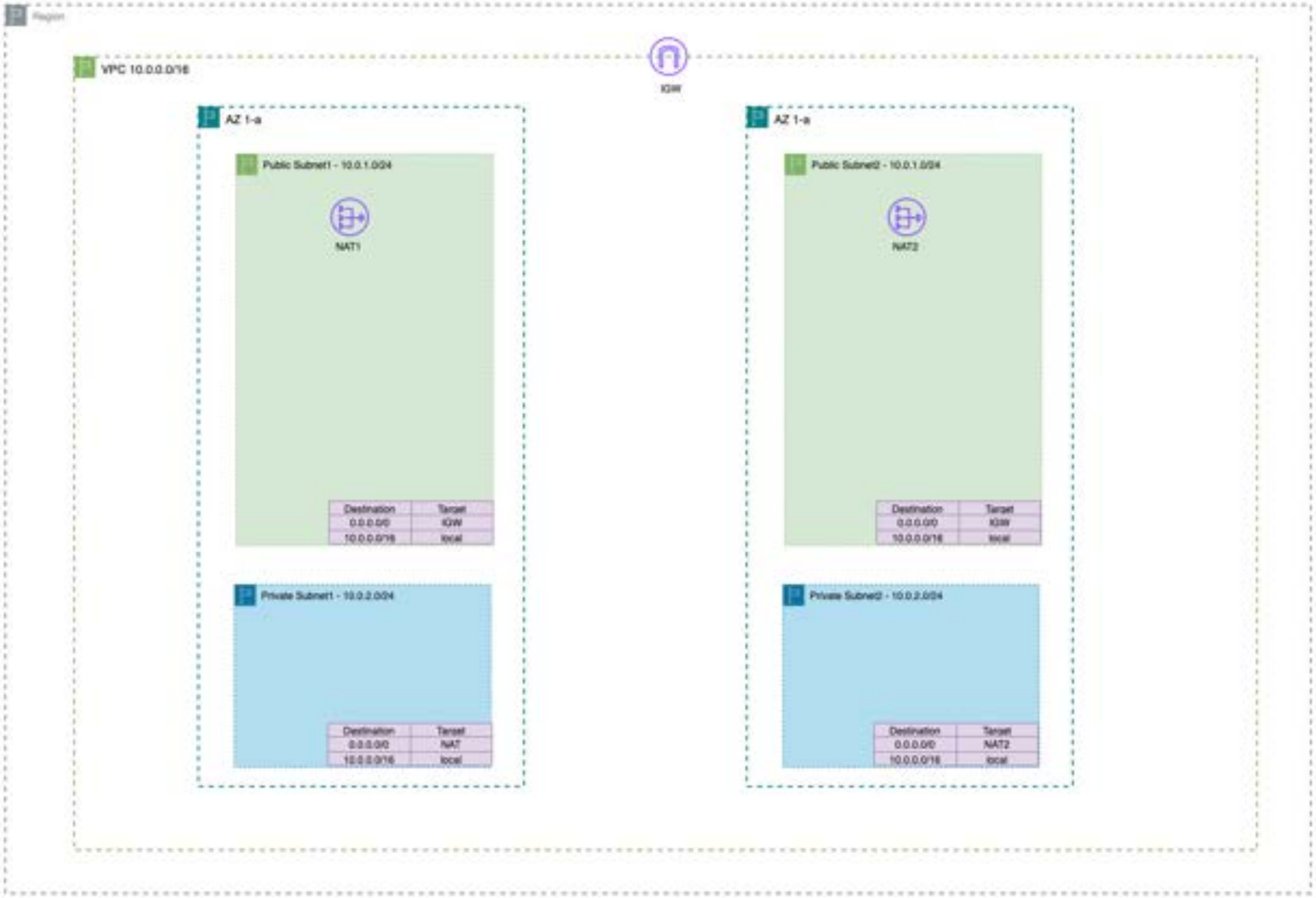
Project4 - HAWA

Jason Zhang

Demo Video: https://broadcast.amazon.com/videos/1204983

VPC
Configuration


Region
VPC 10.0.0.0/16
IGW
AZ 1-a
Public Subnet1 - 10.0.1.0/24
NAT1
Private Subnet1 - 10.0.2.0/24
AZ 1-a
Public Subnet2 - 10.0.1.0/24
NAT2
Private Subnet2 - 10.0.2.0/24

Destination	Target
0.0.0.0/0	IGW
10.0.0.0/16	local

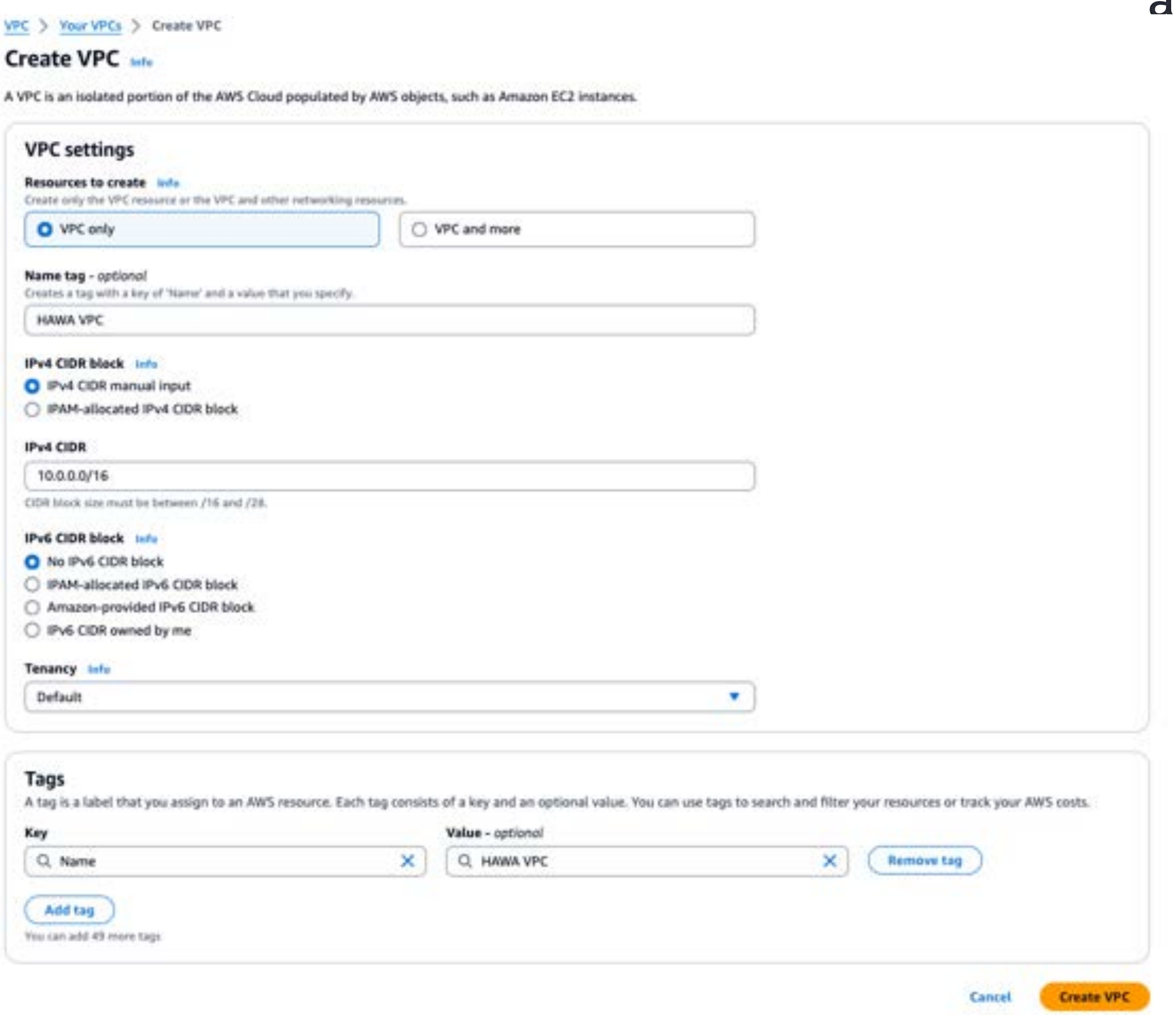
Destination	Target
0.0.0.0/0	NAT
10.0.0.0/16	local

Destination	Target
0.0.0.0/0	IGW
10.0.0.0/16	local

Destination	Target
0.0.0.0/0	NAT2
10.0.0.0/16	local

Step 1 - vpc

```
HAWAVPC:
  Type: AWS::EC2::VPC
  Properties:
    CidrBlock: 10.0.0.0/16
    EnableDnsSupport: true
    EnableDnsHostnames: true
    Tags:
      - Key: Name
        Value: HAWA VPC
```



Create VPC

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create
Create only the VPC resource or the VPC and other networking resources.
● VPC only ○ VPC and more

Name tag - *optional*
Creates a tag with a key of 'Name' and a value that you specify.
HAWA VPC

IPv4 CIDR block
● IPv4 CIDR manual input
○ IPAM-allocated IPv4 CIDR block

IPv4 CIDR
10.0.0.0/16
CIDR block size must be between /16 and /28.

IPv6 CIDR block
● No IPv6 CIDR block
○ IPAM-allocated IPv6 CIDR block
○ Amazon-provided IPv6 CIDR block
○ IPv6 CIDR owned by me

Tenancy
Default

Tags

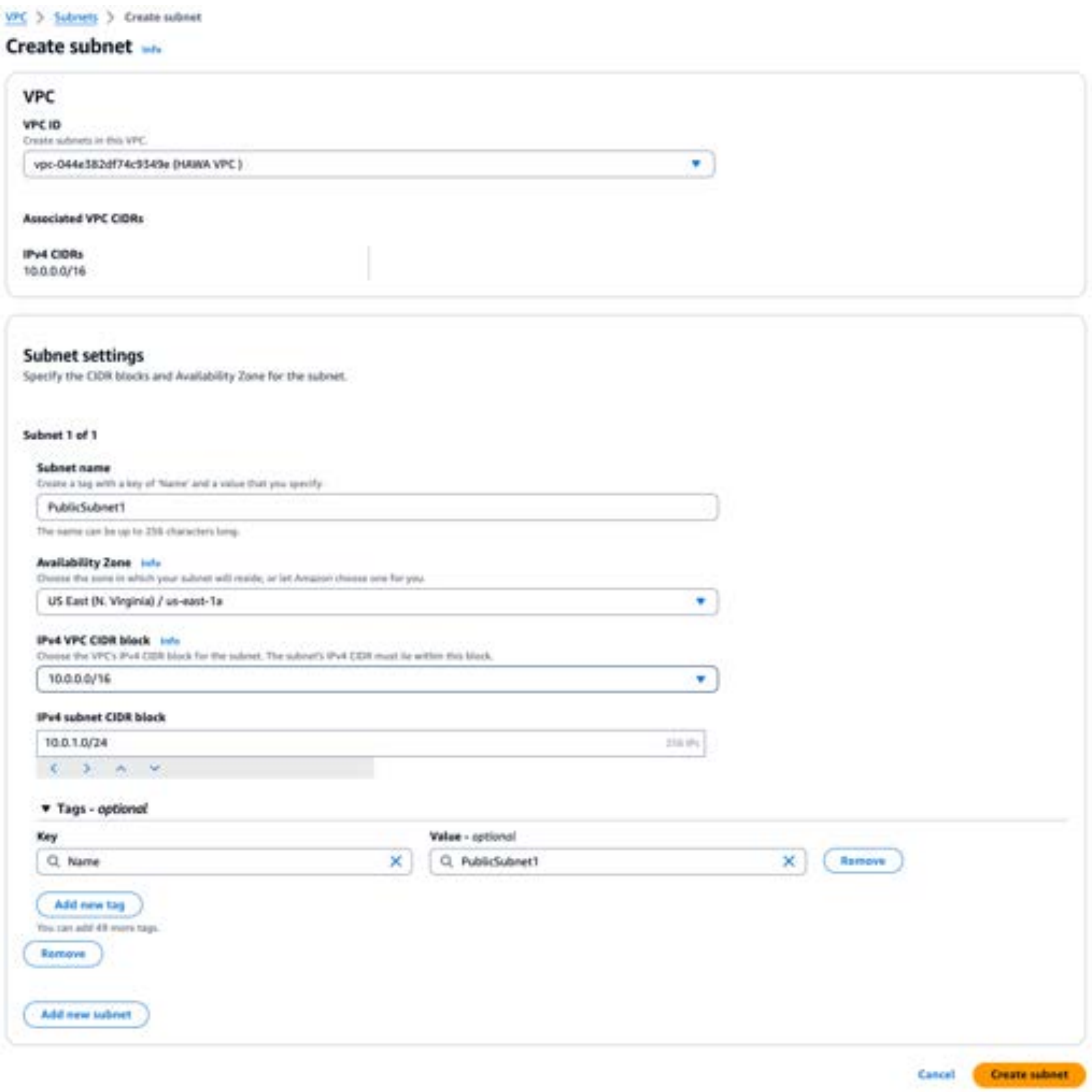
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*
Name	HAWA VPC

You can add 49 more tags

Step 2.1 - vpc

```
PublicSubnet1:
  Type: AWS::EC2::Subnet
  Properties:
    VpcId: !Ref HAWAVPC
    CidrBlock: 10.0.1.0/24
    AvailabilityZone: us-east-1a
    MapPublicIpOnLaunch : true
    Tags:
      - Key: Name
        Value: PublicSubnet1
```



VPC > Subnets > Create subnet

Create subnet

VPC

VPC ID
Create subnets in this VPC.
vpc-044e382df74c9349e (HAWA VPC)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
PublicSubnet1
The name can be up to 256 characters long.

Availability Zone
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1a

IPv4 VPC CIDR block
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.1.0/24
256 IPs

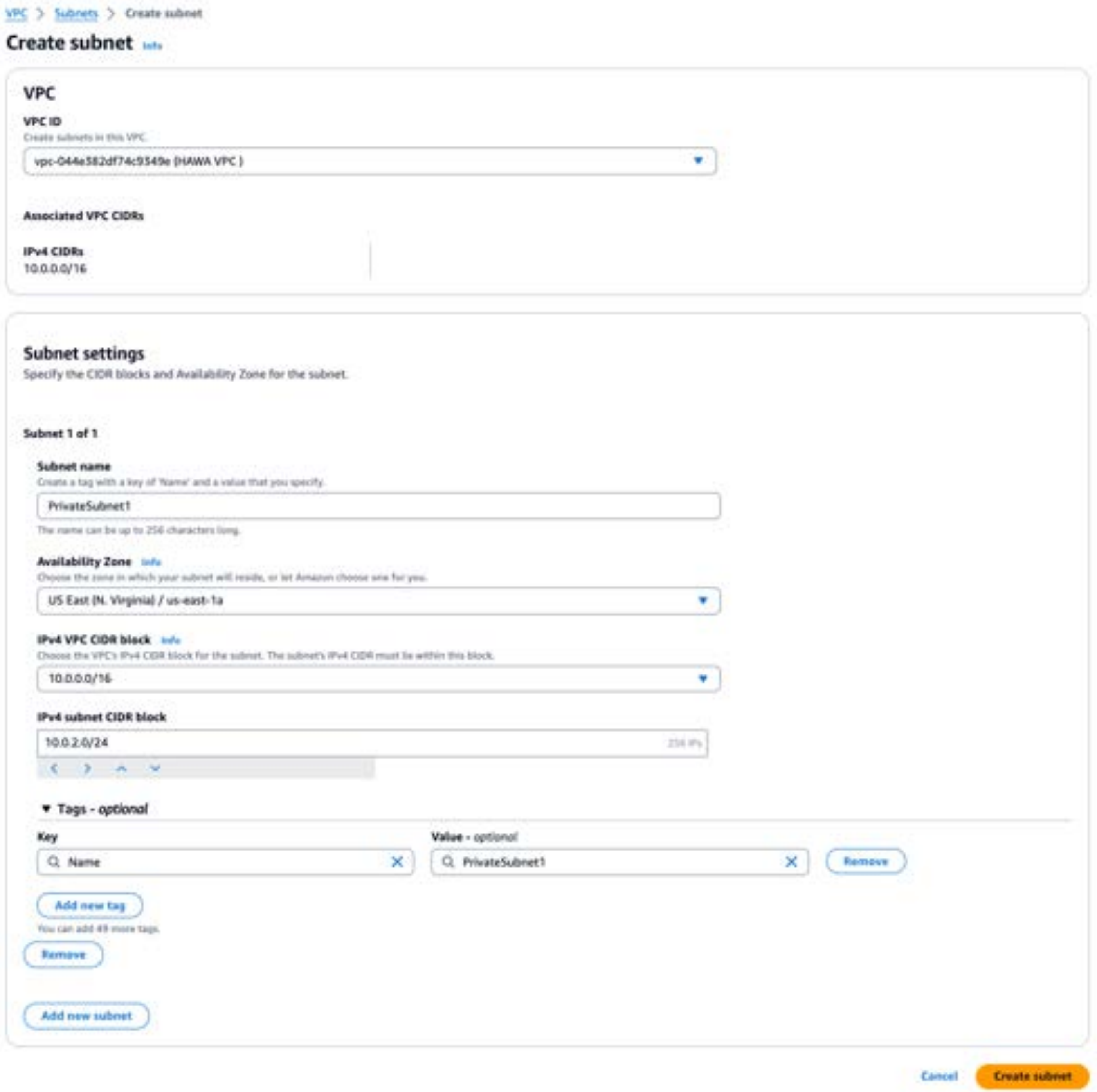
▼ Tags - *optional*

Key	Value - optional
Name	PublicSubnet1

You can add 49 more tags.

Step 2.2 - vpc

```
PrivateSubnet1:
  Type: AWS::EC2::Subnet
  Properties:
    VpcId: !Ref HAWAVPC
    CidrBlock: 10.0.2.0/24
    AvailabilityZone: us-east-1a
    Tags:
      - Key: Name
        Value: PrivateSubnet1
```



VPC > Subnets > Create subnet

Create subnet Info

VPC

VPC ID
Create subnets in this VPC.
vpc-044e382df74c9349e (HAWA VPC)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
PrivateSubnet1
The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1a

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.2.0/24 256 IPs

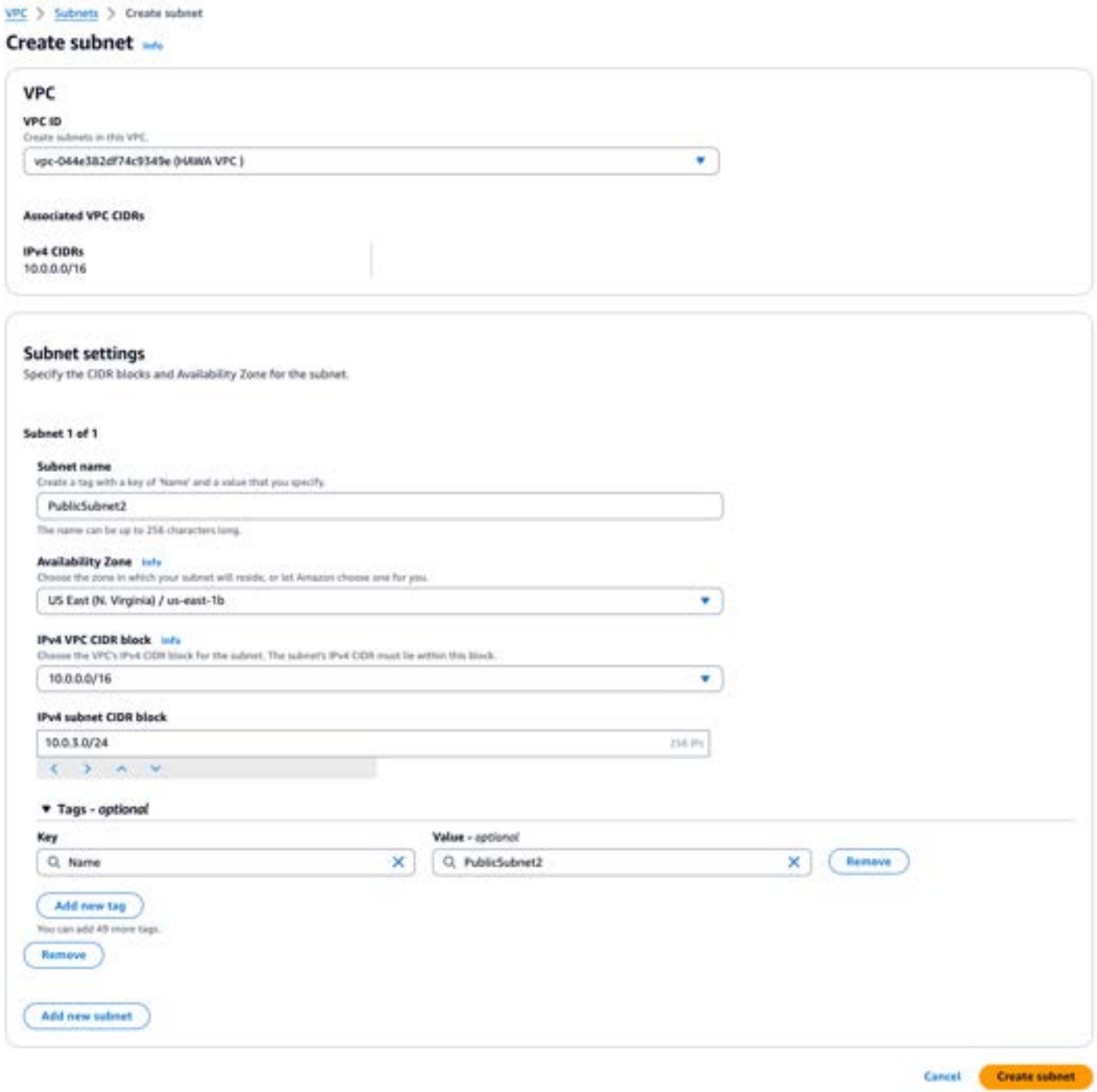
▼ Tags - optional

Key: Name
Value - optional: PrivateSubnet1

You can add 49 more tags.

Step 2.3 - vpc

```yaml
PublicSubnet2:
  Type: AWS::EC2::Subnet
  Properties:
    VpcId: !Ref HAWAVPC
    CidrBlock: 10.0.3.0/24
    AvailabilityZone: us-east-1b
    MapPublicIpOnLaunch : true
    Tags:
      - Key: Name
        Value: PublicSubnet2
```



VPC > Subnets > Create subnet

Create subnet

VPC

VPC ID
Create subnets in this VPC.
vpc-044e382df74c9349e (HAWA VPC)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
PublicSubnet2
The name can be up to 256 characters long.

Availability Zone
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1b

IPv4 VPC CIDR block
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.3.0/24 256 IPs

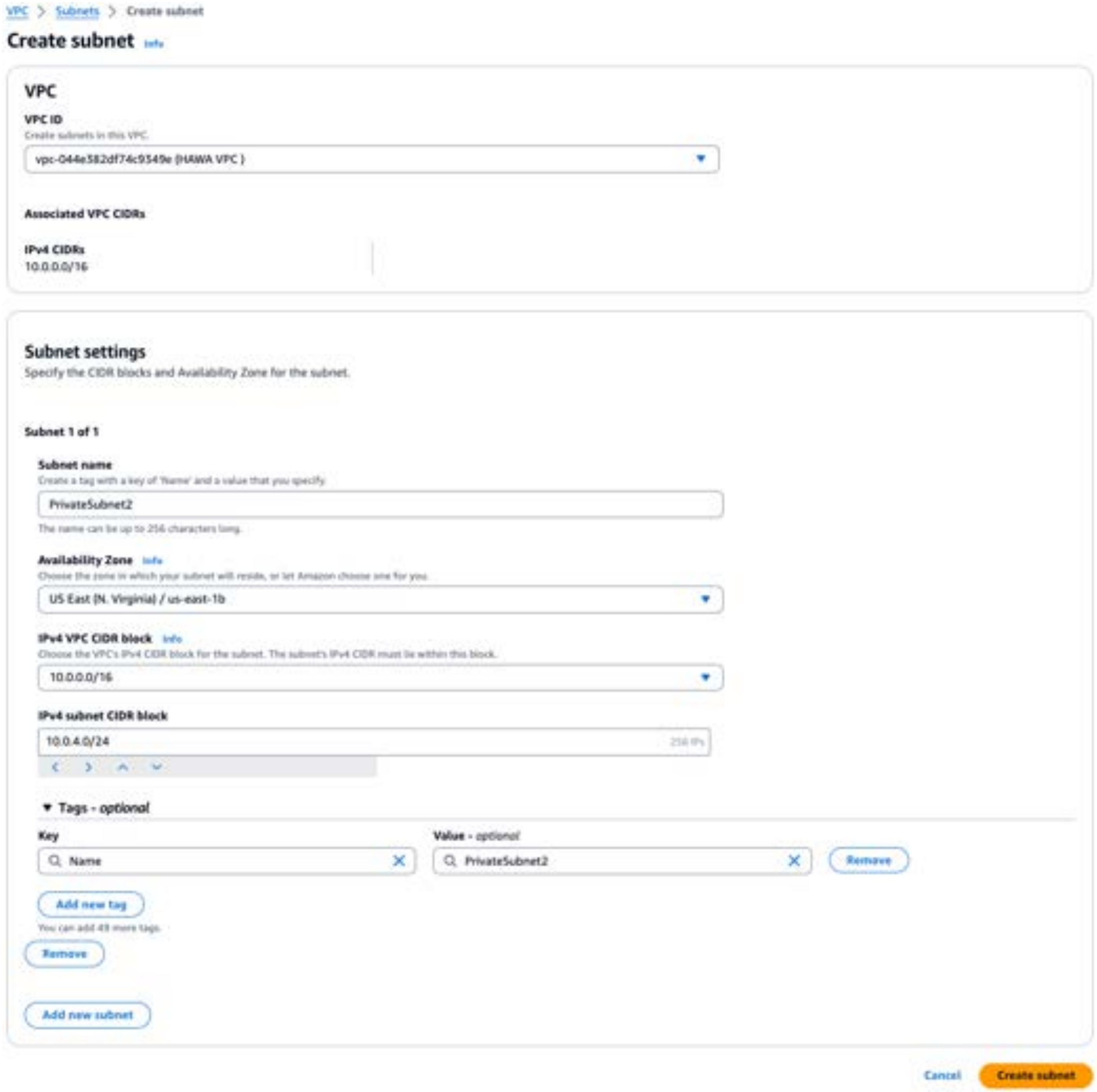
▼ **Tags - optional**

Key	Value - optional
Name	PublicSubnet2

You can add 49 more tags.

Step 2.4 - vpc

```
PrivateSubnet2:
  Type: AWS::EC2::Subnet
  Properties:
    VpcId: !Ref HAWAVPC
    CidrBlock: 10.0.4.0/24
    AvailabilityZone: us-east-1b
    Tags:
      - Key: Name
        Value: PrivateSubnet2
```



Create subnet Info

VPC

VPC ID
Create subnets in this VPC.
vpc-044e382df74c9349e (HAWA VPC)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
PrivateSubnet2
The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1b

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

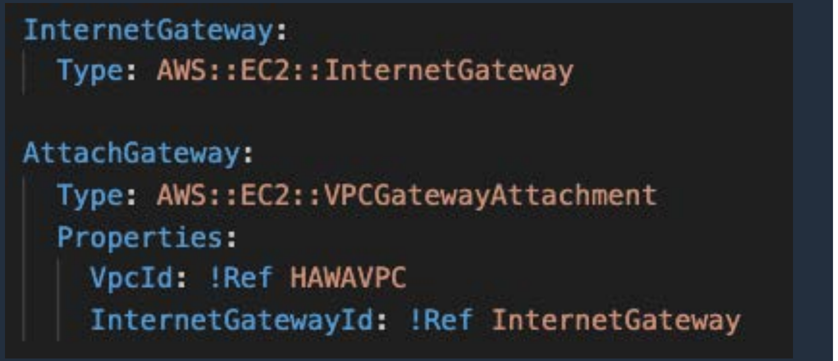
IPv4 subnet CIDR block
10.0.4.0/24 256 IPs

▼ Tags - *optional*

Key	Value - optional
Name	PrivateSubnet2

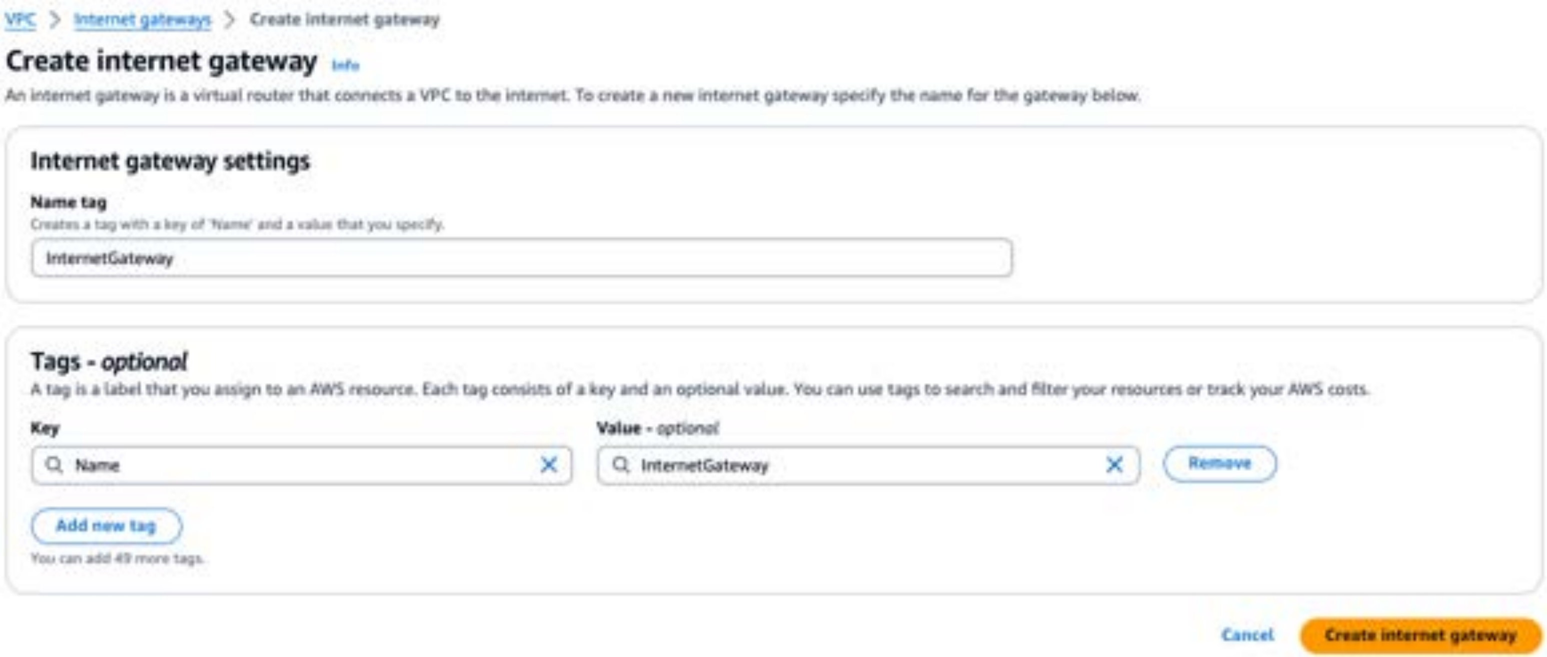
You can add 49 more tags.

Step 3 - vpc



```
InternetGateway:
  Type: AWS::EC2::InternetGateway

AttachGateway:
  Type: AWS::EC2::VPCGatewayAttachment
  Properties:
    VpcId: !Ref HAWAVPC
    InternetGatewayId: !Ref InternetGateway
```



VPC > Internet gateways > Create internet gateway

Create internet gateway

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

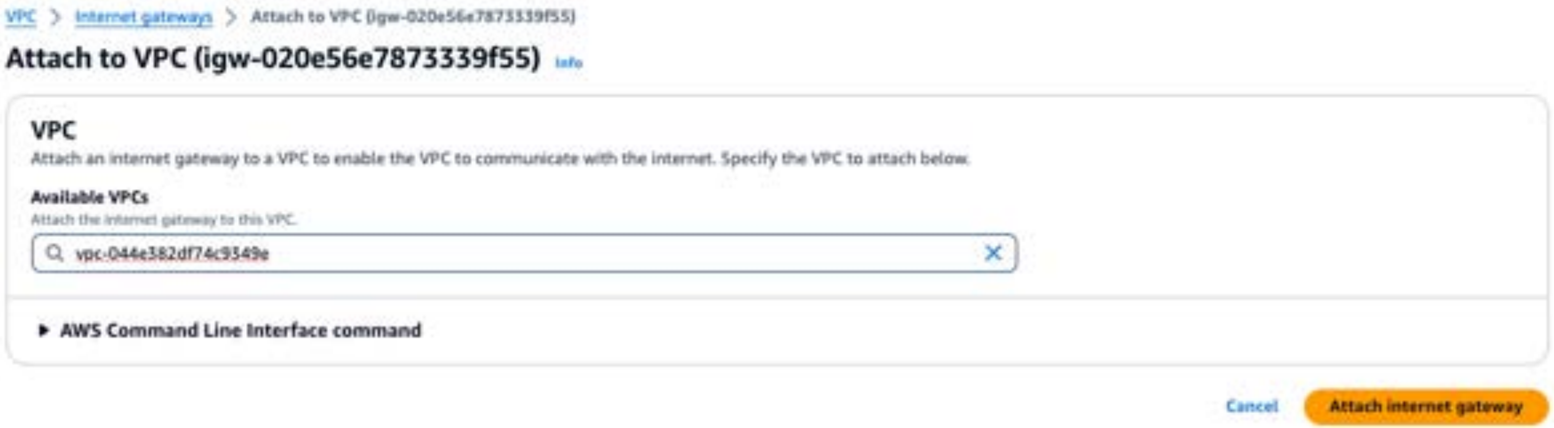
Name tag
Creates a tag with a key of 'Name' and a value that you specify.
InternetGateway

Tags - *optional*

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
Name	InternetGateway

You can add 49 more tags.



VPC > Internet gateways > Attach to VPC (igw-020e56e7873339f55)

Attach to VPC (igw-020e56e7873339f55)

VPC

Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

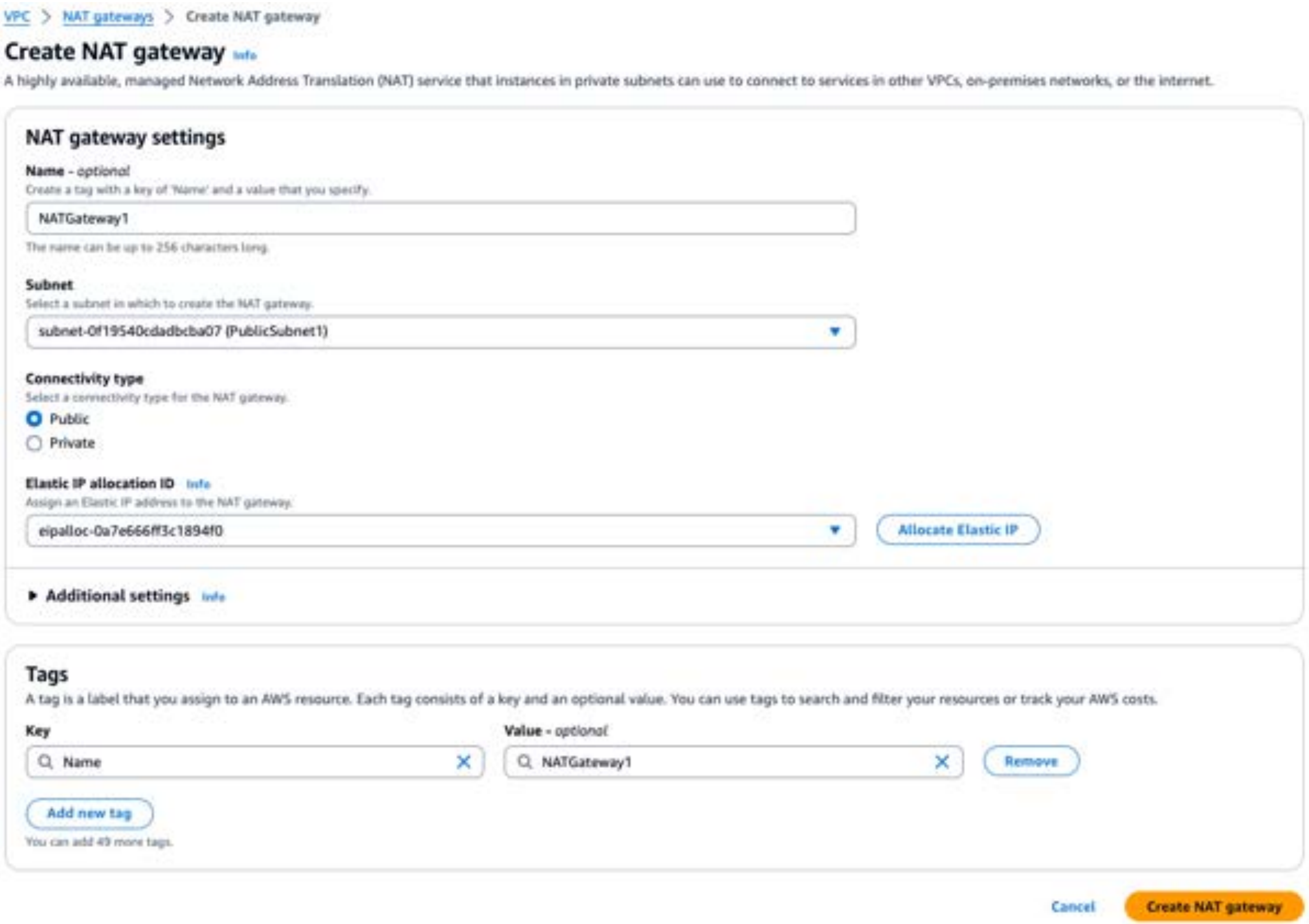
Available VPCs
Attach the internet gateway to this VPC.
vpc-044e382df74c9349e

▸ AWS Command Line Interface command

Step 4.1 - vpc

```yaml
NATGateway1:
  Type: AWS::EC2::NatGateway
  Properties:
    AllocationId: !GetAtt EIP1.AllocationId
    SubnetId: !Ref PublicSubnet1
    Tags:
      - Key: Name
        Value: NATGateway1

EIP1:
  Type: AWS::EC2::EIP
  Properties:
    Domain: vpc
```



VPC > NAT gateways > Create NAT gateway

Create NAT gateway

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
NATGateway1
The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.
subnet-0f19540cdadbcba07 (PublicSubnet1)

Connectivity type
Select a connectivity type for the NAT gateway.
● Public
○ Private

Elastic IP allocation ID
Assign an Elastic IP address to the NAT gateway.
eipalloc-0a7e666ff3c1894f0

Additional settings

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

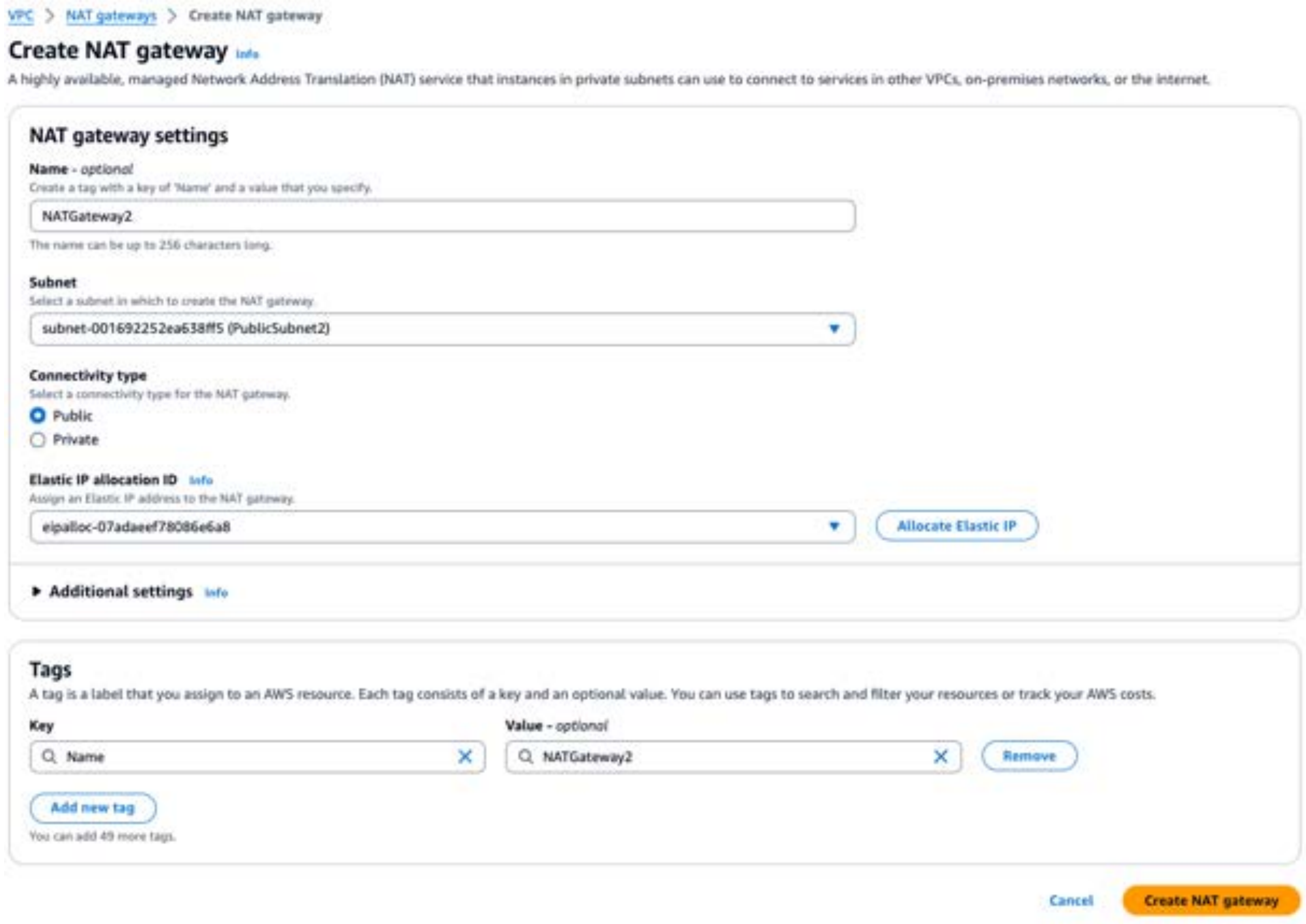
Key: Name
Value - *optional*: NATGateway1

You can add 49 more tags.

Step 4.2 - vpc

```
NATGateway2:
  Type: AWS::EC2::NatGateway
  Properties:
    AllocationId: !GetAtt EIP2.AllocationId
    SubnetId: !Ref PublicSubnet2
    Tags:
      - Key: Name
        Value: NATGateway2

EIP2:
  Type: AWS::EC2::EIP
  Properties:
    Domain: vpc
```



VPC > NAT gateways > Create NAT gateway

Create NAT gateway Info

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
NATGateway2
The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.
subnet-001692252ea638ff5 (PublicSubnet2) ▼

Connectivity type
Select a connectivity type for the NAT gateway.
● Public
○ Private

Elastic IP allocation ID Info
Assign an Elastic IP address to the NAT gateway.
eipalloc-07adaeef78086e6a8 ▼

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*
Name	NATGateway2

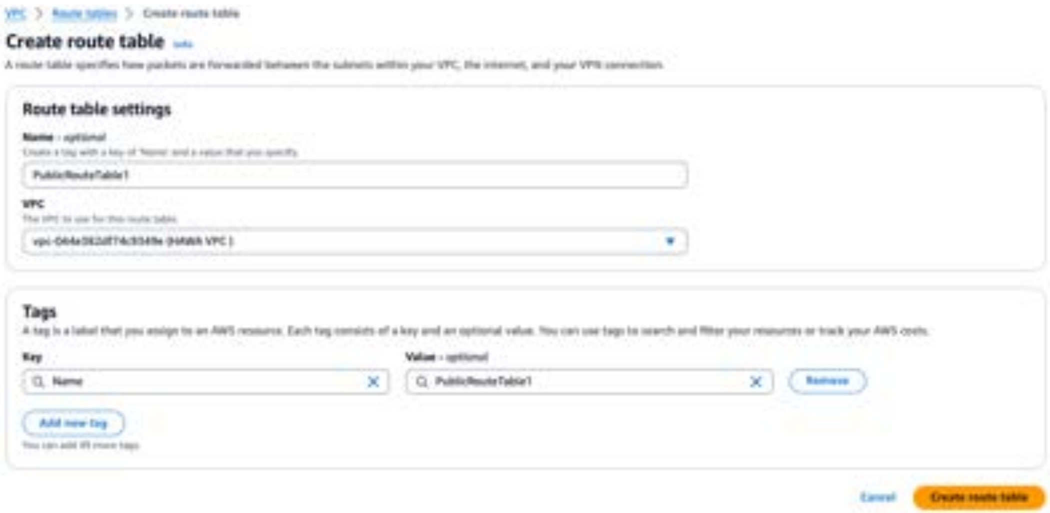
You can add 49 more tags.

Step 5.1 - vpc

```yaml
PublicRouteTable1:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref HAWAVPC
    Tags:
      - Key: Name
        Value: Public Route Table 1

PublicRoute1:
  Type: AWS::EC2::Route
  DependsOn: AttachGateway
  Properties:
    RouteTableId: !Ref PublicRouteTable1
    DestinationCidrBlock: 0.0.0.0/0
    GatewayId: !Ref InternetGateway

PublicSubnet1RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    SubnetId: !Ref PublicSubnet1
    RouteTableId: !Ref PublicRouteTable1
```



VPC > Route tables > Create route table

Create route table

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.
PublicRouteTable1

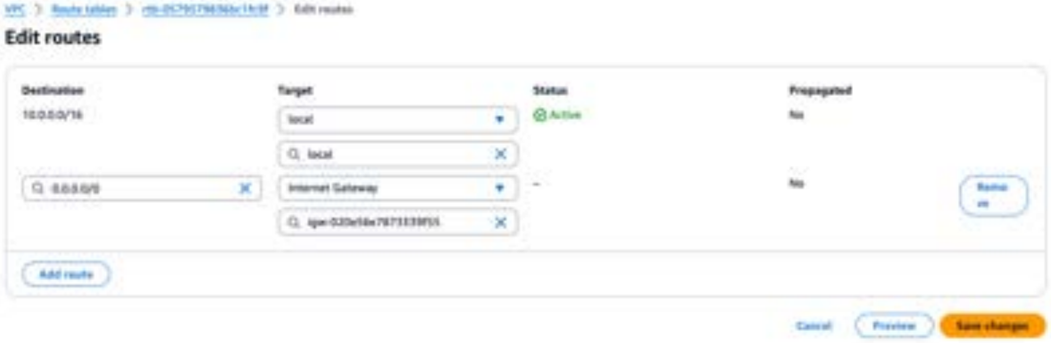
VPC
The VPC to use for this route table.
vpc-[illegible] (HAWA VPC)

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key: Name
Value - optional: PublicRouteTable1

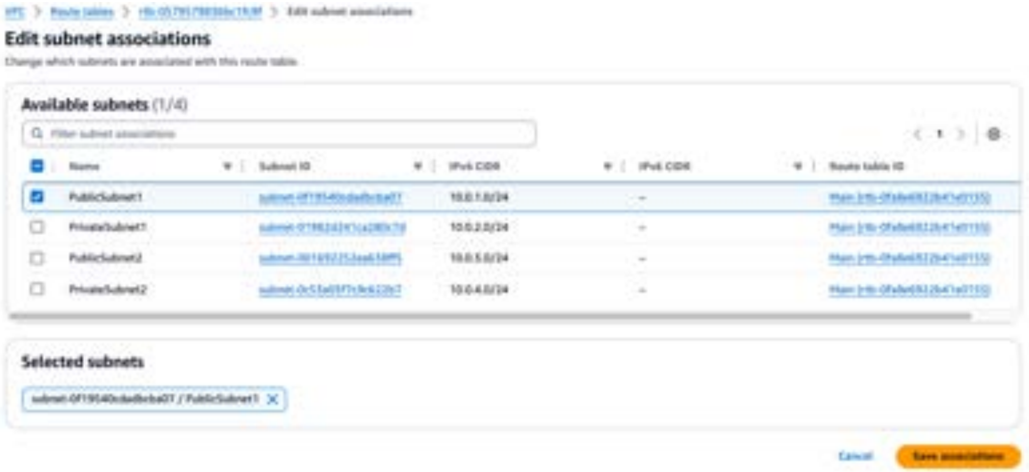
You can add 49 more tags.



VPC > Route tables > rtb-[illegible] > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local local	Active	No
0.0.0.0/0	Internet Gateway igw-[illegible]	-	No



VPC > Route tables > rtb-[illegible] > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/4)

	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
☑	PublicSubnet1	subnet-0f19540cdadbcba07	10.0.1.0/24	–	Main (rtb-[illegible])
□	PrivateSubnet1	subnet-[illegible]	10.0.2.0/24	–	Main (rtb-[illegible])
□	PublicSubnet2	subnet-[illegible]	10.0.5.0/24	–	Main (rtb-[illegible])
□	PrivateSubnet2	subnet-[illegible]	10.0.4.0/24	–	Main (rtb-[illegible])

Selected subnets

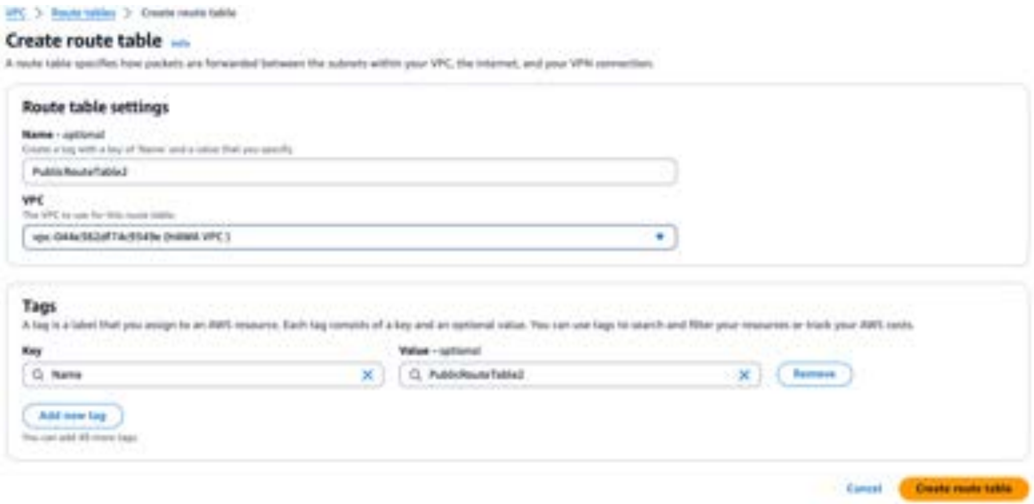
subnet-0f19540cdadbcba07 / PublicSubnet1

Step 5.2 - vpc

```
PublicRouteTable2:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref HAWAVPC
    Tags:
      - Key: Name
        Value: Public Route Table 2

PublicRoute2:
  Type: AWS::EC2::Route
  DependsOn: AttachGateway
  Properties:
    RouteTableId: !Ref PublicRouteTable2
    DestinationCidrBlock: 0.0.0.0/0
    GatewayId: !Ref InternetGateway

PublicSubnet2RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    SubnetId: !Ref PublicSubnet2
    RouteTableId: !Ref PublicRouteTable2
```



VPC > Route tables > Create route table

Create route table

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

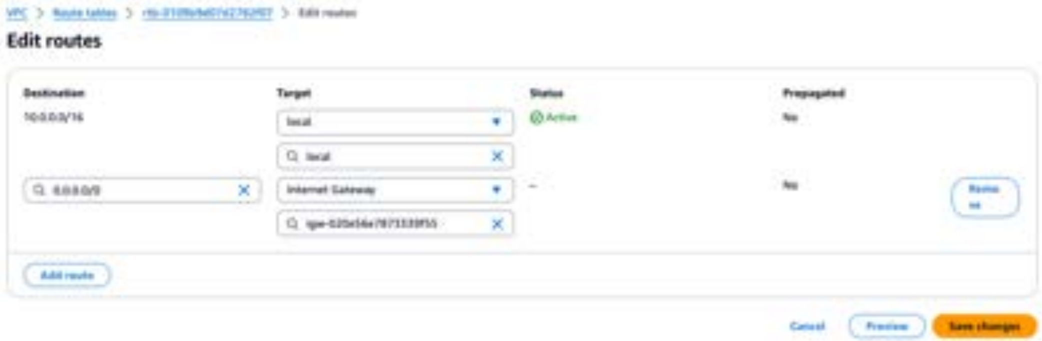
Route table settings

Name - optional: PublicRouteTable2

VPC: vpc-044e582df7 [illegible] (HAWA VPC)

Tags

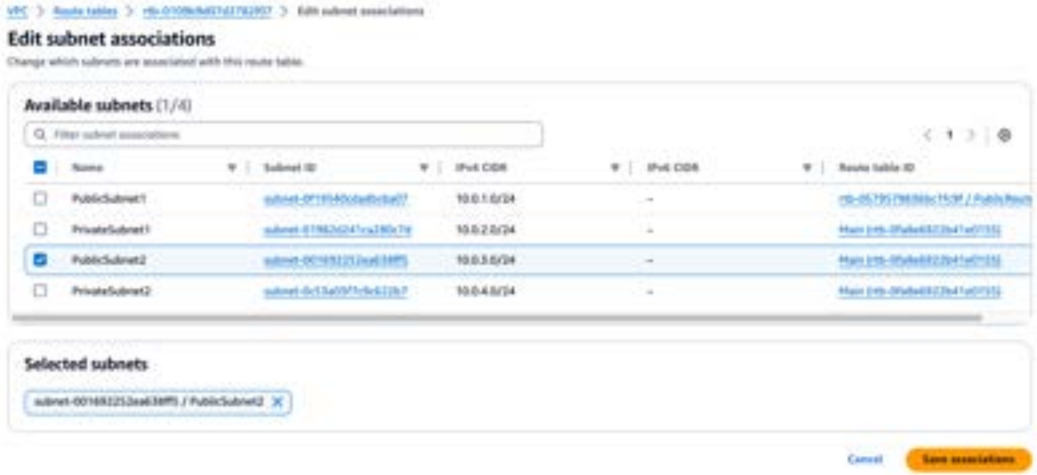
Key	Value - optional
Name	PublicRouteTable2



VPC > Route tables > rtb-0108 [illegible] > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	Internet Gateway igw-020e54e7873339f55	–	No



VPC > Route tables > rtb-0108 [illegible] > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

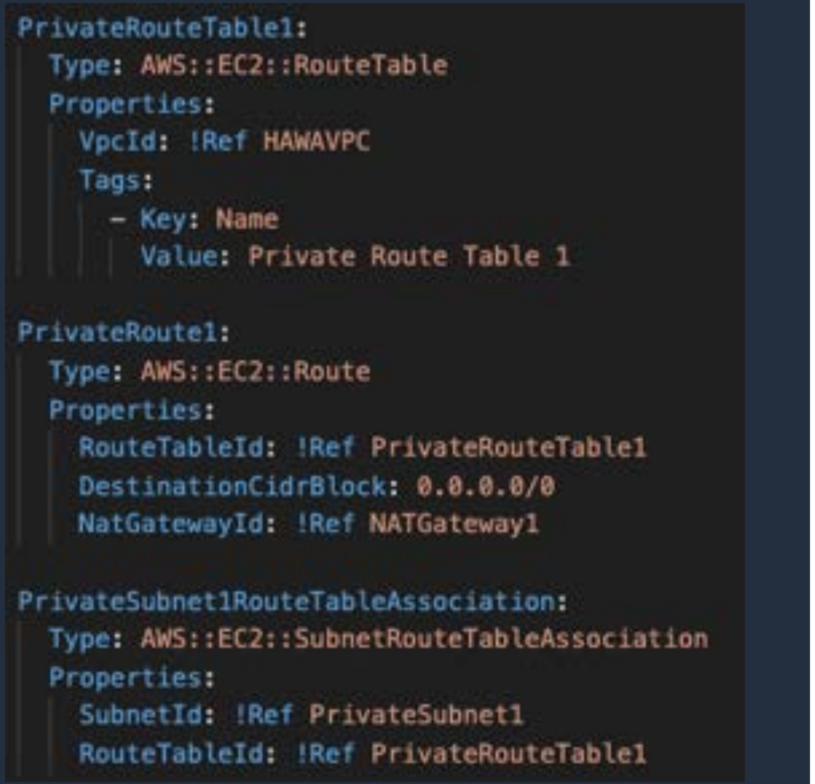
Available subnets (1/4)

	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
□	PublicSubnet1	subnet-0f1 [illegible]	10.0.1.0/24	–	rtb-0579 [illegible] / PublicRout
□	PrivateSubnet1	subnet-0 [illegible]	10.0.2.0/24	–	Main (rtb-0 [illegible])
☑	PublicSubnet2	subnet-0016 [illegible]	10.0.3.0/24	–	Main (rtb-0 [illegible])
□	PrivateSubnet2	subnet-0 [illegible]	10.0.4.0/24	–	Main (rtb-0 [illegible])

Selected subnets

subnet-001692252ea638ff5 / PublicSubnet2

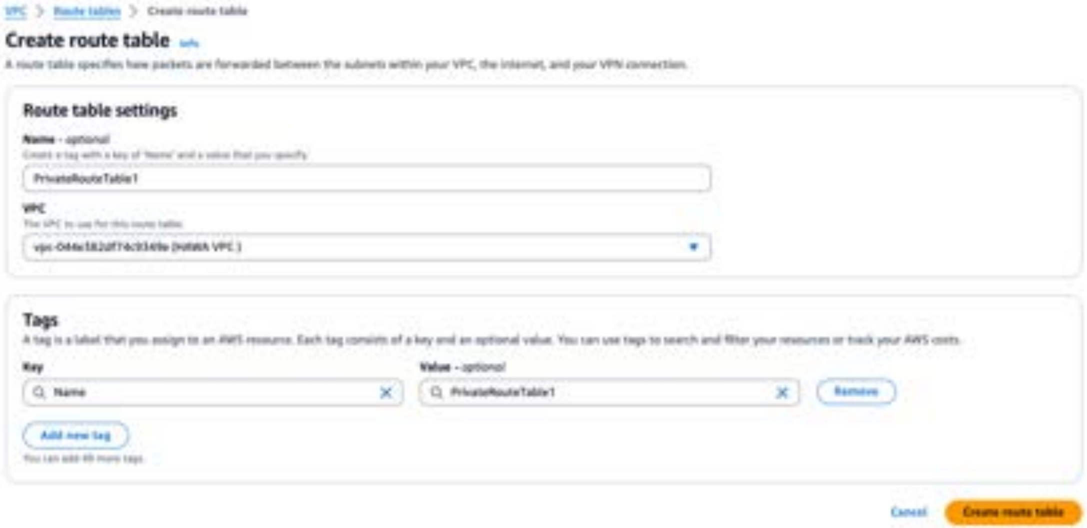
Step 6.1 - vpc



```
PrivateRouteTable1:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref HAWAVPC
    Tags:
      - Key: Name
        Value: Private Route Table 1

PrivateRoute1:
  Type: AWS::EC2::Route
  Properties:
    RouteTableId: !Ref PrivateRouteTable1
    DestinationCidrBlock: 0.0.0.0/0
    NatGatewayId: !Ref NATGateway1

PrivateSubnet1RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    SubnetId: !Ref PrivateSubnet1
    RouteTableId: !Ref PrivateRouteTable1
```



VPC > Route tables > Create route table

Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

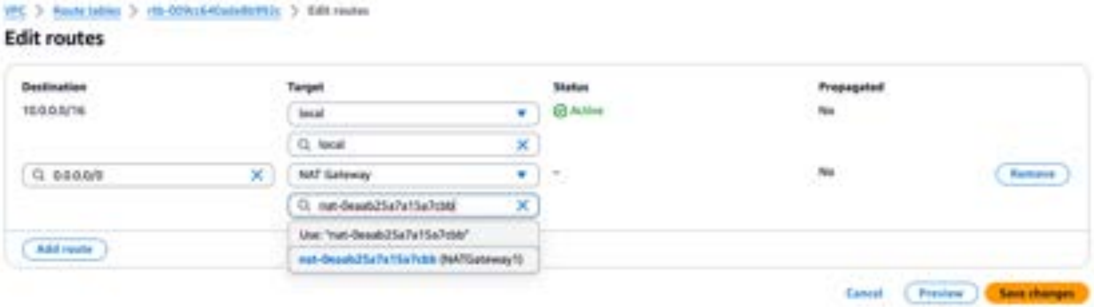
Name - optional
Create a tag with a key of 'Name' and a value that you specify.
PrivateRouteTable1

VPC
The VPC to use for this route table.
vpc-044e382df74c9349e (HAWA VPC)

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
Name	PrivateRouteTable1

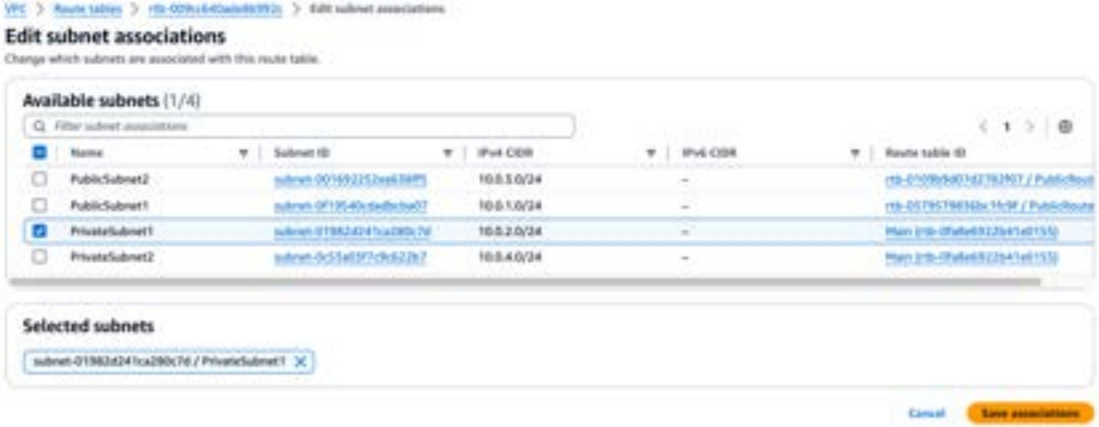


VPC > Route tables > rtb-009c1640ade8b992c > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	NAT Gateway nat-0eaab25a7a15a7cbb	-	No

Use: "nat-0eaab25a7a15a7cbb"
nat-0eaab25a7a15a7cbb (NATGateway1)



VPC > Route tables > rtb-009c1640ade8b992c > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/4)

☑	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
☐	PublicSubnet2	subnet-001692252ea638ff5	10.0.3.0/24	–	rtb-0109b9d01d2782f07 / PublicRout
☐	PublicSubnet1	subnet-0f19540cdadbcba07	10.0.1.0/24	–	rtb-057957f8636bc1fc9f / PublicRoute
☑	PrivateSubnet1	subnet-01982d241ca280c7d	10.0.2.0/24	–	Main (rtb-0fa8a6822b41e0155)
☐	PrivateSubnet2	subnet-0c55a03f7c9c622b7	10.0.4.0/24	–	Main (rtb-0fa8a6822b41e0155)

Selected subnets

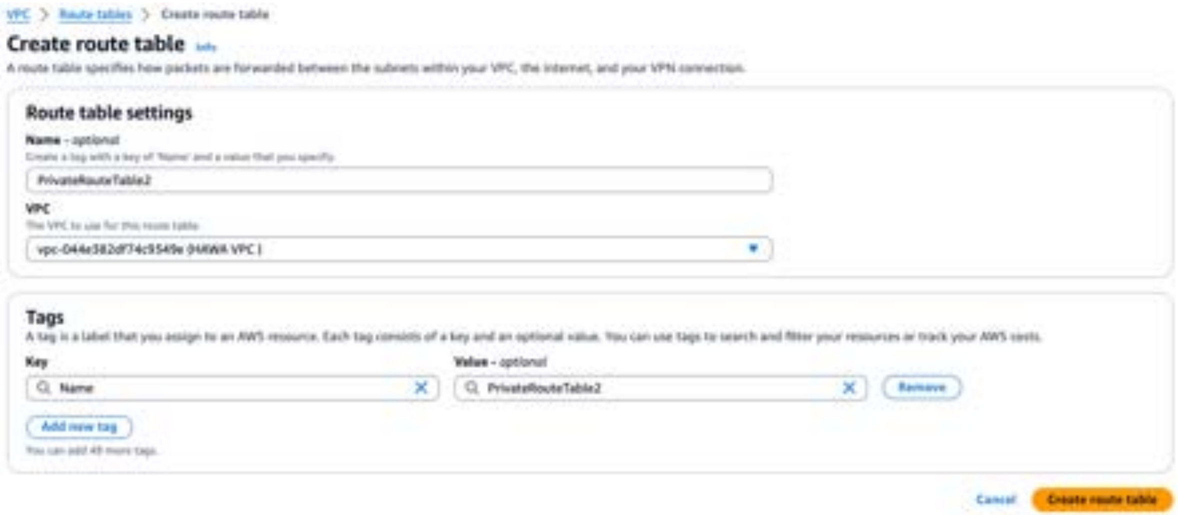
subnet-01982d241ca280c7d / PrivateSubnet1

Step 6.2 - vpc

```
PrivateRouteTable2:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref HAWAVPC
    Tags:
      - Key: Name
        Value: Private Route Table 2

PrivateRoute2:
  Type: AWS::EC2::Route
  Properties:
    RouteTableId: !Ref PrivateRouteTable2
    DestinationCidrBlock: 0.0.0.0/0
    NatGatewayId: !Ref NATGateway2

PrivateSubnet2RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    SubnetId: !Ref PrivateSubnet2
    RouteTableId: !Ref PrivateRouteTable2
```



VPC > Route tables > Create route table

Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

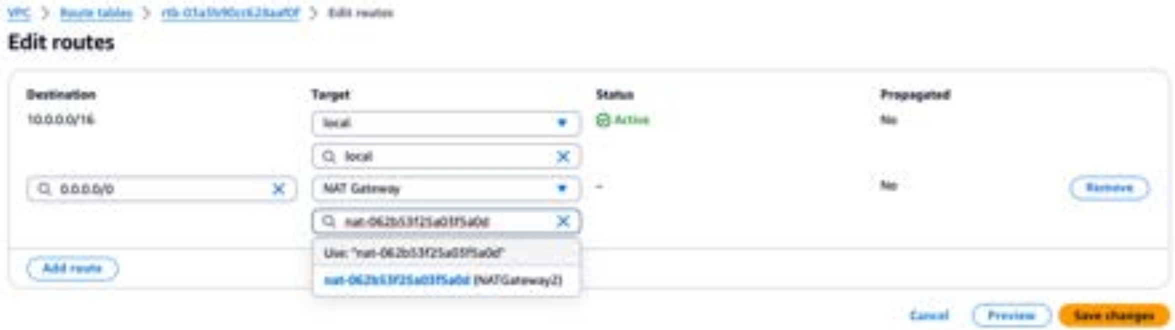
Name - optional
Create a tag with a key of 'Name' and a value that you specify.
PrivateRouteTable2

VPC
The VPC to use for this route table.
vpc-044e382df74c9549e (HAWA VPC)

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key: Name
Value - optional: PrivateRouteTable2

You can add 49 more tags.

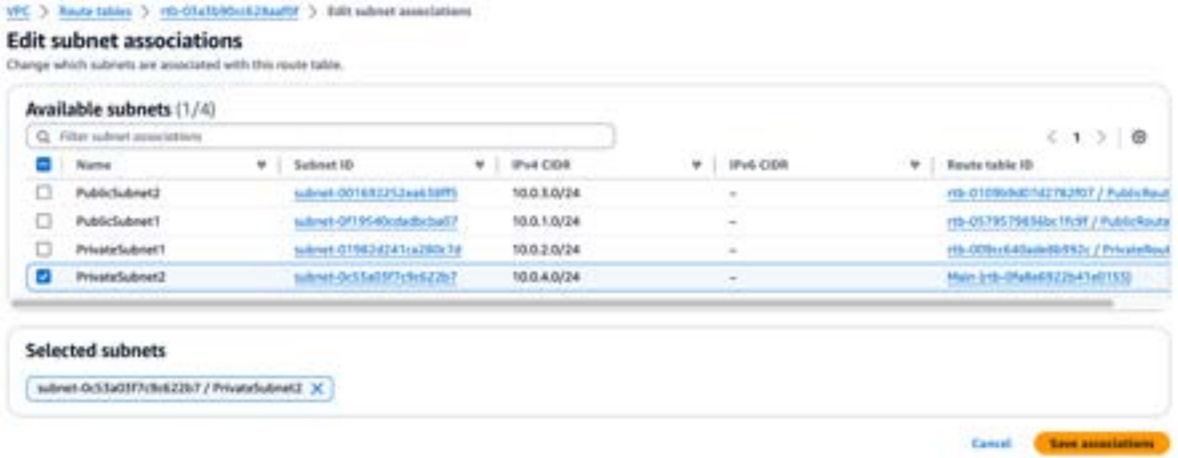


VPC > Route tables > rtb-01a1b90cc628aaf0f > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	NAT Gateway nat-062b53f25a03f5a0d	-	No

Use: "nat-062b53f25a03f5a0d"
nat-062b53f25a03f5a0d (NATGateway2)



VPC > Route tables > rtb-01a1b90cc628aaf0f > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/4)

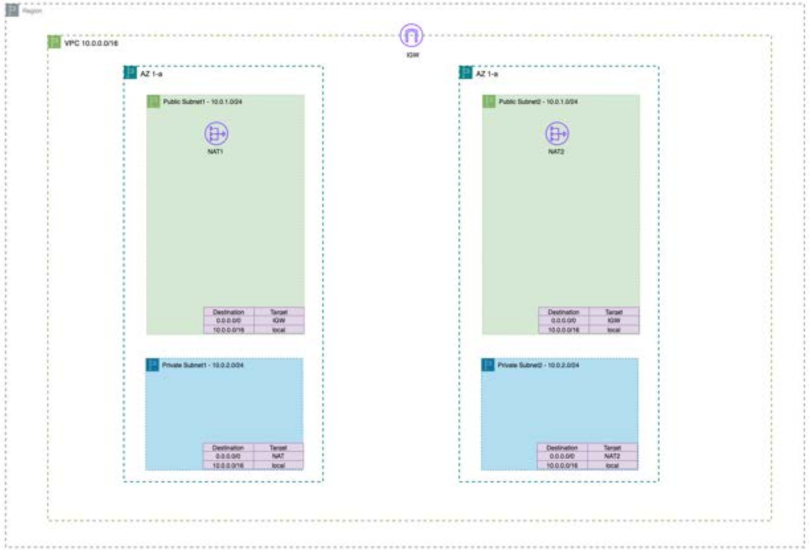
	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
□	PublicSubnet2	subnet-001682252aa638ff5	10.0.3.0/24	-	rtb-0109b9d07d2782f07 / PublicRout
□	PublicSubnet1	subnet-0f19540cdadbcba07	10.0.1.0/24	-	rtb-057957985bc1fc9f / PublicRoute
□	PrivateSubnet1	subnet-01982d241ca280c7d	10.0.2.0/24	-	rtb-009cc640ade8b592c / PrivateRout
☑	PrivateSubnet2	subnet-0c53a03f7c9c622b7	10.0.4.0/24	-	Main (rtb-0fa8a6922b41e0153)

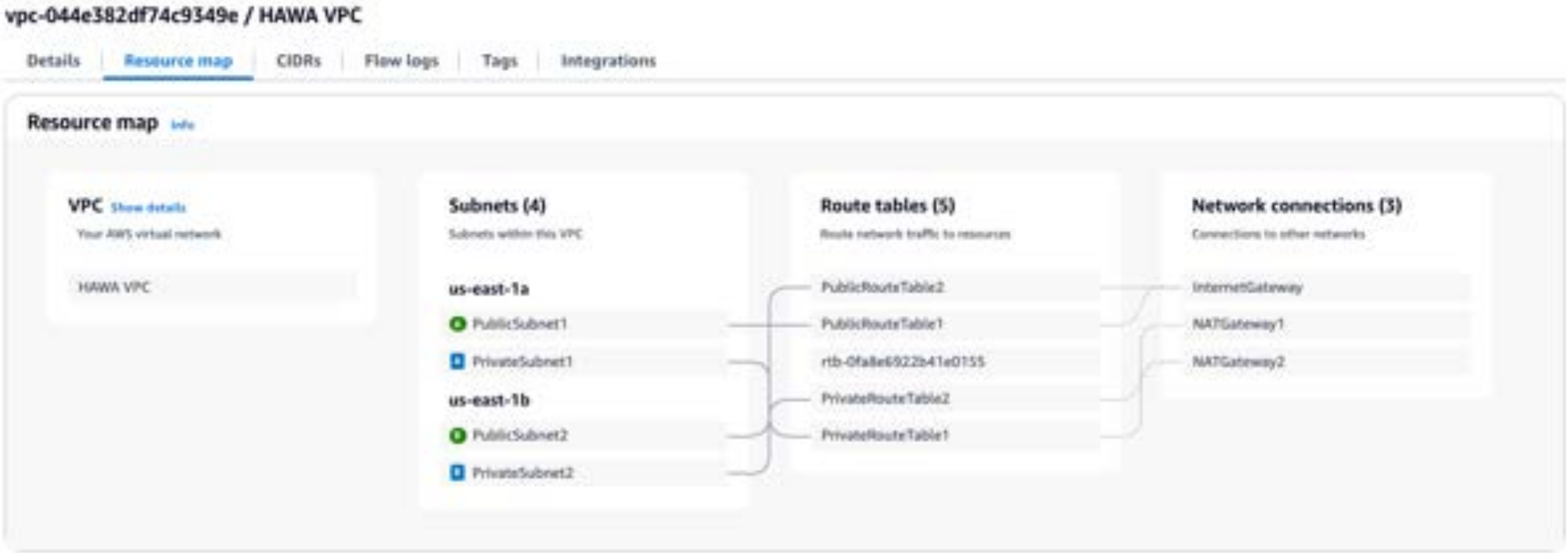
Selected subnets

subnet-0c53a03f7c9c622b7 / PrivateSubnet2

VPC

summary





vpc-044e382df74c9349e / HAWA VPC

Resource map

VPC

Your AWS virtual network

HAWA VPC

Subnets (4)

Subnets within this VPC

us-east-1a

PublicSubnet1

PrivateSubnet1

us-east-1b

PublicSubnet2

PrivateSubnet2

Route tables (5)

Route network traffic to resources

PublicRouteTable2

PublicRouteTable1

rtb-0fa8e6922b41e0155

PrivateRouteTable2

PrivateRouteTable1

Network connections (3)

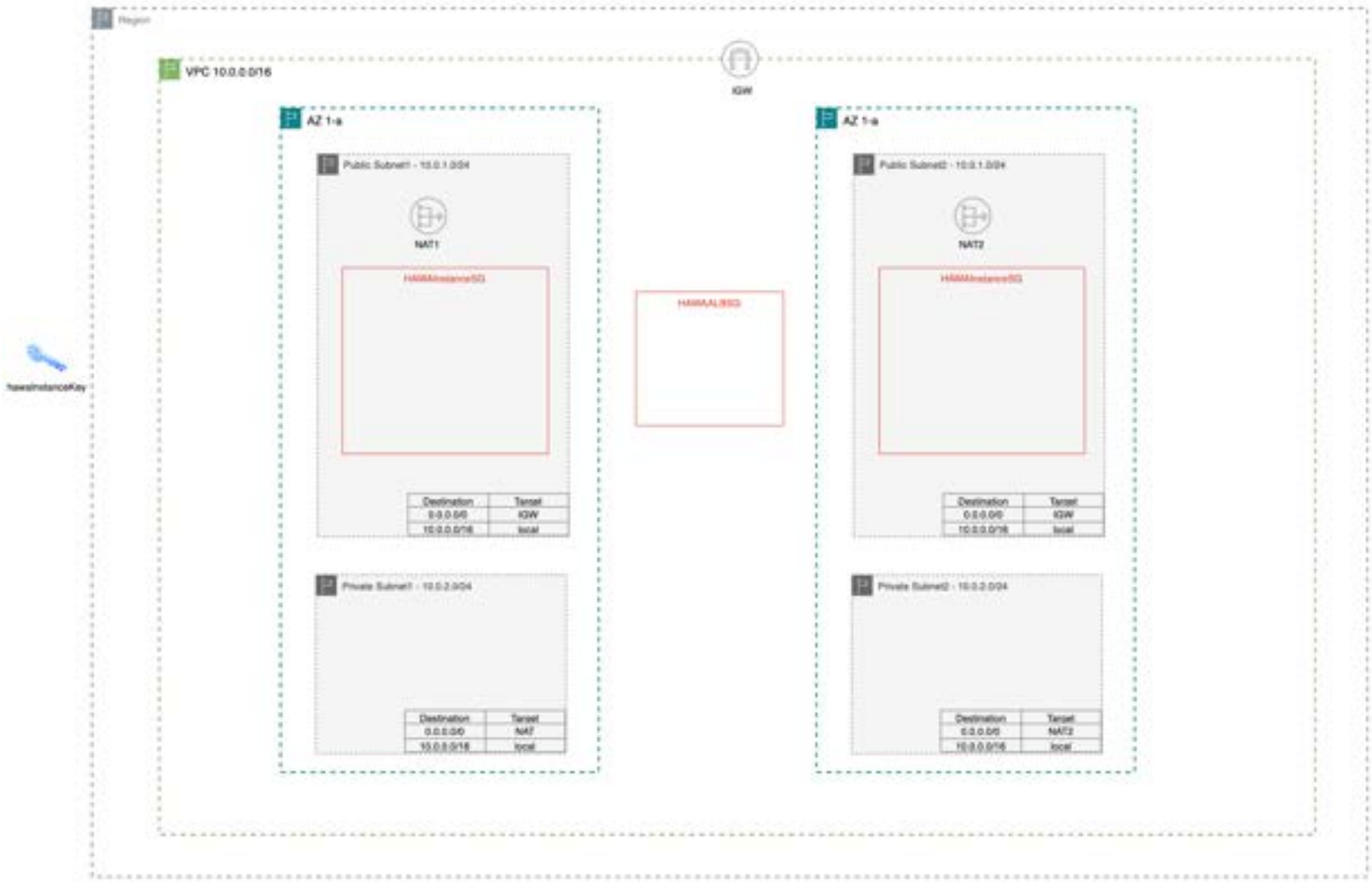
Connections to other networks

InternetGateway

NATGateway1

NATGateway2

SG & Key
Configuration


Region
VPC 10.0.0.0/16
IGW
AZ 1-a
Public Subnet1 - 10.0.1.0/24
NAT1
HAWSInstanceSG
Private Subnet1 - 10.0.2.0/24
HAWSALBSG
AZ 1-a
Public Subnet2 - 10.0.1.0/24
NAT2
HAWSInstanceSG
Private Subnet2 - 10.0.2.0/24
hawsInstanceKey

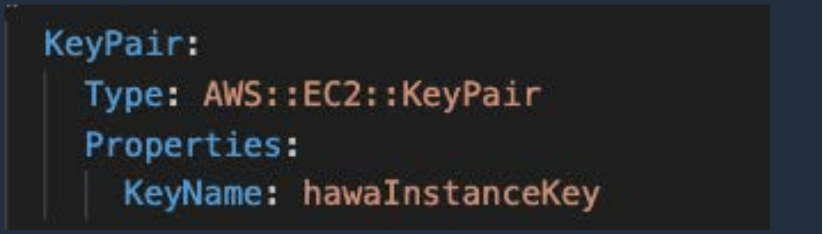
Destination	Target
0.0.0.0/0	IGW
10.0.0.0/16	local

Destination	Target
0.0.0.0/0	NAT
10.0.0.0/16	local

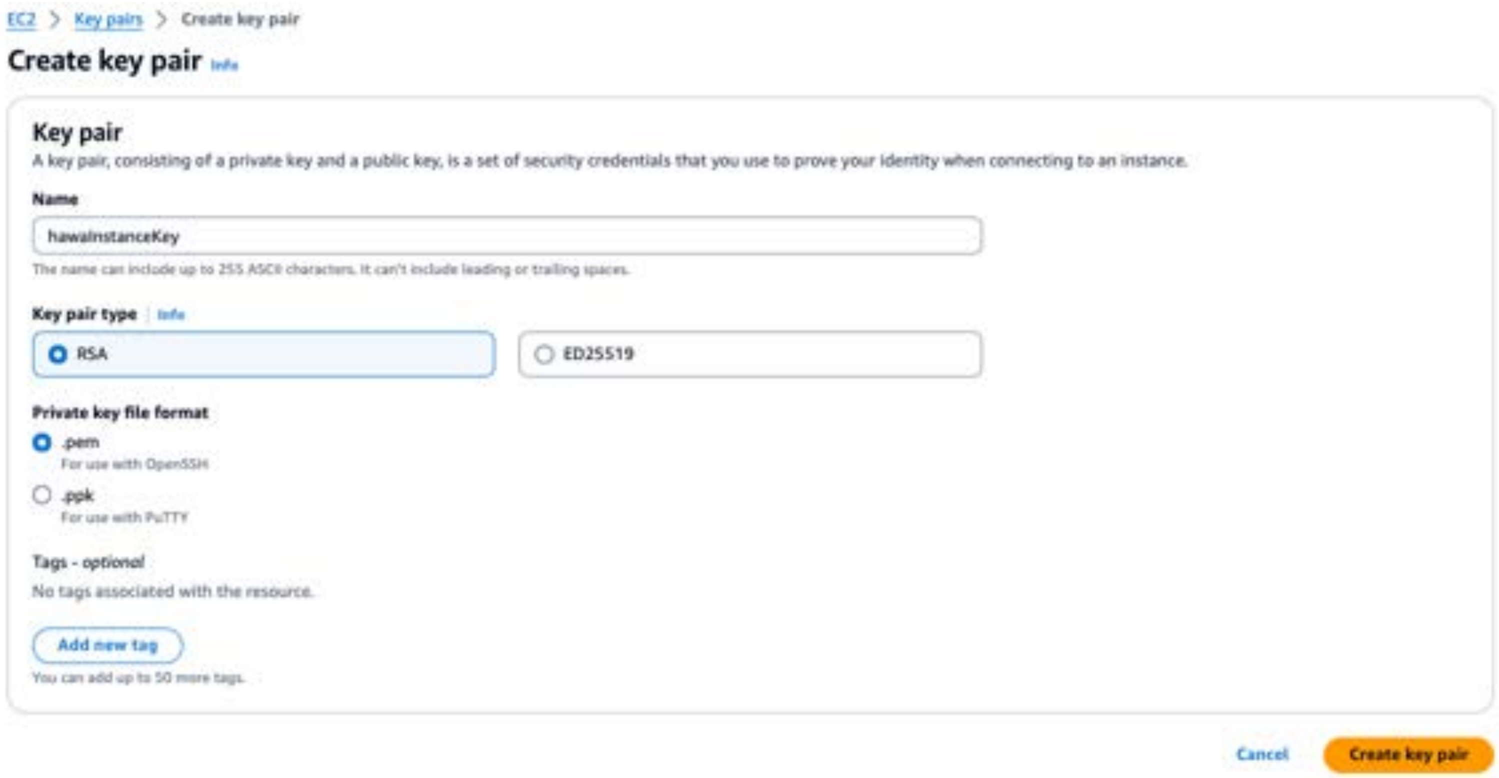
Destination	Target
0.0.0.0/0	IGW
10.0.0.0/16	local

Destination	Target
0.0.0.0/0	NAT2
10.0.0.0/16	local

Step 1 – SG & Key



```
KeyPair:
  Type: AWS::EC2::KeyPair
  Properties:
    KeyName: hawaInstanceKey
```



EC2 > Key pairs > Create key pair

Create key pair Info

Key pair

A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance.

Name

hawaInstanceKey

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

Key pair type Info

● RSA ○ ED25519

Private key file format

● .pem
For use with OpenSSH

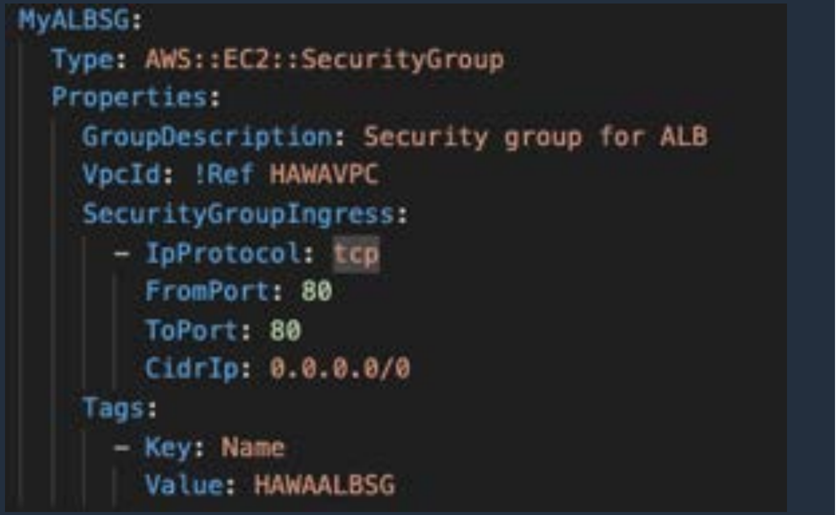
○ .ppk
For use with PuTTY

Tags - optional

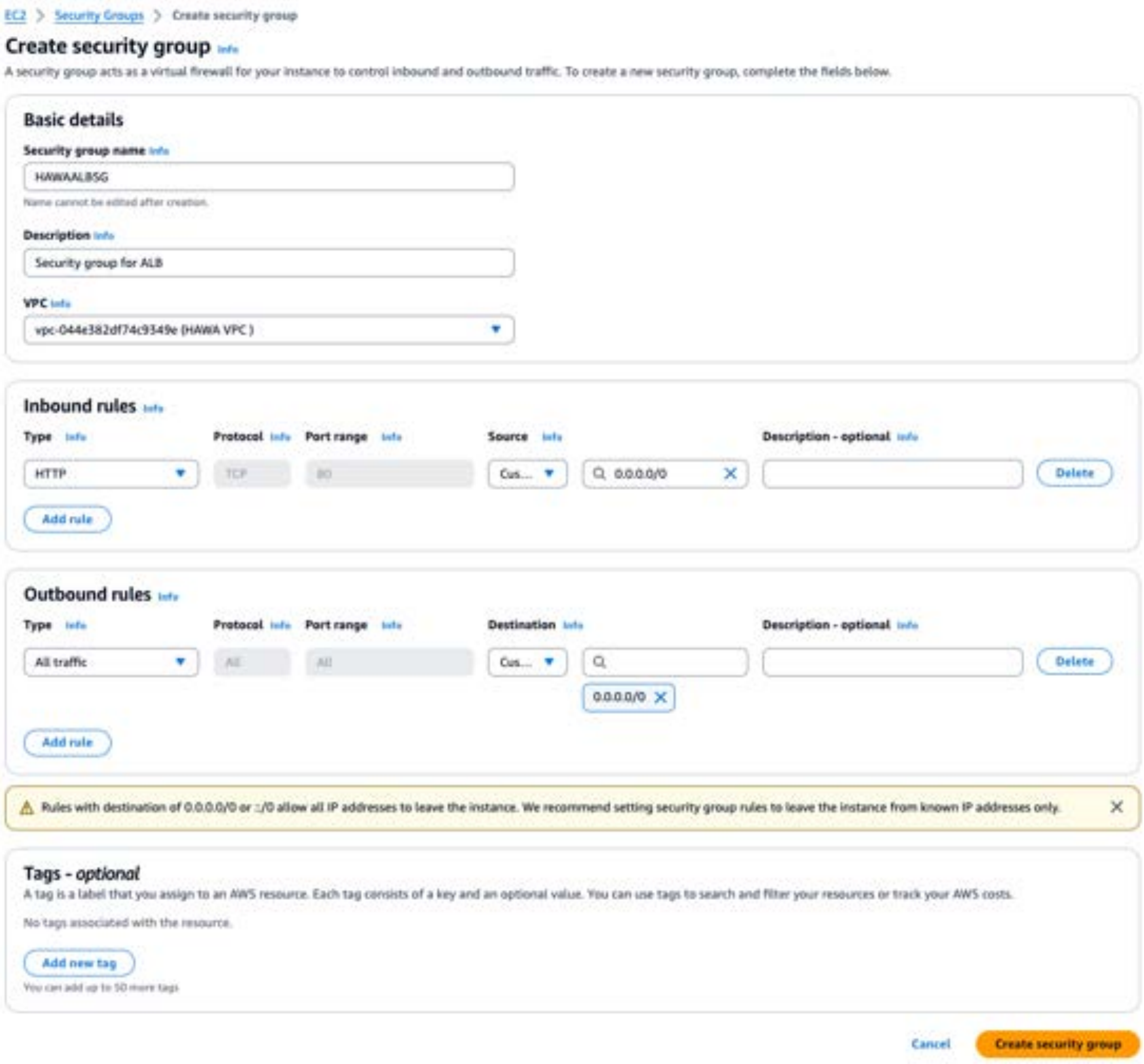
No tags associated with the resource.

You can add up to 50 more tags.

Step 2 – SG & Key



```
MyALBSG:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Security group for ALB
    VpcId: !Ref HAWAVPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 80
        ToPort: 80
        CidrIp: 0.0.0.0/0
    Tags:
      - Key: Name
        Value: HAWAALBSG
```



Create security group

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name: HAWAALBSG

Name cannot be edited after creation.

Description: Security group for ALB

VPC: vpc-044e382df74c9349e (HAWA VPC)

Inbound rules

Type	Protocol	Port range	Source		Description - optional
HTTP	TCP	80	Cus...	0.0.0.0/0	

Outbound rules

Type	Protocol	Port range	Destination		Description - optional
All traffic	All	All	Cus...	0.0.0.0/0	

⚠ Rules with destination of 0.0.0.0/0 or ::/0 allow all IP addresses to leave the instance. We recommend setting security group rules to leave the instance from known IP addresses only.

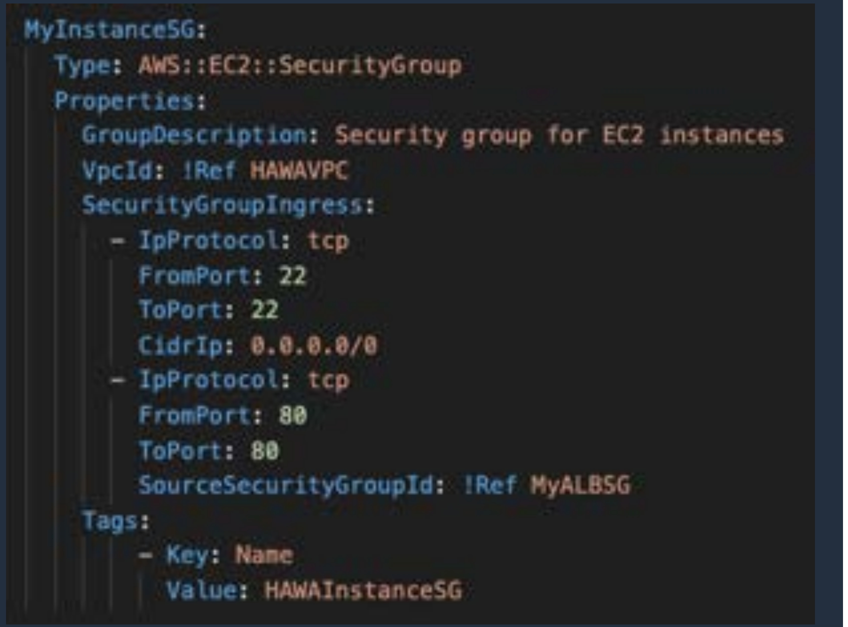
Tags - *optional*

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

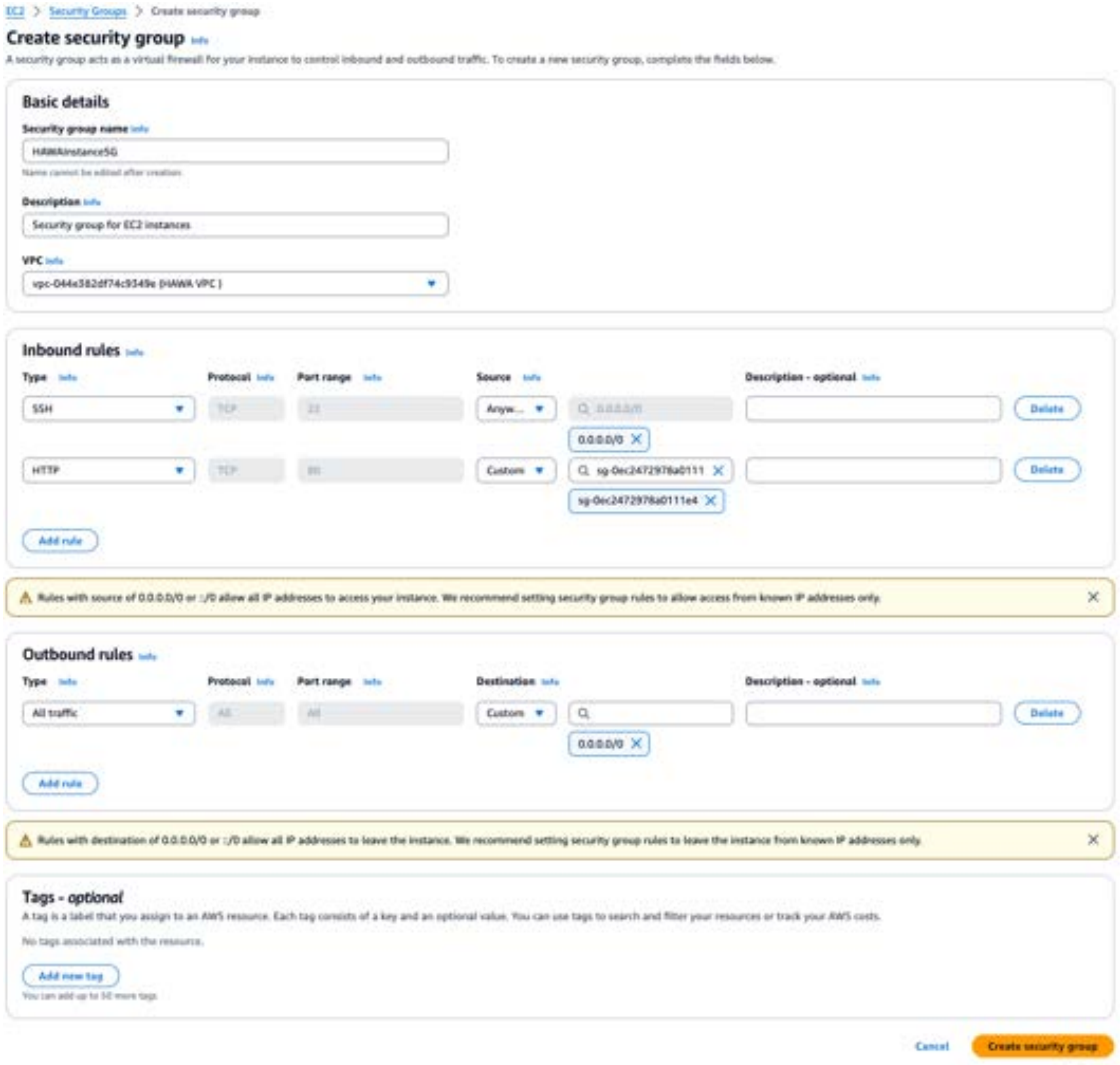
No tags associated with the resource.

You can add up to 50 more tags

Step 3 – SG & Key



```yaml
MyInstanceSG:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Security group for EC2 instances
    VpcId: !Ref HAWAVPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        CidrIp: 0.0.0.0/0
      - IpProtocol: tcp
        FromPort: 80
        ToPort: 80
        SourceSecurityGroupId: !Ref MyALBSG
    Tags:
      - Key: Name
        Value: HAWAInstanceSG
```



EC2 > Security Groups > Create security group

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name Info

HAWAInstanceSG

Name cannot be edited after creation.

Description Info

Security group for EC2 instances

VPC Info

vpc-044e382df74c9349e (HAWA VPC)

Inbound rules Info

Type	Protocol	Port range	Source		Description - optional	
SSH	TCP	22	Anyw...	0.0.0.0/0		Delete
HTTP	TCP	80	Custom	sg-0ec2472978a0111 sg-0ec2472978a0111e4		Delete

Add rule

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Outbound rules Info

Type	Protocol	Port range	Destination		Description - optional	
All traffic	All	All	Custom	0.0.0.0/0		Delete

Add rule

⚠ Rules with destination of 0.0.0.0/0 or ::/0 allow all IP addresses to leave the instance. We recommend setting security group rules to leave the instance from known IP addresses only.

Tags - *optional*

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

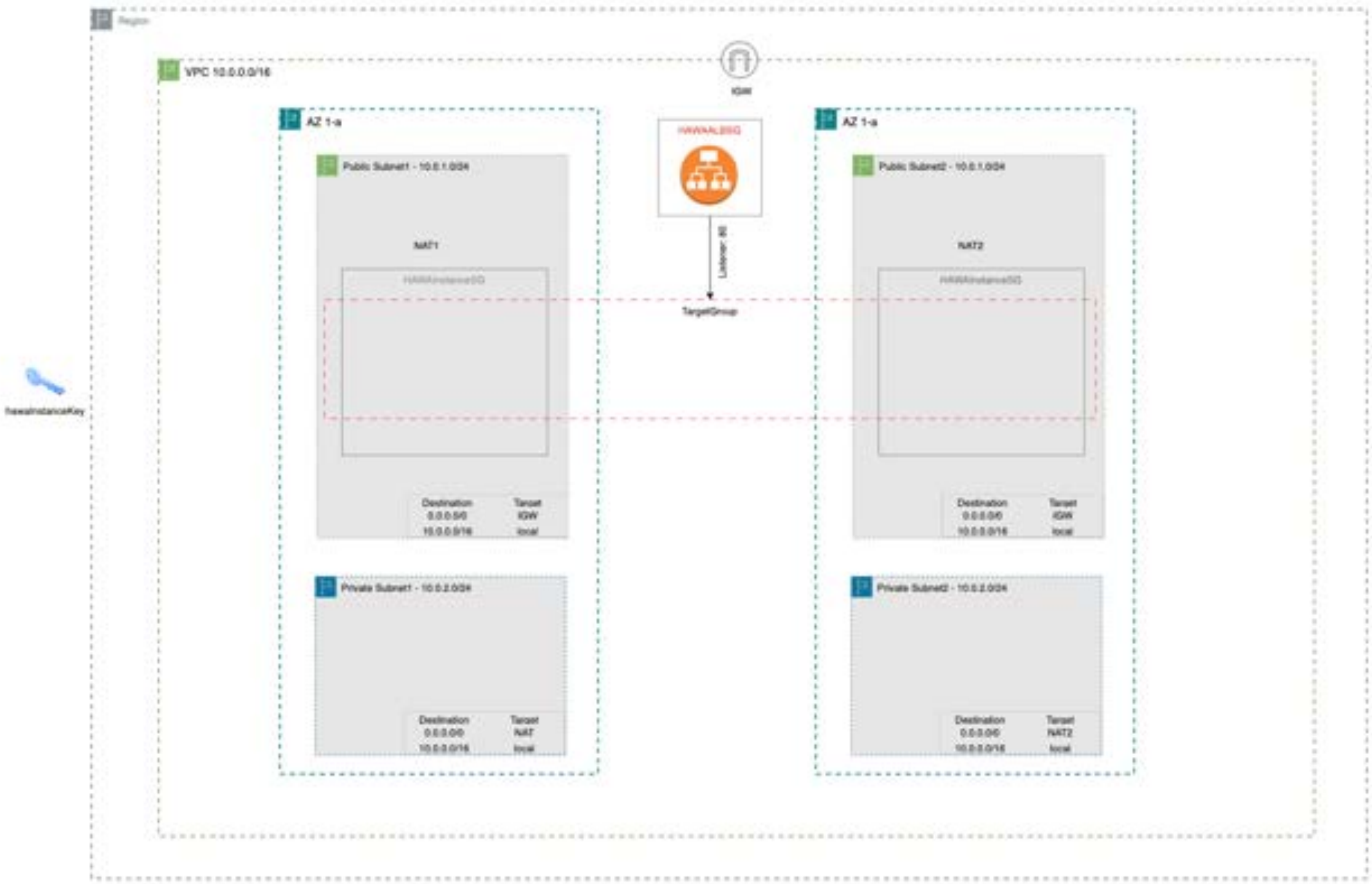
No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

ALB

Configuration



Destination	Target
0.0.0.0/0	IGW
10.0.0.0/16	local

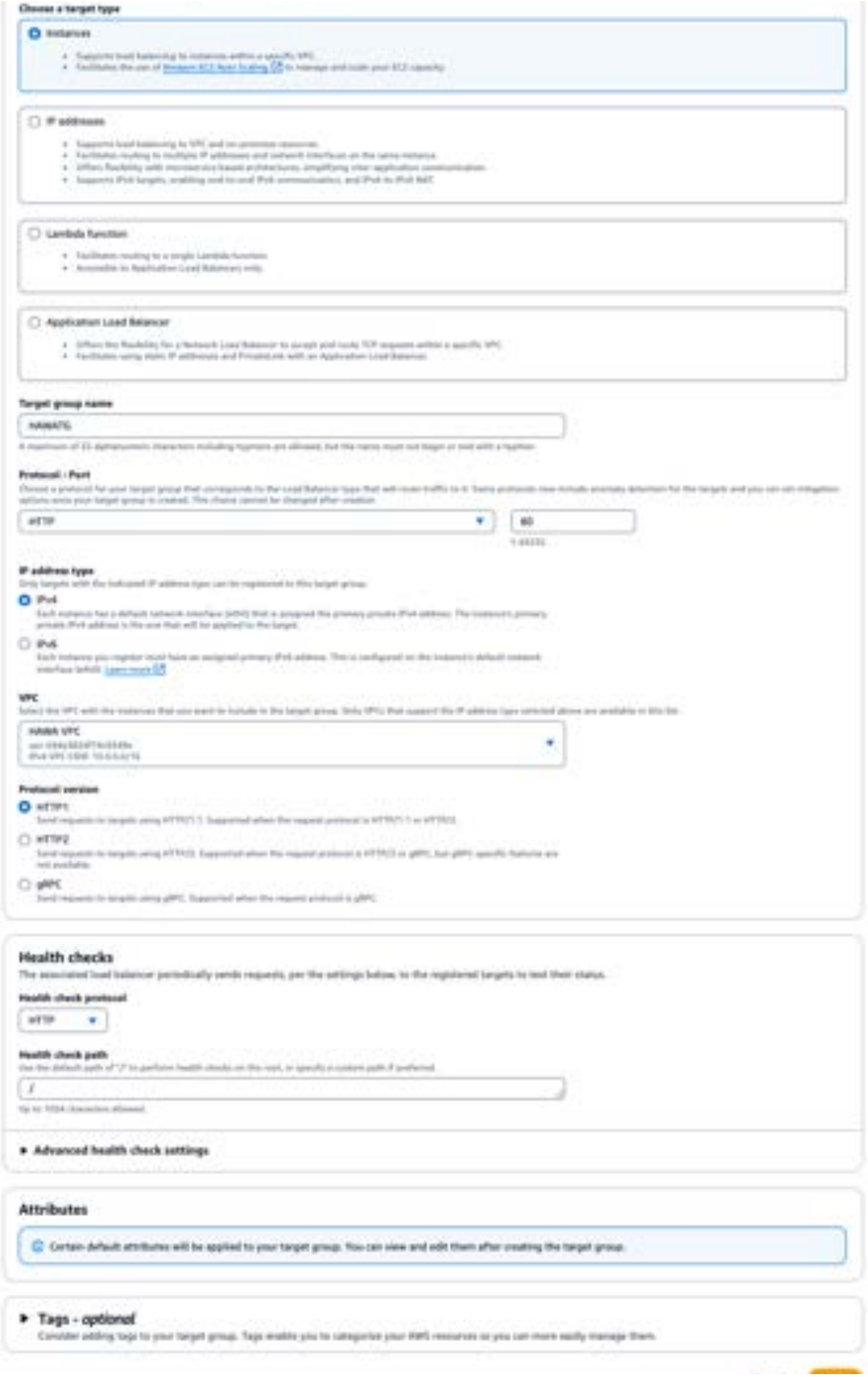
Destination	Target
0.0.0.0/0	IGW
10.0.0.0/16	local

Destination	Target
0.0.0.0/0	NAT
10.0.0.0/16	local

Destination	Target
0.0.0.0/0	NAT2
10.0.0.0/16	local

Step 1 – ALB

```
MyTargetGroup:
  Type: AWS::ElasticLoadBalancingV2::TargetGroup
  Properties:
    Name: MyTargetGroup
    Port: 80
    Protocol: HTTP
    TargetType: instance
    VpcId: !Ref HAWAVPC
```



Choose a target type

● Instances

- Supports load balancing to instances within a specific VPC.
- Facilitates the use of Amazon EC2 Auto Scaling to manage and scale your EC2 capacity.

○ IP addresses

- Supports load balancing to VPC and on-premises resources.
- Facilitates routing to multiple IP addresses and network interfaces on the same instance.
- Offers flexibility with microservice based architectures, simplifying inter-application communication.
- Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

○ Lambda function

- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

○ Application Load Balancer

- Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
- Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Target group name

HAWATG

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol : Port

Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation.

HTTP ▼ 80

1-65535

IP address type

Only targets with the indicated IP address type can be registered to this target group.

● IPv4
Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.

○ IPv6
Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). Learn more

VPC

Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

HAWA VPC
[illegible]
IPv4 VPC CIDR: 10.0.0.0/16 ▼

Protocol version

● HTTP1
Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.

○ HTTP2
Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.

○ gRPC
Send requests to targets using gRPC. Supported when the request protocol is gRPC.

Health checks

The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.

Health check protocol

HTTP ▼

Health check path

Use the default path of "/" to perform health checks on the root, or specify a custom path if preferred.

/

Up to 1024 characters allowed.

▶ Advanced health check settings

Attributes

Certain default attributes will be applied to your target group. You can view and edit them after creating the target group.

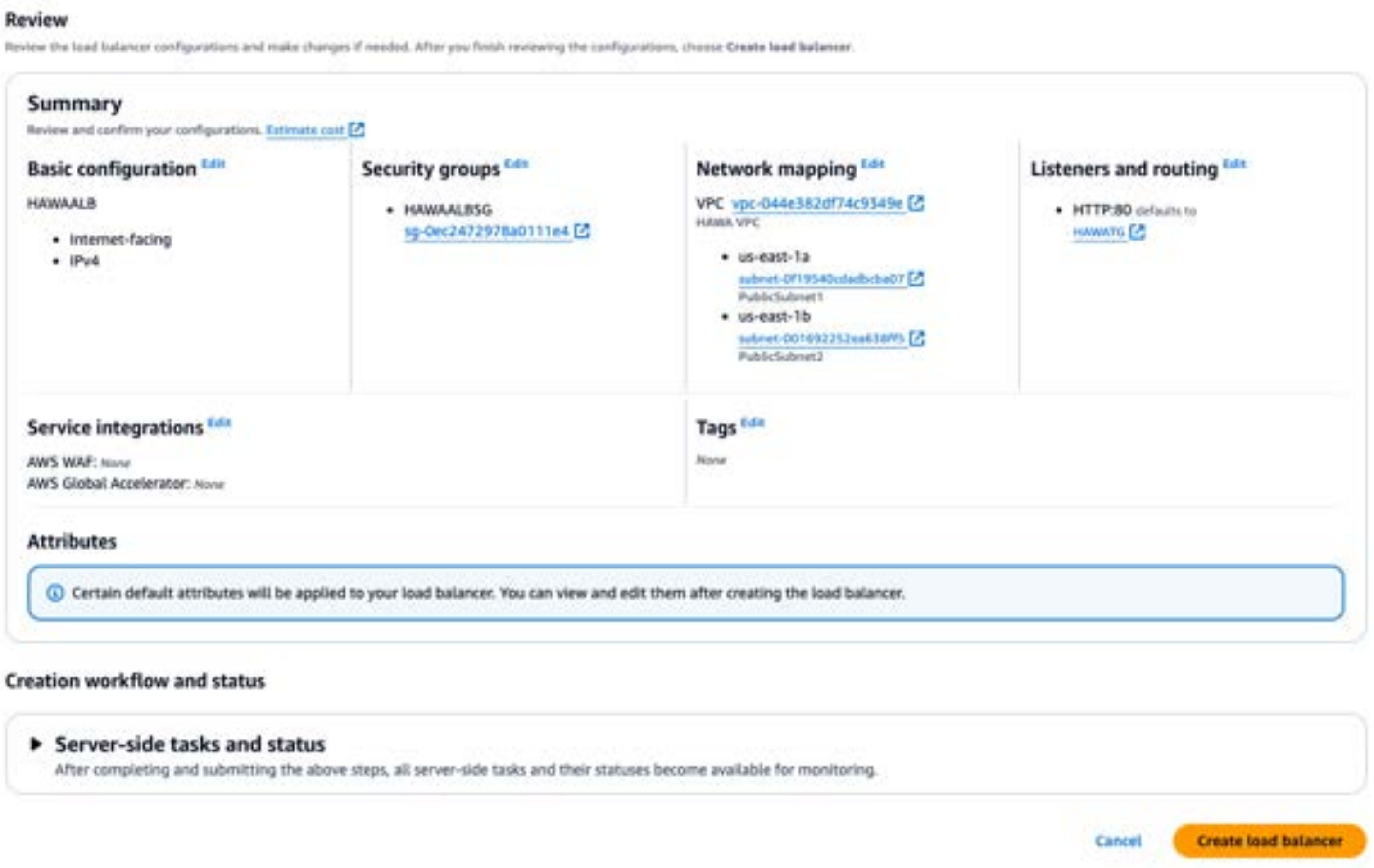
▶ Tags - optional

Consider adding tags to your target group. Tags enable you to categorize your AWS resources so you can more easily manage them.

Step 2 – ALB

```yaml
MyALB:
  Type: AWS::ElasticLoadBalancingV2::LoadBalancer
  Properties:
    Subnets:
      - !Ref PublicSubnet1
      - !Ref PublicSubnet2
    SecurityGroups:
      - !Ref MyALBSG
    Scheme: internet-facing
    LoadBalancerAttributes:
      - Key: idle_timeout.timeout_seconds
        Value: '60'

MyListener:
  Type: AWS::ElasticLoadBalancingV2::Listener
  Properties:
    DefaultActions:
      - Type: forward
        TargetGroupArn: !Ref MyTargetGroup
    LoadBalancerArn: !Ref MyALB
    Port: 80
    Protocol: HTTP
```



Review

Review the load balancer configurations and make changes if needed. After you finish reviewing the configurations, choose **Create load balancer**.

Summary

Review and confirm your configurations. Estimate cost

Basic configuration Edit

HAWAALB

- Internet-facing
- IPv4

Security groups Edit

- HAWAALBSG
  sg-0ec2472978a0111e4

Network mapping Edit

VPC vpc-044e382df74c9349e
HAWA VPC

- us-east-1a
  subnet-0f19540cdadbcba07
  PublicSubnet1
- us-east-1b
  subnet-001692252ea638ff5
  PublicSubnet2

Listeners and routing Edit

- HTTP:80 defaults to
  HAWATG

Service integrations Edit

AWS WAF: None
AWS Global Accelerator: None

Tags Edit

None

Attributes

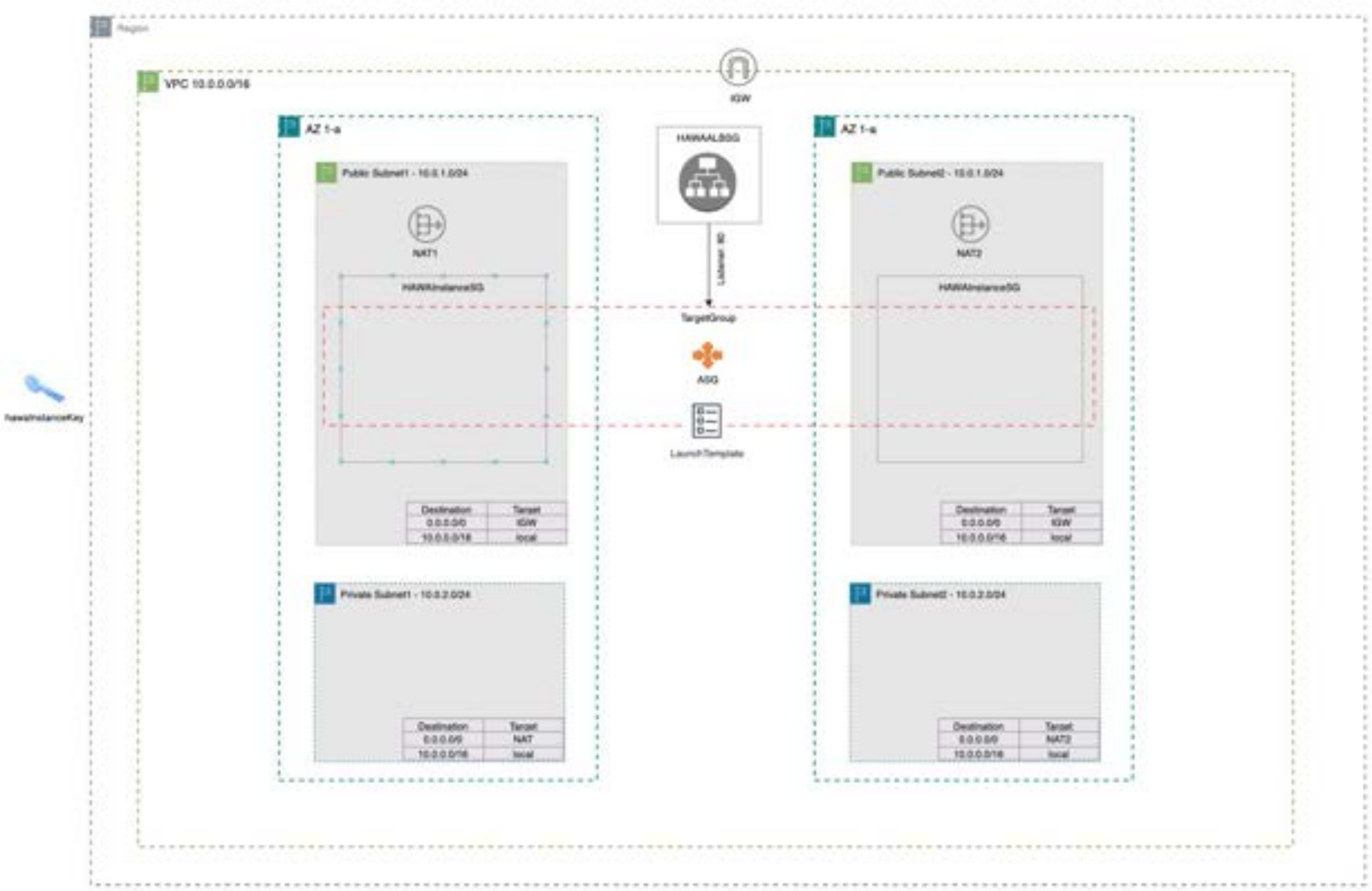
ⓘ Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.

Creation workflow and status

▶ Server-side tasks and status
After completing and submitting the above steps, all server-side tasks and their statuses become available for monitoring.

ASG

Configuration



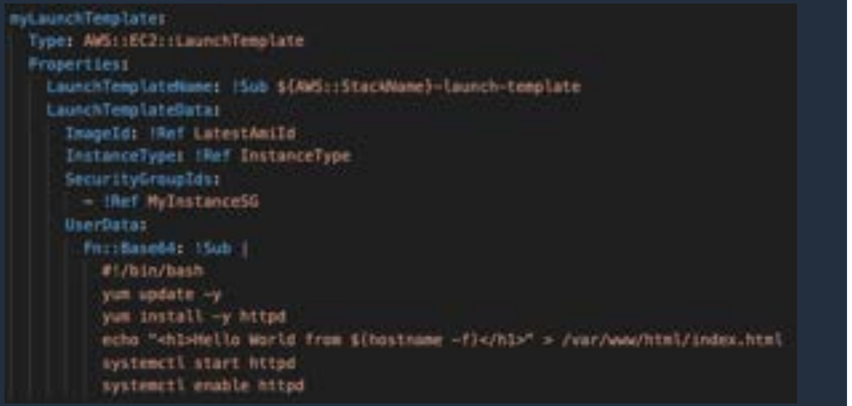
Destination	Target
0.0.0.0/0	IGW
10.0.0.0/16	local

Destination	Target
0.0.0.0/0	IGW
10.0.0.0/16	local

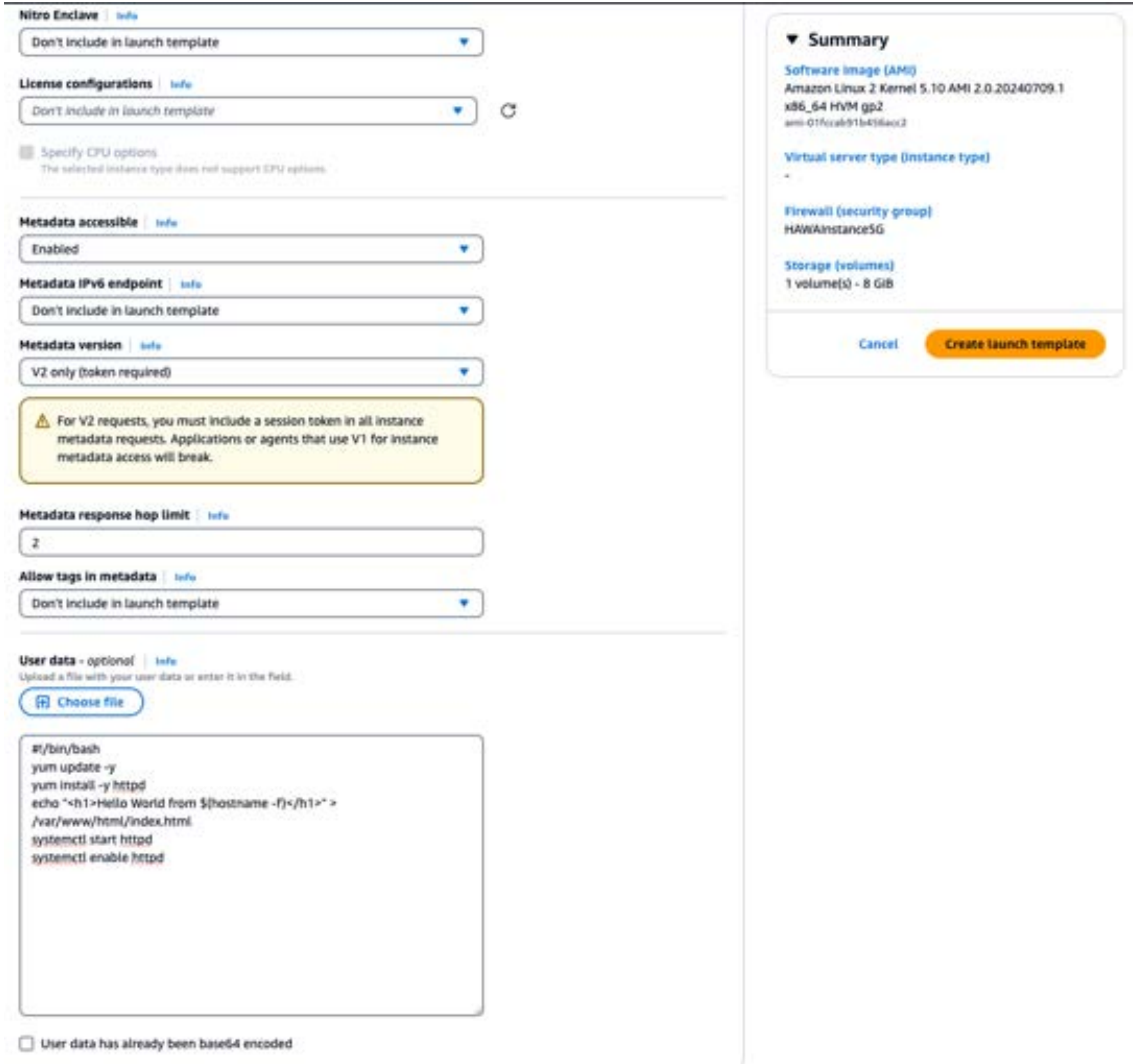
Destination	Target
0.0.0.0/0	NAT
10.0.0.0/16	local

Destination	Target
0.0.0.0/0	NAT2
10.0.0.0/16	local

Step 1 – ASG



```
myLaunchTemplate:
  Type: AWS::EC2::LaunchTemplate
  Properties:
    LaunchTemplateName: !Sub ${AWS::StackName}-launch-template
    LaunchTemplateData:
      ImageId: !Ref LatestAmiId
      InstanceType: !Ref InstanceType
      SecurityGroupIds:
        - !Ref MyInstanceSG
      UserData:
        Fn::Base64: !Sub |
          #!/bin/bash
          yum update -y
          yum install -y httpd
          echo "<h1>Hello World from $(hostname -f)</h1>" > /var/www/html/index.html
          systemctl start httpd
          systemctl enable httpd
```



Nitro Enclave | Info

Don't include in launch template ▼

License configurations | Info

Don't include in launch template ▼

☐ Specify CPU options

The selected instance type does not support CPU options.

Metadata accessible | Info

Enabled ▼

Metadata IPv6 endpoint | Info

Don't include in launch template ▼

Metadata version | Info

V2 only (token required) ▼

⚠ For V2 requests, you must include a session token in all instance metadata requests. Applications or agents that use V1 for instance metadata access will break.

Metadata response hop limit | Info

2

Allow tags in metadata | Info

Don't include in launch template ▼

User data - *optional* | Info

Upload a file with your user data or enter it in the field.

Choose file

```
#!/bin/bash
yum update -y
yum install -y httpd
echo "<h1>Hello World from $(hostname -f)</h1>" >
/var/www/html/index.html
systemctl start httpd
systemctl enable httpd
```

☐ User data has already been base64 encoded

▼ **Summary**

Software image (AMI)

Amazon Linux 2 Kernel 5.10 AMI 2.0.20240709.1 x86_64 HVM gp2

ami-01fccab91b456acc2

Virtual server type (instance type)

-

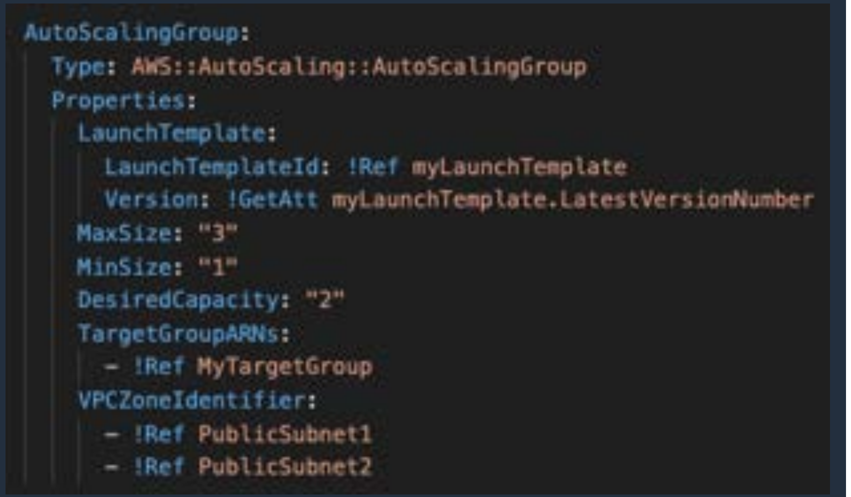
Firewall (security group)

HAWAInstanceSG

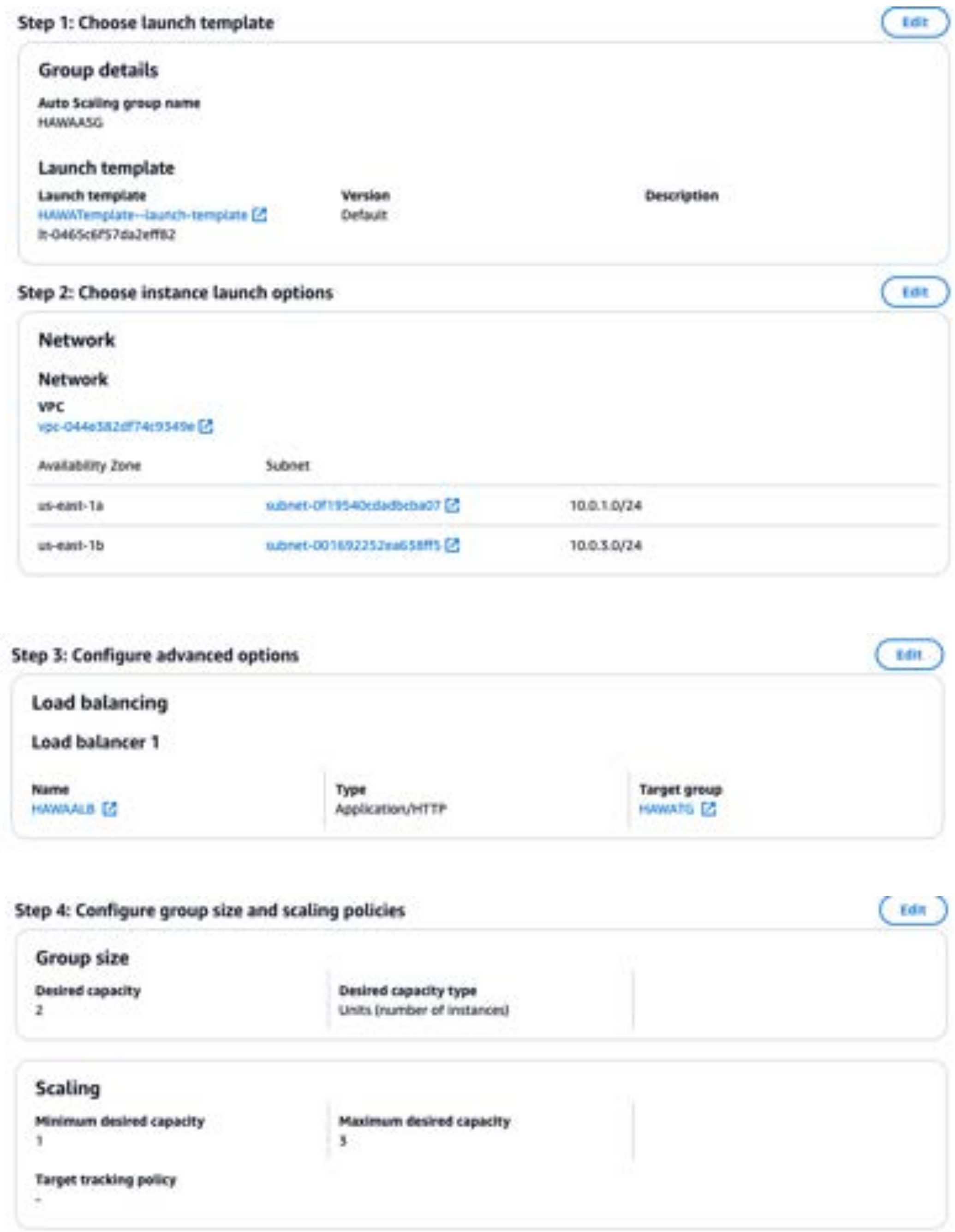
Storage (volumes)

1 volume(s) - 8 GiB

Step 2 – ASG



```yaml
AutoScalingGroup:
  Type: AWS::AutoScaling::AutoScalingGroup
  Properties:
    LaunchTemplate:
      LaunchTemplateId: !Ref myLaunchTemplate
      Version: !GetAtt myLaunchTemplate.LatestVersionNumber
    MaxSize: "3"
    MinSize: "1"
    DesiredCapacity: "2"
    TargetGroupARNs:
      - !Ref MyTargetGroup
    VPCZoneIdentifier:
      - !Ref PublicSubnet1
      - !Ref PublicSubnet2
```



Step 1: Choose launch template

Group details

Auto Scaling group name
HAWAASG

Launch template

Launch template	Version	Description
HAWATemplate--launch-template lt-0465c6f57da2eff82	Default	

Step 2: Choose instance launch options

Network

Network

VPC
vpc-044e382df74c9349e

Availability Zone	Subnet	
us-east-1a	subnet-0f19540cdadbcba07	10.0.1.0/24
us-east-1b	subnet-001692252ea658ff5	10.0.3.0/24

Step 3: Configure advanced options

Load balancing

Load balancer 1

Name	Type	Target group
HAWAALB	Application/HTTP	HAWATG

Step 4: Configure group size and scaling policies

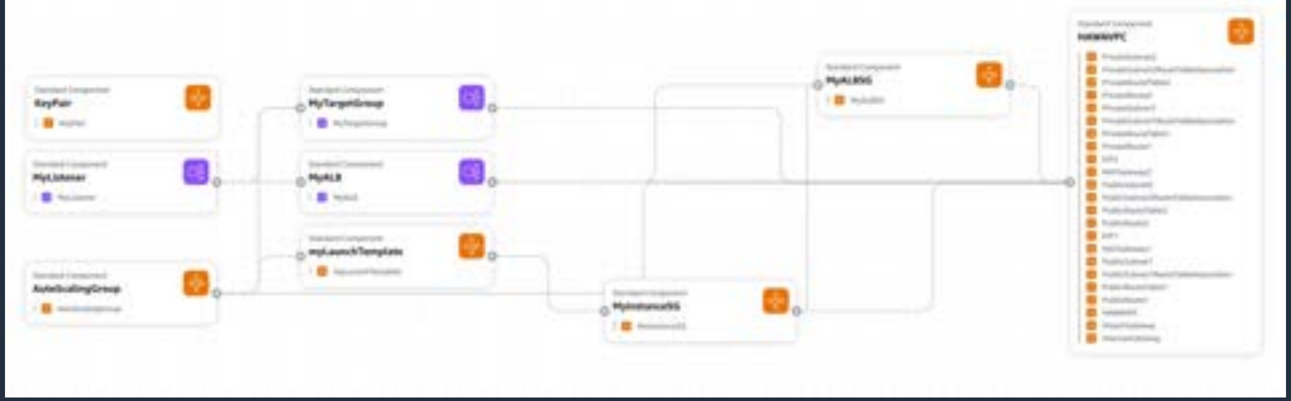
Group size

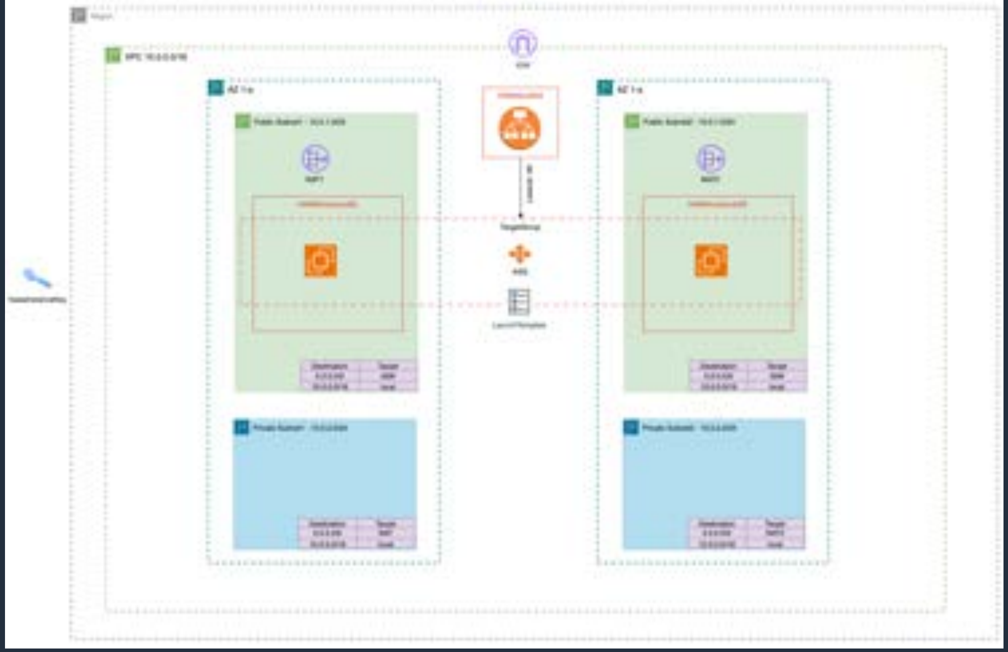
Desired capacity	Desired capacity type
2	Units (number of instances)

Scaling

Minimum desired capacity	Maximum desired capacity
1	3

Target tracking policy
-





Thank You!

Jason Zhang