-Project-BloggingApp-Deployment-Monitoring-PrometheusGrafana-Log... May 7
- singhritesh85/DevOps-Project-BankApplication-Multibranch-MultiCloud May 5
- singhritesh85/DevOps-Project-BankApplication-BlueGreen-Deployment-MultiCloud May 4
- singhritesh85/DevOps-Project-Kubernetes-MultiCluster-MultiCloud May 2
- singhritesh85/hello-world2 private May 2
- singhritesh85/consul-crash-course Apr 30



If you need to apply webhook in Azure DevOps Pipeline then in `azure-pipelines.yaml` trigger: provide the branch name as shown in the screenshot attached below.

```
trigger:
```

```
- main
```

This pipeline triggers for main branch only if you need to apply for other branches then provide the branch names as shown in the screenshot attached below.

```
trigger:
```

```
- main
- stage
- dev
```

OpenJDK Docker Image is deprecated so in this project eclipse-temurin docker image has been used as a base image in the Dockerfile.

To access Docker Image from the Azure Container Registry during Kubernetes Deployment I created a kubernetes Secrets as shown in the screenshot attached below.

```
kubectl create secret docker-registry bankapp-auth --docker-server=https://bankappcontainer24registry.azurecr.io --docker-username=bankappcontainer24registry --docker-password=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX -n bankapp
```

```
[root@yellow ~]# kubectl create ns bankapp
namespace/bankapp created
[root@yellow ~]# kubectl create secret docker-registry bankapp-auth --docker-server=https://bankappcontainer24registry.azurecr.io --docker-username=bankappcontainer24registry --docker-password=yellow -n bankapp
secret/bankapp-auth created
```

I cloned the GitHub Repo <https://github.com/singhritesh85/helm-repo-for-ArgoCD.git> on Azure DevOps Agent Server as shown in the screenshot attached below.

```
[demo@devopsagent-vm ~]$ git clone https://github.com/singhrithesh85/helm-repo-for-ArgoCD.git
Cloning into 'helm-repo-for-ArgoCD'...
remote: Enumerating objects: 17, done.
remote: Counting objects: 100% (17/17), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 17 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (17/17), 9.65 KiB | 9.65 MiB/s, done.
```

For this project I created two Pipelines in Azure DevOps. It is not a good practice to hardcode the variables in the Azure DevOps Pipeline. To run the Azure DevOps Pipeline for **mysql** I created the variable **MYSQL_ROOT_PASSWORD** and **MYSQL_DATABASE** as shown in the screenshot attached below.

The screenshot shows the Azure DevOps Pipeline configuration interface. On the left, a preview of the pipeline YAML is visible:

```
28 Settings
29
30
31
32
33
34
35
36
37
38
39
40
41 :ence.size=1Gi,global.storageClass=managed-csi,primary.service.type=ClusterIP,auth.rootPass
42
```

A modal window titled "New variable" is open on the right, containing the following fields:

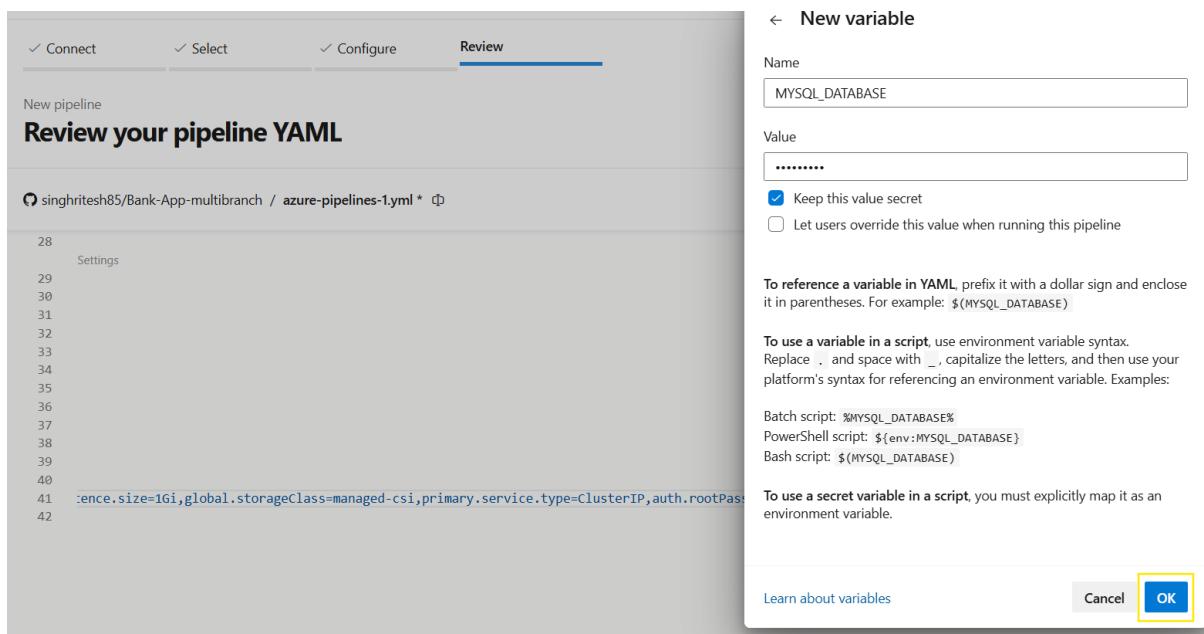
- Name:** MYSQL_ROOT_PASSWORD
- Value:** (redacted)
- Keep this value secret
- Let users override this value when running this pipeline

Below the modal, there is explanatory text about referencing variables in YAML and using them in scripts. At the bottom of the modal are "Learn about variables", "Cancel", and "OK" buttons, with "OK" highlighted.

Below the main pipeline view, another modal window titled "Variables" is open, showing a list of variables:

- Search bar: Search variables
- Variable entry: MYSQL_ROOT_PASSWORD (with a red box around the "+" button)

At the bottom of this modal are "Learn about variables", "Close", and "Save" buttons, with "Close" highlighted.



The two Azure DevOps pipelines named as **mysql** and **bankapp** had been created and ran successfully. First run **mysql** pipeline and then **bankapp** pipeline. As I had not applied webhook in production and so the DevOps Team needs to run it manually. If you want to apply the webhook in production then use the method which I used to apply the webhook in production for Azure DevOps Pipeline (with the help of **trigger:**). Below screenshot shows how to run the Azure DevOps Pipeline manually.

The screenshot shows the Azure DevOps Pipelines page. It displays the 'Recent' tab with two recently run pipelines: 'bankapp' and 'mysql'. Both runs were triggered manually. Below this, the 'mysql' pipeline is selected, showing its details. The 'Run pipeline' button is highlighted with a yellow box.

Pipeline	Last run
bankapp	• Update azure-pipelines-1.yml for Azure Pipelines Manually triggered for main 1m 24s
mysql	• Update azure-pipelines-1.yml for Azure Pipelines Manually triggered for main 1m 46s

mysql

Runs	Branches	Analytics
Description	Stages	
# • Update azure-pipelines-1.yml for Azure Pipelines Manually triggered for main	✓ - ✓	15m ago 1m 46s

Run pipeline
Select parameters below and manually run the pipeline

Branch/tag
 Select a branch from the list or enter the name of a tag as refs/tags/<tagname>

Commit

Advanced options

Variables
This pipeline has no defined variables >

Stages to run
Run as configured >

Resources
Use latest version of all resources >

Enable system diagnostics

Cancel **Run**

← bankapp

Runs Branches Analytics

Description	Stages	Created
# [REDACTED] - Update azure-pipelines-1.yml for Azure Pipelines ↳ Manually triggered for main [REDACTED]	✓-✓-✓	Just now 1m 14s

Run pipeline
Select parameters below and manually run the pipeline

Branch/tag
 Select a branch from the list or enter the name of a tag as refs/tags/<tagname>

Commit

Advanced options

Variables
This pipeline has no defined variables >

Stages to run
Run as configured >

Resources
Use latest version of all resources >

Enable system diagnostics

Cancel **Run**

Finally, the pods had been created as shown in the screenshot attached below.

```
[demo@devopsagent-vm ~]$ kubectl get pods -n bankapp --watch
NAME           READY   STATUS    RESTARTS   AGE
bankapp-folo-  1/1     Running   0          100s
^C[demo@devopsagent-vm ~]$
[ demo@devopsagent-vm ~]$ kubectl get pods -n mysql --watch
NAME           READY   STATUS    RESTARTS   AGE
mysql-primary-0 1/1     Running   0          10m
mysql-secondary-0 1/1     Running   1 (7m47s ago) 10m
^C[ demo@devopsagent-vm ~]$
[ demo@devopsagent-vm ~]$ kubectl get pvc -n mysql --watch
NAME            STATUS  VOLUME                                     CAPACITY  ACCESS MODES  STORAGECLASS  VOLUMEATTRIBUTESCLASS  AGE
data-mysql-primary-0 Bound  pvc-XXXXXXXXXX                               1Gi       RWO          managed-csi  <unset>                10m
data-mysql-secondary-0 Bound  pvc-XXXXXXXXXX                               1Gi       RWO          managed-csi  <unset>                10m
^C[ demo@devopsagent-vm ~]$
[ demo@devopsagent-vm ~]$ kubectl get svc -n bankapp --watch
NAME        TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
bankapp-folo ClusterIP  10.XXXXXX.29   <none>        80/TCP    2m18s
```

Below screenshot shows the ingress rule which I used to access the Bank Application using the URL.

You can use the Ingress rule which I provided in the GitHub Repo

<https://github.com/singhritesh85/DevOps-Project-BankApplication-Multibranch-MultiCloud.git> with the file name **ingress-rule-prod.yaml**. First you need to create the kubernetes secrets as shown in the screenshot attached below.

```
kubectl create secret tls ingress-tls --key mykey.key --cert STAR_singhritesh85_com.crt --namespace bankapp --context=bankapp-cluster
```

```
[root@XXXXXXXXXX ~]# kubectl create secret tls ingress-tls --key mykey.key --cert STAR_singhritesh85_com.crt --namespace bankapp --context=bankapp-cluster
secret/ingress-tls created

[root@XXXXXXXXXX ~]# cat ingress-rule.yaml
# kubectl create secret tls ingress-tls --key mykey.key --cert STAR_singhritesh85_com.crt --namespace bankapp --context=bankapp-cluster
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: bankapp-ingress
  namespace: bankapp
  annotations:
    appgw.ingress.kubernetes.io/ssl-redirect: "true"
spec:
  ingressClassName: azure-application-gateway
  tls:
  - secretName: ingress-tls
  rules:
  - host: bankapp.singhritesh85.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: bankapp-folo
            port:
              number: 80
[root@XXXXXXXXXX ~]# kubectl get ing -A --watch
NAMESPACE   NAME        CLASS      HOSTS          ADDRESS        PORTS   AGE
bankapp     bankapp-ingress  azure-application-gateway  bankapp.singhritesh85.com  51.XXXXXX.201  80, 443  17s
```

```

cat ingress-rule-prod.yaml

# kubectl create secret tls ingress-tls --key mykey.key --cert STAR_singhritesh85_com.crt --namespace
bankapp --context=bankapp-cluster

---

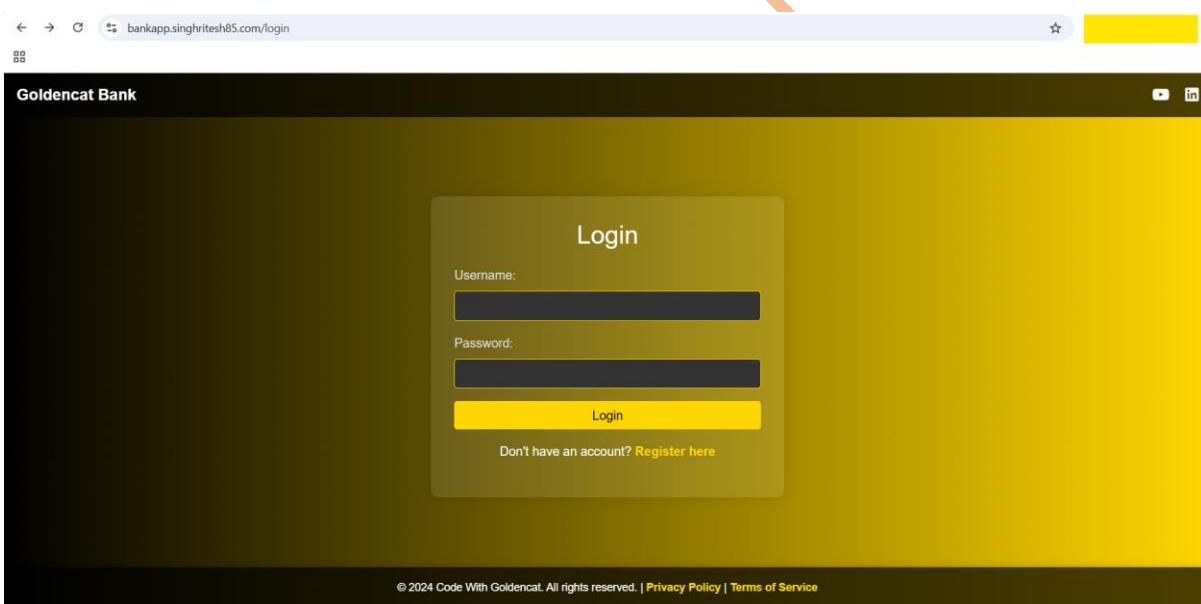
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: bankapp-ingress
  namespace: bankapp
  annotations:
    appgw.ingress.kubernetes.io/ssl-redirect: "true"
spec:
  ingressClassName: azure-application-gateway
  tls:
  - secretName: ingress-tls
  rules:
  - host: bankapp.singhritesh85.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: bankapp-folo
            port:
              number: 80

```

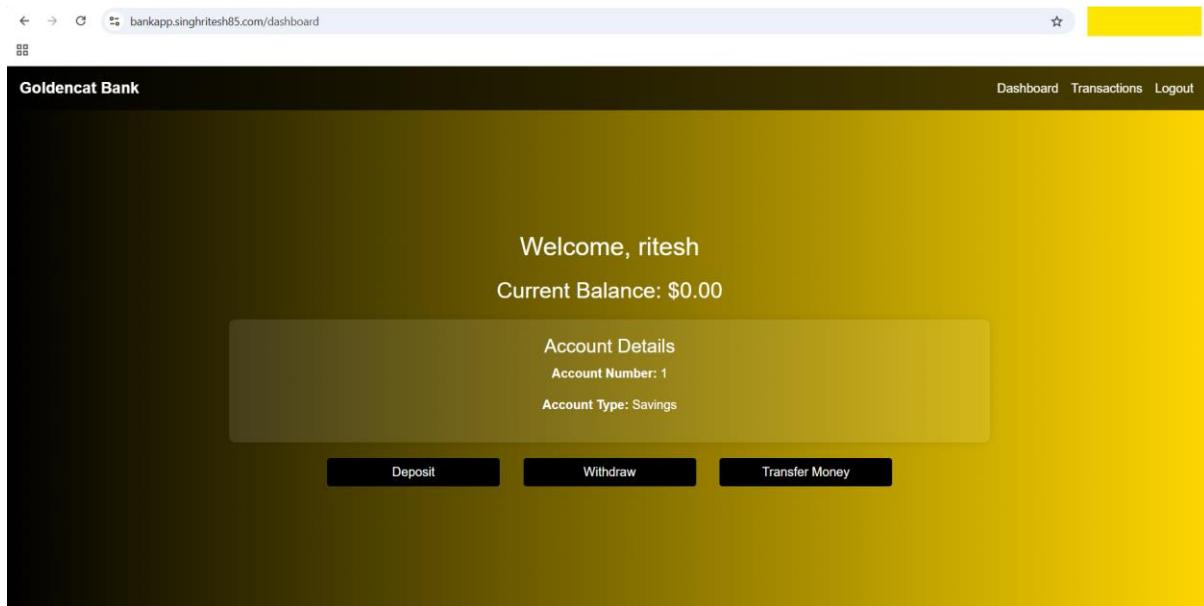
I did the entry for HOST **bankapp.singhritesh85.com** with Public IP Address as shown in the screenshot attached above in the Azure DNS Zone to create the Record Set of A Type.

The screenshot shows the Azure DNS Management portal for the domain singhritesh85.com. On the left, the 'Recordsets' section is selected. A modal window titled 'Add record set' is open, prompting for a name ('bankapp'), type ('A - IPv4 Address records'), and IP address ('51.197.201.128'). The 'Add' button at the bottom left of the modal is highlighted with a yellow box.

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



I registered a user ritesh in the Bank Application as discussed in the non-prod environment and then logged-in with the same user and I had also checked its entry in the MySQL database and found the same as shown in the screenshot attached below.



```
[root@xxxxxxxxx ~]# kubectl exec -it mysql-primary-0 -n mysql /bin/bash
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 565
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;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

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 | [REDACTED] | ritesh |
+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

For the below demonstration I created two Azure DevOps Pipeline named as **bankapp** and **mysql**. I had created Azure DevOps Pipeline in such a way that pipeline **mysql** will run manually and pipeline **bankapp** will run with webhook when Pull Request (PR) comes to approver and approver approves the Pull Request (PR).

To run the Azure DevOps Pipeline manually I already discussed above so here I only discuss about **bankapp** pipeline. You can create a feature branch from dev with name feature-14052025 (name with the date extension) and then merge this branch with the stage branch and then to the main branch.

But for this demonstration I will directly create a release branch from main branch itself and then create a PR to merge it with main. In release branch release-14052025 I changed the line trigger: none to trigger: -main in the file **azure-pipelines.yaml** present in the main branch as shown in the screenshot attached below. It is important to mention here that I kept **pr: none** in the file **azure-pipelines.yaml** and **azure-pipelines-1.yaml** in the **GitHub Repo**

<https://github.com/singhritesh85/Bank-App-multibranch.git> which means whenever a new PR is created in the Repository then Pipeline will not be triggered automatically.

The screenshot shows the GitHub interface for the repository [Bank-App-multibranch](https://github.com/singhritesh85/Bank-App-multibranch).

Code View: The file `azure-pipelines.yaml` is displayed. A yellow box highlights the line `trigger: - none`, with an orange arrow pointing to the word "Older" below it. The code also includes:

```

1   trigger:
2     - none
3
4   pool:
5     name: demo
6     demands:
7       - agent.name -equals demo
8
9   variables:
10    imagePullSecret: 'bankapp-auth'

```

Repository Structure: The repository has three branches: `main`, `3 Branches`, and `0 Tags`. The `main` branch is selected. A modal window titled "Switch branches/tags" is open, showing a search bar with `release-14052025` and a button to "Create branch `release-14052025` from `main`".

Commits: The commit history for the `main` branch is listed, showing the following commits:

- for Azure Pipelines (green checkmark) - 2 hours ago
- Update Jenkinsfile - yesterday
- Add files via upload - last week
- Add files via upload - last week
- Update Jenkinsfile - yesterday
- Update azure-pipelines-1.yaml for Azure Pipelines - 2 hours ago
- Update azure-pipelines.yaml for Azure Pipelines - 2 hours ago
- mvnw - Add files via upload - last week
- mvnw.cmd - Add files via upload - last week
- pom.xml - Add files via upload - last week

The screenshot shows the Azure DevOps pipeline editor for the repository "Bank-App-multibranch". The pipeline file is "azure-pipelines.yml" and it is being edited in the "release-14052025" branch. The "Commit changes" button is highlighted with a yellow box.

Commit message: Update azure-pipelines.yml

Extended description: Add an optional extended description...

Commit options:

- Commit directly to the release-14052025 branch
- Create a new branch for this commit and start a pull request [Learn more about pull requests](#)

Buttons: Cancel, Commit changes

Now to merge the release branch **release-14052025** with the main branch I created a Pull request as shown in the screenshot attached below.

singhritesh85 / Bank-App-multibranch

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Bank-App-multibranch Public

Compare & pull request

About
No description, website, or topics provided.

Activity
0 stars
1 watching
0 forks

Releases
No releases published
[Create a new release](#)

Packages
No packages published
[Publish your first package](#)

Languages

release-14052025 had recent pushes 8 seconds ago

4 Branches **0 Tags** [Go to file](#) [Add file](#) [Code](#)

This branch is **1 commit ahead of main**.

singhritesh85 Update azure-pipelines.yml -1 minute ago 23 Commits

File	Action	Time
mysql	Update Jenkinsfile	yesterday
src	Add files via upload	last week
Dockerfile-Project-1	Add files via upload	last week
Jenkinsfile	Update Jenkinsfile	yesterday
azure-pipelines-1.yml	Update azure-pipelines-1.yml for Azure Pipelines	2 hours ago
azure-pipelines.yml	Update azure-pipelines.yml	1 minute ago

singhritesh85 / Bank-App-multibranch

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Compare & pull request

Filters [ispr isopen](#) [Labels](#) 9 [Milestones](#) 0 [New pull request](#)

Welcome to pull requests!

Pull requests help you collaborate on code with other people. As pull requests are created, they'll appear here in a searchable and filterable list. To get started, you should create a pull request.

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#) or [learn more about diff comparisons](#).

base: main **compare: release-14052025** **Able to merge**. These branches can be automatically merged.

Discuss and review the changes in this comparison with others. [Learn about pull requests](#) [Create pull request](#)

-0 1 commit **1 file changed** **1 contributor**

Commits on May 14, 2025

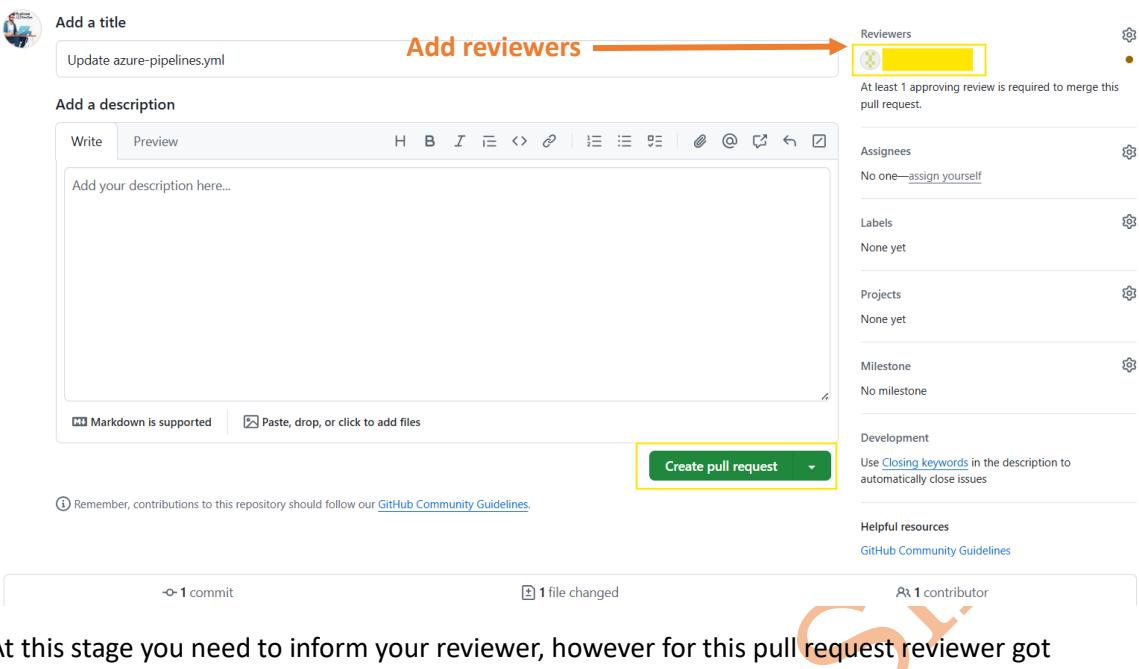
Update azure-pipelines.yml **Verified** **034ec9b** **Split** **Unified**
singhritesh85 authored 3 minutes ago

Showing 1 changed file with 2 additions and 2 deletions.

```

diff --git a/.azure-pipelines.yml b/.azure-pipelines.yml
--- a/.azure-pipelines.yml
+++ b/.azure-pipelines.yml
@@ -1,5 +1,5 @@
 1   trigger:
 2     - none
 3
 4   pool:
 5     name: demo
@@ -68,4 +68,4 @@ stages:
 68   chartType: 'FilePath'

```



At this stage you need to inform your reviewer, however for this pull request reviewer got notification on their email id. The reviewer will review the same PR and after his/her approval you can merge this pull request as shown in the screenshot attached below.

The screenshot shows the GitHub pull request review interface. On the left, there's a diff view of the 'azure-pipelines.yml' file with several changes highlighted. On the right, a modal window titled 'Finish your review' is open, showing a rich text editor for comments and three radio buttons for review actions: 'Comment' (disabled), 'Approve' (selected), and 'Request changes'. The 'Approve' button is highlighted with an orange box.

Update azure-pipelines.yml #2
singhritesh85 wants to merge 1 commit into [main](#) from [release-14052025](#)

Changes approved
1 approving review by reviewers with write access.

All checks have passed
7 successful checks

No conflicts with base branch
Merging can be performed automatically.

Merge pull request You can also merge this with the command line. [View command line instructions](#).

Add a comment

Write Preview Add your comment here... Markdown is supported Paste, drop, or click to add files

None yet

Projects None yet

Milestone No milestone

Development Successfully merging this pull request may close these issues.
None yet

Notifications Customize [Unsubscribe](#)
You're receiving notifications because you're watching this repository.

1 participant

[Lock conversation](#)

In the **azure-pipelines.yaml** file I changed the trigger: - none to trigger: -main and hence bankapp pipeline had been triggered and application pods had been created as shown in the screenshot attached below.

```
[root@[REDACTED] ~]# kubectl get pods -n bankapp
NAME          READY   STATUS    RESTARTS   AGE
bankapp-folo-[REDACTED]   1/1     Running   0          51s
```

After success execution of the Azure DevOps pipelines Email had been triggered to the group Email Id as shown in the screenshot attached below.

[Build succeeded] bankapp - singhritesh85/Bank-App-multibranch:main - [REDACTED] - [REDACTED]

Inbox

Azure DevOps [REDACTED] (9 minutes ago) ⚡ 😊

Microsoft Azure DevOps

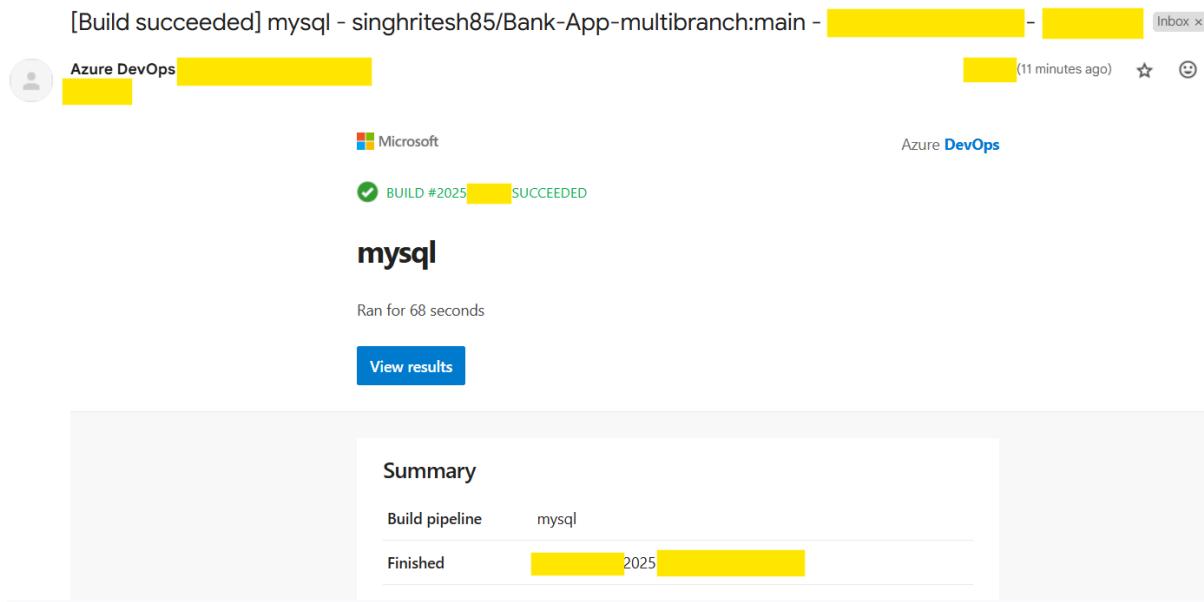
BUILD #20250515.5 SUCCEEDED

bankapp

Ran for 113 seconds

[View results](#)

Summary	
Build pipeline	bankapp
Finished	[REDACTED] 2025 [REDACTED]

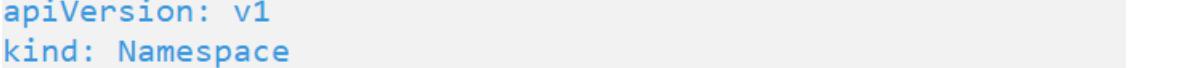
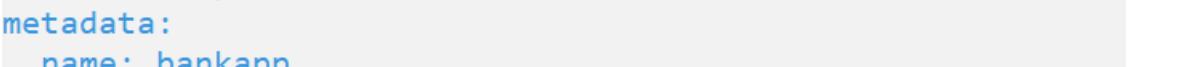
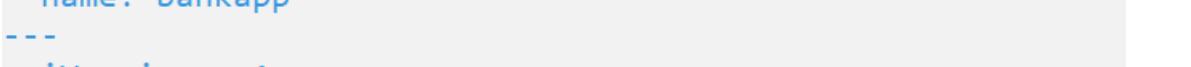
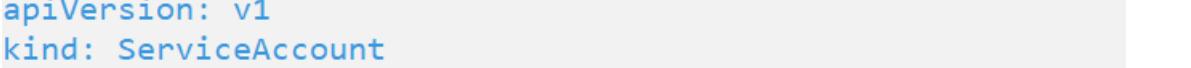


I had provided restricted access to the deployment user **demo** using Service Account, Role and Role Binding as shown in the screenshot attached below. The deployment user had all the accesses in the namespaces **bankapp** and **mysql** but does not have access for the entire AKS cluster. That means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

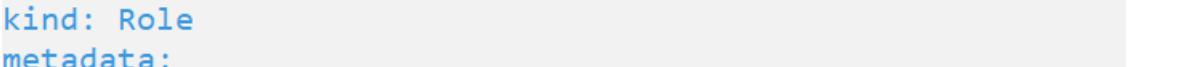
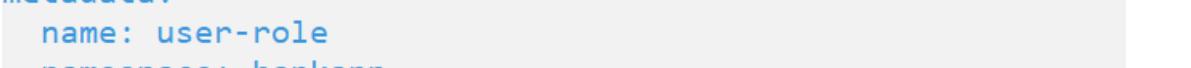
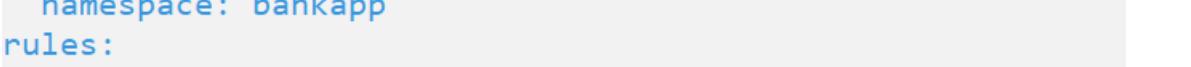
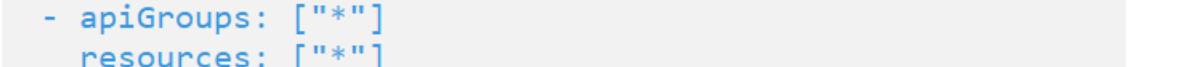
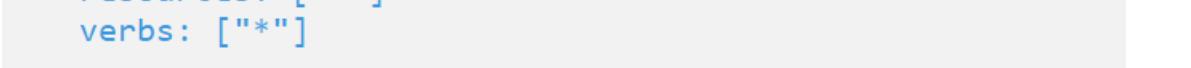
```
[root@REDACTED ~]# kubectl describe secret mysecretname-mysql-aks -n mysql
Name:         mysecretname-mysql-aks
Namespace:    mysql
Labels:       <none>
Annotations: kubernetes.io/service-account.name: devops-agent-mysql-aks
              kubernetes.io/service-account.uid: REDACTED
Type:        kubernetes.io/service-account-token

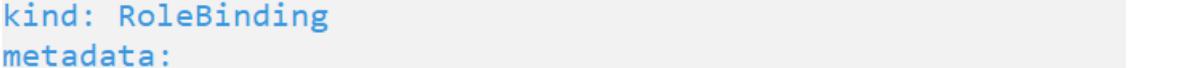
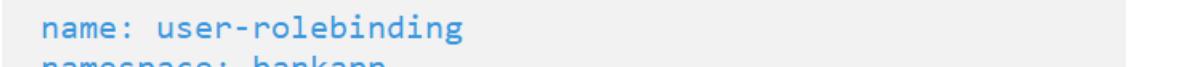
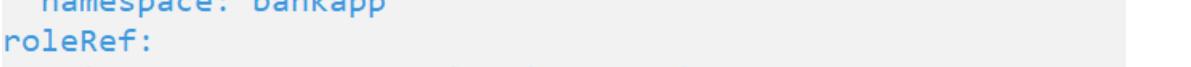
Data
=====
ca.crt:     1761 bytes
namespace:   5 bytes
token:      REDACTED
```

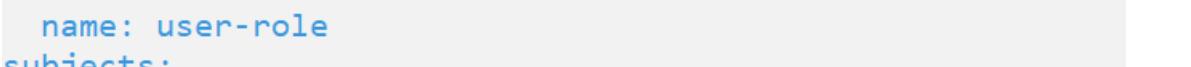
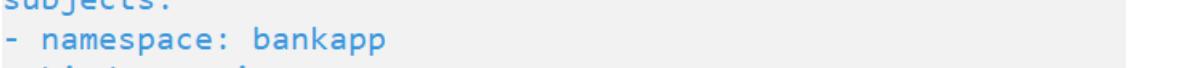





```
[root@REDACTED ~]# cat sa-role-rolebinding-prod.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: devops-agent-bankapp-aks
  namespace: bankapp
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  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: devops-agent-bankapp-aks
```

```
[root@] ~]# cat secrets-prod.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-aks
  namespace: bankapp
  annotations:
    kubernetes.io/service-account.name: devops-agent-bankapp-aks

[root@] ~]# cat sa-role-rolebinding-mysql-prod.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: devops-agent-mysql-aks
  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: devops-agent-mysql-aks
```

```
[root@XXXXXXXXXX ~]# cat secrets-mysql-prod.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-mysql-aks
  namespace: mysql
  annotations:
    kubernetes.io/service-account.name: devops-agent-mysql-aks
```

Ritesh Kumar Singh

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

apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: devops-agent-bankapp-aks
  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: devops-agent-bankapp-aks
```

```
cat secrets-prod.yaml

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

```
cat secrets-mysql-prod.yaml

apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-mysql-aks
  namespace: mysql
  annotations:
    kubernetes.io/service-account.name: devops-agent-mysql-aks
```

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

apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: devops-agent-mysql-aks
  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: devops-agent-mysql-aks
```

The kubeconfig file which I had shared with the deployment user **demo** is as shown in the screenshot attached below.

```
cat ~/.kube/config
apiVersion: v1
clusters:
- cluster:
  certificate-authority-data:
  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
  server: https://bankapp-cluster-dns-
y1drntz2.96d5ebcd80912eb0e8b15cf52a52918.privatelink.eastus.azurek8s.io:443
  name: bankapp-cluster
contexts:
- context:
  cluster: bankapp-cluster
  user: devops-agent-bankapp-aks
  name: devops-agent-bankapp-aks
- context:
  cluster: bankapp-cluster
  user: devops-agent-mysql-aks
  name: devops-agent-mysql-aks
current-context: devops-agent-mysql-aks
kind: Config
preferences: {}
users:
- name: devops-agent-bankapp-aks
  user:
    token:
    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
- name: devops-agent-mysql-aks
  user:
    token:
    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
[demo@devopsagent-vm ~]$ kubectl config get-contexts
CURRENT NAME CLUSTER AUTHINFO NAMESPACE
* devops-agent-bankapp-aks bankapp-cluster devops-agent-bankapp-aks
  devops-agent-mysql-aks bankapp-cluster devops-agent-mysql-aks
[demo@devopsagent-vm ~]$ kubectl get nodes
Error from server (Forbidden): nodes is forbidden: User "system:serviceaccount:mysql:devops-agent-mysql-aks" cannot list resource "nodes" in API group "" at t
he cluster scope
[demo@devopsagent-vm ~]$ kubectl get pods -n mysql
NAME READY STATUS RESTARTS AGE
mysql-primary-0 1/1 Running 0 5h13m
mysql-secondary-0 1/1 Running 1 (5h11m ago) 5h13m
[demo@devopsagent-vm ~]$ kubectl get pvc -n mysql
NAME STATUS VOLUME CAPACITY ACCESS MODES STORAGECLASS VOLUMEATTRIBUTESCLASS AGE
data-mysql-primary-0 Bound pvc-[REDACTED] 1Gi RWO managed-csi <unset> 5h14m
data-mysql-secondary-0 Bound pvc-[REDACTED] 1Gi RWO managed-csi <unset> 5h14m
[demo@devopsagent-vm ~]$ kubectl get pvc -n bankapp
Error from server (Forbidden): persistentvolumeclaims is forbidden: User "system:serviceaccount:mysql:devops-agent-mysql-aks" cannot list resource "persistent
volumeclaims" in API group "" in the namespace "bankapp"
[demo@devopsagent-vm ~]$ kubectl config use-context devops-agent-bankapp-aks
Switched to context "devops-agent-bankapp-aks".
[demo@devopsagent-vm ~]$ kubectl get pods -n bankapp
NAME READY STATUS RESTARTS AGE
bankapp-folio-[REDACTED] 1/1 Running 0 44m
[demo@devopsagent-vm ~]$ kubectl get pods -n mysql
Error from server (Forbidden): pods is forbidden: User "system:serviceaccount:bankapp:devops-agent-bankapp-aks" cannot list resource "pods" in API group "" in
the namespace "mysql"
```

Monitoring using Prometheus and Grafana and Log Aggregation using Loki in Production

To access Grafana using URL I had created record set of A Type in Azure DNS Zone. I had also created a URL for Loki after creating a Record Set of A Type in Azure DNS Zone as shown in the screenshots attached below. For entry in the Azure DNS Zone to create the Record Set of A Type I obtained the Public IPV4 address from the Azure Application Gateway for Grafana and Loki.

The screenshot shows the Azure portal's DNS management interface. On the left, there's a sidebar with options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Settings, DNS Management, and Records. Under DNS Management, the 'Records' section is selected. On the right, there's a modal window titled 'Add record set' for the domain 'singhritesh85.com'. The 'Name' field is set to 'grafana' and the 'Type' field is set to 'A - IPv4 Address records'. Below these, there are fields for 'Alias record set' (set to 'No'), 'TTL' (set to 1), 'TTL unit' (set to 'Hours'), and 'IP address' (containing '172.16.6'). At the bottom of the modal are 'Add', 'Cancel', and 'Give feedback' buttons. In the background, a list of existing record sets for the 'bankapp' zone is visible, including entries for 'app-gtw-ingress-controller', 'bankapp-application-gateway-grafana', and 'bankapp-application-gateway-loki'.

The screenshot shows the Azure DNS Management portal for the domain `singhritesh85.com`. On the left, there's a sidebar with various management options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Settings, DNS Management, and Recordsets. The Recordsets section is currently selected. On the right, a modal window titled "Add record set" is open, prompting for a name ("loki"), type ("A - IPv4 Address records"), and IP address ("172.172.148.239"). The TTL is set to 1 hour.

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 screenshot shows the Grafana configuration interface for creating a new data source. It's a dark-themed page with a navigation bar at the top. The main area is titled "Connections > Data sources > prometheus". It shows a "Settings" tab is active. The "Name" field contains "prometheus". Below it, under "Connection", the "Prometheus server URL" field is filled with "http://10.0.5.90:9090". There's also an "Authentication" section.

The image contains two screenshots of the Grafana interface, both showing the configuration of data sources.

Screenshot 1: Prometheus Data Source Configuration

- The URL is `grafana.singhritesh85.com/connections/datasources/edit/aequqlud83vuob`.
- The "Type" is set to "Prometheus".
- "Incremental querying (beta)" is turned off.
- "Disable recording rules (beta)" is turned off.
- "Other" settings include "Custom query parameters" (Example: `max_source_resolution=5m&timeout`), "HTTP method" (set to "POST"), and "Use series endpoint" (turned off).
- "Exemplars" section has a "+ Add" button.
- A green success message box says: "✓ Successfully queried the Prometheus API." It also suggests building a dashboard or querying data in the Explore view.
- Buttons at the bottom are "Delete" and "Save & test" (highlighted with a yellow box).

Screenshot 2: Loki Data Source Configuration

- The URL is `grafana.singhritesh85.com/connections/datasources/edit/dely4lncz94kc`.
- The "Type" is set to "Loki".
- "Alerting" and "Build a dashboard" buttons are visible in the top right.
- "Settings" tab is selected.
- The "Name" field is set to "loki" (highlighted with a yellow box).
- A note below says: "Before you can use the Loki data source, you must configure it below or in the config file. For detailed instructions, [view the documentation](#)".
- Connection** section shows "URL" set to "`http://loki.singhritesh85.com`".
- Authentication** section shows "Authentication methods" as "No Authentication".

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.

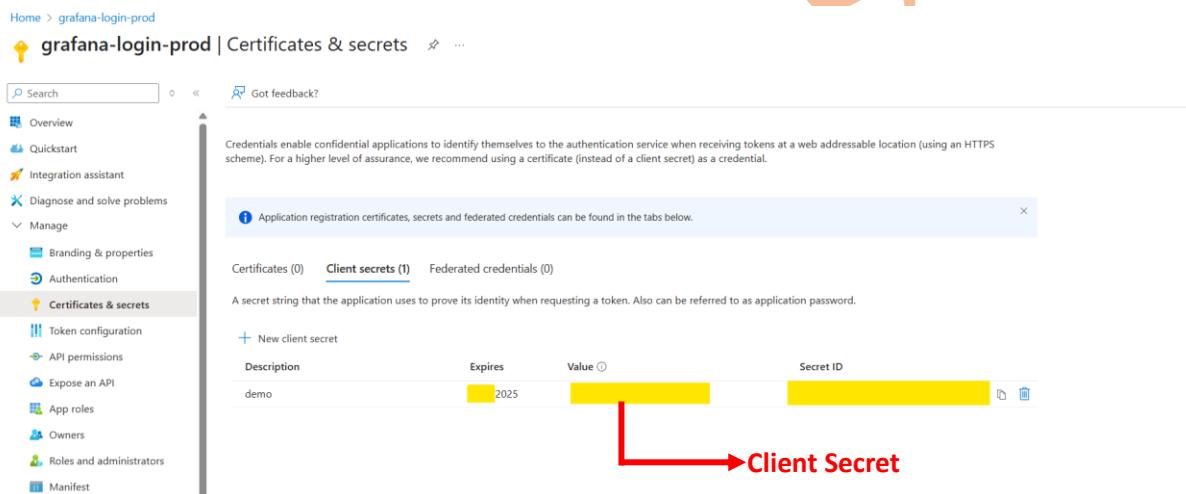
Home > Default Directory | App registrations >
Register an application ...

Supported account types
 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](#)

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



Home > grafana-login-prod

grafana-login-prod | Certificates & secrets

Search

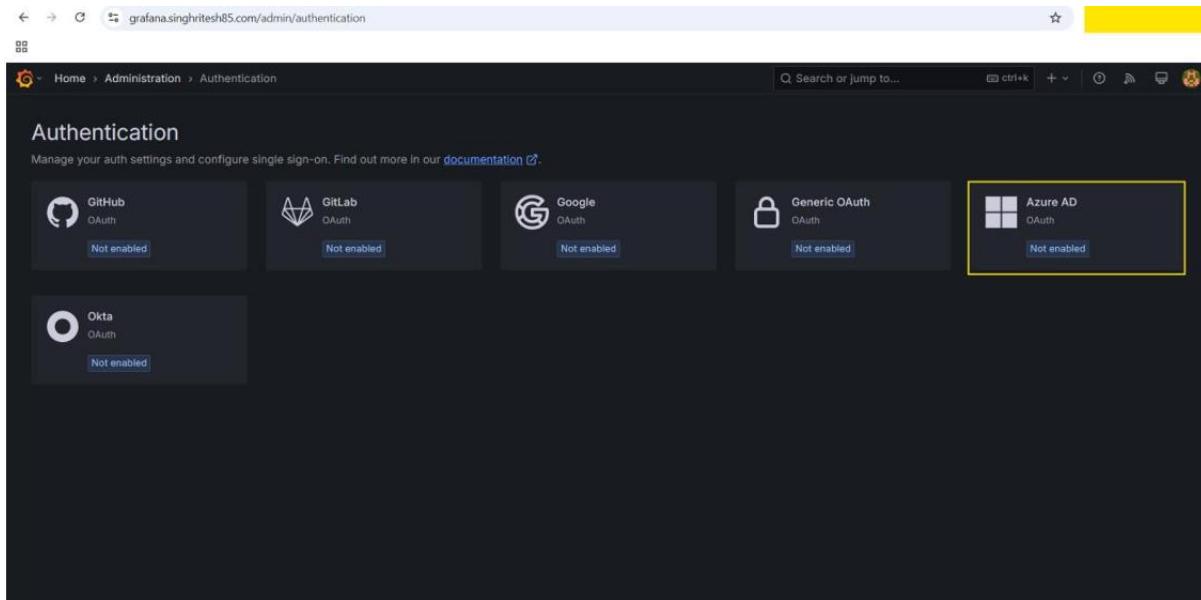
- Overview
- Quickstart
- Integration assistant
- Diagnose and solve problems
- Manage
 - Branding & properties
 - Authentication
 - Certificates & secrets**
 - Token configuration
 - API permissions
 - Expose an API
 - App roles
 - Owners
 - Roles and administrators
 - Manifest

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.

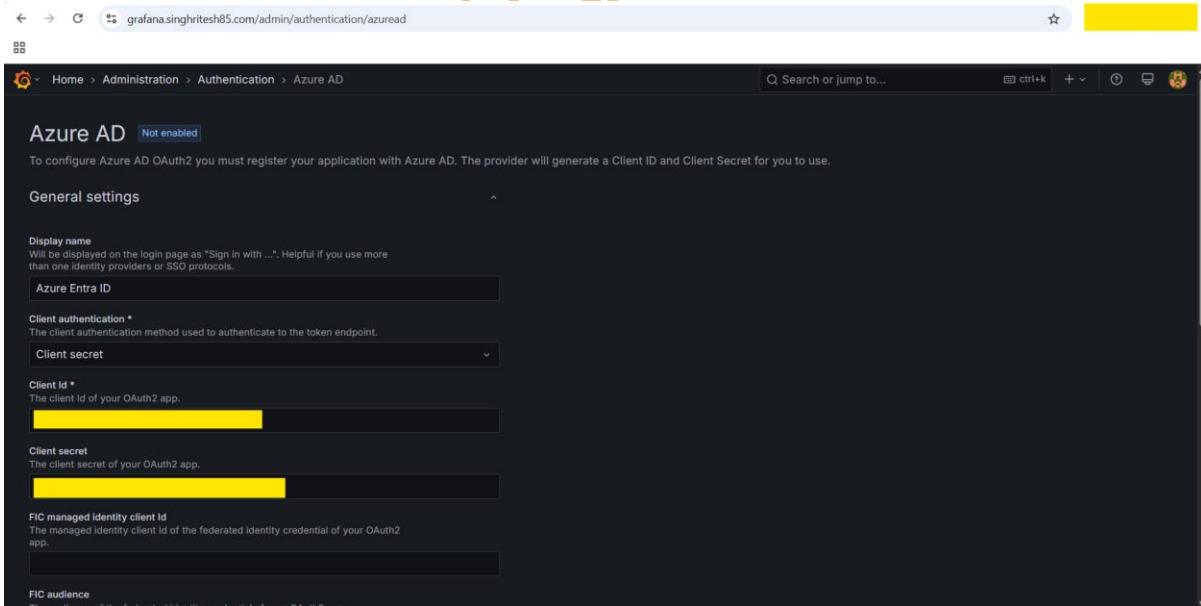
Certificates (0)	Client secrets (1)	Federated credentials (0)
A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.		
+ New client secret Description: demo <input type="button" value="Expires"/> 2025 <input type="button" value="Value"/> <input type="button" value="Secret ID"/>		

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**).



Auth URL *
The authorization endpoint of your OAuth2 provider.
`https://login.microsoftonline.com/[REDACTED]/oauth2/v2.0`

Token URL *
The token endpoint of your OAuth2 provider.
`https://login.microsoftonline.com/[REDACTED]/oauth2/v2.0`

Allow sign up
If not enabled, only existing Grafana users can log in using OAuth.

Auto login
Log in automatically, skipping the login screen.

Sign out redirect URL
The URL to redirect the user to after signing out from Grafana.

User mapping

Role attribute strict mode
If enabled, denies user login if the Grafana role cannot be extracted using Role attribute path.

Organization mapping
List of <GroupId><OrgIdOrName><Role>* mappings.
Enter mappings (my-team:1:Viewer...) and press Enter to add

Allow assign Grafana admin
If enabled, it will automatically sync the Grafana server administrator role.

Skip organization role sync
Prevent synchronizing users' organization roles from your IdP.

Extra security measures

Save and enable **Save** **Discard**

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

Manage your auth settings and configure single sign-on. Find out more in our [documentation](#).

GitHub OAuth Not enabled	GitLab OAuth Not enabled	Google OAuth Not enabled	Generic OAuth Not enabled
Okta OAuth Not enabled	Azure AD OAuth Enabled		

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@grafana-vm ~]# 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

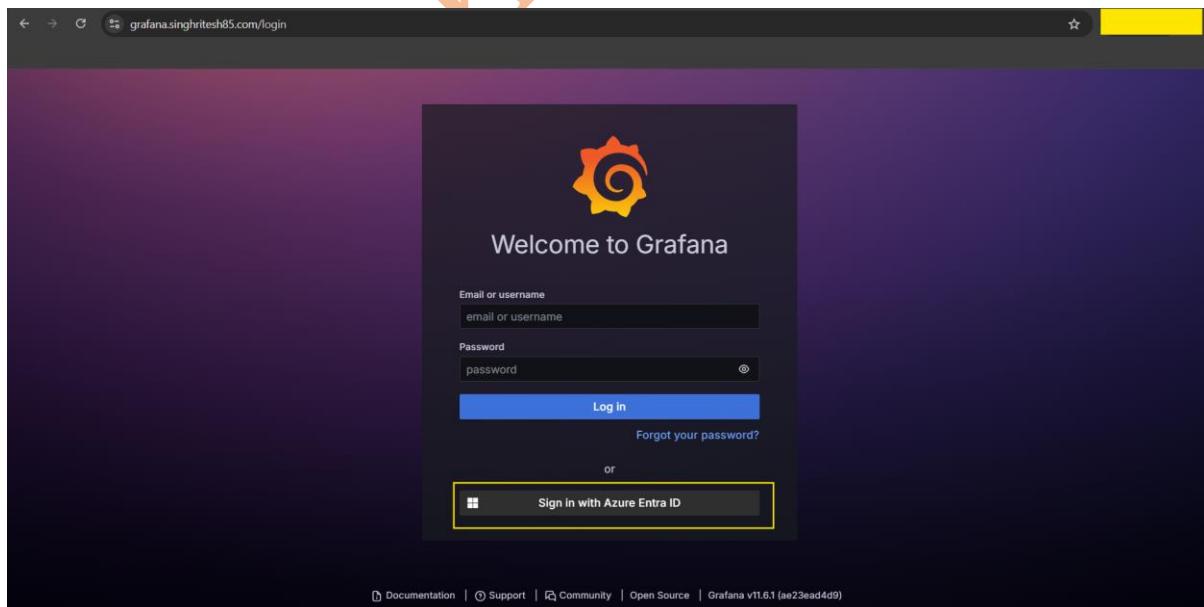
# Redirect to correct domain if host header does not match domain
# Prevents DNS rebinding attacks
;enforce_domain = false

# 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_from_sub_path = false

# Log web requests
;router_logging = false

[root@grafana-vm ~]# systemctl restart grafana-server
[root@grafana-vm ~]# systemctl status grafana-server
● 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
```

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.

The screenshots illustrate the Grafana Admin UI for managing users. The top screenshot shows the 'Users' page with three entries:

Login	Email	Name	Last active	Origin	Provisioned
admin	admin@localhost		6 minutes		
Ritesh		Ritesh	< 1 minute	AzureAD	
user1@singhritesh85.com	user1@singhritesh85.com	User1	2 minutes	AzureAD	

The bottom screenshot shows the detailed settings for the user 'user1@singhritesh85.com'. The 'User information' section shows the following details:

Numerical identifier	3	Synced via AzureAD
Name	user1	Synced via AzureAD
Email	user1@singhritesh85.com	Synced via AzureAD
Username	user1@singhritesh85.com	Synced via AzureAD
Password	*****	Synced via AzureAD

Under 'Permissions', 'Grafana Admin' is set to 'Yes'. Under 'Organizations', 'Main Org.' has 'Admin' selected. There are 'Change role' and 'Remove from organization' buttons.

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

Installation of node-exporter and promtail had been done using the helm chart in the AKS 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: Thu May 15 09:41:15 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
```

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

```
[root@yellow ~]# kubectl get svc -n node-exporter
NAME                  TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
my-prometheus-node-exporter  LoadBalancer  10.yellow.20  128.yellow.144  9100:30603/TCP  3m32s
```

Then I had updated the prometheus configuration file `/etc/prometheus/prometheus.yml` as shown in the screenshot attached below.

```
- job_name: "AKS-Cluster"
  static_configs:
    - targets: ["128.yellow.144:9100"]

[root@prometheus-vm ~]# systemctl restart prometheus.service
[root@prometheus-vm ~]# systemctl status prometheus.service
● prometheus.service - Prometheus
  Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor preset: disabled)
  Active: active (running) since Thu 2025 yellow UTC; 6s ago
```

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
```

```

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.██.10:3100/loki/api/v1/push
  - url: http://10.██.9:3100/loki/api/v1/push
  - url: http://10.██.8:3100/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

```

```

[root@██████ ~]# git clone https://github.com/singhrithesh85/helm-chart-promtail.git
Cloning into 'helm-chart-promtail'...
remote: Enumerating objects: 35, done.
remote: Counting objects: 100% (35/35), done.
remote: Compressing objects: 100% (34/34), done.
remote: Total 35 (delta 7), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (35/35), 24.57 KiB | 8.19 MiB/s, done.
Resolving deltas: 100% (7/7), done.
[root@██████ ~]# vim helm-chart-promtail/
.git/          Chart.yaml      README.md      README.md.gotmpl  ci/           templates/      values.yaml
[root@██████ ~]# vim helm-chart-promtail/values.yaml
[root@██████ ~]# kubectl create ns promtail && helm upgrade --install promtail ./helm-chart-promtail -f ./helm-chart-promtail/values.yaml -n promtail
namespace/promtail created
Release "promtail" does not exist. Installing it now.
NAME: promtail
LAST DEPLOYED: 2025-01-10T10:25:25Z
NAMESPACE: promtail
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
*****
Welcome to Grafana Promtail
Chart version: 6.16.6
Promtail version: 3.0.0
*****
Verify the application is working by running these commands:
* kubectl --namespace promtail port-forward daemonset/promtail 3101
* curl http://127.0.1:3101/metrics

```

```

[root@██████ ~]# kubectl get pods -n promtail --watch
NAME        READY   STATUS    RESTARTS   AGE
promtail-██  1/1     Running   0          2m23s
promtail-██  1/1     Running   0          2m23s
promtail-██  1/1     Running   0          2m23s

```

For Monitoring Tool I had used Prometheus and Grafana. 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.

Prometheus Alerts Graph Status Help

Targets

All scrape pools ▾ All Unhealthy Collapse All Filter by endpoint or labels Unknown Unhealthy Healthy

AKS-Cluster (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://128.1.1.144:9100/metrics	UP	instance="128.1.1.144:9100" job="AKS-Cluster"	12.821s ago	34.853ms	

BlackboxExporter-Server (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.1.1.4:9100/metrics	UP	instance="10.1.1.4:9100" job="BlackboxExporter-Server"	16.1s ago	23.884ms	

DevOpsAgent-Server (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.1.1.6:9100/metrics	UP	instance="10.1.1.6:9100" job="DevOpsAgent-Server"	16.868s ago	21.900ms	

Grafana-Server (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.1.1.7:9100/metrics	UP	instance="10.1.1.7:9100" job="Grafana-Server"	9.49s ago	17.810ms	

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

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.1.1.10:9100/metrics	UP	instance="10.1.1.10:9100" job="Loki-Server-1"	11.196s ago	21.888ms	

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

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.1.1.9:9100/metrics	UP	instance="10.1.1.9:9100" job="Loki-Server-2"	14.776s ago	20.013ms	

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

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.1.1.8:9100/metrics	UP	instance="10.1.1.8:9100" job="Loki-Server-3"	12.316s ago	17.760ms	

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"	15.724s ago	24.659ms	

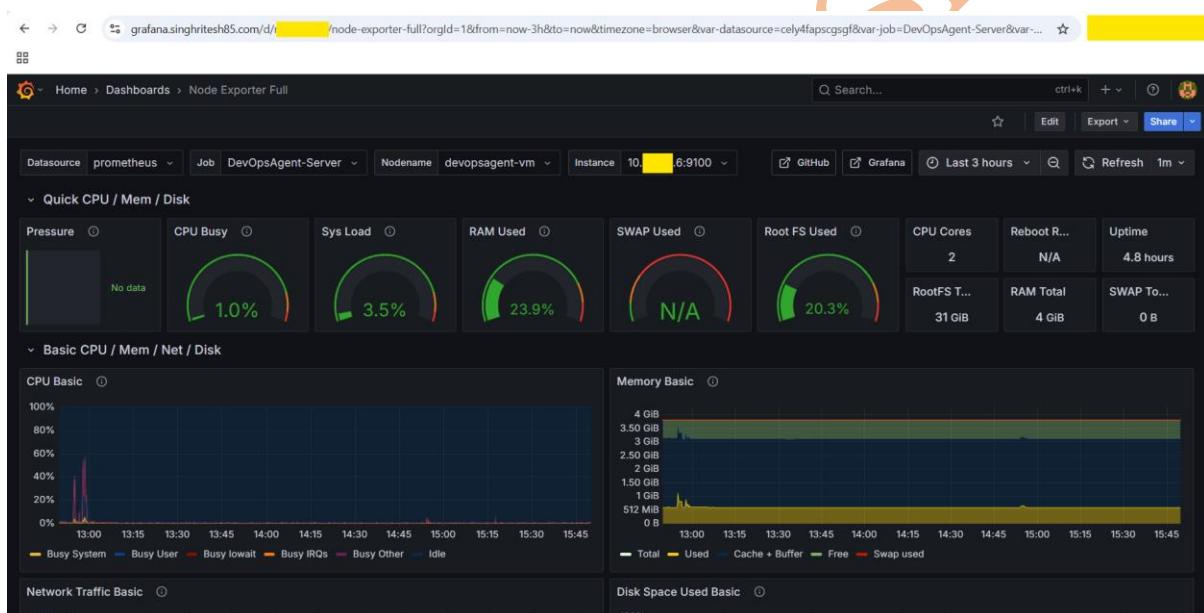
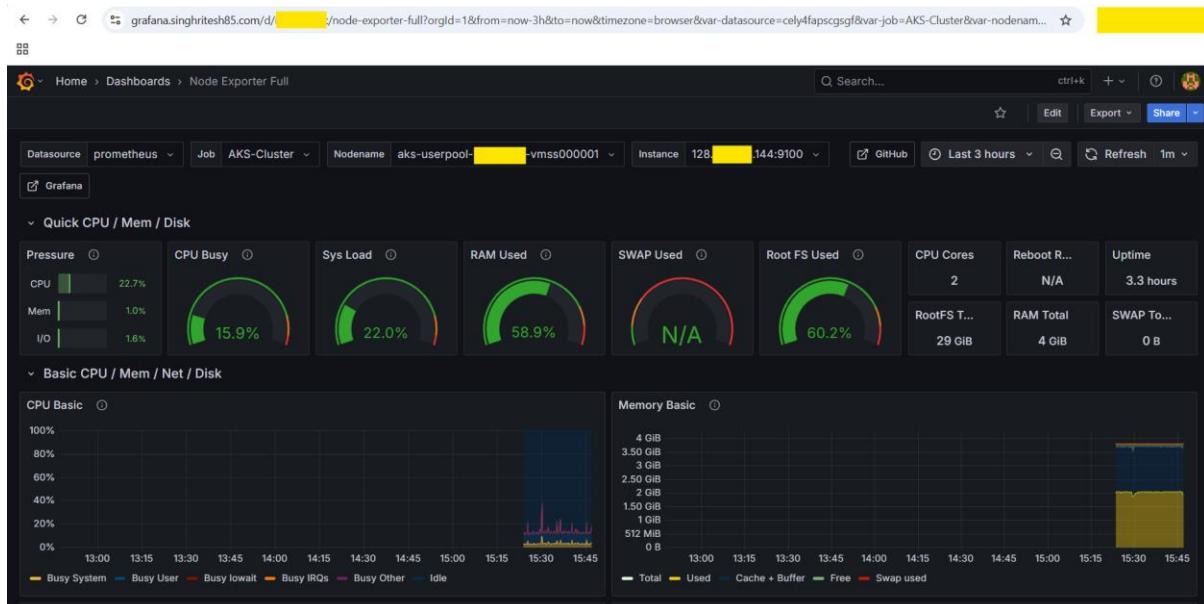
blackbox (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.1.1.9115/probe module="http_2xx_example" target="https://bankapp.singhritesh85.com"	UP	instance="https://bankapp.singhritesh85.com" job="blackbox"	11.906s ago	95.110ms	

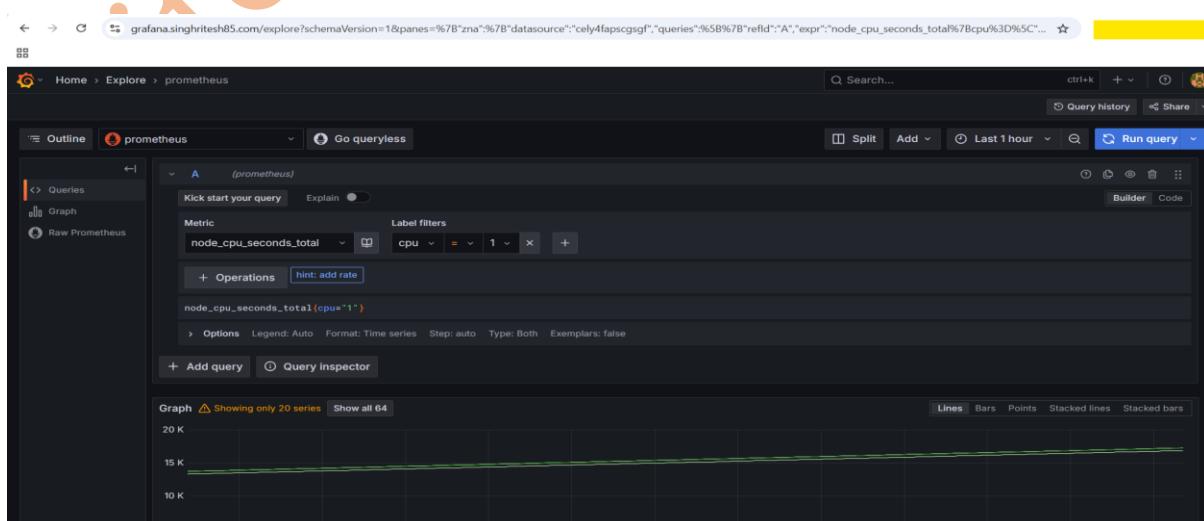
prometheus (1/1 up) show less

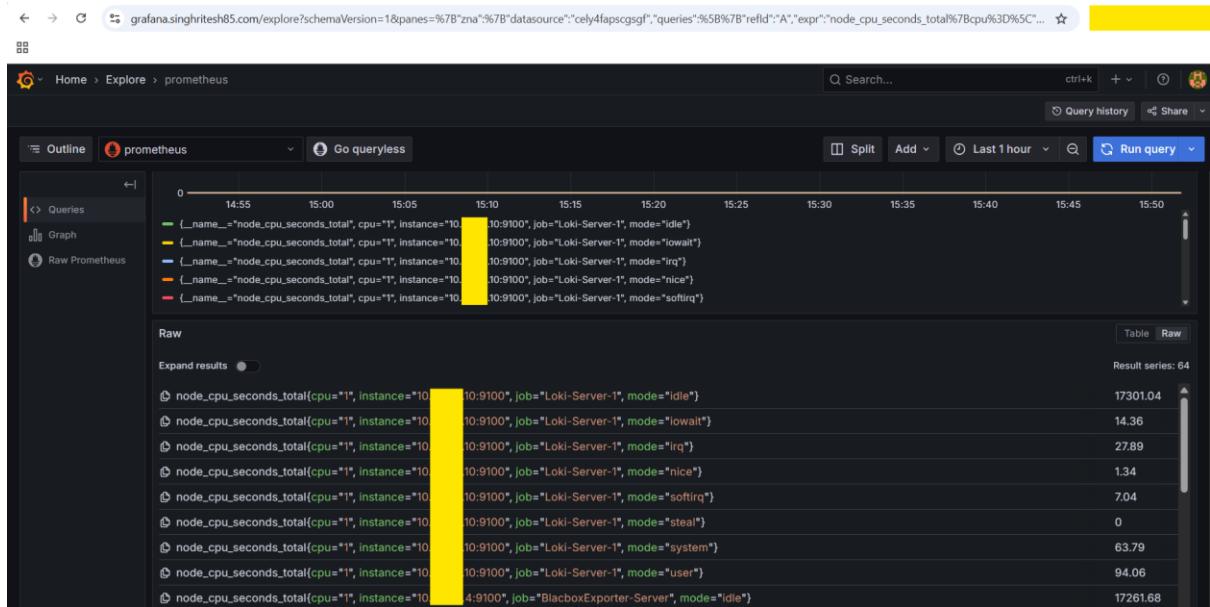
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	14.820s ago	4.842ms	

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

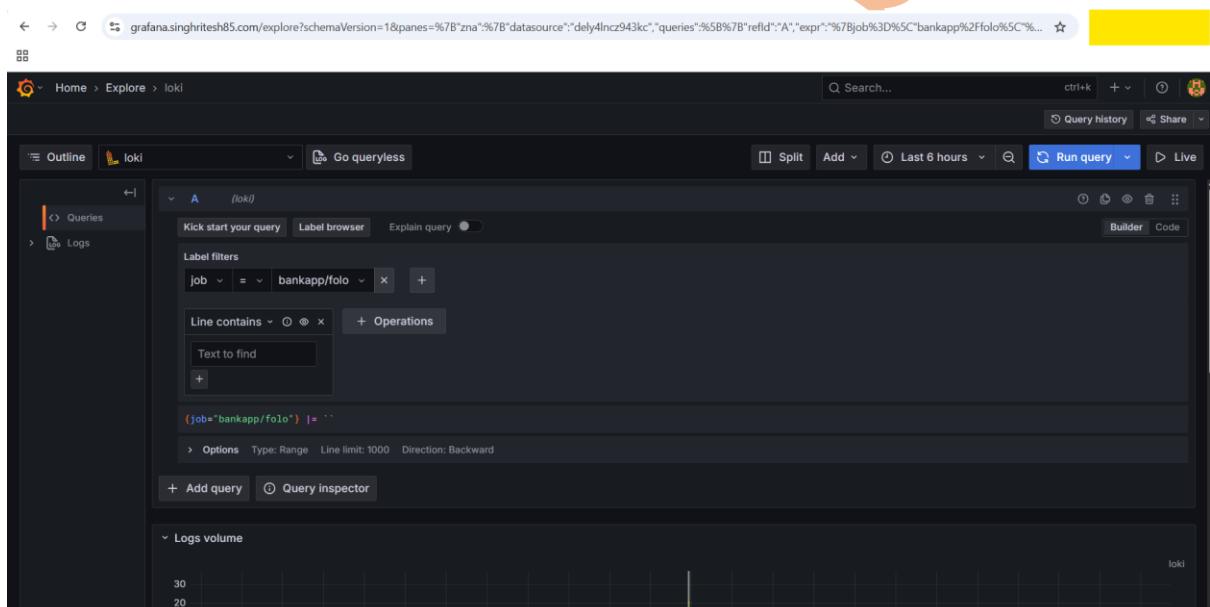


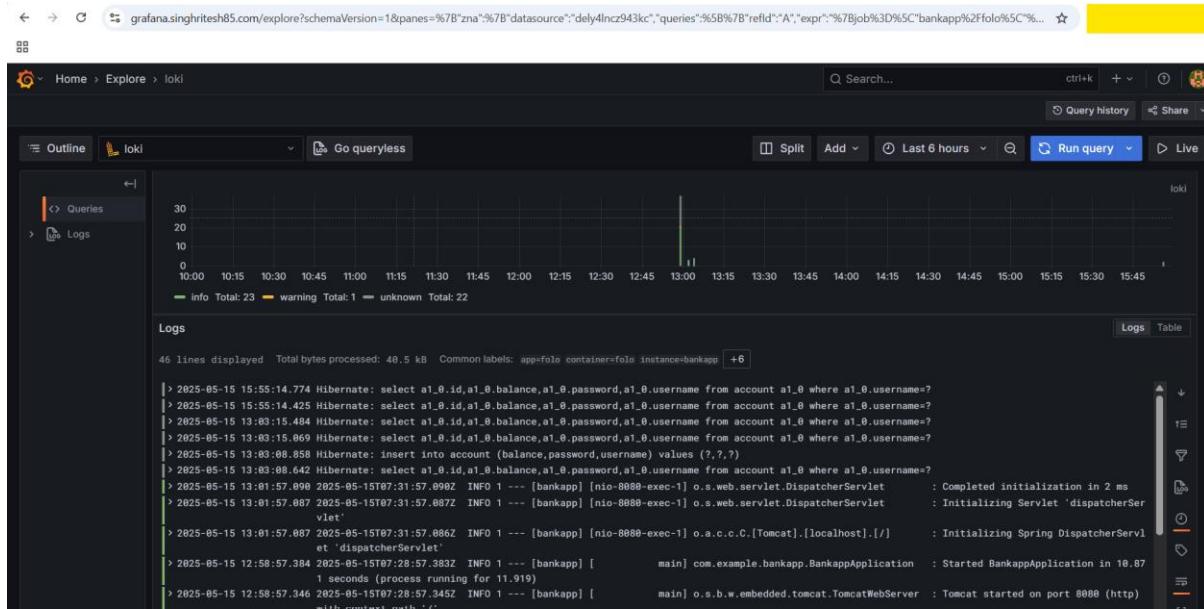
Grafana Metrics for SonarQube I started exploring as shown in the screenshot attached below.





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

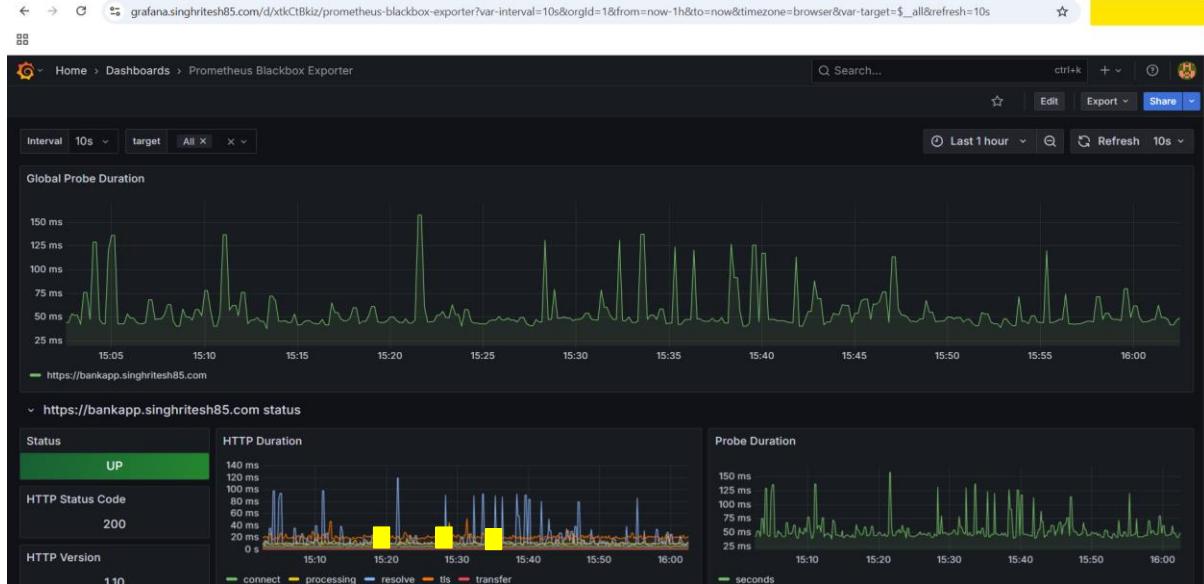




Finally, I was able to perform the synthetic monitoring on the Bankapp Application URL as shown in the screenshot attached below. Application URL <https://bankapp.singhritesh85.com> had been monitored using blackbox exporter.

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.



SSL	SSL Expiry	Average Probe Duration	Average DNS Lookup
YES	months weeks days	42.6 ms	4.34 ms

Configuration of Email to Send notification on Group Email-ID using 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 security interface. On the left, a sidebar lists options like Home, Personal info, Data & privacy, **Security** (which is selected and highlighted in blue), People & sharing, Payments & subscription, and About. A search bar at the top right contains the text "app password". Below the search bar, a dropdown menu titled "Google Account results" lists several items, with "App passwords" being the one highlighted with a yellow box. To the right of the dropdown, there's a shield icon with a checkmark and the text "int secure". Below the dropdown, sections for "Recent security activity" (showing "No security activity or alerts in the last 28 days") and "How you sign in to Google" (with a note to "Make sure you can always access your Google Account by keeping this information up to date") are visible.

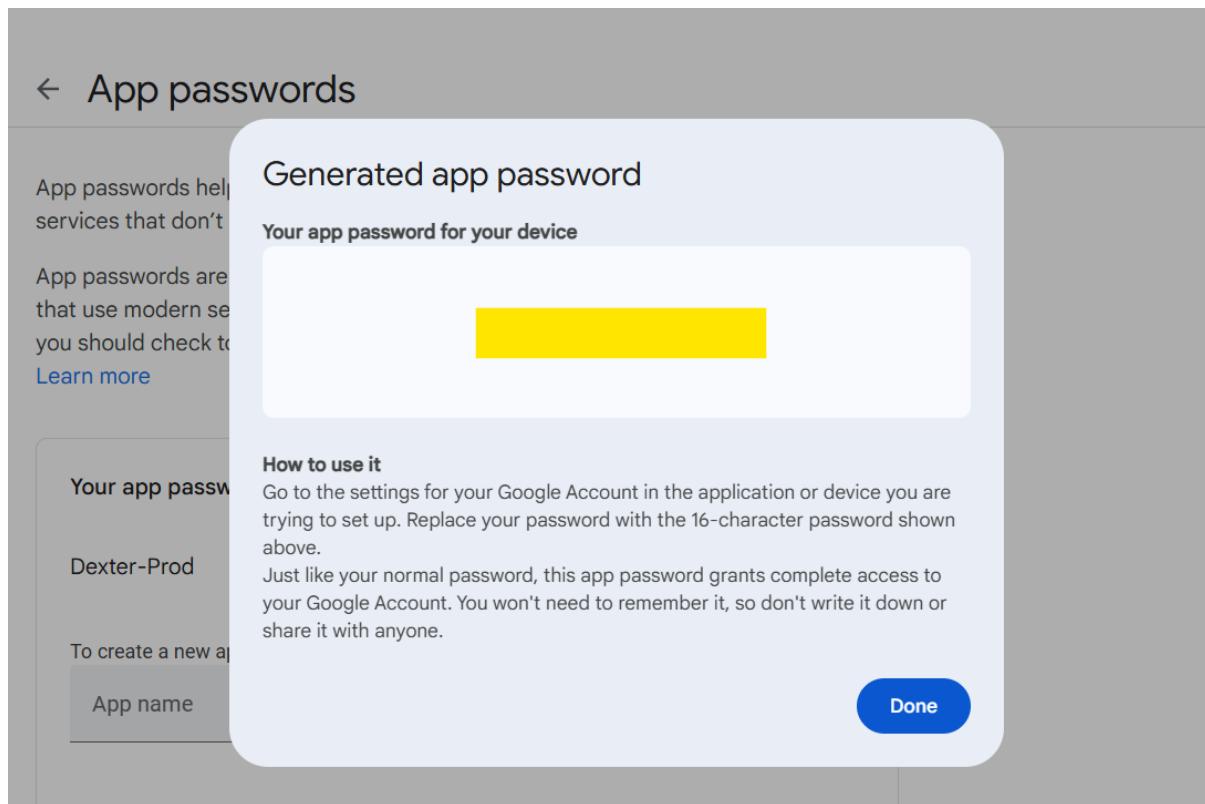
← App passwords

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 a new app password. It starts with a message "You don't have any app passwords." followed by a placeholder text "To create a new app specific password, type a name for it below...". Below this is a text input field containing "App name" and "Dexter-Prod", which is highlighted with a yellow box. At the bottom right of the form is a button labeled "Create" enclosed in a yellow box.



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.

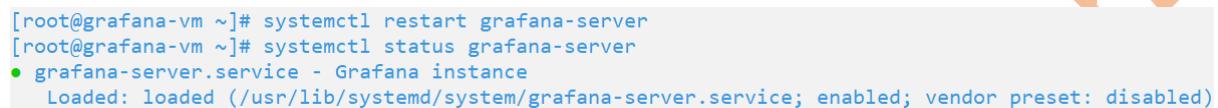
Now, configuration of email to Send notification on Group Email-ID using Jenkins and Grafana is as discussed below.

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 discussed below.

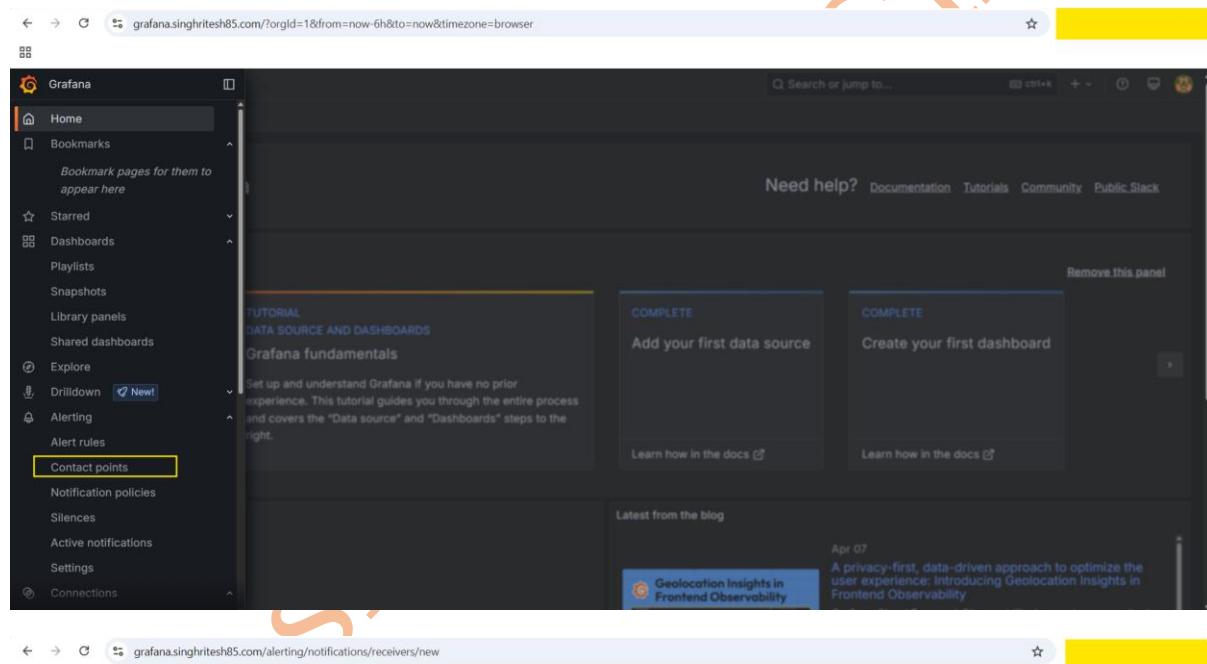
```
#####
##### SMTP / Emailing #####
[smtp]
enabled = true
host = smtp.gmail.com:587
user =
# If the password contains # or ; you have to wrap it with triple quotes. Ex """#password;"""
password =
;cert_file =
;key_file =
skip_verify = true
from_address =
from_name = Grafana Alert for Azure DevOps Pipeline for Production 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 restarted the **grafana-server** service as shown in the screenshot attached below.

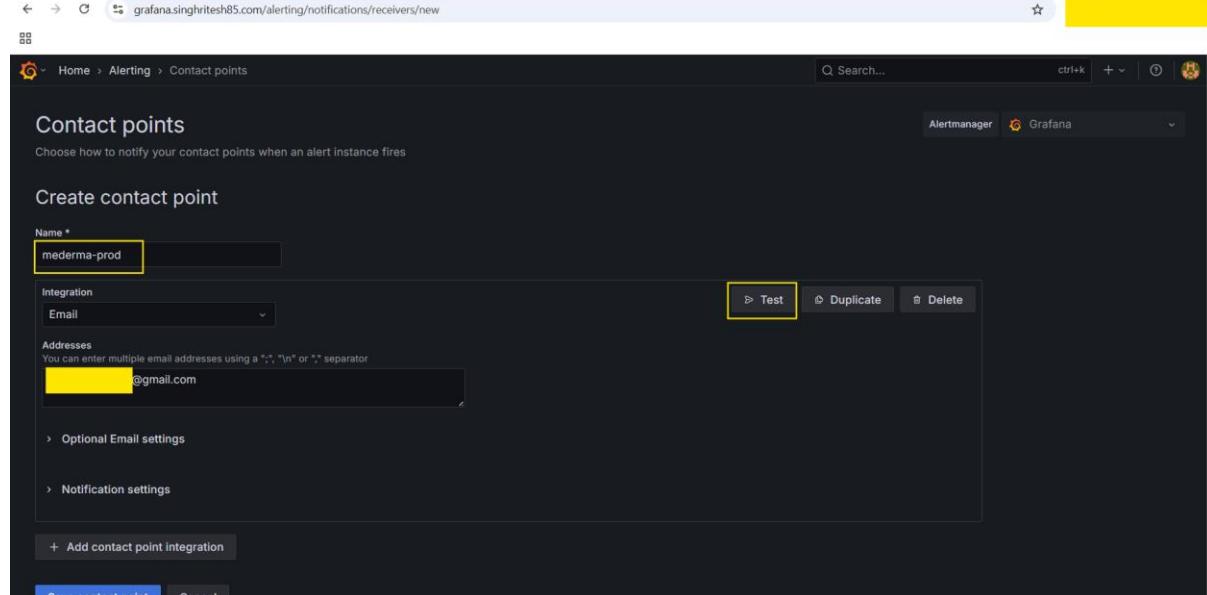


```
[root@grafana-vm ~]# systemctl restart grafana-server
[root@grafana-vm ~]# systemctl status grafana-server
● grafana-server.service - Grafana instance
   Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; vendor preset: disabled)
```

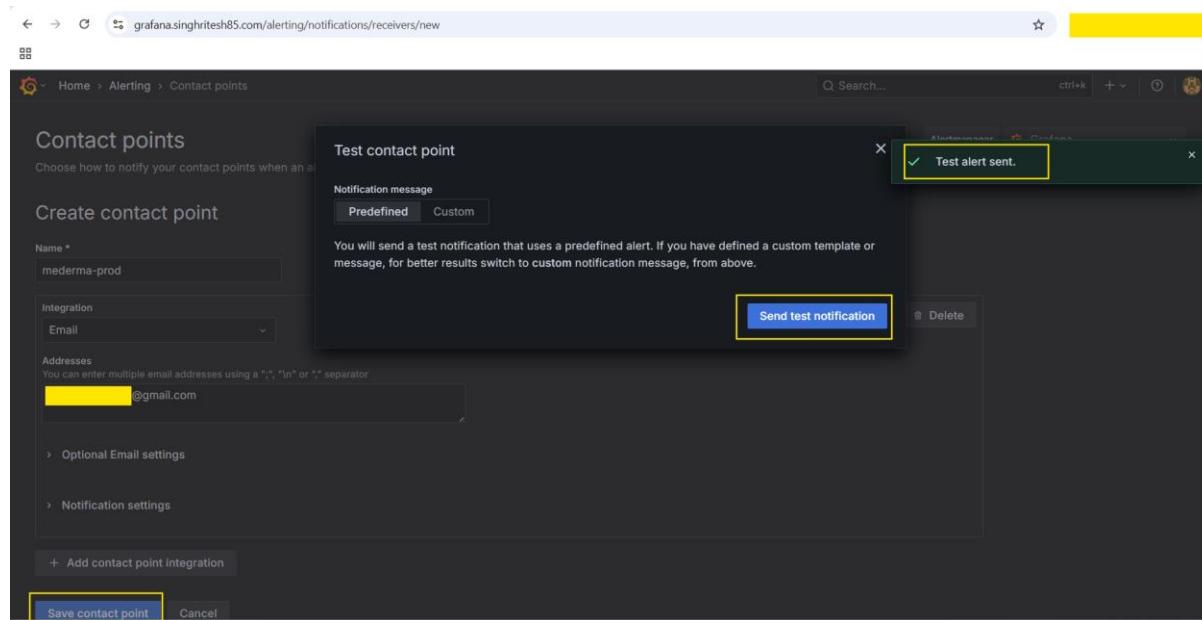
Here I had configured the contact points in Grafana UI as shown in the screenshot attached below.



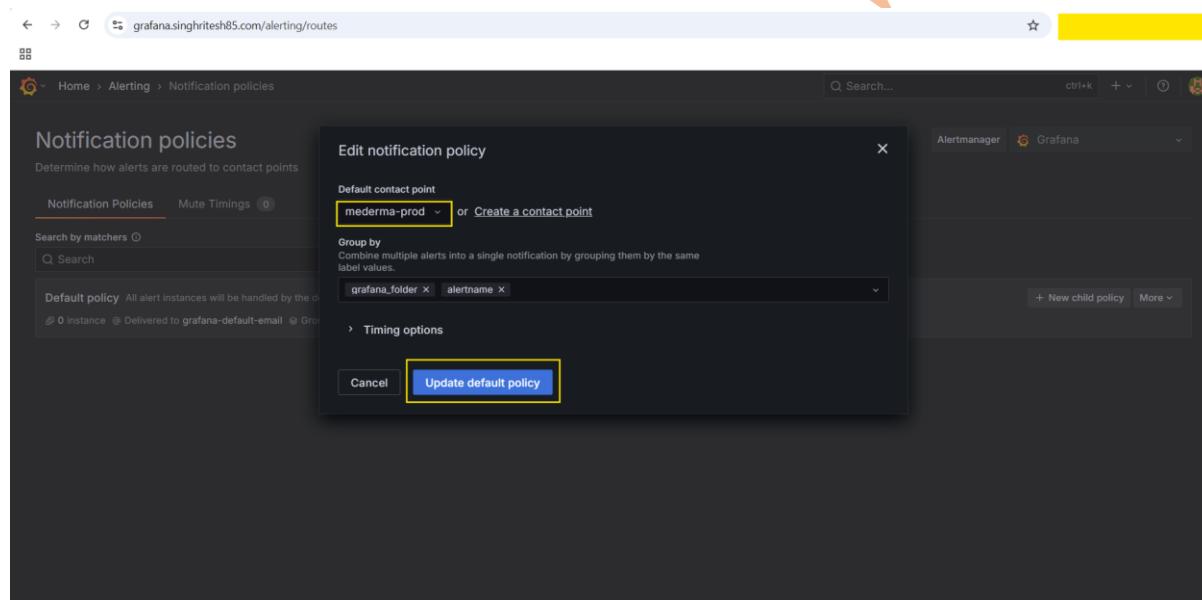
The screenshot shows the Grafana homepage. The left sidebar has a yellow box around the 'Contact points' link under the 'Explore' section. The main content area displays a 'TUTORIAL DATA SOURCE AND DASHBOARDS' section with 'Grafana fundamentals' and a 'COMPLETE Add your first data source' card. There are also cards for 'Create your first dashboard' and 'Learn how in the docs' for both.



The screenshot shows the 'Contact points' configuration page. At the top, it says 'Choose how to notify your contact points when an alert instance fires'. Below that is a 'Create contact point' section. The 'Name' field is filled with 'mederma-prod' and has a yellow box around it. The 'Integration' dropdown is set to 'Email' with a yellow box around it. The 'Addresses' field contains 'mederma-prod@gmail.com' with a yellow box around it. There are buttons for 'Test', 'Duplicate', and 'Delete'. At the bottom, there are sections for 'Optional Email settings' and 'Notification settings', and a 'Save contact point' button.



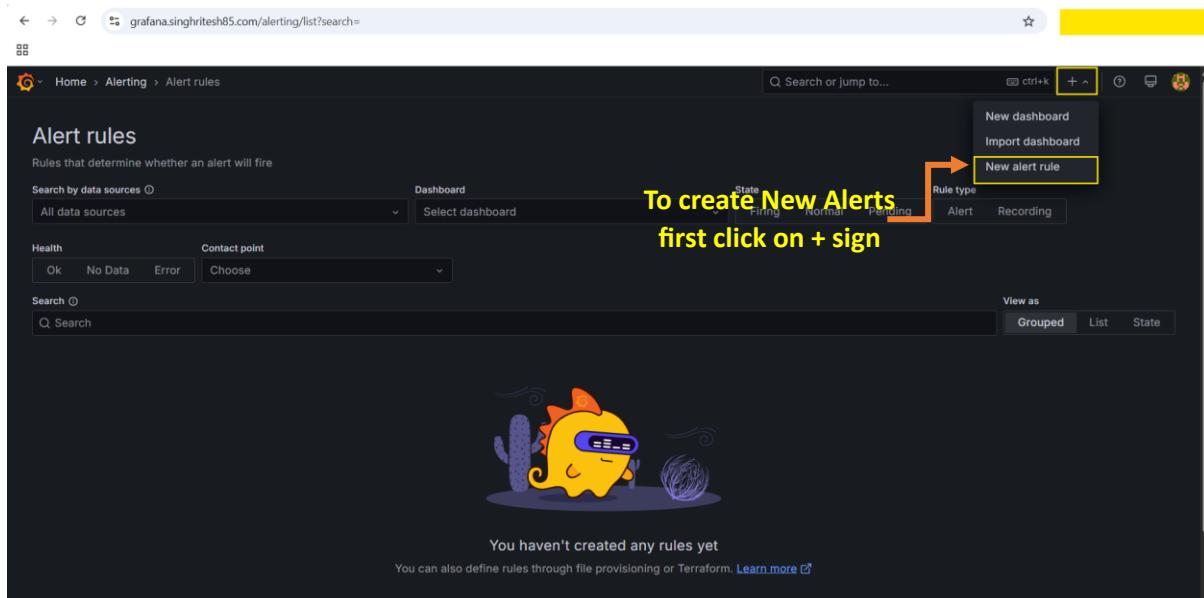
The Default Notification Policy had been changed as shown in the screenshot attached below.



Configure Alert Rule as shown in the screenshot attached below.

grafana.singhritesh85.com/alerting/list?search=

To create New Alerts first click on + sign



grafana.singhritesh85.com/alerting/new

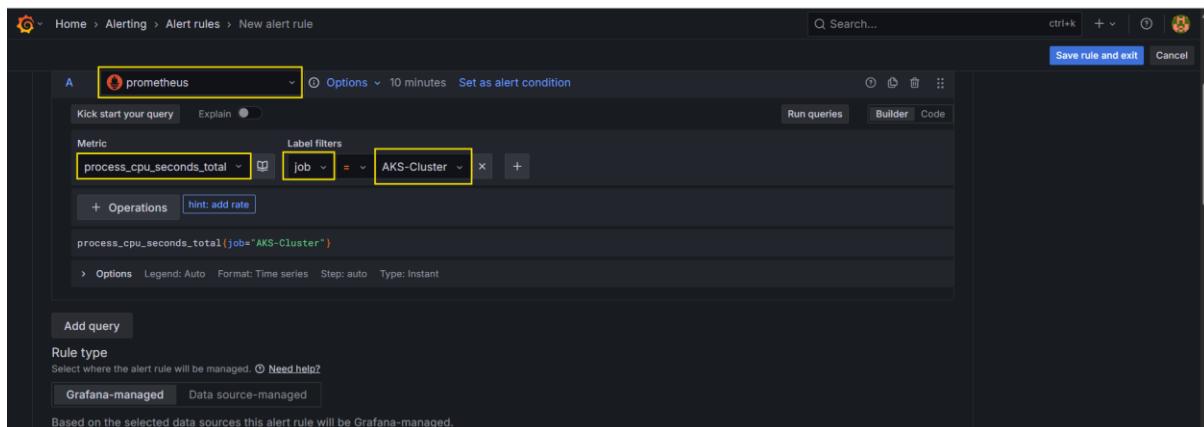
A prometheus Options 10 minutes Set as alert condition

Metric: process_cpu_seconds_total Label filters: job = AKS-Cluster

process_cpu_seconds_total{job="AKS-Cluster"} Options Legend: Auto Format: Time series Step: auto Type: Instant

Add query Rule type: Grafana-managed Data source-managed

Based on the selected data sources this alert rule will be Grafana-managed.



grafana.singhritesh85.com/alerting/new

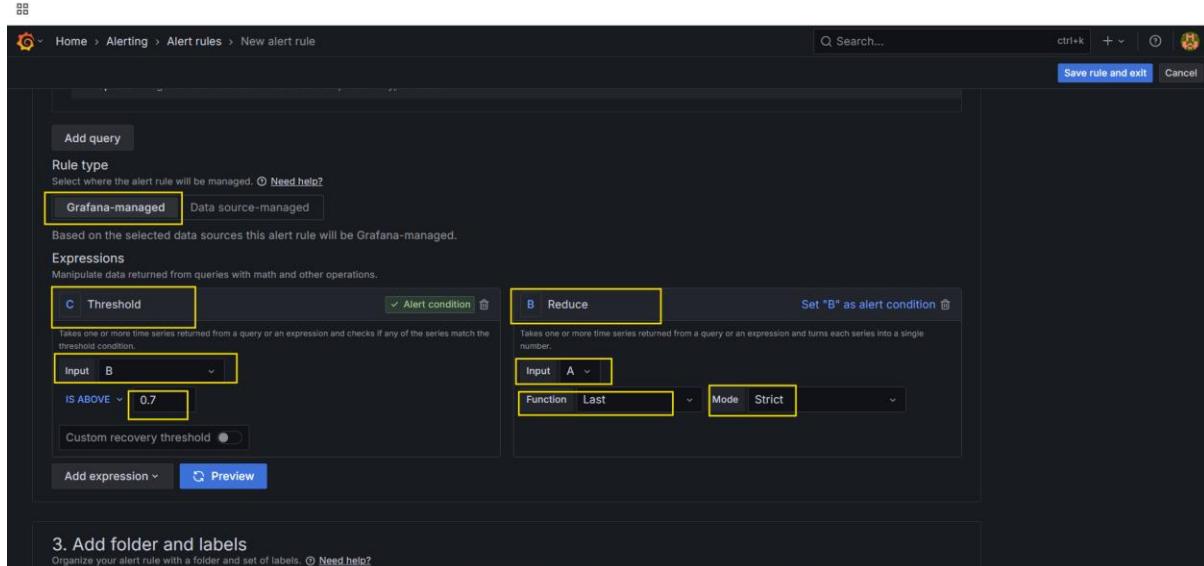
C Threshold Alert condition: IS ABOVE 0.7

B Reduce Input: A Function: Last Mode: Strict

Set "B" as alert condition

3. Add folder and labels

Organize your alert rule with a folder and set of labels. Need help?



The image consists of three vertically stacked screenshots of the Grafana alerting interface, specifically the 'New alert rule' configuration screen.

Screenshot 1: Shows the 'New folder' dialog box open. The 'Folder name' field contains 'CPU time'. The 'Create' button is highlighted with a yellow box.

Screenshot 2: Shows the 'Evaluation group and interval' section. The 'Pending period' dropdown is set to '1m'. The 'Evaluation group' dropdown shows 'Select an evaluation group...' and the '+ New evaluation group' button is highlighted with a yellow box.

Screenshot 3: Shows the 'New evaluation group' dialog box open. The 'Evaluation group name' field contains 'CPU time'. The 'Evaluation interval' dropdown shows '1m' selected, which is highlighted with a yellow box. The 'Create' button is highlighted with a yellow box.

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
mederma-prod

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.

The screenshot shows the Grafana Alerting interface. At the top, there's a search bar with the URL 'grafana.singhritesh85.com/alerting/list'. Below the header, there are several filters: 'Search by data sources' (All data sources), 'Dashboard' (Select dashboard), 'State' (Firing, Normal, Pending, Recovering), and 'Rule type' (Alert, Recording). Under 'Health' and 'Contact point', there are buttons for 'Ok', 'No Data', 'Error' and 'Choose'. A search bar labeled 'Search' is followed by 'View as' buttons for 'Grouped', 'List', and 'State'. A message indicates '1 rule 1 firing'. The main table lists one alert: 'Grafana-managed' for 'CPU time > CPU time'. The alert details are: State (Firing for 24s), Name (mederma-prod-alert), Health (ok), Summary (ok), Next evaluation (in a few seconds), and Actions (View, Edit, More). A large orange arrow points from the 'Firing' button in the table to the 'Firing' status in the alert summary.

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

The screenshot shows an email inbox with a single message from 'Grafana Alert for Azure DevOps Pipeline for Production BankApp <[REDACTED]@gmail.com>' to the user. The subject is '[FIRING:1] mederma-prod-alert CPU time (128.128.144.9100 AKS-Cluster)'. The email body contains the Grafana alert details: 'CPU time > mederma-prod-alert', '1 firing instances', 'Firing' (button), 'mederma-prod-alert' (label), 'View alert' (button), 'Values' (A=14.32, B=14.32, C=1), and 'Labels' (alertname: mederma-prod-alert, grafana_folder: CPU time). A large orange arrow points from the 'Firing' button in the alert summary to the 'Firing' status in the email subject line.



📁 CPU time › mederma-prod-alert

🔥 1 firing instances

Firingmederma-prod-alertView alert

Values

```
A=14.32 B=14.32 C=1
```

Labels

alertname	mederma-prod-alert
grafana_folder	CPU time
instance	128.199.144.9100
job	AKS-Cluster

Silence

Ritesh

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

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

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

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

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

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

Ritesh Kumar Singh