

Kubernetes & VPC

RAJESH JAGARLAMUDI

rajesh@srishsoft.com , 9948867777

SRISH TECHNOLOGIES, KPHB, Beside Arjun Theater

Introduction

Topics to be discussed

- Basics
- VPC - Subnets - Routing table - Internet Gateway
- Baston - Host configuration (configuring AWS CLI)
- Secure Systems Manager
- Kubernetes installation at Client
- Configuring EKS (Elastic Kubernetes Service)
- Managing Nodes
- Pod's - Templates - Services with examples
- Service types (ClusterIP, Node IP, Load Balancer)
- Integrating below as Project

GITLAB - JENKINS - ECR - EKS (DOCKER - CONTAINERS)

VPC - Subnets - Route Tables - Internet Gateway

Login to aws console → Services → Networking & Content Delivery → VPC

The screenshot shows the AWS Management Console interface for the VPC service. The top navigation bar has 'Services' highlighted with a red box. Below it, the main content area is titled 'Find a service by name or feature (for example, EC2, S3 or VM, storage.)'. On the left, a sidebar lists various services: History, VPC, EC2, EKS, ECS, Systems Manager, and IAM. The 'Networking & Content Delivery' section is expanded, showing 'VPC' (which is also highlighted with a red box) and other services like CloudFront, Route 53, API Gateway, Direct Connect, AWS App Mesh, and VPC Flow Logs. To the right, several other service categories are listed under their respective icons: Personal Health Dashboard, AWS Chatbot, Launch Wizard, AWS Compute Optimizer (under Migration & Transfer); Elastic Transcoder, Kinesis Video Streams, MediaConnect, MediaConvert, MediaLive, MediaPackage, MediaStore, MediaTailor, Elemental Appliances & Software (under Media Services); Step Functions, Amazon EventBridge, Amazon MQ, Simple Notification Service, Simple Queue Service, SWF (under Application Integration); and AR & VR, Amazon Sumerian (under AR & VR). A red arrow points from the 'VPC' link in the sidebar to the 'VPC' link in the 'Networking & Content Delivery' section.

VPC - Subnets - Route Tables - Internet Gateway

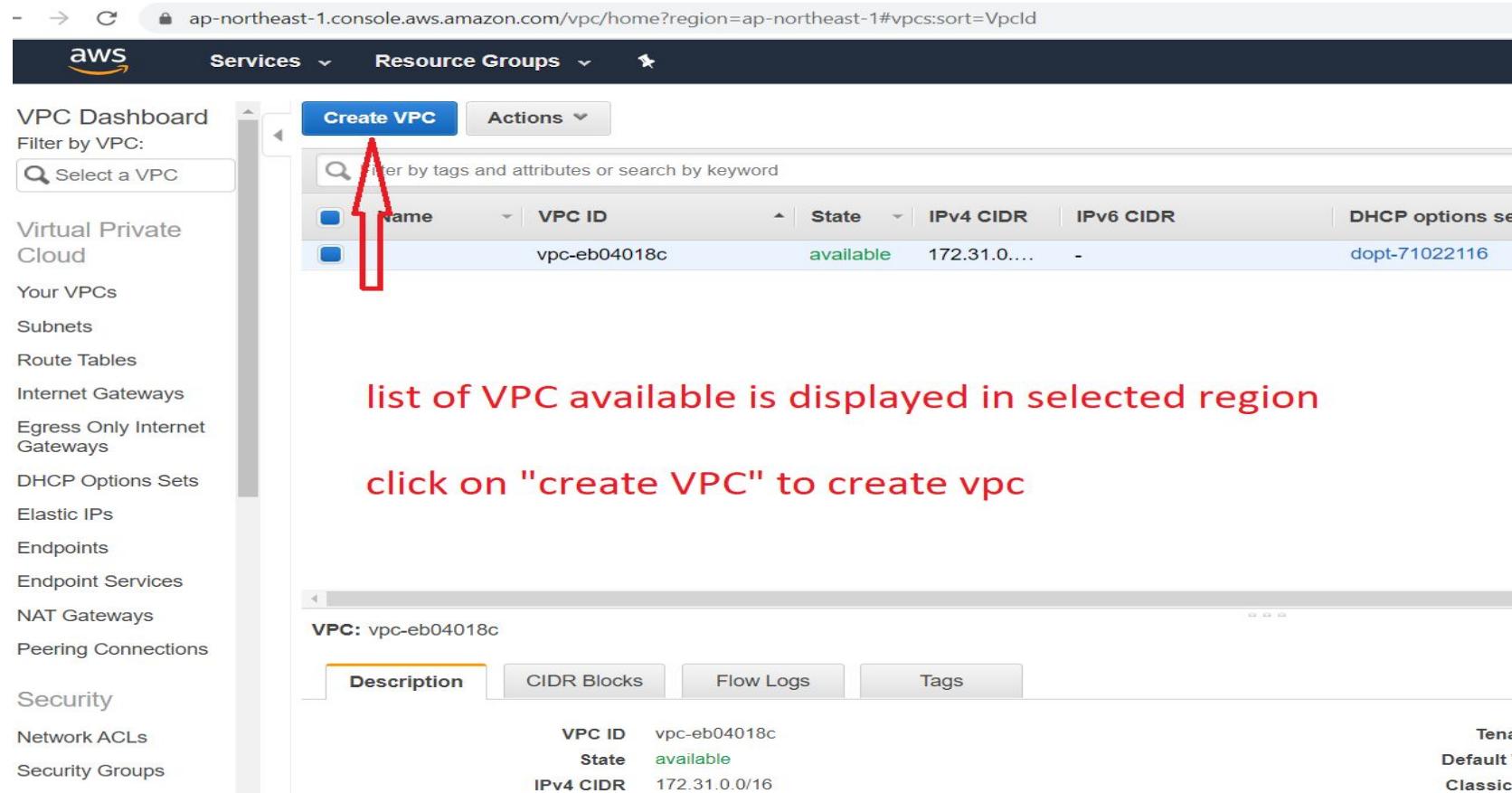
- → C 🔒 ap-northeast-1.console.aws.amazon.com/vpc/home?region=ap-northeast-1#dashboard:

The screenshot shows the AWS VPC Dashboard for the Asia Pacific (Tokyo) region. The left sidebar lists various VPC-related services. The main area displays 'Resources by Region' with the following data:

Resource Type	Region	Count
VPCs	Tokyo	1
NAT Gateways	Tokyo	0
Subnets	Tokyo	3
VPC Peering Connections	Tokyo	0
Route Tables	Tokyo	1
Network ACLs	Tokyo	1
Internet Gateways	Tokyo	1
Security Groups	Tokyo	1
Egress-only Internet Gateways	Tokyo	0
Customer Gateways	Tokyo	0

A red annotation on the right side of the dashboard reads: "this shows complete info vpc, subnets, RT, IG etc". Another red annotation in the center says "click on VPC".

VPC - Subnets - Route Tables - Internet Gateway



The screenshot shows the AWS VPC Dashboard. On the left, a sidebar lists various VPC-related services: VPC Dashboard, Virtual Private Cloud, Your VPCs, Subnets, Route Tables, Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets, Elastic IPs, Endpoints, Endpoint Services, NAT Gateways, Peering Connections, Security, Network ACLs, and Security Groups. A red arrow points from the text "list of VPC available is displayed in selected region" to the search bar at the top of the main content area. The main content area displays a table of VPC resources. The table has columns: Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR, and DHCP options set. One row is visible: Name is "vpc-eb04018c", VPC ID is "vpc-eb04018c", State is "available", IPv4 CIDR is "172.31.0.0/16", IPv6 CIDR is "-", and DHCP options set is "dopt-71022116". Below the table, a red arrow points from the text "click on \"create VPC\" to create vpc" to the "Create VPC" button.

list of VPC available is displayed in selected region

click on "create VPC" to create vpc

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options set
vpc-eb04018c	vpc-eb04018c	available	172.31.0.0/16	-	dopt-71022116

VPC: vpc-eb04018c

Description CIDR Blocks Flow Logs Tags

VPC ID: vpc-eb04018c
State: available
IPv4 CIDR: 172.31.0.0/16

Ten...
Default...
Classic

VPC - Subnets - Route Tables - Internet Gateway

← → 🔍 ap-northeast-1.console.aws.amazon.com/vpc/home?region=ap-northeast-1#CreateVpc:VpcId=vpc-eb04018c

AWS Services Resource Groups 🔍

Rajesh DevOps Tokyo Support

VPCs > Create VPC

Create VPC

A VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances. You must specify an IPv4 address range for your VPC. Specify the IPv4 address range as a Classless Inter-Domain Routing (CIDR) block; for example, 10.0.0.0/16. You cannot specify an IPv4 CIDR block larger than /16. You can optionally associate an Amazon-provided IPv6 CIDR block with the VPC.

Name tag eks enter vpc name

IPv4 CIDR block* 192.168.1.0/24 enter CICR in entered CICR, we will get 256 ip address

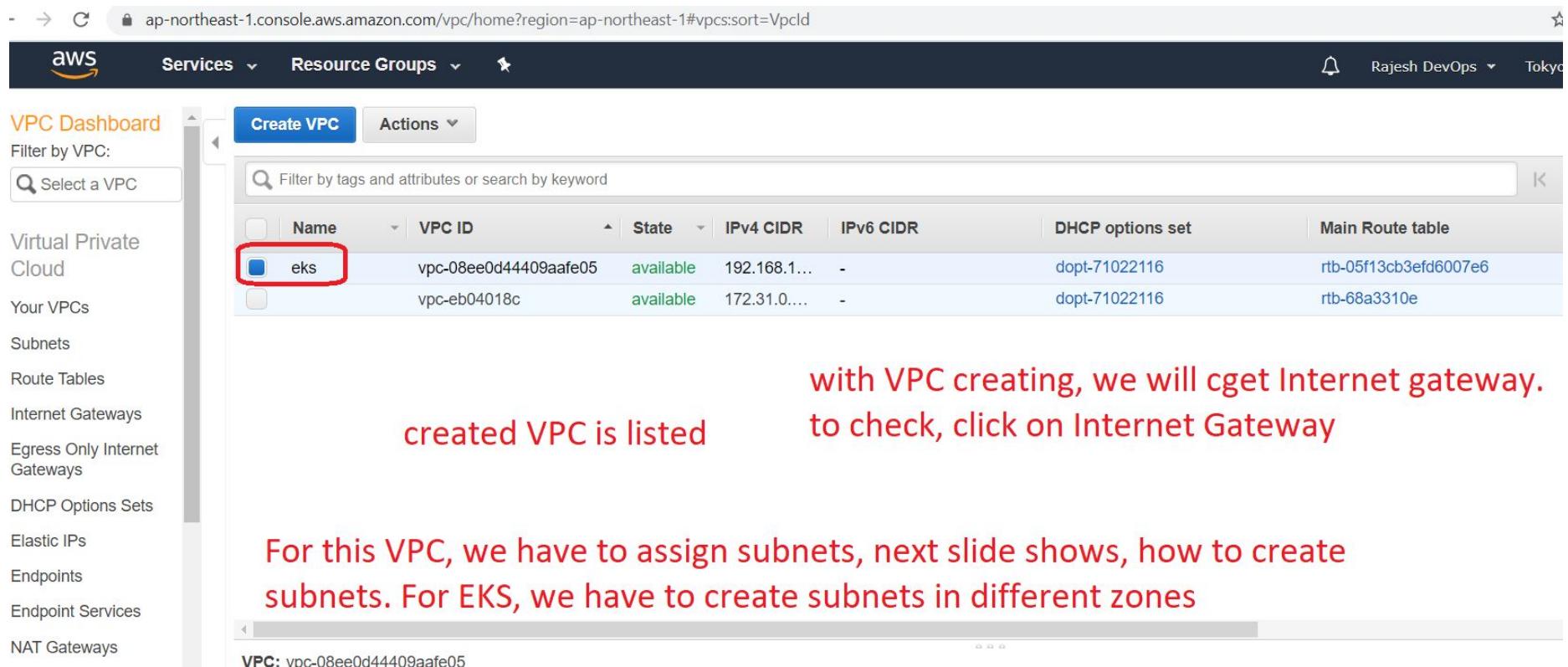
IPv6 CIDR block No IPv6 CIDR Block Amazon provided IPv6 CIDR block

Tenancy Default click on create

* Required Cancel Create



VPC - Subnets - Route Tables - Internet Gateway



The screenshot shows the AWS VPC Dashboard. On the left sidebar, there are several navigation links: VPC Dashboard, Virtual Private Cloud, Your VPCs, Subnets, Route Tables, Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets, Elastic IPs, Endpoints, Endpoint Services, and NAT Gateways. The main content area displays a table of VPCs. The table has columns: Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR, DHCP options set, and Main Route table. Two VPCs are listed: 'eks' (VPC ID: `vpce-08ee0d44409aafe05`, State: available, CIDRs: 192.168.1..., 172.31.0...), and another unnamed VPC (VPC ID: `vpce-eb04018c`, State: available, CIDRs: 172.31.0....). A red box highlights the 'eks' row. Above the table, there are buttons for 'Create VPC' and 'Actions'. A search bar at the top says 'Filter by tags and attributes or search by keyword'.

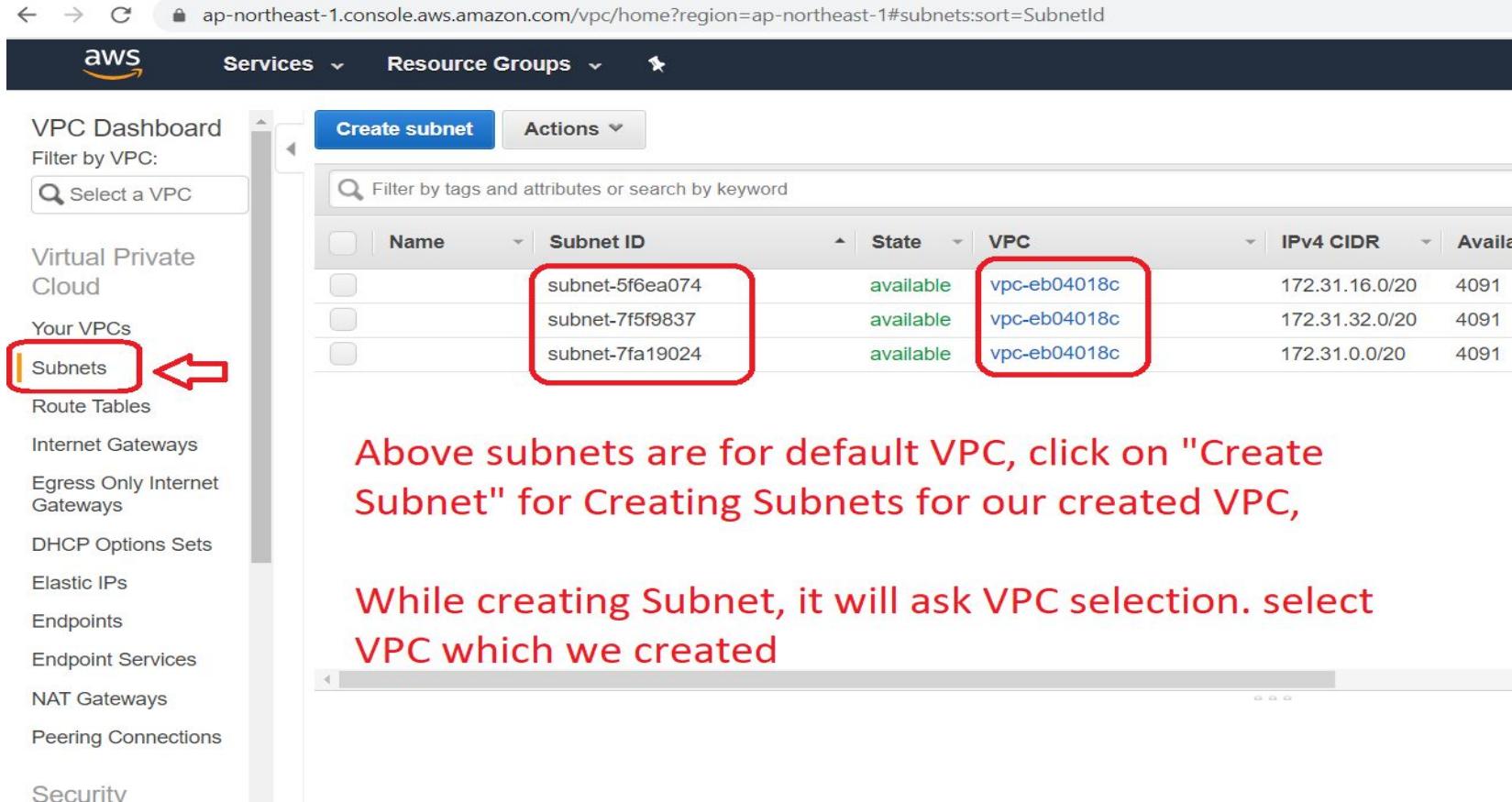
created VPC is listed

with VPC creating, we will get Internet gateway.
to check, click on Internet Gateway

For this VPC, we have to assign subnets, next slide shows, how to create subnets. For EKS, we have to create subnets in different zones

VPC: `vpce-08ee0d44409aafe05`

VPC - Subnets - Route Tables - Internet Gateway



The screenshot shows the AWS VPC Subnets page. On the left sidebar, under 'Your VPCs', the 'Subnets' option is highlighted with a red box and a red arrow pointing to it. The main content area displays a table of subnets:

	Name	Subnet ID	State	VPC	IPv4 CIDR	Available
<input type="checkbox"/>		subnet-5f6ea074	available	vpc-eb04018c	172.31.16.0/20	4091
<input type="checkbox"/>		subnet-7f5f9837	available	vpc-eb04018c	172.31.32.0/20	4091
<input type="checkbox"/>		subnet-7fa19024	available	vpc-eb04018c	172.31.0.0/20	4091

Above subnets are for default VPC, click on "Create Subnet" for Creating Subnets for our created VPC,

While creating Subnet, it will ask VPC selection. select VPC which we created

VPC - Subnets - Route Tables - Internet Gateway

Screenshot of the AWS Subnets 'Create subnet' page.

Subnets > Create subnet

Create subnet

Specify your subnet's IP address block in CIDR format; for example, 10.0.0.0/24. IPv4 block sizes must be between a /16 netmask and /28 netmask, and can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag	eks - subnet - 1	<i>enter subnet name</i>	
VPC*	vpc-08ee0d44409aafe05	<i>select VPC from dropdown which you created</i>	
VPC CIDRs	CIDR	Status	Status Reason
	192.168.1.0/24	associated	
Availability Zone	ap-northeast-1a	<i>select zone, in which you want create subnet</i>	
IPv4 CIDR block*	192.168.1.0/25	<i>this give 128 ip address</i> $2^{(32-25)} = 2^7 = 128$ ip address	

* Required

click on create to create subnet

in AWS, 5 ip's are used for internal routing purpose. so we will get 123 ip's

Create



VPC - Subnets - Route Tables - Internet Gateway

← → ⌂ ap-northeast-1.console.aws.amazon.com/vpc/home?region=ap-northeast-1#CreateSubnet:

aws Services Resource Groups ★

Subnets > Create subnet

Second Subnet

Create subnet

Specify your subnet's IP address block in CIDR format; for example, 10.0.0.0/24. IPv4 block sizes must be between a /16 netmask and /28 netmask, and can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag	eks - subnet - 2	i
VPC*	vpc-08ee0d44409aafe05	i

Same create for second subnet. CIDR changes from Subnet -1 to subnet -2

VPC CIDRs	CIDR	Status	Status Reason
	192.168.1.0/24	associated	

Availability Zone ap-northeast-1d i

IPv4 CIDR block* 192.168.1.128/25 i

* Required

Cancel Create



VPC - Subnets - Route Tables - Internet Gateway

The screenshot shows the AWS VPC Dashboard with the Subnets section selected. A red box highlights the second subnet in the list, which is then expanded to show its detailed configuration.

Subnet Details:

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR
eks - subne...	subnet-04b9d26c9720fb7d0	available	vpc-08ee0d44409aafe05 eks	192.168.1.12...	123	-
eks - subne...	subnet-084bda23a83538960	available	vpc-08ee0d44409aafe05 eks	192.168.1.0/25	123	-
	subnet-5f6ea074	available	vpc-eb04018c	172.31.16.0/20	4091	-
	subnet-7f5f9837	available	vpc-eb04018c	172.31.32.0/20	4091	-
	subnet-7fa19024	available	vpc-eb04018c	172.31.0.0/20	4091	-

Subnet Info:

Subnet: subnet-084bda23a83538960

Description	Flow Logs	Route Table	Network ACL	Tags	Sharing
Subnet ID: subnet-084bda23a83538960					
VPC: vpc-08ee0d44409aafe05 eks					
Available IPv4 Addresses: 123					
Availability Zone: ap-northeast-1a (apne1-az4)					
Network ACL: acl-0a037b4a092f0e5d1					
Auto-assign public IPv4 address: No					
Outpost ID: -					
State: available					
IPv4 CIDR: 192.168.1.0/25					
IPv6 CIDR: -					
Route Table: rtb-05f13cb3efd6007e6					
Default subnet: No					
Auto-assign IPv6 address: No					
Owner: 642975937704					

Text Overlay:

created 2 subnets are listed above, after select any one subnet, you can see below info

VPC - Subnets - Route Tables - Internet Gateway

The screenshot shows the AWS VPC Dashboard. On the left sidebar, under the 'Subnets' section, the 'Subnets' item is selected. The main content area displays a list of subnets. One subnet, 'eks - subnet-084bda23a83538960', is selected and highlighted with a blue border. Below the subnet list, there is a 'Route Table' tab. Under the 'Route Table' tab, there is a table showing route associations. One row in this table has a red box drawn around the 'Target' column, which contains the value 'local'. To the right of the screenshot, there is a descriptive text block.

After selecting subnet, in below, click on Route Table.

only for Internal Create Internet Gateway

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR
eks - subnet-04b9d26c9720fb7d0	subnet-04b9d26c9720fb7d0	available	vpc-08ee0d44409aafe05 ...	192.168.1.12...	123	-
eks - subnet-084bda23a83538960	subnet-084bda23a83538960	available	vpc-08ee0d44409aafe05 ...	192.168.1.0/25	123	-
	subnet-5f6ea074	available	vpc-eb04018c	172.31.16.0/20	4091	-
	subnet-7f5f9837	available	vpc-eb04018c	172.31.32.0/20	4091	-
	subnet-7fa19024	available	vpc-eb04018c	172.31.0.0/20	4091	-

VPC - Subnets - Route Tables - Internet Gateway

The screenshot shows the AWS VPC Dashboard. On the left sidebar, under the 'Internet Gateways' section, there is a red box and a red arrow pointing to the 'Create Internet gateway' button at the top of the main content area. The main content area displays a table with one row of data:

Name	ID	State	VPC	Owner
igw-67929303	igw-67929303	attached	vpc-eb04018c	642975937704

Below the table, a red box and a red arrow point to the 'Create on Internet Gateway' button.

The screenshot shows the 'Create internet gateway' wizard. At the top, there is a red box and a red arrow pointing to the 'Name tag' input field, which contains the value 'eks'. To the right of the input field is a blue 'Create' button with a red arrow pointing to it. Below the input field, there is a note: '* Required' and the instruction 'Enter gateway name and click on create'.

VPC - Subnets - Route Tables - Internet Gateway

← → C ap-northeast-1.console.aws.amazon.com/vpc/home?region=ap-northeast-1#igws:sort=internetGatewayId

aws Services Resource Groups ⚙️ 🔔 Rajesh

VPC Dashboard Filter by VPC: Select a VPC

Virtual Private Cloud Your VPCs Subnets Route Tables Internet Gateways Egress Only Internet Gateways DHCP Options Sets Elastic IPs Endpoints Endpoint Services NAT Gateways Peering Connections

Create internet gateway Actions

Filter by tags and attributes or search by keyword

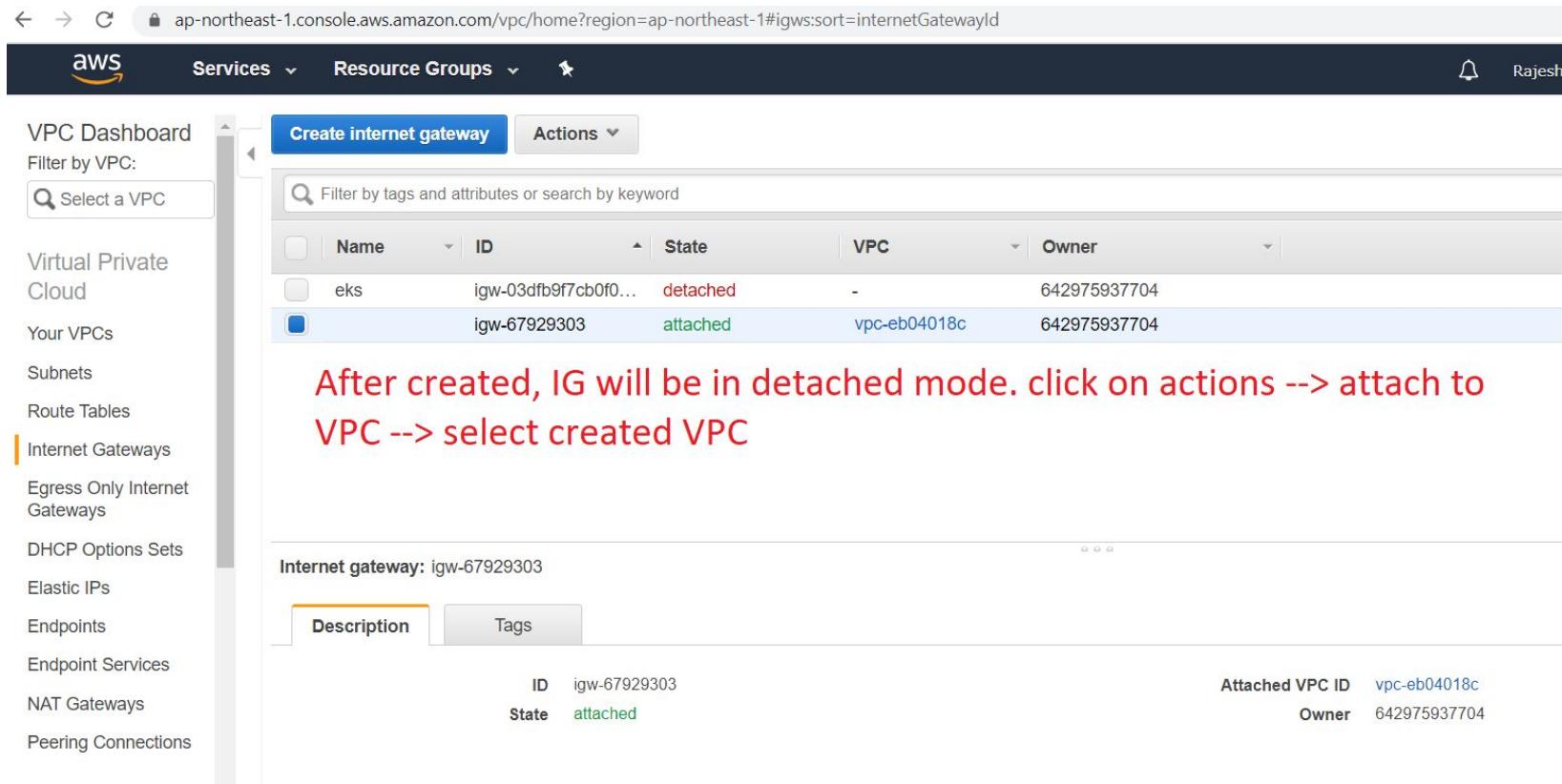
Name	ID	State	VPC	Owner
eks	igw-03dfb9f7cb0f...	detached	-	642975937704
<input checked="" type="checkbox"/>	igw-67929303	attached	vpc-eb04018c	642975937704

After created, IG will be in detached mode. click on actions --> attach to VPC --> select created VPC

Internet gateway: igw-67929303

Description Tags

ID	igw-67929303	Attached VPC ID	vpc-eb04018c
State	attached	Owner	642975937704



VPC - Subnets - Route Tables - Internet Gateway

← → 🔒 ap-northeast-1.console.aws.amazon.com/vpc/home?region=ap-northeast-1#Attach%20to%20VPC:internetGatewayId=igw-03dfb9f7cb0f05e29

aws Services Resource Groups 🔍

Rajesh DevOps

Internet gateways > Attach to VPC

Attach to VPC

Attach an internet gateway to a VPC to enable communication with the internet. Specify the VPC you would like to attach below.

VPC* ⓘ

▶ AWS Command Line Interface command

select created vpc and click on attach

* Required

Cancel **Attach**

VPC - Subnets - Route Tables - Internet Gateway

The screenshot shows the AWS VPC Route Tables - Edit routes interface. A red box highlights the 'Destination' field containing '0.0.0.0/0'. A red arrow labeled 'created gateway' points to the 'Target' dropdown menu, which is set to 'igw-03dfb9f7cb0f05e29'. Another red arrow points to the 'Save routes' button at the bottom right.

Route Tables > Edit routes

Edit routes

Destination	Target	Status	Propagated
192.168.1.0/24	local	active	No
0.0.0.0/0	igw-03dfb9f7cb0f05e29		No

Add route

* Required

Cancel Save routes

enter record as above and click on "save routes"

VPC - Subnets - Route Tables - Internet Gateway

The screenshot shows the AWS VPC Subnets page. On the left sidebar, the 'Subnets' option is highlighted with a red arrow. In the main content area, a subnet named 'eks - subnet...' is selected, indicated by a red box around its checkbox. A context menu is open over this subnet, with the 'Modify auto-assign IP settings' option highlighted in red. Red annotations on the right side of the screen read: 'click on subnets --> select subnet which is created --> right click --> modify auto assign --> check on ipv4'. At the bottom of the page, there is a red box around the 'Edit route table association' button.

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR
eks - subnet...	subnet-04b9d26c9720fb7d0	Active	vpc-08ee0d44409aafe05	192.168.1.12/20	123	-
eks - subnet...	subnet-084bda2e0f3a2a20	Active	vpc-08ee0d44409aafe05	192.168.1.0/25	123	-
eks - subnet...	subnet-5f6ea074a2a2a20	Active	vpc-eb04018c	172.31.16.0/20	4091	-
eks - subnet...	subnet-7f5f9837a2a2a20	Active	vpc-eb04018c	172.31.32.0/20	4091	-
eks - subnet...	subnet-7fa19024a2a2a20	Active	vpc-eb04018c	172.31.0.0/20	4091	-

Subnet: subnet-04b9d26c9720fb7d0

Description Flow Logs **Route Table** Network ACL Tags Sharing

Edit route table association

Route Table: rtb-05f13cb3efd6007e6

--- VPC CREATION COMPLETED ---

Bastion - Host Launching and Configuration

Server or instance which acts as medium between user/client and EKS is known as Baston host

Launch EC2 Instance and select VPC which we created while launching.

The screenshot shows the AWS EC2 Instance Creation Wizard. The top navigation bar includes 'Services' and 'Resource Groups'. Below it, a horizontal menu bar lists steps: 1. Choose AMI (highlighted with a red box), 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The main content area is titled 'Step 1: Choose an Amazon Machine Image (AMI)'. A sub-instruction says: 'An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select a public AMI or you can select one of your own AMIs.' A search bar at the bottom left contains the placeholder text 'Search for an AMI by entering a search term e.g. "Windows"'.

select AMI as Amazon Linux 2

Quick Start
My AMIs
AWS Marketplace
Community AMIs
<input type="checkbox"/> Free tier only (i)

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-068a6cefc24c301d2 (64-bit x86) / ami-0501686af
Amazon Linux 2 comes with five years support. It provides Linux kernel 4.11 tuned for optimal performance on Amazon EC2. 2.29.1, and the latest software packages through extras.
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-0ab3e16f9c414dee7
The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Docker, PHP, MySQL, PostgreSQL, and other packages.
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Microsoft Windows Server 2016 Base - ami-0ca1h6337a71c2fad

Baston - Host Launching and Configuration

← → C ap-northeast-1.console.aws.amazon.com/ec2/v2/home?region=ap-northeast-1#LaunchInstanceWizard:

aws Services Resource Groups ⚡

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower

Number of instances Launch into Auto Scaling Group (i)

Purchasing option Request Spot instances

select VPC which is created and subnet

Network (i) C Create new VPC

Subnet (i) C Create new subnet
123 IP Addresses available

Auto-assign Public IP (i)

Placement group Add instance to placement group

Capacity Reservation (i) C Create new Capacity Reservation

IAM role (i) ... C Create new IAM Role

Baston - Host Launching and Configuration

File systems ⓘ Add file system Add to user data C Create new file system

▼ Network interfaces ⓘ

Device	Network Interface	Subnet	Primary IP	Secondary IP addresses	IPv6 IF
eth0	New network interface ▾	subnet-04b9d26c ▾	Auto-assign	Add IP	Add IF

Add Device

Advanced Details

By Using Advanced Details, can install service as below.
below script is for installing SSM agent in baston host server

User data ⓘ

As text As file Input is already base64 encoded

```
#!/bin/bash
sudo yum install -y https://s3.amazonaws.com/ec2-downloads-
windows/SSMAgent/latest/1/x86_64/amazon-ssm-agent.rpm
systemctl start amazon-ssm-agent
systemctl enable amazon-ssm-agent
```

SSM --> Secure Session Manager

Bastion - Host Launching and Configuration

launched baston - host instance

Instance: i-07bca58c8db510c4a Public IP: 18.182.20.17

Description	Status Checks	Monitoring	Tags
Instance ID	i-07bca58c8db510c4a		
Instance state	running		
Instance type	t2.micro		
Elastic IPs			
Availability zone	ap-northeast-1d		
Security groups	default, view inbound rules, view outbound rules		
Scheduled events	No scheduled events		
AMI ID	amzn2-ami-hvm-2.0.20191116.0-x86_64-gp2 (ami-068a6cefc24c301d2)		
Platform	-		
IAM role	-		
Key pair name	tokio		
Owner	642975937704		
Launch time	December 22, 2019 at 8:43:12 PM UTC+5:30 (less than 1 hour ago)		
Public DNS (IPv4)	18.182.20.17		
IPv4 Public IP	18.182.20.17		
IPv6 IPs			
Private DNS	ip-192-168-1-237.ap-northeast-1.compute.internal		
Private IPs	192.168.1.237		
Secondary private IPs			
VPC ID	vpc-08ee0d44409aafe05 (eks)		
Subnet ID	subnet-04b9d26c9720fb7d0 (eks - subnet - 2)		
Network interfaces	eth0		
Source/dest. check	True		
T2/T3 Unlimited	Disabled		
EBS-optimized	False		
Root device type	ebs		

IAM - Identity & Access Management

Create a role (IAM) for SSM login and attach to Baston - host EC2 instance

AWS Login → Services → Security, Identity & Compliances → IAM (click), below page appears

The screenshot shows the AWS IAM Dashboard. On the left, there's a navigation sidebar with links like Dashboard, Access management, and Access reports. The main area displays a welcome message and summary statistics: Users: 0, Groups: 0, Roles: 7 (which is highlighted with a red box), Customer Managed Policies: 0, and Identity Providers: 0. Below this, there's a 'Security Status' section with five items, each preceded by a yellow warning icon:

- Delete your root access keys
- Activate MFA on your root account
- Create individual IAM users
- Use groups to assign permissions
- Apply an IAM password policy

A red box also highlights the 'Roles: 7' text. A large red annotation in the center-right says "click on roles --> for creating role".

IAM - Identity & Access Management

console.aws.amazon.com/iam/home?region=ap-northeast-1#/roles

aws Services Resource Groups Rajesh

Identity and Access Management (IAM)

Dashboard

Access management

Groups

Users

Roles

Policies

Identity providers

Account settings

Access reports

Access analyzer

Archive rules

Analyzer details

Credential report

Organization activity

Service control policies (SCPs)

Common Scenarios for Roles

Create role Delete role

available roles appears below and click on create to create new role

Role name	Trusted entities	Last activity
AmazonSSMRoleForAutomationAssumeQuickSetup	AWS service: ssm	None
AmazonSSMRoleForInstancesQuickSetup	AWS service: ec2	None
AWSServiceRoleForAmazonSSM	AWS service: ssm (Service-Linked role)	Today
AWSServiceRoleForRDS	AWS service: rds (Service-Linked role)	11 days
AWSServiceRoleForSupport	AWS service: support (Service-Linked role)	None
AWSServiceRoleForTrustedAdvisor	AWS service: trustedadvisor (Service-Linked ...)	None
rds-monitoring-role	AWS service: monitoring.rds	11 days

IAM - Identity & Access Management

→ C [console.aws.amazon.com/iam/home?region=ap-northeast-1#/roles\\$new?step=type](https://console.aws.amazon.com/iam/home?region=ap-northeast-1#/roles$new?step=type)

aws Services Resource Groups ⌂ Rajesh DevOps

Create role

Select type of trusted entity

1 2 3 4

AWS service EC2, Lambda and others 

Another AWS account Belonging to you or 3rd party

Web identity Cognito or any OpenID provider

SAML 2.0 federation Your corporate directory

Allows AWS services to perform actions on your behalf. [Learn more](#)

Choose the service that will use this role

EC2  Allows EC2 instances to call AWS services on your behalf. **click on EC2**

Lambda Allows Lambda functions to call AWS services on your behalf.

API Gateway	CodeBuild	EKS	Kinesis	S3
AWS Backup	CodeDeploy	EMR	Lambda	SMS
AWS Chatbot	CodeStar Notifications	ElastiCache	Lex	SNS
AWS Support	Comprehend	Elastic Beanstalk	License Manager	SWF
Amplify	Config	Elastic Container Service	Machine Learning	SageMaker
AppStream 2.0	Connect	Elastic Transcoder	Macie	Security Hub

IAM - Identity & Access Management

console.aws.amazon.com/iam/home?region=ap-northeast-1#/roles\$new?step=permissions&commonUseCase=EC2%2BEC2&selectedUseCase=EC2

Rajesh DevOps

Create role

Attach permissions policies

Choose one or more policies to attach to your new role.

Showing 13 results

	Policy name	Used as
<input type="checkbox"/>	AmazonEC2RoleforSSM	None
<input type="checkbox"/>	AmazonSSMAutomationApproverAccess	None
<input type="checkbox"/>	AmazonSSMAutomationRole	None
<input type="checkbox"/>	AmazonSSMDirectoryServiceAccess	None
<input checked="" type="checkbox"/>	AmazonSSMFullAccess	None
<input type="checkbox"/>	AmazonSSMMaintenanceWindowRole	None
<input type="checkbox"/>	AmazonSSMManagedInstanceCore	Permissions policy (1)
<input type="checkbox"/>	AmazonSSMReadOnlyAccess	None

* Required

IAM - Identity & Access Management

This screenshot shows the first step of creating a new role in AWS IAM. It's titled 'Create role' and has four steps numbered 1 to 4. Step 1 is selected. A table lists a single tag: 'name' with 'SSM role' as the value. Below the table, it says 'You can add 49 more tags.' The URL in the address bar is [console.aws.amazon.com/iam/home?region=ap-northeast-1#/roles\\$new?step=tags&commonUseCase=EC2%2BEC2&selectedUseCase=EC2&policies=arn:aws:iam::awspolicy%2FAmazonSSMRolePolicy](https://console.aws.amazon.com/iam/home?region=ap-northeast-1#/roles$new?step=tags&commonUseCase=EC2%2BEC2&selectedUseCase=EC2&policies=arn:aws:iam::awspolicy%2FAmazonSSMRolePolicy).

Create role

1 2 3 4

Add tags (optional)

IAM tags are key-value pairs you can add to your role. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this role. [Learn more](#)

Key	Value (optional)	Remove
name	SSM role	x
Add new key		

You can add 49 more tags.

add tag name

click on review



Cancel Previous Next: Review

This screenshot shows the second step of creating a new role in AWS IAM, titled 'Review'. It includes fields for 'Role name' (set to 'ssmrole') and 'Role description' (set to 'Allows EC2 instances to call AWS services on your behalf'). The URL in the address bar is [console.aws.amazon.com/iam/home?region=ap-northeast-1#/roles\\$new?step=review&commonUseCase=EC2%2BEC2&selectedUseCase=EC2&policies=arn:aws:iam::awspolicy%2FAmazonSSMRolePolicy](https://console.aws.amazon.com/iam/home?region=ap-northeast-1#/roles$new?step=review&commonUseCase=EC2%2BEC2&selectedUseCase=EC2&policies=arn:aws:iam::awspolicy%2FAmazonSSMRolePolicy).

Create role

1 2 3 4

Review

Provide the required information below and review this role before you create it.

Role name*

ssmrole

Use alphanumeric and '+=_@-' characters. Maximum 64 characters.

Role description

Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use alphanumeric and '+=_@-' characters.

Trusted entities AWS service: ec2.amazonaws.com

Policies [AmazonSSMFullAccess](#)

Permissions boundary Permissions boundary is not set

The new role will receive the following tag

Key	Value
name	SSM role

* Required

enter name and click on
"create role"



Cancel Previous Create role

Attaching Created IAM role (SSM) to Baston instance

console.aws.amazon.com/fam/home?region=ap-northeast-1#roles

Rajesh DevOps

Services Resource Groups

Create role Delete role

Role name ▾ Trusted entities Last activity ▾

- AmazonSSMRoleForAutomationAssumeQuickSetup AWS service: ssm None
- AmazonSSMRoleForInstancesQuickSetup AWS service: ec2 None
- AWSServiceRoleForAmazonSSM AWS service: ssm (Service-Linked role) Today
- AWSServiceRoleForRDS AWS service: rds (Service-Linked role) 11 days
- AWSServiceRoleForSupport AWS service: support (Service-Linked role) None
- AWSServiceRoleForTrustedAdvisor AWS service: trustedadvisor (Service-Linked ...) None
- rds-monitoring-role AWS service: monitoring.rds 11 days
- ssmrole** AWS service: ec2 None

Created role appears

aws Services Resource Groups

New EC2 Experience Tell us what you think

EC2 Dashboard New

Launch Instance

Filter by tags and attributes or search by keyword

Name Instance ID Instance Type Availability Zone Instance State Status Checks Alarm Status Public

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public
i-07bca58c8db510c4a	t2.micro	ap-northeast-1d	terminated				
i-0d1d1ca119796737			east-1d	running	2/2 checks ...	None	

Connect Get Windows Password Create Template From Instance Launch More Like This

Description Status Checks

Instance State

Instance Settings

- Add/Edit Tags
- Attach to Auto Scaling Group
- Attach/Replace IAM Role
- Change Instance Type
- Change Termination Protection
- View/Change User Data
- Change Shutdown Behavior
- Change T2/T3 Unlimited
- Get System Log
- Get Instance Screenshot
- Modify Instance Placement
- Modify Capacity Reservation Settings

Instance ID: i-0d1d1ca119796737
Instance state: running
Instance type: t2.micro
Elastic IPs: ap-northeast-1d
Availability zone: default, view inbound rules, view outbound rules
Security groups: No scheduled events
AMI ID: amzn2-ami-hvm-2.0.20191116-x86_64-068a6cefc24c301d2
Platform: -
IAM role: -
Key pair name: tokio

Public DNS (IPv4): -
IPv4 Public IP: 3.112.108.56
Private DNS: ip-192-168-1-193.a
Private IPs: 192.168.1.193
Secondary private IPs:
VPC ID: vpc-08ee0d44409a
Subnet ID: subnet-04bd26c9f
Network interfaces: eth0
Source/dest. check: True
T2/T3 Unlimited: Disabled

goto EC2 instance --> select baston host instance -->
Instance Settings --> Attach / Replace IAM Role

Attaching Created IAM role (SSM) to Baston instance

← → C ap-northeast-1.console.aws.amazon.com/ec2/v2/home?region=ap-northeast-1#AttachReplaceIAMRole: ☆ S O | :

AWS Services Resource Groups Rajesh DevOps Tokyo Support

Instances > Attach/Replace IAM Role

Attach/Replace IAM Role

Select an IAM role to attach to your instance. If you don't have any IAM roles, choose Create new IAM role to create a role in the IAM console. If an IAM role is already attached to your instance, the IAM role you choose will replace the existing role.

Instance ID i-0d1d1ca1197969737 () ⓘ

IAM role* No Role ⏺ C Create new IAM role ⓘ

* Required

Filter by attributes

Profile Name

- No Role
- AmazonSSMRoleForInstancesQuickSetup
- ssmrole

Cancel Apply



Configuring System Manager

ap-northeast-1.console.aws.amazon.com/ec2/v2/home?region=ap-northeast-1#Instances:sort=instanceId

aws Services Resource Groups

History

EC2

VPC

Systems Manager

EKS

ECS

IAM

services --> search by System Manager

System Manage

Compute

- EC2
- Lightsail
- ECR
- ECS
- EKS
- Lambda
- Batch
- Elastic Beanstalk
- Serverless Application Repository
- AWS Outposts
- EC2 Image Builder

Customer Enablement

- AWS IQ
- Support
- Managed Services

Blockchain

- Amazon Managed Blockchain

Satellite

- Ground Station

Quantum Technologies

ap-northeast-1.console.aws.amazon.com/systems-manager/home?region=ap-northeast-1

aws Services Resource Groups

AWS Systems Manager X

Quick Setup

Operations Management

Explorer New

OpsCenter

CloudWatch Dashboard

Trusted Advisor & PHD

An Ath EM Clo Ela Kin Qui Dal AW MS

Application Management

Resource Groups

AppConfig New

Parameter Store

Actions & Change

Automation

Change Calendar New

Maintenance Windows

MANAGEMENT TOOLS

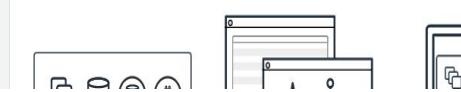
click on "get started with systems manager"

Gain Operational Insight on AWS Resources.

Get Started with Systems Manager

View operational data for groups of resources, so you can quickly identify and use those resources.

How it works



Configuring System Manager

The screenshot shows the AWS Systems Manager Quick Setup interface. On the left, a sidebar lists various AWS services like CloudWatch Metrics, Lambda, CloudWatch Metrics Insights, CloudWatch Metrics Dashboard, CloudWatch Metrics Advisor & PHD, CloudWatch Metrics Store, CloudWatch Metrics Groups, CloudWatch Metrics Config, CloudWatch Metrics Store, CloudWatch Metrics & Change, CloudFormation, CloudWatch Metrics Calendar, and CloudWatch Metrics Maintenance Windows. The main content area is titled "Configure required security roles and commonly used Systems Manager capabilities". It includes sections for "Permissions (Required)" and "Assume role for Systems Manager". Under "Permissions (Required)", there are two options: "Instance profile role" (radio button) and "Choose an existing role" (radio button, highlighted with a red box). Under "Assume role for Systems Manager", there are also two options: "Use the default role" (radio button) and "Choose an existing role" (radio button, highlighted with a red box). Both "ssmrole" and "ssmrole" are listed in dropdown menus under their respective sections. A large red box highlights the text "click on \"Set up systems manager\"".

Configure required security roles and commonly used Systems Manager capabilities.

Permissions (Required)
Use the following options to configure two roles that give Systems Manager permission to access your instances and run commands on them.

Instance profile role

Use the default role
Quick Setup creates a new instance profile that uses a secure IAM permissions policy. Quick Setup assigns the profile to the instances that you specify.

Choose an existing role
Uses an existing instance profile. The instance profile must contain the required permissions policy. Choose the instance profile from the following list.

IAM Roles
ssmrole

Assume role for Systems Manager

Use the default role
Quick Setup creates a new assume role that enables Systems Manager to securely run commands on your instances.

Choose an existing role
Uses an existing service role. The role must contain the required permissions policy. Choose the role from the following list.

IAM Roles
ssmrole

click on "Set up systems manager"

Configuring System Manager

The screenshot shows the AWS Systems Manager Session Manager interface. On the left, the navigation pane is open, showing various services like Application Management, Resource Groups, and Session Manager. The 'Session Manager' option is highlighted with a red box and a red arrow pointing to it from the bottom-left. The main panel displays a table titled 'Target instances' with one row. The row contains the following information:

Instance name	Instance ID	Agent version	Instance state	Availability zone	Platform
	i-0d1d1ca1197969737	2.3.786.0	running	ap-northeast-1d	Amazon Linux

At the bottom right of the main panel are two buttons: 'Cancel' and 'Start session'. A red arrow points from the bottom-right towards the 'Start session' button.

The screenshot shows a terminal session within the AWS Systems Manager Session Manager. The terminal window has a black background and white text. At the top, the URL is visible: ap-northeast-1.console.aws.amazon.com/systems-manager/session-manager/i-0d1d1ca1197969737?region=ap-northeast-1. Below the URL, the session ID is 'Session ID: root-0d96d3e55c964dd5f' and the instance ID is 'Instance ID: i-0d1d1ca1197969737'. On the far right of the terminal window, there is a red 'Terminate' button with a red arrow pointing to it from the bottom-right.

sh-4.2\$

click on "Terminate" to logout from instance

login to Baston host

AWS - EKS

AWS → Services → Compute → EKS → clusters (left menu) → create cluster

The screenshot shows the 'Create cluster' wizard in the AWS EKS console. The 'General configuration' step is selected. It includes fields for 'Cluster name' (with placeholder 'Enter a unique name for your Amazon EKS cluster.'), 'Kubernetes version' (set to '1.14'), and 'Role name' (with a red box highlighting the 'Info' link). Below this, there is a note: 'Select the IAM Role to allow Amazon EKS and the Kubernetes control plane to manage AWS resources on your behalf.' The 'Networking' section is partially visible at the bottom.

**create an IAM role & VPC for eks
steps for creating was taught at
respective topics**

General configuration

Cluster name

Enter a unique name for your Amazon EKS cluster.

Kubernetes version

Select the Kubernetes version to install.

1.14

Role name Info

Select the IAM Role to allow Amazon EKS and the Kubernetes control plane to manage AWS resources on your behalf.

Networking Info

VPC Info

Select a VPC to use for your EKS Cluster resources.

vpc-eb04018c - 172.31.0.0/16



Services ▾

Resource Groups ▾

Select the Kubernetes version to install.

1.14

Role name Info

Select the IAM Role to allow Amazon EKS and the Kubernetes control plane to manage AWS resources on your behalf.

myeksiam

after creating iam role for eks, select iam role

Networking Info

VPC

Select a VPC to use for your EKS Cluster resources.

vpc-08ee0d44409aafe05 - 192.168.1.0/24

select vpc which is created

Subnets

Choose the subnets in your VPC where your worker nodes will run.

Find subnet

subnets

<input checked="" type="checkbox"/>	Subnet	Name	Availability Zone	Subnet IPv4 CIDR
<input checked="" type="checkbox"/>	subnet-04b9d26c9720fb7d0	eks - subnet - 2	ap-northeast-1d	192.168.1.128/25
<input checked="" type="checkbox"/>	subnet-084bda23a83538960	eks - subnet - 1	ap-northeast-1a	192.168.1.0/25

Security groups Info



Services ▾

Resource Groups ▾



Private access

Enable private API server endpoint access

Disabled

Public access

Enable public API server endpoint access

Enabled

enable public access

▶ Advanced Settings

Logging

CloudWatch log group

EKS automatically creates a CloudWatch log group for you when you enable logging.

API server

Logs pertaining to API requests to the cluster

Enabled

enable logs

Audit

Logs pertaining to cluster access via the Kubernetes API

Enabled

click on below create option

Authenticator

Logs pertaining to authentication requests into the cluster

Enabled

Controller manager

Logs pertaining to state of cluster controllers

Enabled

Created EKS cluster

ap-northeast-1.console.aws.amazon.com/eks/home?region=ap-northeast-1#/clusters/myekscluster

Rajesh DevOps Tokyo Support

EKS > Clusters > myekscluster

myekscluster

General configuration

Kubernetes version 1.14

Platform version eks.7

Status ACTIVE

API server endpoint https://920F07F15102D80D79256F105B31A1E3.yl4.ap-northeast-1.eks.amazonaws.com

OpenID Connect provider URL https://oidc.eks.ap-northeast-1.amazonaws.com/id/920F07F15102D80D79256F105B31A1E3

Cluster ARN arn:aws:eks:ap-northeast-1:642975937704:cluster/myekscluster

Certificate authority LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUN5RENDQWJDZ0F3SUJBZ0LCQURBTkJna3Foa2lHOXcwQkFRc0ZBREFWTVJNd0VRWURWUVFERXdwcmRXSmwKY201bGRHVnpNQjRYRFRNU1USXlOakV4TlRVd01Gb1hEVek1TVRJeU

click on "add node group" to add node (servers / instance) for this cluster

Cluster IAM Role ARN arn:aws:iam::642975937704:role/myeksiam

Add node group

Node Groups (0)

IAM Role for Adding Nodes to cluster

Creating the Amazon EKS Worker Node IAM Role

If you created your worker nodes by following the steps in the [Getting Started with the AWS Management Console](#) or [Getting Started with eksctl](#) topics, then the worker node role account already exists and you don't need to manually create it. You can use the following procedure to create the Amazon EKS worker node role if you do not already have one for your account.

To create your Amazon EKS worker node IAM role

this slide for creating IAM role for adding nodes

1. Open the AWS CloudFormation console at <https://console.aws.amazon.com/cloudformation>.
2. Choose **Create stack**.
3. For **Choose a template**, select **Specify an Amazon S3 template URL**.
4. Paste the following URL into the text area and choose **Next**:

```
https://amazon-eks.s3-us-west-2.amazonaws.com/cloudformation/2019-11-15/amazon-eks-nodegroup-role.yaml
```



5. On the **Specify Details** page, fill out the parameters accordingly, and then choose **Next**.
 - **Stack name:** Choose a stack name for your AWS CloudFormation stack. For example, you can call it **eks-node-group-instance-role**.
6. (Optional) On the **Options** page, you can choose to tag your stack resources. Choose **Next**.
7. On the **Review** page, check the box in the **Capabilities** section and choose **Create stack**.
8. When your stack is created, select it in the console and choose **Outputs**.
9. Record the **NodeInstanceRole** value for the IAM role that was created. You need this when you create your node group.

Adding Nodes to Cluster

Configure node group

A node group is a group of EC2 instances that supply compute capacity to your Amazon EKS cluster. You can add multiple node groups to your cluster. [Info](#)

Group configuration
These properties cannot be changed after the node group is created.

Name
Assign a unique name for this node group

Node IAM Role Name [Info](#) **Select IAM, which is created for nodes**
Select the IAM Role that will be used by the nodes.

Subnets [Info](#) **Select subnets**
Specify the subnets in your VPC where your nodes will run

Allow remote access to nodes [Info](#)
Without remote access enabled you will not be able to directly connect to nodes after they are created.

SSH key pair [Info](#) **Select pem key**
Select an SSH key pair to allow secure remote access to your nodes.

click on next

EKS > Clusters > myeks-cluster > Add node group

Step 1
Configure node group

Step 2
Set compute configuration
Node compute configuration
These properties cannot be changed after the node group is created.

AMI type [Info](#) **Select worker node AMI**
Select the EKS-optimized Amazon Machine Image for nodes

Instance type [Info](#) **Select instance type**
Select the EC2 instance type for nodes

Disk size
Select the size of the attached EBS volume for each node
 GiB **volume / block for worker instance**



Adding Nodes to Cluster

Resource Groups ▾ 🔍

EKS > Clusters > myeks-cluster > Add node group

Step 1
Configure node group

Step 2
Set compute configuration

Step 3
Set scaling configuration

Step 4
Review and create

Set scaling configuration

Group size

Minimum size
Set the minimum number of nodes that the group can scale in to
 Nodes

Maximum size
Set the maximum number of nodes that the group can scale out to
 Nodes

Desired size
Set the desired number of nodes that the group should launch with initially
 Nodes

give min, max and desired size for auto scaling the worker nodes

Cancel Previous **Next**



Adding Nodes to Cluster

ie.aws.amazon.com/eks/home?region=ap-northeast-1#/clusters/myeks-cluster/add-node-group

Resource Groups ▾ ★

Rajesh DevOps ▾ Tokyo

EKS > Clusters > myeks-cluster > Add node group

Step 1
Configure node group

Step 2
Set compute configuration

Step 3
Set scaling configuration

Step 4
Review and create

Review and create

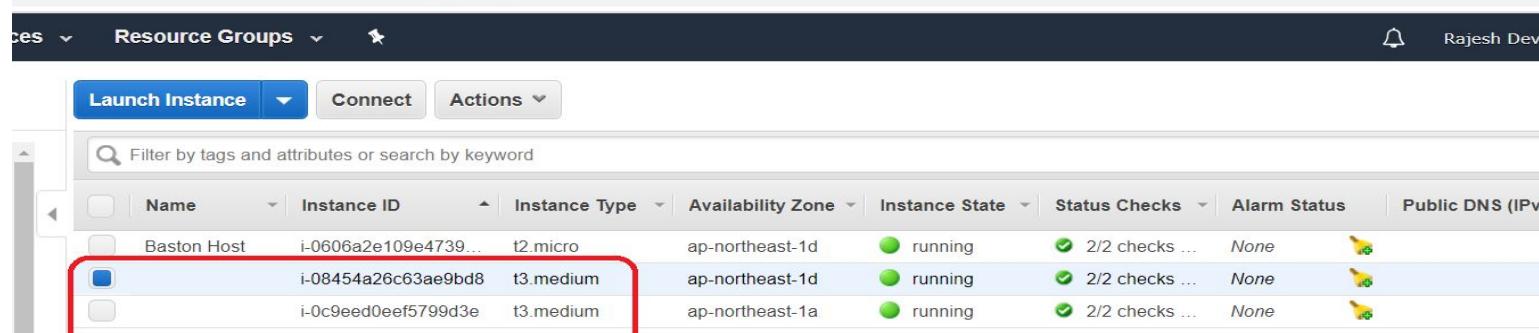
Step 1: Configure node group Edit

Group configuration

nodes group created and added worker nodes

Name	myeks-nodes	Allow remote access to nodes
Node IAM Role Name	eks-node-group-instance-role-NodeInstanceRole-QACZE2CDTE0T	Enabled
Subnets	subnet-04b9d26c9720fb7d0 subnet-084bda23a83538960	SSH key pair tokio
		Allow remote access from All

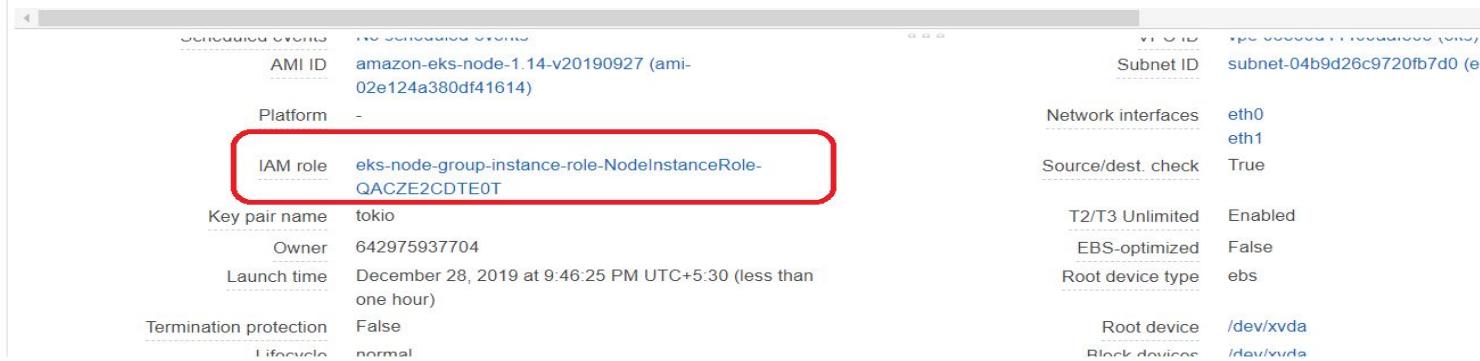
For Cross Checking Created Worker Nodes



A screenshot of the AWS Lambda console. At the top, there are navigation tabs for 'Services' (with a dropdown arrow), 'Resource Groups' (with a dropdown arrow), and a search icon. On the right, there is a notification bell icon and the name 'Rajesh Dev'. Below the header is a toolbar with three buttons: 'Launch Instance' (blue), 'Connect' (grey), and 'Actions' (grey with a dropdown arrow). A search bar with the placeholder 'Filter by tags and attributes or search by keyword' is positioned above a table. The table has columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS (IPv4). There are four rows of data:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
Bastion Host	i-0606a2e109e4739...	t2.micro	ap-northeast-1d	running	2/2 checks ...	None	
<input checked="" type="checkbox"/> Bastion Host	i-08454a26c63ae9bd8	t3.medium	ap-northeast-1d	running	2/2 checks ...	None	
<input type="checkbox"/> Bastion Host	i-0c9eed0eef5799d3e	t3.medium	ap-northeast-1a	running	2/2 checks ...	None	

worker nodes created



A screenshot of the AWS Lambda console showing the configuration details for a Lambda function. The configuration page has sections for 'Scheduled events', 'IAM role', 'Platform', 'Key pair name', 'Owner', 'Launch time', 'Termination protection', and 'Lifecycle'. The 'IAM role' section is highlighted with a red box. The IAM role listed is 'eks-node-group-instance-role-NodeInstanceRole-QACZE2CDTE0T'. Other visible details include the AMI ID 'amazon-eks-node-1.14-v20190927 (ami-02e124a380df41614)', Platform 'AWS Lambda', Network interfaces 'eth0 eth1', Source/dest. check 'True', T2/T3 Unlimited 'Enabled', EBS-optimized 'False', Root device type 'ebs', Root device '/dev/xvda', and Block devices '/dev/xvda'.

Scheduled events	AMI ID	Subnet ID	
	amazon-eks-node-1.14-v20190927 (ami-02e124a380df41614)	subnet-04b9d26c9720fb7d0 (eu-central-1)	
Platform	Network interfaces	eth0 eth1	
IAM role	eks-node-group-instance-role-NodeInstanceRole-QACZE2CDTE0T	Source/dest. check	True
Key pair name	tokio	T2/T3 Unlimited	Enabled
Owner	642975937704	EBS-optimized	False
Launch time	December 28, 2019 at 9:46:25 PM UTC+5:30 (less than one hour)	Root device type	ebs
Termination protection	False	Root device	/dev/xvda
Lifecycle	normal	Block devices	/dev/xvda

Configuring AWS CLI on Client

Check for python if installed or not

```
[root@ip-192-168-1-212 /]# python --version  
Python 2.7.16  
[root@ip-192-168-1-212 /]# █
```

Install pip tool, which is from python

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py  
python get-pip.py
```

For confirmation → #pip

Configure awscli by using below command / snip

```
#pip install awscli
```

```
#aws configure
```

Access key id :

Access Secret key :

Region :

For Cross Checking, execute below command.

#aws s3 ls

Display if any s3 buckets available.

Install Kubernetes client in client (baston - host)

Enabling Repo

```
cat <<EOF > /etc/yum.repos.d/kubernetes.repo
[kubernetes]
name=Kubernetes
baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-x86_64
enabled=1
gpgcheck=1
repo_gpgcheck=1
gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg
https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
EOF
```

Copy & paste above snip in client for enabling repo

Install Kubernetes client in client (baston - host)

Installing Kubectl

```
yum-config-manager --enable kubernetes  
yum install kubectl  
curl -o kubectl  
https://amazon-eks.s3-us-west-2.amazonaws.com/1.14.6/2019-08-22/bin/linux/amd64/kubectl  
chmod +x ./kubectl  
mkdir -p $HOME/bin && cp ./kubectl $HOME/bin/kubectl && export PATH=$HOME/bin:$PATH  
kubectl version --short --client
```

Install AWS IAM Authenticator

```
curl -o aws-iam-authenticator  
https://amazon-eks.s3-us-west-2.amazonaws.com/1.14.6/2019-08-22/bin/linux/amd64/aws-iam-authenticator  
chmod +x ./kubectl  
chmod +x ./aws-iam-authenticator  
chmod 777 /root/bin/aws-iam-authenticator
```

To check Where AWS Credentials are stored and which keys are stored

→ #cat /root/.aws/credentials

To configure EKS cluster in node,

→ #aws eks update-kubeconfig --name <your eks name>

```
[root@ip-192-168-1-218 ~]# aws eks update-kubeconfig --name myeks-cluster
Added new context arn:aws:eks:ap-northeast-1:642975937704:cluster/myeks-cluster to /root/.kube/config
Warning: aws-iam-authenticator is not installed properly or is not in your path.
Refer to the AWS Documentation to download it at https://docs.aws.amazon.com/eks/latest/userguide/configure-kubectl.html
[root@ip-192-168-1-218 ~]#
```

EKS Cluster details & configuration is stored in blow dir

→ cat /root/.kube/config

To list all worker nodes available / attached to cluster

→ #kubectl get nodes

```
[root@ip-192-168-1-218 /]#  
[root@ip-192-168-1-218 /]# kubectl get nodes  
NAME                      STATUS    ROLES      AGE     VERSION  
ip-192-168-1-113.ap-northeast-1.compute.internal   Ready     <none>   53m    v1.14.7-eks-1861c5  
ip-192-168-1-241.ap-northeast-1.compute.internal   Ready     <none>   54m    v1.14.7-eks-1861c5  
[root@ip-192-168-1-218 /]#
```

to list all pods available in cluster

→ #kubectl get pods → for listing user defined

→ #kubectl get pods --all-namespaces → for listing all

```
[root@ip-192-168-1-218 /]#  
[root@ip-192-168-1-218 /]# kubectl get pods --all-namespaces  
NAMESPACE     NAME           READY   STATUS    RESTARTS   AGE  
kube-system   aws-node-8gkdv   1/1     Running   0          57m  
kube-system   aws-node-dc8tq   1/1     Running   0          57m  
kube-system   coredns-58986cd576-2775q   1/1     Running   0          65m  
kube-system   coredns-58986cd576-cvlws   1/1     Running   0          65m  
kube-system   kube-proxy-5492b   1/1     Running   0          57m  
kube-system   kube-proxy-m4fnq   1/1     Running   0          57m  
[root@ip-192-168-1-218 /]#
```

Userful Commands

1. Kubectl get nodes
2. Kubectl get pods
3. Kubectl describe deployment <deployment file name>
4. Kubectl describe pods <pod name>
5. Kubectl apply -f <deployment file>
6. Kubectl apply -f <service file name>
7. Kubectl get deployments
8. Kubectl get svc
9. Kubectl describe pods <pod name>
10. Kubectl delete deployment <deployment name> (**if you delete deployment, pods will be deleted**)
11. Kubectl delete svc <service name>
12. Kubectl edit deployment <deployment name>
13. Kubectl get nodes --watch