

# SYST35144 Final Project

Luka Necajev And Owen Ross

## Objective

The objective of the project is to validate the functional knowledge of building out, using, and debugging cloud infrastructure. The main part of the project should be done using the Infrastructure as Code (IaC) approach (Terraform) to create a repeatable, flexible, and maintainable configuration that can be reliably deployed anytime anywhere.

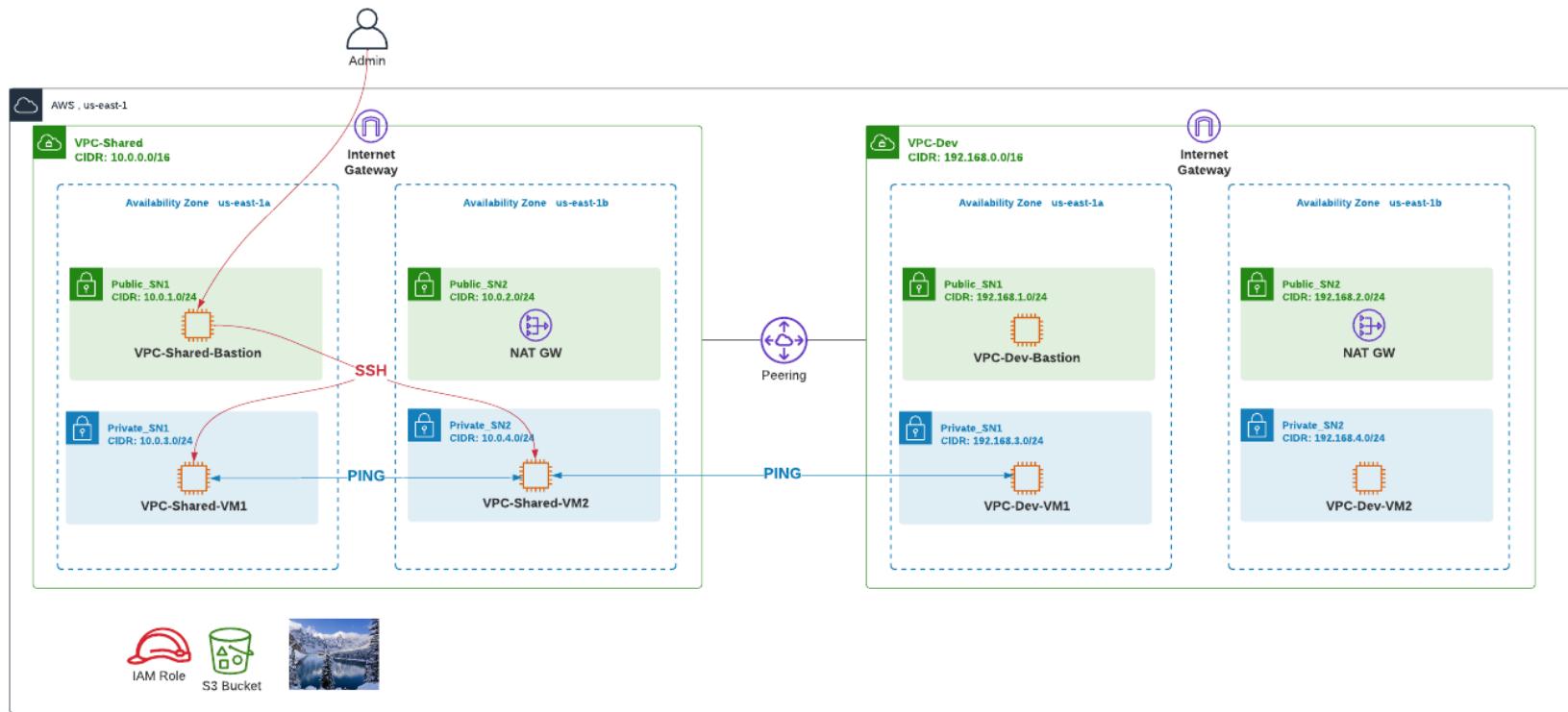
## Architecture and Functional Requirements

The design diagram below depicts the real-life scenario where one VPC needs to have network connectivity to another VPC. For example, one VPC can have a Jenkins server that deploys infrastructure and application changes to all the other VPCs (dev, non-prod, and prod) and requires network connectivity to these VPCs in a hub and spoke layout.

The diagram below is **for demonstration purposes only. You are not deploying this design diagram.**

## Architecture

In the scope of this assignment, you will deploy 2 VPCs (VPC-Jenkins and VPC-dev) in us-east-1 region with 2 private and 2 public subnets each interconnected via VPC peering. Detailed design diagram and deployment spec are below.



## Requirements

We are deploying 2 VPCs where some connectivity is permitted and the rest is blocked. Here are the requirements for your assignment.

1. Creating our VPCs

Code to Create the VPC's

VPC1

The screenshot shows a file tree for a project named 'FinalPro1 - /home'. Under the 'terraforming' directory, there are two sub-directories: '01-VPC' and '02-VPC'. In '01-VPC', the 'main.tf' file is selected. The code in 'main.tf' is as follows:

```
1 # AZ data source
2 data "aws_availability_zones" "available" {
3     state = "available"
4 }
5
6 # Create a new VPC
7 resource "aws_vpc" "vpc-tf" {
8     cidr_block          = var.vpc_cidr
9     enable_dns_hostnames = true
10    enable_dns_support   = true
11    tags = merge(
12        var.default_tags,
13        {
14            Name = "VPC-Shared"
15        }
16    )
17 }
```

Creating the first VPC with the name VPC-Shared

## VPC2

The screenshot shows a file tree for the same project 'FinalPro1 - /home'. Under the 'terraforming' directory, there are two sub-directories: '01-VPC' and '02-VPC'. In '02-VPC', the 'main.tf' file is selected. The code in 'main.tf' is as follows:

```
1 # AZ data source
2 data "aws_availability_zones" "available" {
3     state = "available"
4 }
5
6 # Create a new VPC
7 resource "aws_vpc" "vpc-tf" {
8     cidr_block          = var.vpc_cidr
9     enable_dns_hostnames = true
10    enable_dns_support   = true
11    tags = merge(
12        var.default_tags,
13        {
14            Name = "${var.prefix}-test"
15        }
16    )
17 }
```

Creating the second VPC with the name VPC2-test

## Showing our VPC's

The screenshot shows the AWS VPC console interface. At the top, there is a search bar and navigation links for 'N. Virginia' and the user 'voclabs/user1598691=Necajev,Luka @ 3051-3998-4171'. Below the search bar is a header with 'Your VPCs (3)' and an 'Info' link. To the right are buttons for 'Actions' and 'Create VPC'. A small info icon is also present. The main area is a table titled 'Your VPCs' with three rows. The columns are: Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR (Network border group), and IPv6. The data is as follows:

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR (Network border group)	IPv6
-	vpc-0626e552532f8b7c4	Available	172.31.0.0/16	-	-
VPC-Shared	vpc-05d3b8371e4a72cae	Available	10.0.0.0/16	-	-
VPC2-test	vpc-0526aa1e6447cccb0	Available	192.168.0.0/16	-	-

As you can see we have the default VPC, VPC1 (Shared VPC) and VPC2-(Dev VPC)

## VPC1:

The screenshot shows the AWS VPC console interface. At the top, there is a search bar and a navigation bar with options like 'Actions' and 'Create VPC'. Below the search bar, the title 'Your VPCs (1/3) Info' is displayed. A table lists three VPCs: one with a checkmark and two others without. The selected VPC is 'VPC-Shared' with VPC ID 'vpc-05d3b8371e4a72cae'. The table columns include Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR (Network border group), and IPv6. Below the table, the specific details for 'VPC-Shared' are shown, including its VPC ID, state, DHCP options set, IP ranges, and other configurations.

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR (Network border group)	IPv6
-	vpc-0626e552532f8b7c4	Available	172.31.0.0/16	-	-
<input checked="" type="checkbox"/> VPC-Shared	vpc-05d3b8371e4a72cae	Available	10.0.0.0/16	-	-
<input type="checkbox"/> VPC2-test	vpc-0526aa1e6447cccb0	Available	192.168.0.0/16	-	-

**vpc-05d3b8371e4a72cae / VPC-Shared**

**Details**    CIDRs    Flow logs    Tags

Details			
VPC ID vpc-05d3b8371e4a72cae	State Available	DNS hostnames Enabled	DNS resolution Enabled
Tenancy Default	DHCP options set dopt-09420f577faac01a2	Main route table rtb-0c95a688123c0fe85	Main network ACL acl-0c22ab3a8e5611b0c
Default VPC No	IPv4 CIDR 10.0.0.0/16	IPv6 pool -	IPv6 CIDR (Network border group) -
Route 53 Resolver DNS Firewall rule groups Failed to load rule groups	Owner ID 305139984171		

We set the first VPC to be in the 10.0.0.0/16 CIDR

## VPC2:

The screenshot shows the AWS VPC console interface. At the top, there's a search bar with placeholder text "Search for services, features, blogs, docs, and more" and a keyboard shortcut "[Alt+S]". To the right of the search bar are icons for refresh, help, and account information, followed by "N. Virginia" and a user profile for "vocabs/user1598691=Necajev,Luka @ 3051-3998-417".

The main area displays "Your VPCs (1/3) Info". A search bar labeled "Filter VPCs" is present. On the right, there are buttons for "Actions" and "Create VPC". Below the search bar is a table with columns: Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR (Network border group), and IPv6. The table contains three rows:

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR (Network border group)	IPv6
-	vpc-0626e552532f8b7c4	Available	172.31.0.0/16	-	-
VPC-Shared	vpc-05d3b8371e4a72cae	Available	10.0.0.0/16	-	-
VPC2-test	vpc-0526aa1e6447cccb0	Available	192.168.0.0/16	-	-

Below the table, the details for the selected VPC ("VPC2-test") are shown. The title is "vpc-0526aa1e6447cccb0 / VPC2-test". There are tabs for "Details", "CIDRs", "Flow logs", and "Tags", with "Details" being the active tab.

The "Details" section contains the following information:

VPC ID	State	DNS hostnames	DNS resolution
vpc-0526aa1e6447cccb0	Available	Enabled	Enabled
Tenancy	DHCP options set	Main route table	Main network ACL
Default	dopt-09420f577faac01a2	rtb-086ac002b1a90090a	acl-02b4ab76652eb4cac
Default VPC	IPv4 CIDR	IPv6 pool	IPv6 CIDR (Network border group)
No	192.168.0.0/16	-	-
Route 53 Resolver DNS Firewall rule groups	Owner ID		
Failed to load rule groups	305139984171		

We set the second VPC to be in the 192.168.0.0/16 CIDR

## 2. Subnets

To Create our subnets we ran terraform code that created two private and two public subnets in EACH VPC.

```
37
38  # Creating VPC-Jenkins
39  module "networking_VPC1" {
40    source      = "./01-VPC"
41    prefix      = "VPC1"
42    vpc_cidr   = "10.0.0.0/16"
43    public_cidrs = ["10.0.1.0/24", "10.0.2.0/24"]
44    private_cidrs = ["10.0.3.0/24", "10.0.4.0/24"]
45  }
46
47  # Creating VPC-Dev
48  module "networking_VPC2" {
49    source      = "./02-VPC"
50    prefix      = "VPC2"
51    vpc_cidr   = "192.168.0.0/16"
52    public_cidrs = ["192.168.1.0/24", "192.168.2.0/24"]
53    private_cidrs = ["192.168.3.0/24", "192.168.4.0/24"]
54  }
55
56
```

Showing the Subnets that were created in our VPC.

Search for services, features, blogs, docs, and more [Alt+S] N. Virginia v vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171 ▾

**Subnets (8/14) Info**

Filter subnets

Actions Create subnet

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
–	subnet-053642fc36a578548	Available	vpc-0626e552532f8b7c4	172.31.16.0/20	–
–	subnet-0c7486a743c7d4a41	Available	vpc-0626e552532f8b7c4	172.31.64.0/20	–
–	subnet-09e5379285e376f7b	Available	vpc-0626e552532f8b7c4	172.31.80.0/20	–
–	subnet-099a5cdb49ce85645	Available	vpc-0626e552532f8b7c4	172.31.0.0/20	–
–	subnet-074724371be9fc260	Available	vpc-0626e552532f8b7c4	172.31.32.0/20	–
–	subnet-02fed070b97aa637b	Available	vpc-0626e552532f8b7c4	172.31.48.0/20	–
<input checked="" type="checkbox"/> \$VPC-Shared-Private-SN1	subnet-0ffed8f6ca39cfda6	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.3.0/24	–
<input checked="" type="checkbox"/> \$VPC-Shared-Private-SN2	subnet-0779e9b40c9c2d0f7	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.4.0/24	–
<input checked="" type="checkbox"/> VPC-Shared-Public-SN1	subnet-069612a78407c9465	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.1.0/24	–
<input checked="" type="checkbox"/> VPC-Shared-Public-SN2	subnet-0f72302662eff87e9	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.2.0/24	–
<input checked="" type="checkbox"/> VPC2-Private-SN1	subnet-035725cb5d4db3bc7	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.3.0/24	–
<input checked="" type="checkbox"/> VPC2-Private-SN2	subnet-0f027487a0801afcc	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.4.0/24	–
<input checked="" type="checkbox"/> VPC2-Public-SN1	subnet-044d6cbc13d242cc8	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.1.0/24	–
<input checked="" type="checkbox"/> VPC2-Public-SN2	subnet-057c18508a27a172a	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.2.0/24	–

Showing The Subnet variables. Note that We are in the right CIDR, availability zone and using the correct routing table

## VPC1-Private SN1

Search for services, features, blogs, docs, and more [Alt+S]

N. Virginia vocabs/user1598691=Necajev,Luka @ 3051-3998-4171

Subnets (1/14) Info

Actions Create subnet

Filter subnets

<input type="checkbox"/>	-	subnet-074724371be9fc260	Available	vpc-0626e552532f8b7c4	172.31.32.0/20
<input type="checkbox"/>	-	subnet-02fed070b97aa637b	Available	vpc-0626e552532f8b7c4	172.31.48.0/20
<input checked="" type="checkbox"/>	\$VPC-Shared-Private-SN1	subnet-0ffed8f6ca39cfda6	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.3.0/24
<input type="checkbox"/>	\$VPC-Shared-Private-SN2	subnet-0779e9b40c9c2d0f7	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.4.0/24
<input type="checkbox"/>	VPC-Shared-Public-SN1	subnet-069612a78407c9465	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.1.0/24
<input type="checkbox"/>	VPC-Shared-Public-SN2	subnet-0f72302662eff87e9	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.2.0/24
<input type="checkbox"/>	VPC2-Private-SN1	subnet-035725cb5d4db3bc7	Available	vpc-0526aa1e6447cccb0   VPC-Shared-Private-RT	192.168.3.0/24
<input type="checkbox"/>	VPC2-Private-SN2	subnet-0f027487a0801afcc	Available	vnr-0526aa1e6447cccb0   VPC-Shared-Private-RT	192.168.4.0/24

subnet-0ffed8f6ca39cfda6 / \$VPC-Shared-Private-SN1

Details Flow logs Route table Network ACL CIDR reservations Sharing Tags

Details

Subnet ID subnet-0ffed8f6ca39cfda6	Subnet ARN arn:aws:ec2:us-east-1:305139984171:subnet/subnet-0ffed8f6ca39cfda6	State Available	IPv4 CIDR 10.0.3.0/24
Available IPv4 addresses 250	IPv6 CIDR -	Availability Zone us-east-1a	Availability Zone ID use1-az1
Network border group us-east-1	VPC vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	Route table rtb-050d3807a95f12092   VPC-Shared-Private-RT	Network ACL acl-0c22ab3a8e5611b0c
Default subnet No	Auto-assign public IPv4 address No	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No
Customer-owned IPv4 pool -	Outpost ID -	IPv4 CIDR reservations -	IPv6 CIDR reservations -
IPv6-only No	Hostname type IP name	Resource name DNS A record Disabled	Resource name DNS AAAA record Disabled
DNS64 Disabled	Owner 305139984171		

**Orange** is Showing the correct CIDR for private network 1. (10.0.3.0)

**Blue** is showing the correct Availability zone (us-east-1a)

**Red** is showing that we are associated with the correct routing table. (VPC-Shared-Private-RT)

## VPC1-Private SN2

Search for services, features, blogs, docs, and more [Alt+S] N. Virginia vocabs/user1598691=Necajev,Luka @ 3051-3998-4171

**Subnets (1/14) Info**

<input type="checkbox"/>	-	subnet-074724371be9fc260	<span>Available</span>	vpc-0626e552532f8b7c4	172.31.32.0/20	-
<input type="checkbox"/>	-	subnet-02fed070b97aa637b	<span>Available</span>	vpc-0626e552532f8b7c4	172.31.48.0/20	-
<input type="checkbox"/>	\$VPC-Shared-Private-SN1	subnet-0ffed8f6ca39cfda6	<span>Available</span>	vpc-05d3b8371e4a72cae   VPC...	10.0.3.0/24	-
<input checked="" type="checkbox"/>	\$VPC-Shared-Private-SN2	subnet-0779e9b40c9c2d0f7	<span>Available</span>	vpc-05d3b8371e4a72cae   VPC...	10.0.4.0/24	-
<input type="checkbox"/>	VPC-Shared-Public-SN1	subnet-069612a78407c9465	<span>Available</span>	vpc-05d3b8371e4a72cae   VPC...	10.0.1.0/24	-
<input type="checkbox"/>	VPC-Shared-Public-SN2	subnet-0f72302662eff87e9	<span>Available</span>	vpc-05d3b8371e4a72cae   VPC...	10.0.2.0/24	-
<input type="checkbox"/>	VPC2-Private-SN1	subnet-035725cb5d4db3bc7	<span>Available</span>	vpc-0526aa1e6447cccb0   VPC...	192.168.3.0/24	-
<input type="checkbox"/>	VPC2-Private-SN2	subnet-0f027487a0801afcc	<span>Available</span>	vnr-0526aa1e6447ccch0   VPC...	192.168.4.0/24	-

subnet-0779e9b40c9c2d0f7 / \$VPC-Shared-Private-SN2

**Details** Flow logs Route table Network ACL CIDR reservations Sharing Tags

**Details**

Subnet ID <a href="#">subnet-0779e9b40c9c2d0f7</a>	Subnet ARN <a href="#">arn:aws:ec2:us-east-1:305139984171:subnet/subnet-0779e9b40c9c2d0f7</a>	State <span>Available</span>	IPv4 CIDR <a href="#">10.0.4.0/24</a>
Available IPv4 addresses <a href="#">250</a>	IPv6 CIDR -	Availability Zone <a href="#">us-east-1b</a>	Availability Zone ID <a href="#">use1-az2</a>
Network border group <a href="#">us-east-1</a>	VPC <a href="#">vpc-05d3b8371e4a72cae   VPC-Shared-Private</a>	Route table <a href="#">rtb-050d3807a95f12092   VPC-Shared-Private-RT</a>	Network ACL <a href="#">acl-0c22ab3a8e5611b0c</a>
Default subnet No	Auto-assign public IPv4 address No	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No
Customer-owned IPv4 pool -	Outpost ID -	IPv4 CIDR reservations -	IPv6 CIDR reservations -
IPv6-only No	Hostname type IP name	Resource name DNS A record Disabled	Resource name DNS AAAA record Disabled
DNS64 Disabled	Owner <a href="#">305139984171</a>		

Orange is Showing the correct CIDR for private network 1. (10.0.4.0)

Blue is showing the correct Availability zone (us-east-1b)

Red is showing that we are associated with the correct routing table. (VPC-Shared-Private-RT)

### Routing table

The screenshot shows two pages from the AWS VPC console.

**Subnets (1/14) Page:**

- Shows 14 subnets listed in a table.
- The subnet `subnet-0ffed8f6ca39cfda6` is selected, highlighted with a blue border.
- Details for this subnet:
  - Subnet ID: `subnet-0ffed8f6ca39cfda6`
  - Availability Zone: `Available` (blue checkmark)
  - VPC: `vpc-0626e552532f8b7c4`
  - CIDR: `172.31.0.0/20`

**Route table: rtb-050d3807a95f12092 / VPC-Shared-Private-RT Page:**

- Shows a route table with 2 routes.
- Details for the routes:

Destination	Target
<code>10.0.0.0/16</code>	<code>local</code>
<code>0.0.0.0/0</code>	<code>nat-05f477143056ef0ed</code>

## VPC1-Public SN1

Search for services, features, blogs, docs, and more [Alt+S]

N. Virginia vocabs/user1598691=Necajev,Luka @ 3051-3998-4171

**Subnets (1/14) Info**

Filter subnets

<input type="checkbox"/>	-	subnet-074724371be9fc260	Available	vpc-0626e552532f8b7c4	172.31.32.0/20	-
<input type="checkbox"/>	-	subnet-02fed070b97aa637b	Available	vpc-0626e552532f8b7c4	172.31.48.0/20	-
<input checked="" type="checkbox"/>	\$VPC-Shared-Private-SN1	subnet-0ffed8f6ca39cfda6	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.3.0/24	-
<input type="checkbox"/>	\$VPC-Shared-Private-SN2	subnet-0779e9b40c9c2d0f7	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.4.0/24	-
<input type="checkbox"/>	VPC-Shared-Public-SN1	subnet-069612a78407c9465	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.1.0/24	-
<input type="checkbox"/>	VPC-Shared-Public-SN2	subnet-0f72302662eff87e9	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.2.0/24	-
<input type="checkbox"/>	VPC2-Private-SN1	subnet-035725cb5d4db3bc7	Available	vpc-0526aa1e6447cccb0   VPC-Shared-Private-RT	192.168.3.0/24	-
<input type="checkbox"/>	VPC2-Private-SN2	subnet-0f027487a0801afcc	Available	vnr-0526aa1e6447cccb0   VPC-Shared-Private-RT	192.168.4.0/24	-

**subnet-0ffed8f6ca39cfda6 / \$VPC-Shared-Private-SN1**

Details Flow logs Route table Network ACL CIDR reservations Sharing Tags

**Details**

Subnet ID <a href="#">subnet-0ffed8f6ca39cfda6</a>	Subnet ARN <a href="#">arn:aws:ec2:us-east-1:305139984171:subnet/subnet-0ffed8f6ca39cfda6</a>	State <a href="#">Available</a>	IPv4 CIDR <a href="#">10.0.3.0/24</a>
Available IPv4 addresses <a href="#">250</a>	IPv6 CIDR -	Availability Zone <a href="#">us-east-1a</a>	Availability Zone ID <a href="#">use1-az1</a>
Network border group <a href="#">us-east-1</a>	VPC <a href="#">vpc-05d3b8371e4a72cae   VPC-Shared-Private</a>	Route table <a href="#">rtb-050d3807a95f12092   VPC-Shared-Private-RT</a>	Network ACL <a href="#">acl-0c22ab3a8e5611b0c</a>
Default subnet No	Auto-assign public IPv4 address No	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No
Customer-owned IPv4 pool -	Outpost ID -	IPv4 CIDR reservations -	IPv6 CIDR reservations -
IPv6-only No	Hostname type IP name	Resource name DNS A record Disabled	Resource name DNS AAAA record Disabled
DNS64 Disabled	Owner <a href="#">305139984171</a>		

Orange is Showing the correct CIDR for private network 1. (10.0.3.0)

Blue is showing the correct Availability zone (us-east-1a)

Red is showing that we are associated with the correct routing table. (VPC-Shared-Private-RT)

## Routing table

The screenshot shows two related AWS VPC management pages.

**Subnets (1/14) Info**

ID	Name	Status	VPC	CIDR	Availability Zone
subnet-0779e9b40c9c2d0f7	\$VPC-Shared-Private-SN1	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.3.0/24	-
subnet-02fed07b97aa637b		Available	vpc-0626e52532f8b7c4	172.31.48.0/20	-
subnet-0ffed8f6ca39cfda6		Available	vpc-05d3b8371e4a72cae   VPC-Shared-Public-RT	10.0.4.0/24	-
subnet-0779e9b40c9c2d0f7	\$VPC-Shared-Private-SN1	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.4.0/24	-
subnet-069612a78407c9465		Available	vpc-05d3b8371e4a72cae   VPC-Shared-Public-RT	10.0.1.0/24	-
subnet-0f72302662eff87e9		Available	vpc-05d3b8371e4a72cae   VPC-Shared-Public-RT	10.0.2.0/24	-
subnet-035725cb5d4db3bc7		Available	vpc-0526aa1e6447cccb0   VPC-Shared-Public-RT	192.168.3.0/24	-
subnet-0f027487a0801afcc		Available	vpc-0526aa1e6447cccb0   VPC-Shared-Public-RT	192.168.4.0/24	-

**subnet-0779e9b40c9c2d0f7 / \$VPC-Shared-Private-SN1**

Route table: rtb-050d3807a95f12092 / VPC-Shared-Private-RT

Routes (2)

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	nat-05f477143056ef0ed

## VPC1-Public SN1

Search for services, features, blogs, docs, and more [Alt+S] NL Virginia v vodabs/user1598691-NecajevLuka @ 3051-3998-4171 v

### Subnets (1/14) Info

Actions Create subnet

	Name	Status	Owner	CIDR	Range
<input type="checkbox"/>	-	Available	vpc-0626e552532f8b7c4	172.31.32.0/20	-
<input type="checkbox"/>	-	Available	vpc-0626e552532f8b7c4	172.31.48.0/20	-
<input type="checkbox"/>	SVPC-Shared-Private-SN1	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.3.0/24	-
<input type="checkbox"/>	SVPC-Shared-Private-SN2	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.4.0/24	-
<input checked="" type="checkbox"/>	VPC-Shared-Public-SN1	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.1.0/24	-
<input type="checkbox"/>	VPC-Shared-Public-SN2	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.2.0/24	-
<input type="checkbox"/>	VPC2-Private-SN1	Available	vpc-0526aa1e6447ccb0   VPC...	192.168.3.0/24	-
<input type="checkbox"/>	VPC2-Private-SN2	Available	vpc-0526aa1e6447ccb0   VPC...	192.168.4.0/24	-

subnet-069612a78407c9465 / VPC-Shared-Public-SN1

Details Flow logs Route table Network ACL CIDR reservations Sharing Tags

#### Details

Subnet ID <a href="#">subnet-069612a78407c9465</a>	Subnet ARN <a href="#">arn:aws:ec2:us-east-1:305139984171:subnet/subnet-069612a78407c9465</a>	State <span style="color: green;">Available</span>	IPv4 CIDR <a href="#">10.0.1.0/24</a>
Available IPv4 addresses 250	IPv6 CIDR -	Availability Zone <a href="#">us-east-1a</a>	Availability Zone ID <a href="#">use1-az1</a>
Network border group <a href="#">us-east-1</a>	VPC <a href="#">vpc-05d3b8371e4a72cae   VPC-Shared-Public-RT</a>	Route table <a href="#">rtb-0aa89a494c54bfd64   VPC-Shared-Public-RT</a>	Network ACL <a href="#">acl-0c22ab3a8e5611b0c</a>
Default subnet No	Auto-assign public IPv4 address No	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No
Customer-owned IPv4 pool -	Outpost ID -	IPv4 CIDR reservations -	IPv6 CIDR reservations -
IPv6-only No	Hostname type IP name	Resource name DNS A record Disabled	Resource name DNS AAAA record Disabled
DNS64 Disabled	Owner <a href="#">305139984171</a>		

Orange is Showing the correct CIDR for private network 1. (10.0.1.0)

Blue is showing the correct Availability zone (us-east-1a)

Red is showing that we are associated with the correct routing table. (VPC-Shared-Public-RT)

### Routing table

The screenshot shows two pages from the AWS Management Console:

**Subnets (1/14) - Info**

This page lists 14 subnets. The subnet **subnet-069612a78407c9465 / VPC-Shared-Public-SN1** is selected, indicated by a checked checkbox in the first column. This subnet is associated with the **VPC-Shared-Public-RT** (Route Table), which is highlighted in red.

Subnet ID	Subnet Name	Status	Route Table	CIDR Range	Tags
subnet-0779e9b40c9c2d0f7	\$VPC-Shared-Private-SN1	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Private-RT	10.0.4.0/24	-
subnet-069612a78407c9465	VPC-Shared-Public-SN1	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Public-RT	10.0.1.0/24	-
subnet-0f72302662eff87e9	VPC-Shared-Public-SN2	Available	vpc-05d3b8371e4a72cae   VPC-Shared-Public-RT	10.0.2.0/24	-
subnet-035725cb5d4db3bc7	VPC2-Private-SN1	Available	vpc-0526aa1e6447cccb0   VPC2-Private-RT	192.168.3.0/24	-
subnet-0f027487a0801afcc	VPC2-Private-SN2	Available	vpc-0526aa1e6447cccb0   VPC2-Private-RT	192.168.4.0/24	-
subnet-044d6cbc13d242cc8	VPC2-Public-SN1	Available	vpc-0526aa1e6447cccb0   VPC2-Public-RT	192.168.1.0/24	-
subnet-057c18508a27a172a	VPC2-Public-SN2	Available	vpc-0526aa1e6447cccb0   VPC2-Public-RT	192.168.2.0/24	-

**Route table: rtb-0aa89a494c54bfd64 / VPC-Shared-Public-RT**

This page displays the routes for the selected route table. It shows two entries:

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-0809e94771070e78f

Buttons at the bottom right of the route table page include "Edit route table association" and "Run Reachability Analyzer".

## VPC1-Public SN2

The screenshot shows the AWS VPC Subnets page with 14 subnets listed. The subnet **subnet-0f72302662eff87e9 / VPC-Shared-Public-SN2** is selected, highlighted with a blue border. The subnet details are displayed in a large modal window.

**Subnets (1/14) Info**

Subnet ID	Subnet ARN	State	IPv4 CIDR
subnet-0f72302662eff87e9	arn:aws:ec2:us-east-1:305139984171:subnet/subnet-0f72302662eff87e9	Available	10.0.2.0/24
250	IPv6 CIDR	Availability Zone	Availability Zone ID
us-east-1	-	us-east-1b	use1-az2
VPC	Route table	Network ACL	
vpc-05d3b8371e4a72cae   VPC-Shared-Public-SN2	rtb-0aa89a494c54bfbd64   VPC-Shared-Public-RT	ad-0c22ab3a8e5611b0c	
No	Auto-assign IPv6 address	Auto-assign customer-owned IPv4 address	
No	No	No	
-	Outpost ID	IPv4 CIDR reservations	
IPv6-only	-	-	
No	Hostname type	Resource name DNS A record	
DNS64	IP name	Disabled	
Disabled	Owner	Resource name DNS AAAA record	
	305139984171	Disabled	

**Details** | Flow logs | Route table | Network ACL | CIDR reservations | Sharing | Tags

**Details**

Subnet ID subnet-0f72302662eff87e9	Subnet ARN arn:aws:ec2:us-east-1:305139984171:subnet/subnet-0f72302662eff87e9	State Available	IPv4 CIDR 10.0.2.0/24
Available IPv4 addresses 250	IPv6 CIDR	Availability Zone us-east-1b	Availability Zone ID use1-az2
Network border group us-east-1	VPC	Route table rtb-0aa89a494c54bfbd64   VPC-Shared-Public-RT	Network ACL ad-0c22ab3a8e5611b0c
Default subnet No	Auto-assign public IPv4 address No	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No
Customer-owned IPv4 pool -	Outpost ID	IPv4 CIDR reservations -	IPv6 CIDR reservations -
IPv6-only No	Hostname type	Resource name DNS A record	Resource name DNS AAAA record
DNS64 Disabled	IP name	Disabled	Disabled
	Owner		
	305139984171		

Orange is Showing the correct CIDR for private network 1. (10.0.2.0)

Blue is showing the correct Availability zone (us-east-1b)

Red is showing that we are associated with the correct routing table. (VPC-Shared-Public-RT)

## Routing table

The screenshot shows the AWS VPC Routing Table details page for a specific route table. At the top, there's a navigation bar with tabs for Details, Flow logs, Route table (which is selected and highlighted in orange), Network ACL, CIDR reservations, Sharing, and Tags. Below the tabs, a message says "You can now check network connectivity with Reachability Analyzer" with a "Run Reachability Analyzer" button.

The main section displays the "Route table: rtb-0aa89a494c54bfd64 / VPC-Shared-Public-RT". It includes a "Routes (2)" table with two entries:

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-0809e94771070e78f

At the bottom right of the main content area, there's a "Edit route table association" button.

## VPC2-Private SN1

Search for services, features, blogs, docs, and more [Alt+S] N. Virginia v vodabs/user1598691=NecajevLuka @ 3051-3998-4171

Subnets (1/14) [Info](#)

Actions [Create subnet](#)

Filter subnets

Subnet ID	Subnet ARN	State	IPv4 CIDR
\$VPC-Shared-Private-SN2	subnet-0779e9b40c9c2d0f7	Available	vpc-05d3b8371e4a72cae   VP...
VPC-Shared-Public-SN1	subnet-069612a78407c9465	Available	vpc-05d3b8371e4a72cae   VP...
VPC-Shared-Public-SN2	subnet-0f772302662eff87e9	Available	vpc-05d3b8371e4a72cae   VP...
<input checked="" type="checkbox"/> VPC2-Private-SN1	subnet-035725cb5d4db3bc7	Available	vpc-0526aa1e6447cccb0   VPC...
VPC2-Private-SN2	subnet-0f027487a0801afcc	Available	vpc-0526aa1e6447cccb0   VPC...
VPC2-Public-SN1	subnet-044d6cbc13d242cc8	Available	vpc-0526aa1e6447cccb0   VPC...
VPC2-Public-SN2	subnet-057c18508a27a172a	Available	vpc-0526aa1e6447cccb0   VPC...

subnet-035725cb5d4db3bc7 / VPC2-Private-SN1

Details Flow logs Route table Network ACL CIDR reservations Sharing Tags

Details

Subnet ID <a href="#">subnet-035725cb5d4db3bc7</a>	Subnet ARN <a href="#">arn:aws:ec2:us-east-1:305139984171:subnet/subnet-035725cb5d4db3bc7</a>	State <a href="#">Available</a>	IPv4 CIDR <a href="#">192.168.3.0/24</a>
Available IPv4 addresses <a href="#">250</a>	IPv6 CIDR —	Availability Zone <a href="#">us-east-1a</a>	Availability Zone ID <a href="#">use1-az1</a>
Network border group <a href="#">us-east-1</a>	VPC <a href="#">vpc-0526aa1e6447cccb0   VPC2-test</a>	Route table <a href="#">rtb-09e6f4958cd74e0aa   VPC2-Private-RT</a>	Network ACL <a href="#">aci-02b4ab76652eb4cac</a>
Default subnet No	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No	Auto-assign customer-owned IPv4 address No
Customer-owned IPv4 pool —	Auto-assign public IPv4 address No	IPv4 CIDR reservations —	IPv6 CIDR reservations —
IPv6-only No	Outpost ID —	Resource name DNS A record Disabled	Resource name DNS AAAA record Disabled
DNS64 Disabled	IP name Owner <a href="#">305139984171</a>		

Orange is Showing the correct CIDR for private network 1. (192.168.3.0/24)

Blue is showing the correct Availability zone (us-east-1a)

Red is showing that we are associated with the correct routing table. (VPC2 -Private-RT)

## Routing table

The screenshot shows the AWS VPC console interface. At the top, there's a navigation bar with links for services, features, blogs, docs, and more, and a search bar labeled [Alt+S]. The top right shows the region as N. Virginia and the user as vodabs/user1598691=Necajev,Luka @ 3051-3998-4171.

**Subnets (1/14) Info**

A table lists 14 subnets. One subnet is selected: "VPC2-Private-SN1" (subnet-035725cb5d4db3bc7). The table columns include:

	Name	ID	Status	VPC	CIDR	Availability Zone
<input type="checkbox"/>	\$VPC-Shared-Priva...	subnet-0779e9b40c9c2d0f7	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.4.0/24	-
<input type="checkbox"/>	VPC-Shared-Public-...	subnet-069612a78407c9465	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.1.0/24	-
<input type="checkbox"/>	VPC-Shared-Public-...	subnet-0f72302662eff87e9	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.2.0/24	-
<input checked="" type="checkbox"/>	VPC2-Private-SN1	subnet-035725cb5d4db3bc7	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.3.0/24	-
<input type="checkbox"/>	VPC2-Private-SN2	subnet-0f027487a0801afcc	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.4.0/24	-
<input type="checkbox"/>	VPC2-Public-SN1	subnet-044d6cbc13d242cc8	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.1.0/24	-
<input type="checkbox"/>	VPC2-Public-SN2	subnet-057c18508a27a172a	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.2.0/24	-

**subnet-035725cb5d4db3bc7 / VPC2-Private-SN1**

Details Flow logs **Route table** Network ACL CIDR reservations Sharing Tags

A message box says: "You can now check network connectivity with Reachability Analyzer". It includes a "Run Reachability Analyzer" button and a close button.

**Route table: rtb-09e6f4958cd74e0aa / VPC2-Private-RT**

Edit route table association

**Routes (2)**

Destination	Target
192.168.0.0/16	local
0.0.0.0/0	nat-05b7b519e3d8f6b65

## VPC2-Private SN2

for services, features, blogs, docs, and more [Alt+5] N. Virginia v vodlabs/user1598691+Necajev,Luka @ 3051-3998-4

Subnets (1/14) Info Actions Create subnet

Filter subnets

Subnet ID	Name	State	IPv4 CIDR	IPv6 CIDR
\$VPC-Shared-Private-SN2	subnet-0779e9b40c9c2d0f7	Available	vpc-05d3b8371e4a72cae   VP...	10.0.4.0/24
VPC-Shared-Public-SN1	subnet-069612a78407c9465	Available	vpc-05d3b8371e4a72cae   VP...	10.0.1.0/24
VPC-Shared-Public-SN2	subnet-0f72302662ef87e9	Available	vpc-05d3b8371e4a72cae   VP...	10.0.2.0/24
VPC2-Private-SN1	subnet-035725cb5d4db3bc7	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.3.0/24
<b>VPC2-Private-SN2</b>	<b>subnet-0f027487a0801afcc</b>	<b>Available</b>	<b>vpc-0526aa1e6447cccb0   VPC...</b>	<b>192.168.4.0/24</b>
VPC2-Public-SN1	subnet-044d6cbc13d242cc8	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.1.0/24
VPC2-Public-SN2	subnet-057c18508a27a172a	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.2.0/24

subnet-0f027487a0801afcc / VPC2-Private-SN2

Details Flow logs Route table Network ACL CIDR reservations Sharing Tags

**Details**

Subnet ID	Subnet ARN	State	IPv4 CIDR
subnet-0f027487a0801afcc	arn:aws:ec2:us-east-1:305139984171:subnet/subnet-0f027487a0801afcc	Available	192.168.4.0/24
Available IPv4 addresses	250	Availability Zone	Availability Zone ID
250	IPv6 CIDR	us-east-1b	use1-az2
Network border group	-	Route table	Network ACL
us-east-1	VPC	rtb-09e6f4958cd74e0aa   VPC2-Private-RT	acl-02b4ab76652eb4cac
Default subnet	vpc-0526aa1e6447cccb0   VPC2-test	No	Auto-assign customer-owned IPv4 address
No	Auto-assign public IPv4 address	No	No
Customer-owned IPv4 pool	No	IPv4 CIDR reservations	IPv6 CIDR reservations
-	Outpost ID	-	-
IPv6-only	-	Resource name DNS A record	Resource name DNS AAAA record
No	Hostname type	Disabled	Disabled
DNS64	IP name		
Disabled	Owner		
	305139984171		

Orange is Showing the correct CIDR for private network 1. (192.168.4.0/24)

Blue is showing the correct Availability zone (us-east-1b)

Red is showing that we are associated with the correct routing table. (VPC2 -Private-RT)

### Routing Table

The screenshot shows two pages from the AWS Management Console related to VPC routing.

**Subnets (1/14) Page:**

- Shows a list of subnets under VPC2-Private-SN2.
- The subnet `subnet-0f027487a0801afcc` is selected, highlighted with a blue border.
- Details for this subnet are shown in the modal below:

  - Route table:** `rtb-09e6f4958cd74e0aa / VPC2-Private-RT`
  - Details tab:** Shows basic subnet information.
  - Route table tab:** Shows the association with the correct route table.
  - Message:** "You can now check network connectivity with Reachability Analyzer" with a "Run Reachability Analyzer" button.

**Route Table Association Page:**

- Shows the association between the subnet and the route table `rtb-09e6f4958cd74e0aa / VPC2-Private-RT`.
- Routes (2) Table:**

Destination	Target
192.168.0.0/16	local
0.0.0.0/0	<code>nat-05b7b519e3d8f6b65</code>

## VPC2-Public SN1

The screenshot shows the AWS VPC Subnets page with the following details:

Subnet ID	Subnet ARN	State	IPv4 CIDR
subnet-044d6cbc13d242cc8	arn:aws:ec2:us-east-1:305139984171:subnet/subnet-044d6cbc13d242cc8	Available	192.168.1.0/24
Available IPv4 addresses	249		
Network border group	us-east-1	Availability Zone	use1-az1
Default subnet	No	Route table	rtb-0cfb9e3a9c3e61e48   VPC2-Public-RT
Customer-owned IPv4 pool	-	Auto-assign IPv6 address	No
IPv6-only	No	Auto-assign public IPv4 address	No
DNS64	Disabled	IPv4 CIDR reservations	-
		Resource name DNS A record	Disabled
		IP name	
		Hostname type	
		Owner	305139984171

Annotations in the screenshot:

- Orange** highlights the "IPv4 CIDR" field (192.168.1.0/24).
- Blue** highlights the "Availability Zone" field (us-east-1a).
- Red** highlights the "Route table" field (rtb-0cfb9e3a9c3e61e48 | VPC2-Public-RT).

Orange is Showing the correct CIDR for private network 1. (192.168.1.0/24)

Blue is showing the correct Availability zone (us-east-1a)

Red is showing that we are associated with the correct routing table. (VPC2 -Public-RT)

## Route Table

for services, features, blogs, docs, and more [Alt+S] N. Virginia vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171

Subnets (1/14) Info

Filter subnets

Subnet ID	Name	Status	Associated VPC	CIDR Range
subnet-0779e9b40c9c2d0f7	\$VPC-Shared-Private-SN1	Available	vp-05d3b8371e4a72cae   VPC...	10.0.4.0/24
subnet-069612a78407c9465	VPC-Shared-Public-SN1	Available	vp-05d3b8371e4a72cae   VPC...	10.0.1.0/24
subnet-0f72302662eff87e9	VPC-Shared-Public-SN2	Available	vp-05d3b8371e4a72cae   VPC...	10.0.2.0/24
subnet-035725cb5d4db3bc7	VPC2-Private-SN1	Available	vp-0526aa1e6447cccb0   VPC...	192.168.3.0/24
subnet-0f027487a0801afcc	VPC2-Private-SN2	Available	vp-0526aa1e6447cccb0   VPC...	192.168.4.0/24
subnet-044d6cbc13d242cc8	VPC2-Public-SN1	Available	vp-0526aa1e6447cccb0   VPC...	192.168.1.0/24
subnet-057c18508a27a172a	VPC2-Public-SN2	Available	vp-0526aa1e6447cccb0   VPC...	192.168.2.0/24

subnet-044d6cbc13d242cc8 / VPC2-Public-SN1

Details Flow logs Route table Network ACL CIDR reservations Sharing Tags

You can now check network connectivity with Reachability Analyzer Run Reachability Analyzer

Route table: rtb-0efb9e3a9c3e61e48 / VPC2-Public-RT Edit route table association

Routes (2)

Filter routes

Destination	Target
192.168.0.0/16	local
0.0.0.0/0	igw-0effc15f10c03f0bd

## VPC2-Public SN2

or services, features, blogs, docs, and more [ALL+5] N. Virginia vodabs/user1598691-NecajevLuka @ 3051-3998-41

### Subnets (1/14) Info

Filter subnets

	Name	Status	ARN	CIDR	
<input type="checkbox"/>	\$VPC-Shared-Private-SN2	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.4.0/24	-
<input type="checkbox"/>	VPC-Shared-Public-SN1	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.1.0/24	-
<input type="checkbox"/>	VPC-Shared-Public-SN2	Available	vpc-05d3b8371e4a72cae   VPC...	10.0.2.0/24	-
<input type="checkbox"/>	VPC2-Private-SN1	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.3.0/24	-
<input type="checkbox"/>	VPC2-Private-SN2	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.4.0/24	-
<input type="checkbox"/>	VPC2-Public-SN1	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.1.0/24	-
<input checked="" type="checkbox"/>	VPC2-Public-SN2	Available	vpc-0526aa1e6447cccb0   VPC...	192.168.2.0/24	-

subnet-057c18508a27a172a / VPC2-Public-SN2

Details Flow logs Route table Network ACL CIDR reservations Sharing Tags

#### Details

Subnet ID	subnet-057c18508a27a172a	Subnet ARN	arn:aws:ec2:us-east-1:305139984171:subnet/subnet-057c18508a27a172a	State	Available	IPv4 CIDR	192.168.2.0/24
Available IPv4 addresses	251	IPv6 CIDR	-	Availability Zone	us-east-1b	Availability Zone ID	use1-az2
Network border group	us-east-1	VPC	vpc-0526aa1e6447cccb0   VPC2-test	Route table	rtb-0efb9e3a9c3e61e48   VPC2-Public-RT	Network ACL	acl-02b4ab76652eb4cac
Default subnet	No	Auto-assign public IPv4 address	No	Auto-assign IPv6 address	No	Auto-assign customer-owned IPv4 address	No
Customer-owned IPv4 pool	-	Auto-assign public IPv4 address	No	IPv4 CIDR reservations	-	IPv6 CIDR reservations	-
IPv6-only	No	Outpost ID	-	Resource name DNS A record	Disabled	Resource name DNS AAAA record	Disabled
DNS64	Disabled	Hostname type	IP name	Owner	305139984171		

Orange is Showing the correct CIDR for private network 1. (192.168.2.0/24)

Blue is showing the correct Availability zone (us-east-1b)

Red is showing that we are associated with the correct routing table. (VPC2 -Public-RT)

## Route Table

subnet-057c18508a27a172a / VPC2-Public-SN2

Details | Flow logs | **Route table** | Network ACL | CIDR reservations | Sharing | Tags

You can now check network connectivity with Reachability Analyzer [Run Reachability Analyzer](#)

Route table: [rtb-0efb9e3a9c3e61e48 / VPC2-Public-RT](#) [Edit route table association](#)

Routes (2)	
<input type="text"/> Filter routes	
Destination	Target
192.168.0.0/16	local
0.0.0.0/0	<a href="#">igw-0effc15f10c03f0bd</a>

As you can see the highlighted sections, We

### 3. Route Tables

#### Code to Create Routing Tables and Associations

Code to create public and private routing table for VPC1 (Shared)

```
75
76 # Create Public Route Table
77 resource "aws_route_table" "rt-public" {
78   vpc_id = aws_vpc.vpc-tf.id
79   route {
80     cidr_block = "0.0.0.0/0"
81     gateway_id = aws_internet_gateway.igw.id
82   }
83   tags = merge(
84     var.default_tags,
85     {
86       Name = "VPC-Shared-Public-RT"
87     }
88   )
89 }
90
91 # Create Private Route Table
92 resource "aws_route_table" "rt-private" {
93   vpc_id = aws_vpc.vpc-tf.id
94   route {
95     cidr_block = "0.0.0.0/0"
96     gateway_id = aws_nat_gateway.nat.id
97   }
98   tags = merge(
99     var.default_tags,
100    {
101      Name = "VPC-Shared-Private-RT"
102    }
103  )
104 }
```

Code associating the routing tables to a subnet

```
105  # Create Route Table Association
106  resource "aws_route_table_association" "association-pub" {
107    count          = var.counter
108    subnet_id     = aws_subnet.public.*.id[count.index]
109    route_table_id = aws_route_table.rt-public.id
110  }
111
112  resource "aws_route_table_association" "association-pr" {
113    count          = var.counter
114    subnet_id     = aws_subnet.private.*.id[count.index]
115    route_table_id = aws_route_table.rt-private.id
116  }
```

Code Creating VPC2 (DEV) Routing tables and Associations

```
74
75  # Create Public Route Table
76  ✓ resource "aws_route_table" "rt-public" {
77    vpc_id = aws_vpc.vpc-tf.id
78    ✓ route {
79      cidr_block = "0.0.0.0/0"
80      gateway_id = aws_internet_gateway.igw.id
81    }
82    ✓ tags = merge(
83      var.default_tags,
84      {
85        Name = "${var.prefix}-Public-RT"
86      }
87    )
88  }
89  # Create Private Route Table
90  ✓ resource "aws_route_table" "rt-private" {
91    vpc_id = aws_vpc.vpc-tf.id
92    ✓ route {
93      cidr_block = "0.0.0.0/0"
94      gateway_id = aws_nat_gateway.nat.id
95    }
96    ✓ tags = merge(
97      var.default_tags,
98      {
99        Name = "${var.prefix}-Private-RT"
100      }
101    )
102  }
103
```

Then Create the Route Table associations for VPC2 Public and Private fields

```

104  # Create Route Table Association
105  resource "aws_route_table_association" "association-pub" {
106    count      = var.counter
107    subnet_id  = aws_subnet.public.*.id[count.index]
108    route_table_id = aws_route_table.rt-public.id
109  }
110
111 resource "aws_route_table_association" "association-pr" {
112   count      = var.counter
113   subnet_id  = aws_subnet.private.*.id[count.index]
114   route_table_id = aws_route_table.rt-private.id
115 }
116

```

## Using Console

Search for services, features, blogs, docs, and more [Alt+S]

N. Virginia v vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171 ▾

The screenshot shows the AWS VPC Route Tables page. At the top, there's a search bar and navigation links for services, features, blogs, and docs. The region is set to N. Virginia, and the user is vodlabs/user1598691=Necajev,Luka. Below the header, there's a button to 'Create route table'. The main area displays a table of route tables with the following columns: Select, Name, Route table ID, Explicit subnet associations, Edge associations, Main, VPC, and Owner ID. There are 7 rows listed:

	Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC	Owner ID
<input type="checkbox"/>	VPC2-Public-RT	rtb-0efb9e3a9c3e61e48	2 subnets	—	No	vpc-0526aa1e6447ccb0   VPC...	305139984...
<input type="checkbox"/>	—	rtb-086ac002b1a90090a	—	—	Yes	vpc-0526aa1e6447ccb0   VPC...	305139984...
<input type="checkbox"/>	VPC-Shared-Public-...	rtb-0aa89a494c54bfd64	2 subnets	—	No	vpc-05d3b8371e4a72cae   VP...	305139984...
<input type="checkbox"/>	—	rtb-04251ef9510165f4c	—	—	Yes	vpc-0626e552532f8b7c4	305139984...
<input type="checkbox"/>	VPC2-Private-RT	rtb-09e6f4958cd74e0aa	2 subnets	—	No	vpc-0526aa1e6447ccb0   VPC...	305139984...
<input type="checkbox"/>	VPC-Shared-Private...	rtb-050d3807a95f12092	2 subnets	—	No	vpc-05d3b8371e4a72cae   VP...	305139984...
<input type="checkbox"/>	—	rtb-0c95a688123c0fe85	—	—	Yes	vpc-05d3b8371e4a72cae   VP...	305139984...

## VPC2 Public Routing table

### Route tables (1/7) [Info](#)

Filter route tables

<input type="checkbox"/>	Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC	Owner ID
<input checked="" type="checkbox"/>	VPC2-Public-RT	rtb-0efb9e3a9c3e61e48	2 subnets	—	No	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input type="checkbox"/>	—	rtb-086ac002b1a90090a	—	—	Yes	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input type="checkbox"/>	VPC-Shared-Public-...	rtb-0aa89a494c54bfd64	2 subnets	—	No	vpc-05d3b8371e4a72cae   VP...	305139984...
<input type="checkbox"/>	—	rtb-04251ef9510165f4c	—	—	Yes	vpc-0626e552532f8b7c4	305139984...
<input type="checkbox"/>	VPC2-Private-RT	rtb-09e6f4958cd74e0aa	2 subnets	—	No	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input type="checkbox"/>	VPC-Shared-Private...	rtb-050d3807a95f12092	2 subnets	—	No	vpc-05d3b8371e4a72cae   VP...	305139984...
<input type="checkbox"/>	—	rtb-0c95a688123c0fe85	—	—	Yes	vpc-05d3b8371e4a72cae   VP...	305139984...

rtb-0efb9e3a9c3e61e48 / VPC2-Public-RT

[Details](#) [Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

#### Explicit subnet associations (2)

Find subnet association

Subnet ID	IPv4 CIDR	IPv6 CIDR
subnet-044d6cbc13d242cc8 / VPC2-Public-SN1	192.168.1.0/24	—
subnet-057c18508a27a172a / VPC2-Public-SN2	192.168.2.0/24	—

rtb-0efb9e3a9c3e61e48 / VPC2-Public-RT

Details    Routes    Subnet associations    Edge associations    Route propagat

ⓘ You can now check network connectivity with Reachability Analyzer

### Details

Route table ID	Main
<a href="#">rtb-0efb9e3a9c3e61e48</a>	<input checked="" type="checkbox"/> No
VPC	Owner ID
<a href="#">vpc-0526aa1e6447cccb0   VPC2-test</a>	<input checked="" type="checkbox"/> 305139984171

Showing we are connected to the right VPC (VPC2-test)

VPC2 Private Routing table associations

Route tables (1/7) [Info](#)

Filter route tables

[Actions ▾](#) [Create route table](#)

-	Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC	Owner ID
<input type="checkbox"/>	VPC2-Public-RT	rtb-0efb9e3a9c3e61e48	2 subnets	–	No	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input type="checkbox"/>	–	rtb-086ac002b1a90090a	–	–	Yes	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input type="checkbox"/>	VPC-Shared-Public-...	rtb-0aa89a494c54bfd64	2 subnets	–	No	vpc-05d3b8371e4a72cae   VP...	305139984...
<input type="checkbox"/>	–	rtb-04251ef9510165f4c	–	–	Yes	vpc-0626e552532f8b7c4	305139984...
<input checked="" type="checkbox"/>	VPC2-Private-RT	rtb-09e6f4958cd74e0aa	2 subnets	–	No	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input type="checkbox"/>	VPC-Shared-Private...	rtb-050d3807a95f12092	2 subnets	–	No	vpc-05d3b8371e4a72cae   VP...	305139984...
<input type="checkbox"/>	–	rtb-0c95a688123c0fe85	–	–	Yes	vpc-05d3b8371e4a72cae   VP...	305139984...

rtb-09e6f4958cd74e0aa / VPC2-Private-RT

[Details](#) [Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

[Edit subnet associations](#)

Find subnet association

Subnet ID	IPv4 CIDR	IPv6 CIDR
subnet-035725cb5d4db3bc7 / VPC2-Private-SN1	192.168.3.0/24	–
subnet-0f027487a0801afcc / VPC2-Private-SN2	192.168.4.0/24	–

rtb-09e6f4958cd74e0aa / VPC2-Private-RT

Details    Routes    Subnet associations    Edge associations    Route propagation    Tags

ⓘ You can now check network connectivity with Reachability Analyzer Run Reachability Analyzer X

**Details**

Route table ID <span>rtb-09e6f4958cd74e0aa</span>	Main <span>No</span>	Explicit subnet associations 2 subnets	Edge associations -
VPC <span>vpc-0526aa1e6447cccb0   VPC2-test</span>	Owner ID <span>305139984171</span>		

Showing we are connected to the right VPC (VPC2-test)

VPC1 Public Routing table associations

for services, features, blogs, docs, and more [Alt+S] N. Virginia v vocabs/user1598691=Necajev,Luka @ 3051-3998-4171

### Route tables (1/7) [Info](#)

Filter route tables < 1 > ⚙

-	Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC	Owner ID
<input type="checkbox"/>	VPC2-Public-RT	rtb-0efb9e3a9c3e61e48	2 subnets	-	No	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input type="checkbox"/>	-	rtb-086ac002b1a90090a	-	-	Yes	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input checked="" type="checkbox"/>	VPC-Shared-Public-...	rtb-0aa89a494c54bfd64	2 subnets	-	No	vpc-05d3b8371e4a72cae   VP...	305139984...
<input type="checkbox"/>	-	rtb-04251ef9510165f4c	-	-	Yes	vpc-0626e552532f8b7c4	305139984...
<input type="checkbox"/>	VPC2-Private-RT	rtb-09e6f4958cd74e0aa	2 subnets	-	No	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input type="checkbox"/>	VPC-Shared-Private...	rtb-050d3807a95f12092	2 subnets	-	No	vpc-05d3b8371e4a72cae   VP...	305139984...
<input type="checkbox"/>	-	rtb-0c95a688123c0fe85	-	-	Yes	vpc-05d3b8371e4a72cae   VP...	305139984...

rtb-0aa89a494c54bfd64 / VPC-Shared-Public-RT

Details [Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

#### Explicit subnet associations (2)

Find subnet association < 1 > ⚙

Subnet ID	IPv4 CIDR	IPv6 CIDR
subnet-0f72302662eff87e9 / VPC-Shared-Public-SN2	10.0.2.0/24	-
subnet-069612a78407c9465 / VPC-Shared-Public-SN1	10.0.1.0/24	-

rtb-0aa89a494c54bfd64 / VPC-Shared-Public-RT

Details    Routes    Subnet associations    Edge associations    Route prop

*(i) You can now check network connectivity with Reachability Analyzer*

**Details**

Route table ID	Main
<input checked="" type="checkbox"/> rtb-0aa89a494c54bfd64	<input type="checkbox"/> No
VPC	Owner ID
<a href="#">vpc-05d3b8371e4a72cae</a>   VPC-Shared	<input checked="" type="checkbox"/> 305139984171

Showing we are connected to the right VPC (VPC-Shared)

VPC1 Private Routing table associations

### Route tables (1/7) [Info](#)

Actions [Create route table](#)

<input type="checkbox"/>	Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC	Owner ID
<input type="checkbox"/>	VPC2-Public-RT	rtb-0efb9e3a9c3e61e48	2 subnets	–	No	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input type="checkbox"/>	–	rtb-086ac002b1a90090a	–	–	Yes	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input type="checkbox"/>	VPC-Shared-Public-...	rtb-0aa89a494c54bfd64	2 subnets	–	No	vpc-05d3b8371e4a72cae   VP...	305139984...
<input type="checkbox"/>	–	rtb-04251ef9510165f4c	–	–	Yes	vpc-0626e552532f8b7c4	305139984...
<input type="checkbox"/>	VPC2-Private-RT	rtb-09e6f4958cd74e0aa	2 subnets	–	No	vpc-0526aa1e6447cccb0   VPC...	305139984...
<input checked="" type="checkbox"/>	VPC-Shared-Private...	rtb-050d3807a95f12092	2 subnets	–	No	vpc-05d3b8371e4a72cae   VP...	305139984...
<input type="checkbox"/>	–	rtb-0c95a688123c0fe85	–	–	Yes	vpc-05d3b8371e4a72cae   VP...	305139984...

rtb-050d3807a95f12092 / VPC-Shared-Private-RT

[Details](#) [Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

#### Explicit subnet associations (2)

Edit subnet associations

Subnet ID	IPv4 CIDR	IPv6 CIDR
subnet-0779e9b40c9c2d0f7 / \$VPC-Shared-Private-SN2	10.0.4.0/24	–
subnet-0ffed8f6ca39cfda6 / \$VPC-Shared-Private-SN1	10.0.3.0/24	–

rtb-050d3807a95f12092 / VPC-Shared-Private-RT

Details    Routes    Subnet associations    Edge associations    Route propagation    Tag

*(i) You can now check network connectivity with Reachability Analyzer*

**Details**

Route table ID <a href="#">rtb-050d3807a95f12092</a>	Main <input type="checkbox"/> No	Explicit s 2 subnet
VPC <a href="#">vpc-05d3b8371e4a72cae   VPC-Shared</a>	Owner ID <input type="checkbox"/> 305139984171	

Showing we are connected to the right VPC (VPC-Shared)

#### 4. NAT

Showing Code Creating NATS

```
63 # Create NAT GW
64 resource "aws_nat_gateway" "nat" {
65   allocation_id = aws_eip.nat-eip.id
66   subnet_id     = aws_subnet.public[1].id
67
68   tags = merge(
69     var.default_tags,
70     {
71       Name = "VPC-Shared-NAT-GW"
72     }
73   )
74 }
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173 }
```

## Showing all NATS

The screenshot shows the AWS CloudWatch Metrics interface with the title "Showing all NATS". The top navigation bar includes links for services, features, blogs, docs, and more, along with a search bar and user information. The main content area displays a table titled "NAT gateways (1/2) Info" with one item listed. The table columns are: Name, NAT gateway ID, Connectivity, State, State message, Elastic IP address, and Private IP address. The first row shows a selected NAT gateway named "VPC-Shared-NAT-GW" with ID "nat-05f477143056ef0ed", which is Public and Available. Its Elastic IP is 100.25.127.109 and its Private IP is 10.0.2.205.

Name	NAT gateway ID	Connectivity	State	State message	Elastic IP address	Private IP address
VPC-Shared-NAT-GW	nat-05f477143056ef0ed	Public	Available	-	100.25.127.109	10.0.2.205
VPC2-NAT-GW-VPC2	nat-05b7b519e3d8f6b65	Public	Available	-	54.81.173.17	192.168.1.232

Showwng Nat For VPC1

for services, features, blogs, docs, and more [Alt+S] N. Virginia v vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171 ▾

### NAT gateways (1/2) [Info](#)

Filter NAT gateways

Name	NAT gateway ID	Connectivity type	State	State message	Elastic IP address	Private IP address
VPC-Shared-NAT-GW	nat-05f477143056ef0ed	Public	Available	-	100.25.127.109	10.0.2.205
VPC2-NAT-GW-VPC2	nat-05b7b519e3d8f6b65	Public	Available	-	54.81.173.17	192.168.1.232

nat-05f477143056ef0ed / VPC-Shared-NAT-GW

[Details](#) [Monitoring](#) [Tags](#)

#### Details

NAT gateway ID <a href="#">nat-05f477143056ef0ed</a>	Connectivity type Public	State Available	State message -
Elastic IP address <a href="#">100.25.127.109</a>	Private IP address <a href="#">10.0.2.205</a>	Network interface ID <a href="#">eni-063f7b1bd1e147599</a>	VPC <a href="#">vpc-05d3b8371e4a72cae / VPC-Shared</a>
Subnet <a href="#">subnet-0f72302662eff87e9 / VPC-Shared-Public-SN2</a>	Created <a href="#">2021/12/05 13:41 GMT-5</a>	Deleted -	

Making sure its connected to VPC-Shared

Showing the NAT tags

nat-05f477143056ef0ed / VPC-Shared-NAT-GW

Details    Monitoring    **Tags**

**Tags**    Manage tags

Search tags

Key	Value
Project	FinalProject
Environment	VPC1
Owner	Luka_Owen
Name	VPC-Shared-NAT-GW

**Showing NAT for VPC2**

NAT gateways (1/2) <a href="#">Info</a>						
Name	NAT gateway ID	Connectivity type	State	State message	Elastic IP address	Private IP address
<a href="#">VPC-Shared-NAT-GW</a>	nat-05f477143056ef0ed	Public	<span>Available</span>	-	100.25.127.109	10.0.2.205
<a href="#">VPC2-NAT-GW-VPC2</a>	nat-05b7b519e3d8f6b65	Public	<span>Available</span>	-	54.81.173.17	192.168.1.232

nat-05b7b519e3d8f6b65 / VPC2-NAT-GW-VPC2

[Details](#) [Monitoring](#) [Tags](#)

Details			
NAT gateway ID <a href="#">nat-05b7b519e3d8f6b65</a>	Connectivity type Public	State <span>Available</span>	State message -
Elastic IP address <a href="#">54.81.173.17</a>	Private IP address <a href="#">192.168.1.232</a>	Network interface ID <a href="#">eni-03545d1157e38fee0</a>	VPC <a href="#">vpc-0526aa1e6447ccb0 / VPC2-test</a>
Subnet <a href="#">subnet-044d6cbc13d242cc8 / VPC2-Public-SN1</a>	Created <a href="#">2021/12/05 13:41 GMT-5</a>	Deleted -	

Making sure its connected to VPC2-Test

nat-05b7b519e3d8f6b65 / VPC2-NAT-GW-VPC2

Details    Monitoring    **Tags**

Tags

Manage tags

< 1 >    ⚙

Key	Value
Project	FinalProject
Environment	VPC2
Owner	Owen_Luka
Name	VPC2-NAT-GW-VPC2

Showing the NAT tags

## 5. Internet gateway

Showing Code Creating Internet Gateway

```
19  # Create Internet Gateway
20  resource "aws_internet_gateway" "igw" {
21    vpc_id = aws_vpc.vpc-tf.id
22    tags = merge(
23      var.default_tags,
24      {
25        Name = "${var.prefix}-IGW-VPC2"
26      }
27    )
28 }
```

```
19  # Create Internet Gateway
20  resource "aws_internet_gateway" "igw" {
21    vpc_id = aws_vpc.vpc-tf.id
22    tags = merge(
23      var.default_tags,
24      {
25        Name = "VPC-Shared-IGW"
26      }
27    )
28  }
```

Showing Internet Gateway for VPC1

Search for services, features, blogs, docs, and more [Alt+S]

N. Virginia v vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171

### Internet gateways (1/3) Info

Filter internet gateways

C Actions ▾ Create internet gateway

Name	Internet gateway ID	State	VPC ID	Owner
–	igw-061a6c5a90eff35f4	Attached	vpc-0626e552532f8b7c4	305139984171
<input checked="" type="checkbox"/> VPC-Shared-IGW	igw-0809e94771070e78f	Attached	vpc-05d3b8371e4a72cae   VPC-Shared	305139984171
<input type="checkbox"/> VPC2-IGW-VPN2	igw-0effc15f10c03f0bd	Attached	vpc-0526aa1e6447cccb0   VPC2-test	305139984171

igw-0809e94771070e78f / VPC-Shared-IGW

Details Tags

#### Details

Internet gateway ID igw-0809e94771070e78f	State Attached	VPC ID vpc-05d3b8371e4a72cae   VPC-Shared	Owner 305139984171
--	-------------------	--	-----------------------

Making sure its connected to VPC-Shared

Showing related tags

igw-0809e94771070e78f / VPC-Shared-IGW

Details

Tags

Tags

Manage tags

Search tags

< 1 >

Key	Value
Name	VPC-Shared-IGW
Project	FinalProject
Environment	VPC1
Owner	Luka_Owen

Showing Internet Gateway for VPC2

Search for services, features, blogs, docs, and more [Alt+S]

N. Virginia v vocabs/user1598691=Necajev,Luka @ 3051-3998-4171

### Internet gateways (1/3) [Info](#)

Filter internet gateways

Name	Internet gateway ID	State	VPC ID	Owner
-	igw-061a6c5a90eff35f4	Attached	vpc-0626e552532f8b7c4	305139984171
VPC-Shared-IGW	igw-0809e94771070e78f	Attached	vpc-05d3b8371e4a72cae   VPC-Shared	305139984171
<input checked="" type="checkbox"/> VPC2-IGW-VPC2	igw-0efffc15f10c03f0bd	Attached	vpc-0526aa1e6447cccb0   VPC2-test	305139984171

igw-0efffc15f10c03f0bd / VPC2-IGW-VPC2

[Details](#) [Tags](#)

#### Details

Internet gateway ID <a href="#">igw-0efffc15f10c03f0bd</a>	State Attached	VPC ID vpc-0526aa1e6447cccb0   VPC2-test	Owner 305139984171
---	-------------------	---	-----------------------

Making sure its connected to VPC2-Test

Showing related tags

Tags	
<input type="text"/> Search tags	
Key	Value
Name	VPC2-IGW-VPC2
Owner	Owen_Luka
Environment	VPC2
Project	FinalProject

6. S3 bucket

- a. an object uploaded to the bucket,

features, blogs, docs, and more [Alt+S]

Amazon S3

Account snapshot

Last updated: Dec 4, 2021 by Storage Lens. Metrics are generated every 24 hours. [Learn more](#)

Total storage Object count Avg. object size You can enable advanced metrics in the "default-account-dashboard" configuration.

3.6 MB 6 617.4 KB

Buckets (1) [Info](#)

Buckets are containers for data stored in S3. [Learn more](#)

Find buckets by name

Name	AWS Region	Access	Creation date
tf-luka-owen-final	US East (N. Virginia) us-east-1	Bucket and objects not public	December 5, 2021, 13:27:12 (UTC-05:00)

[View Storage Lens dashboard](#)

[Create bucket](#)

The screenshot shows the AWS S3 console interface. At the top, there's a header bar with various icons and the user's email address: vodlabs/user1598691=NecajevLuka @ 3051-3998-4171. Below the header is a section titled 'Account snapshot' which provides a quick overview of storage usage. It shows 'Total storage' at 3.6 MB, 'Object count' at 6, and 'Avg. object size' at 617.4 KB. A note indicates that advanced metrics can be enabled via a configuration file. The main content area is titled 'Buckets (1)' and shows a single bucket named 'tf-luka-owen-final'. This bucket is located in the 'US East (N. Virginia)' region and was created on December 5, 2021, at 13:27:12 UTC-05:00. The bucket status is listed as 'Bucket and objects not public'. There are buttons for 'Copy ARN', 'Empty', and 'Delete' along with a prominent orange 'Create bucket' button. A search bar at the top allows users to find specific buckets by name.

Adding the images

Services Search for services, features, blogs, docs, and more [Alt+S]

Upload succeeded  
View details below.

## Upload: status

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

### Summary

Destination	Succeeded	Failed
s3://tf-luka-owen-final	2 files, 2.1 MB (100.00%)	0 files, 0 B (0%)

Files and folders Configuration

#### Files and folders (2 Total, 2.1 MB)

Name	Folder	Type	Size	Status	Error
LukaOneCard.jpg	-	image/jpeg	2.1 MB	Succeeded	-
Sully.jpg	-	image/jpeg	38.5 KB	Succeeded	-

Showing the images and RF file

Services  [Alt+S] Global v vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171

Amazon S3 > tf-luka-owen-final

## tf-luka-owen-final Info

Objects Properties Permissions Metrics Management Access Points

### Objects (3)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	<a href="#">LukaOneCard.jpg</a>	jpg	December 5, 2021, 15:19:34 (UTC-05:00)	2.1 MB	Standard
<input type="checkbox"/>	<a href="#">Sully.jpg</a>	jpg	December 5, 2021, 15:19:34 (UTC-05:00)	38.5 KB	Standard
<input type="checkbox"/>	<a href="#">terraform.tfstate</a>	tfstate	December 5, 2021, 13:46:20 (UTC-05:00)	84.8 KB	Standard

**The bucket should be accessible by identities in your account only. The bucket should not be publicly accessible.**

**Block Public Access settings for this bucket**

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

**Block all public access**  
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through *new* access control lists (ACLs)**  
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through *any* access control lists (ACLs)**  
S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through *new* public bucket or access point policies**  
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**  
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

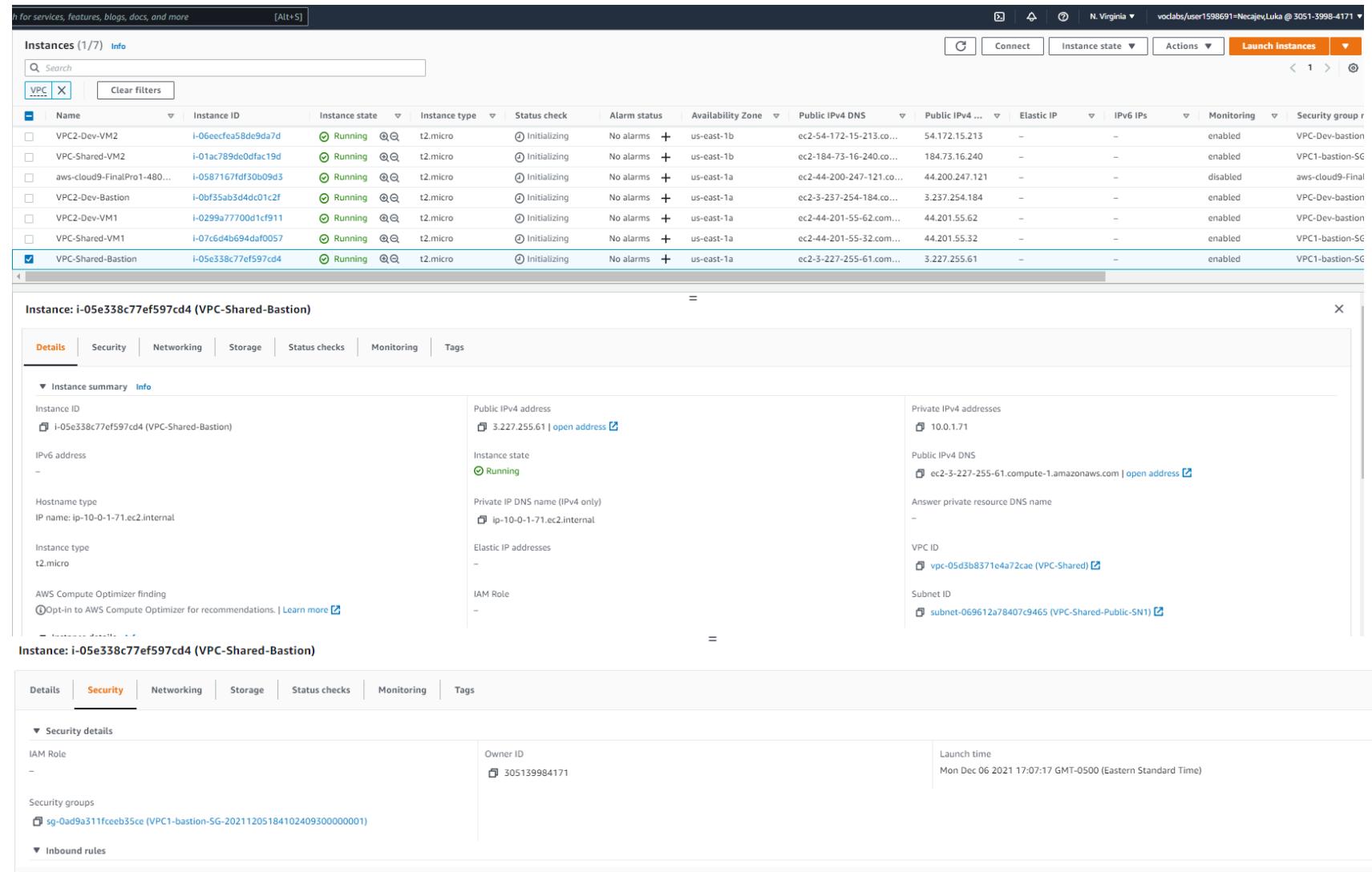
Showing the bucket has all public access blocked

7. **Bastion hosts and VM's should be deployed via Terraform.**

8. **All Instance photo's as proof they were created.**

# VPC1

## VPC-Shared-Bastion



The screenshot shows the AWS CloudWatch Metrics console with the following details:

**Metrics Insights Search:**

- Region: N. Virginia
- User: vocabs/user1598691=Necajev,Luka @ 3051-3998-4171
- Search bar: `for services, features, blogs, docs, and more [Alt+S]`
- Filter: `VPC`
- Actions: `Launch Instances`

**Instances (1/7) Info:**

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs	Monitoring	Security group
VPC2-Dev-VM2	i-06eefcefa58de9da7d	Running	t2.micro	Initializing	No alarms	us-east-1b	ec2-54-172-15-213.co...	54.172.15.213	-	-	enabled	VPC-Dev-bastion
VPC-Shared-VM2	i-01ac789de0dfac19d	Running	t2.micro	Initializing	No alarms	us-east-1b	ec2-184-73-16-240.co...	184.73.16.240	-	-	enabled	VPC1-bastion-SG
aws-cloud9-FinalPro1-480...	i-0587167fd30b09d3	Running	t2.micro	Initializing	No alarms	us-east-1a	ec2-44-200-247-121.co...	44.200.247.121	-	-	disabled	aws-cloud9-final
VPC2-Dev-Bastion	i-0bf35ab3d4dc01c2f	Running	t2.micro	Initializing	No alarms	us-east-1a	ec2-3-237-254-184.co...	3.237.254.184	-	-	enabled	VPC-Dev-bastion
VPC2-Dev-VM1	i-0299aa77700d1cf911	Running	t2.micro	Initializing	No alarms	us-east-1a	ec2-44-201-55-62.com...	44.201.55.62	-	-	enabled	VPC-Dev-bastion
VPC-Shared-VM1	i-07c6d4b6594da0057	Running	t2.micro	Initializing	No alarms	us-east-1a	ec2-44-201-55-32.com...	44.201.55.32	-	-	enabled	VPC1-bastion-SG
<b>VPC-Shared-Bastion</b>	<b>i-05e338c77ef597cd4</b>	<b>Running</b>	<b>t2.micro</b>	<b>Initializing</b>	<b>No alarms</b>	<b>us-east-1a</b>	<b>ec2-3-227-255-61.com...</b>	<b>3.227.255.61</b>	<b>-</b>	<b>-</b>	<b>enabled</b>	<b>VPC1-bastion-SG</b>

**Instance: i-05e338c77ef597cd4 (VPC-Shared-Bastion)**

**Details** | Security | Networking | Storage | Status checks | Monitoring | Tags

**Instance summary** | **Info**

Instance ID <a href="#">i-05e338c77ef597cd4 (VPC-Shared-Bastion)</a>	Public IPv4 address <a href="#">3.227.255.61   open address</a>	Private IPv4 addresses <a href="#">10.0.1.71</a>
IPv6 address -	Instance state <a href="#">Running</a>	Public IPv4 DNS <a href="#">ec2-3-227-255-61.compute-1.amazonaws.com   open address</a>
Hostname type IP name: ip-10-0-1-71.ec2.internal	Private IP DNS name (IPv4 only) <a href="#">ip-10-0-1-71.ec2.internal</a>	Answer private resource DNS name -
Instance type t2.micro	Elastic IP addresses -	VPC ID <a href="#">vpc-05d3b8371e4a72cae (VPC-Shared)</a>
AWS Compute Optimizer finding <a href="#">Opt-in to AWS Compute Optimizer for recommendations.   Learn more</a>	IAM Role -	Subnet ID <a href="#">subnet-069612a78407c9465 (VPC-Shared-Public-SN)</a>

**Instance: i-05e338c77ef597cd4 (VPC-Shared-Bastion)**

**Details** | **Security** | Networking | Storage | Status checks | Monitoring | Tags

**Security details**

IAM Role -	Owner ID <a href="#">305139984171</a>	Launch time Mon Dec 06 2021 17:07:17 GMT-0500 (Eastern Standard Time)
Security groups <a href="#">sg-0ad9a311fceeb35ce (VPC1-bastion-SG-20211205184102409300000001)</a>		

Notice that we are in the correct security group (VPC1-bastion-SG)

## VPC-Shared-VM1

The screenshot shows the AWS CloudWatch Metrics console with the search bar at the top containing the query: "search for services, features, blogs, docs, and more [Alt+5]". Below the search bar is a navigation bar with tabs: Instances (1/7), Info, and a search bar. The main content area displays a table of EC2 instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs	Monitoring	Security group
VPC2-Dev-VM2	i-06ecceaf58de9da7d	Running	t2.micro	Initializing	No alarms	us-east-1b	ec2-54-172-15-213.co...	54.172.15.213	-	-	enabled	VPC-Dev-bastion-
VPC-Shared-VM2	i-01ac789de0dfac19d	Running	t2.micro	Initializing	No alarms	us-east-1b	ec2-184-73-16-240.co...	184.73.16.240	-	-	enabled	VPC1-bastion-SG-
aws-cloud9-FinalPro1-480...	i-0587167fdf30b09d3	Running	t2.micro	Initializing	No alarms	us-east-1a	ec2-44-200-247-121.co...	44.200.247.121	-	-	disabled	aws-cloud9-Final
VPC2-Dev-Bastion	i-0ff55ab3d4dc01c2f	Running	t2.micro	Initializing	No alarms	us-east-1a	ec2-3-237-254-184.co...	3.237.254.184	-	-	enabled	VPC-Dev-bastion-
VPC2-Dev-VM1	i-0299a7700d1cf911	Running	t2.micro	Initializing	No alarms	us-east-1a	ec2-44-201-55-62.com...	44.201.55.62	-	-	enabled	VPC-Dev-bastion-
<b>VPC-Shared-VM1</b>	<b>i-07c6d4b694daf0057</b>	<b>Running</b>	<b>t2.micro</b>	<b>Initializing</b>	<b>No alarms</b>	<b>us-east-1a</b>	<b>ec2-44-201-55-32.com...</b>	<b>44.201.55.32</b>	<b>-</b>	<b>-</b>	<b>enabled</b>	<b>VPC1-bastion-SG-</b>
VPC-Shared-Bastion	i-05e338c77ef597cd4	Running	t2.micro	Initializing	No alarms	us-east-1a	ec2-3-227-255-61.com...	3.227.255.61	-	-	enabled	VPC1-bastion-SG-

Below the table, a detailed view of the selected instance (i-07c6d4b694daf0057) is shown:

**Instance: i-07c6d4b694daf0057 (VPC-Shared-VM1)**

**Details** | Security | Networking | Storage | Status checks | Monitoring | Tags

**Instance summary**

Instance ID i-07c6d4b694daf0057 (VPC-Shared-VM1)	Public IPv4 address 44.201.55.32   open address	Private IPv4 addresses 10.0.3.181
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-44-201-55-32.compute-1.amazonaws.com   open address
Hostname type IP name: ip-10-0-3-181.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-3-181.ec2.internal	Answer private resource DNS name -
Instance type t2.micro	Elastic IP addresses -	VPC ID vpc-05d3b8371e4a72cae (VPC-Shared)
AWS Compute Optimizer finding ①Opt-in to AWS Compute Optimizer for recommendations.   Learn more	IAM Role -	Subnet ID subnet-0fbed8f6ca39cfda6 (\$VPC-Shared-Private-SN1)

## Instance: i-07c6d4b694daf0057 (VPC-Shared-VM1)

X

Details	Security	Networking	Storage	Status checks	Monitoring	Tags
<p>▼ Security details</p>						
IAM Role		Owner ID			Launch time	
-		 305139984171			Mon Dec 06 2021 17:07:17 GMT-0500 (Eastern Standard Time)	
Security groups						
 <a href="#">sg-0404ba6409b6daebb (VPC1-VM-SG)</a>						

VPC-Shared-VM2

for services, features, blogs, docs, and more [Alt+S]

N. Virginia v vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171

Security groups for eni-0cc04146221732336 changed successfully

Instances (1/7) Info

Search VPC X Clear filters

Name Instance ID Instance state Instance type Status check Alarm status Availability Zone Public IPv4 D

VPC2-Dev-VM2	i-06eecfea58de9da7d	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1b	ec2-54-172-1
VPC-Shared-VM2	i-01ac789de0dfac19d	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1b	ec2-184-73-1
aws-cloud9-FinalPro1-480...	i-0587167fdf30b09d3	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1a	ec2-44-200-2
VPC2-Dev-Bastion	i-0bf35ab3d4dc01c2f	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1a	ec2-3-237-25
VPC2-Dev-VM1	i-0299a77700d1cf911	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1a	ec2-44-201-5
VPC-Shared-VM1	i-07c6d4b694daf0057	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1a	ec2-44-201-5
VPC-Shared-Bastion	i-05e338c77ef597cd4	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1a	ec2-3-227-25

### Instance: i-01ac789de0dfac19d (VPC-Shared-VM2)

Details Security Networking Storage Status checks Monitoring Tags

Instance summary Info

Instance ID <a href="#">i-01ac789de0dfac19d (VPC-Shared-VM2)</a>	Public IPv4 address <a href="#">184.73.16.240   open address</a>	Private IPv4 addresses <a href="#">10.0.4.185</a>
IPv6 address -	Instance state <span>Running</span>	Public IPv4 DNS <a href="#">ec2-184-73-16-240.compute-1.amazonaws.com   open address</a>
Hostname type IP name: ip-10-0-4-185.ec2.internal	Private IP DNS name (IPv4 only) <a href="#">ip-10-0-4-185.ec2.internal</a>	Answer private resource DNS name -
Instance type t2.micro	Elastic IP addresses -	VPC ID <a href="#">vpc-05d3b8371e4a72cae (VPC-Shared)</a>
AWS Compute Optimizer finding <a href="#">Opt-in to AWS Compute Optimizer for recommendations.</a>	IAM Role -	Subnet ID <a href="#">subnet-0779e9b40c9c2d0f7 (\$VPC-Shared-Private-SN2)</a>

### Instance: i-01ac789de0dfac19d (VPC-Shared-VM2)

X

#### ▼ Security details

IAM Role

-

Owner ID

 305139984171

Launch time

Mon Dec 06 2021 17:07:17 GMT-0500 (Eastern Standard Time)

Security groups

 [sg-0404ba6409b6daebb \(VPC1-VM-SG\)](#)

#### ▼ Inbound rules

VPC2

VPC-Dev-Bastion

for services, features, blogs, docs, and more [Alt+S]

N. Virginia v vodlabs/user1598691=NecajevLuka @ 3051-3998-4171 ▾

### Instances (1/7) Info

Search Clear filters VPC X

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs	Monitoring	Security group
VPC2-Dev-VM2	i-06eefcea58de9da7d	<span>Running</span>	t2.micro	<span>Initializing</span>	No alarms	+ us-east-1b	ec2-54-172-15-213.co...	54.172.15.213	-	-	enabled	VPC-Dev-bastion-1
VPC-Shared-VM2	i-01ac789de0dfac19d	<span>Running</span>	t2.micro	<span>Initializing</span>	No alarms	+ us-east-1b	ec2-184-73-16-240.co...	184.73.16.240	-	-	enabled	VPC1-bastion-SG-
aws-cloud9-FinalPro1-480...	i-0587167fdf30b09d3	<span>Running</span>	t2.micro	<span>Initializing</span>	No alarms	+ us-east-1a	ec2-44-200-247-121.co...	44.200.247.121	-	-	disabled	aws-cloud9-FinalP
<span>✓</span> VPC2-Dev-Bastion	i-0bf35ab3d4dc01c2f	<span>Running</span>	t2.micro	<span>Initializing</span>	No alarms	+ us-east-1a	ec2-3-237-254-184.co...	3.237.254.184	-	-	enabled	VPC-Dev-bastion-1
VPC2-Dev-VM1	i-0299a77700d1cf911	<span>Running</span>	t2.micro	<span>Initializing</span>	No alarms	+ us-east-1a	ec2-44-201-55-62.co...	44.201.55.62	-	-	enabled	VPC-Dev-bastion-1
VPC-Shared-VM1	i-07d64b694da0057	<span>Running</span>	t2.micro	<span>Initializing</span>	No alarms	+ us-east-1a	ec2-44-201-55-32.co...	44.201.55.32	-	-	enabled	VPC1-bastion-SG-
VPC-Shared-Bastion	i-05e338c77ef597cd4	<span>Running</span>	t2.micro	<span>Initializing</span>	No alarms	+ us-east-1a	ec2-3-227-255-61.co...	3.227.255.61	-	-	enabled	VPC1-bastion-SG-

Instance: i-0bf35ab3d4dc01c2f (VPC2-Dev-Bastion)

Details Security Networking Storage Status checks Monitoring Tags

▼ Instance summary Info

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0bf35ab3d4dc01c2f (VPC2-Dev-Bastion)	3.237.254.184   open address	192.168.1.93
IPv6 address	Instance state	Public IPv4 DNS
-	<span>Running</span>	ec2-3-237-254-184.compute-1.amazonaws.com   open address
Hostname type	Private IP DNS name (IPv4 only)	Answer private resource DNS name
IP name: ip-192-168-1-93.ec2.internal	ip-192-168-1-93.ec2.internal	-
Instance type	Elastic IP addresses	VPC ID
t2.micro	-	vpc-0526aa1e6447cccb0 (VPC2-test)
AWS Compute Optimizer finding	IAM Role	Subnet ID
<span>ⓘ</span> Opt-in to AWS Compute Optimizer for recommendations.   Learn more	-	subnet-044d6cbc13d242cc8 (VPC2-Public-SN1)

▼ Instance details Info

Instance: i-0bf35ab3d4dc01c2f (VPC2-Dev-Bastion)

Details Security Networking Storage Status checks Monitoring Tags

▼ Security details

IAM Role	Owner ID	Launch time
-	305139984171	Mon Dec 06 2021 17:07:17 GMT-0500 (Eastern Standard Time)
Security groups		
<span>sg-02c26e99c67d65b64 (VPC-Dev-bastion-SG-20211205184102633300000002)</span>		

▼ Inbound rules

Filter rules

Security group rule ID	Port range	Protocol	Source	Security groups
sgr-0ba31a6bba4224cae	22	TCP	0.0.0.0/0	VPC-Dev-bastion-SG-20211205184102633300000002

Notice that we are in the correct security group (VPC1-bastion-SG)

## VPC2-Dev-VM1

The screenshot shows the AWS CloudWatch Metrics console. At the top, there is a search bar with placeholder text "Search for services, features, blogs, docs, and more [Alt+S]" and a dropdown menu showing "N. Virginia". Below the search bar, the user "voclabs/user1598691=NecajevLuka @ 3051-3998-4171" is listed. The main area displays a table titled "Instances (1/6) Info" with 6 rows. The first five rows are unselected, and the last row, "VPC2-Dev-VM1", is selected, indicated by a blue border around the entire row. The columns in the table are: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, and Public. The "Status check" column shows "2/2 checks passed" for all instances. The "Alarm status" column shows "No alarms" for all instances. The "Availability Zone" column shows "us-east-1a" for all instances. The "Public IPv4 DNS" and "Public" columns show various AWS IP addresses. Below the table, a modal window titled "Instance: i-0c2705e48b77becb0 (VPC2-Dev-VM1)" is open. It has tabs for Details, Security, Networking, Storage, Status checks, Monitoring, and Tags. The Details tab is selected. Under "Instance summary", there are several sections: Instance ID (i-0c2705e48b77becb0), Public IPv4 address (3.219.240.190), Private IPv4 addresses (192.168.3.155), IPv6 address (-), Instance state (Running), Hostname type (IP name: ip-192-168-3-155.ec2.internal), IP name (ip-192-168-3-155.ec2.internal), Instance type (t2.micro), AWS Compute Optimizer finding (Opt-in to AWS Compute Optimizer for recommendations), IAM Role (-), VPC ID (vpc-00b37bf94e3aff2c4 (VPC2-test)), and Subnet ID (subnet-0097c59c0b77a2263 (VPC2-Private-SN1)). Under "Instance details", there are sections for Platform (Amazon Linux (Inferred)), AMI ID (ami-061ac2e015473fbe2), AMI name (amazon2ami.20211201.0.vRF.64.0.0), Monitoring (detailed), and Termination protection (Disabled).

Showing the security Group

## Instance: i-0c2705e48b77becb0 (VPC2-Dev-VM1)

Details | **Security** | Networking | Storage | Status checks | Monitoring | Tags

▼ Security details

IAM Role -	Owner ID  305139984171	Launch time Wed Dec 08 2021 19:26:07 GMT-0500 (Eastern Standard Time)
Security groups  sg-0bdb907c7c2d885e1 (VPC2-Dev-VM-SG-20211208200557384400000001)		

▼ Inbound rules

Filter rules				
Security group rule ID	Port range	Protocol	Source	Security groups
sgr-0fdb8e691de50ff4d	22	TCP	0.0.0.0/0	VPC2-Dev-VM-SG-20211208200557384400000001
sgr-04dba6de68456e691	All	ICMP	10.0.0.0/16	VPC2-Dev-VM-SG-20211208200557384400000001

▼ Outbound rules

Filter rules				
Security group rule ID	Port range	Protocol	Destination	Security groups
sgr-077981d26762a99ab	All	All	0.0.0.0/0	VPC2-Dev-VM-SG-20211208200557384400000001
sgr-07e1e90b96d32b7ac	All	All	::/0	VPC2-Dev-VM-SG-20211208200557384400000001

## VPC2-Dev-VM2

Search for services, features, blogs, docs, and more [Alt+S]

N. Virginia ▾ vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171 ▾

### Instances (1/6) Info

Search Clear filters

Instance state = running X Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IP
VPC-Shared-VM2	i-08ca23b6e418e6729	<span>Running</span>	t2.micro	<span>2/2 checks passed</span>	No alarms	us-east-1b	ec2-35-171-17-43.com...	35.171.
<span>✓</span> VPC2-Dev-VM2	i-0ed9dfdbf6d18e14c	<span>Running</span>	t2.micro	<span>2/2 checks passed</span>	No alarms	us-east-1b	ec2-34-201-128-159.co...	34.201.
VPC-Shared-Bastion	i-036a94cd1ffedf68d	<span>Running</span>	t2.micro	<span>2/2 checks passed</span>	No alarms	us-east-1a	ec2-44-200-115-102.co...	44.200.
VPC-Shared-VM1	i-061f58704ac7235e6	<span>Running</span>	t2.micro	<span>2/2 checks passed</span>	No alarms	us-east-1a	ec2-3-231-161-237.co...	3.231.1
VPC2-Dev-Bastion	i-01125b778c088b1c9	<span>Running</span>	t2.micro	<span>2/2 checks passed</span>	No alarms	us-east-1a	ec2-3-239-172-241.co...	3.239.1
VPC2-Dev-VM1	i-0c2705e48b77becb0	<span>Running</span>	t2.micro	<span>2/2 checks passed</span>	No alarms	us-east-1a	ec2-3-219-240-190.co...	3.219.2

Instance: i-0ed9dfdbf6d18e14c (VPC2-Dev-VM2)

Details Security Networking Storage Status checks Monitoring Tags

Instance summary Info

Instance ID <a href="#">i-0ed9dfdbf6d18e14c (VPC2-Dev-VM2)</a>	Public IPv4 address <a href="#">34.201.128.159   open address</a>	Private IPv4 addresses <a href="#">192.168.4.129</a>
IPv6 address -	Instance state <span>Running</span>	Public IPv4 DNS <a href="#">ec2-34-201-128-159.compute-1.amazonaws.com   open address</a>
Hostname type IP name: ip-192-168-4-129.ec2.internal	Private IP DNS name (IPv4 only) <a href="#">ip-192-168-4-129.ec2.internal</a>	Answer private resource DNS name -
Instance type t2.micro	Elastic IP addresses -	VPC ID <a href="#">vpc-00b37bf94e3aff2c4 (VPC2-test)</a>
AWS Compute Optimizer finding <small>Opt-in to AWS Compute Optimizer for recommendations.   Learn more</small>	IAM Role -	Subnet ID <a href="#">subnet-0a24cb38b88ff48e6 (VPC2-Private-SN2)</a>

Instance details Info

Platform <a href="#">Amazon Linux (Inferred)</a>	AMI ID <a href="#">ami-061ac2e015473fbe2</a>	Monitoring detailed
---	---	------------------------

Showing the security group for VPC2 VM2

Instance: i-0ed9dfdbf6d18e14c (VPC2-Dev-VM2)

Details | **Security** | Networking | Storage | Status Checks | Monitoring | Tags

▼ Security details

IAM Role	Owner ID	Launch time
-	305139984171	Wed Dec 08 2021 19:26:07 GMT-0500 (Eastern Standard Time)
Security groups		
sg-0bdb907c7c2d885e1 (VPC2-Dev-VM-SG-20211208200557384400000001)		

▼ Inbound rules

Filter rules	<	1	>	
Security group rule ID	Port range	Protocol	Source	Security groups
sgr-0fdbd8e691de50ff4d	22	TCP	0.0.0.0/0	VPC2-Dev-VM-SG-20211208200557384400000001
sgr-04dba6de68456e691	All	ICMP	10.0.0.0/16	VPC2-Dev-VM-SG-20211208200557384400000001

▼ Outbound rules

Filter rules	<	1	>	
Security group rule ID	Port range	Protocol	Destination	Security groups
sgr-077981d26762a99ab	All	All	0.0.0.0/0	VPC2-Dev-VM-SG-20211208200557384400000001
sgr-07e1e90b96d32b7ac	All	All	::/0	VPC2-Dev-VM-SG-20211208200557384400000001

## 9. Security Groups

### VPC-Shared-VM-SG

Inbound rules [Info](#)

inbound rules control the incoming traffic that's allowed to reach the instance.

Security group rule ID	Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>	Source <a href="#">Info</a>	Description - optional <a href="#">Info</a>
sgr-0ac593e46b288915b	All ICMP - IPv4	ICMP	All	Custom	Ping from VM1 to VM2
sgr-0ac3281c7a9c4e6fc	SSH	TCP	22	Custom	Access from bastion to VMs
sgr-0cde206b305a1bda2	All ICMP - IPv4	ICMP	All	Custom	Allowing Access from VM2 CIDR range

[Add rule](#)

[Cancel](#) [Preview changes](#) [Save rules](#)

1. We are allowing ICMP only in the security groups that our VM's in VPC1 are inside. So we are specifying all ICMP to all instances inside of "VPC-Shared-VM-SG"
2. We are allowing SSH Inbound from our bastion host in VPC1. We are doing this by specifying the Bastion hosts security group. If the bastion host had a static IP we would only specify our bastion hosts IP, but since it is not static, we will reference the whole security group.
3. We are allowing all ICMP traffic to come in from the IP address range (CIDR) of our second VPC (Ip == 192.168.0.0/16)

## VPC1-bastion-SG

SWS | Services | Search for services, features, blogs, docs, and more [Alt+S] | N. Virginia | vclabs/user1598691=Necajev,Luka @ 3051-3998-4171 ▾ | ⓘ

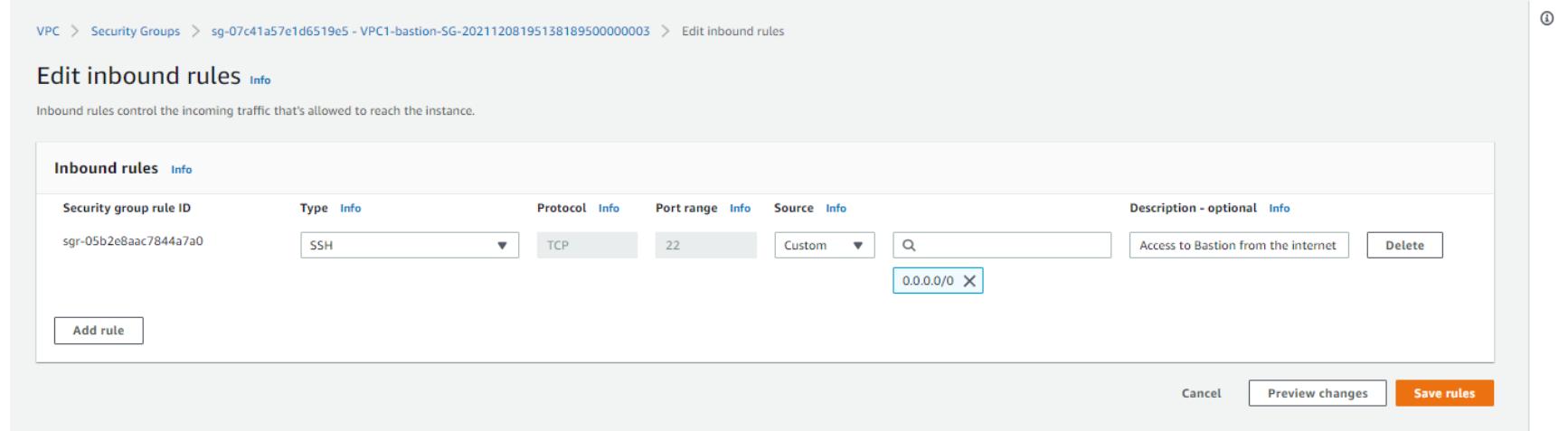
VPC > Security Groups > sg-07c41a57e1d6519e5 - VPC1-bastion-SG-20211208195138189500000003 > Edit inbound rules

## Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules <small>Info</small>					
Security group rule ID	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>
sgr-05b2e8aac7844a7a0	SSH	TCP	22	Custom	Access to Bastion from the internet <small>X</small>
<input type="button" value="Add rule"/>					

Add rule Cancel Preview changes Save rules



1. We are allowing SSH Inbound to our bastion host from anywhere on the internet. So any admin with its ip address can access it.

## VPC2-Dev-VM-SG

The screenshot shows the AWS Management Console interface for managing security groups. The top navigation bar includes 'Services', a search bar, and account information ('N. Virginia' and 'vocabs/user1598691=NecajevLuka @ 3051-3998-4171'). The current page is 'Edit inbound rules' for a specific security group ('sg-0bdb907c7c2d885e1 - VPC2-Dev-VM-SG-20211208200557384400000001').

The main content area displays the 'Inbound rules' table. It has columns for 'Security group rule ID', 'Type', 'Protocol', 'Port range', 'Source', and 'Description - optional'. There are two rows:

- Row 1: Security group rule ID 'sgr-0fdb8e691de50ff4d', Type 'SSH', Protocol 'TCP', Port range '22', Source 'Custom' (0.0.0.0/0), Description 'Access from bastion to VMs'.
- Row 2: Security group rule ID 'sgr-04dba6de68456e691', Type 'All ICMP - IPv4', Protocol 'ICMP', Port range 'All', Source 'Custom' (10.0.0.0/16), Description 'Allowing Access from VM1 CIDR ran'.

At the bottom of the table are buttons for 'Add rule', 'Cancel', 'Preview changes', and 'Save rules'.

1. We are allowing SSH Inbound from our bastion host in VPC2. We are doing this by specifying the Bastion hosts security group. If the bastion host had a static IP we would only specify our bastion hosts IP, but since it is not static, we will reference the whole security group.
2. We are allowing all ICMP traffic to come in from the IP address range (CIDR) of our first VPC (Ip == 10.0.0.0/16)

## VPC2-Dev-bastion-SG

The screenshot shows the AWS Management Console interface for managing VPC security groups. The user is navigating through the 'VPC' > 'Security Groups' > 'sg-0f9eb9be0a29cfbb0 - VPC-Dev-bastion-SG-20211208195137452300000001' > 'Edit inbound rules' section. The 'Inbound rules' table lists one rule:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0d459775409c76aba	SSH	TCP	22	Custom	Access to Bastion from the internet 0.0.0.0/0

At the bottom right of the table are buttons for 'Cancel', 'Preview changes', and a prominent orange 'Save rules' button.

1. We are allowing SSH Inbound to our bastion host from anywhere on the internet. So any admin with its ip address can access it.

## 10. Network connectivity

- a. Admin user should be able to ssh to VPC-Shared-Bastion and through it to VPC-Shared-VM1 and VPC-Shared-VM2.

### SSH into VPC bastion

```
ddd_v1_w_MvbI_882554@runweb44202:~$ ssh -i ~/.ssh/labsuser.pem ec2-user@3.227.255.61
Last login: Mon Dec  6 22:29:05 2021 from ec2-34-220-150-248.us-west-2.compute.amazonaws.com
```

```
 _|_ _|_
_| ( / Amazon Linux 2 AMI
__|_\_|_ |
```

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-1-71 ~]$
[ec2-user@ip-10-0-1-71 ~]$
[ec2-user@ip-10-0-1-71 ~]$
[ec2-user@ip-10-0-1-71 ~]$ █
```

### SSH into VPC-Shared-VM1

Create the key file and place it into our bastion host. Then change the settings so its executable

```
[ec2-user@ip-10-0-1-71 ~]$ nano key.pem
[ec2-user@ip-10-0-1-71 ~]$ sudo chmod 777 key.pem
[ec2-user@ip-10-0-1-71 ~]$ ssh -i key.pem ec2-user@3.227.255.61
Last login: Mon Dec 6 22:32:34 2021 from ec2-34-228-158-248.us-west-2.compute.amazonaws.com
[ec2-user@ip-10-0-1-71 ~]_
[ec2-user@ip-10-0-1-71 ~]_
```

<https://aws.amazon.com/amazon-linux-2/>

This means that we will need port 22 open on the bastion sg to allow outside users to ssh to the bastion

This means that we will need port 22 open on our vm sg to allow bastion host sg in

- b. VPC-Shared-VM1 should be able to ping VPC-Shared-VM2 and vice versa

We will need to allow ICMP to all instances within the VPC1-VM-SG vm security group

The screenshot shows the AWS VPC Security Groups console. The URL in the address bar is [https://console.aws.amazon.com/vpc/home?region=us-east-1#security-groups:group/sg-06a01f91379849eb7/editInboundRules](#). The page title is "Edit inbound rules" for the security group "sg-06a01f91379849eb7".

The table displays two inbound rules:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional	Action
sgr-0ac593e46b288915b	All ICMP - IPv4	ICMP	All	Custom	Ping from VM1 to VM2	Delete
sgr-0ac3281c7a9c4e6fc	SSH	TCP	22	Custom	Access from bastion to VMs	Delete

At the bottom of the table, there is a "Add rule" button. At the bottom right of the page, there are "Cancel", "Preview changes", and "Save rules" buttons.

```
--- 10.0.4.179 ping statistics ---
48 packets transmitted, 48 received, 0% packet loss, time 47989ms
rtt min/avg/max/mdev 0.708/0.881/1.447/0.126 ms
[ec2-user@ip-10-0-3-42 ~]$ ping 10.0.4.179
PING 10.0.4.179 (10.0.4.179) 56(84) bytes of data.
64 bytes from 10.0.4.179: icmp_seq=1 ttl=255 time=0.675 ms
64 bytes from 10.0.4.179: icmp_seq=2 ttl=255 time=0.818 ms
64 bytes from 10.0.4.179: icmp_seq=3 ttl=255 time=0.734 ms
```

- c. **Bonus!** VPC-Shared-VM1 should be able to copy an image from the S3 bucket you created.

**aws s3api get-object --bucket tf-luka-owen-final --key Sully.jpg sully.jpg**

```
ddd_v1_w_MvbI_882554@runweb44484:~$ aws s3api get-object --bucket tf-luka-owen-final --key Sully.jpg sully.jpg
{
  "AcceptRanges": "bytes",
  "LastModified": "2021-12-08T20:17:08+00:00",
  "ContentLength": 39373,
  "ETag": "\"8dbc2ff353c410691977bd542524e328\"",
  "ContentType": "image/jpeg",
  "Metadata": {}
}
ddd_v1_w_MvbI_882554@runweb44484:~$ ls
sully.jpg
ddd_v1_w_MvbI_882554@runweb44484:~$
```

- d. VPC-Shared-VM2 should be able to ping VPC-Dev-VM1; VPC-Shared-VM1 should not be able to ping VPC-Dev-VM1

**Creating the peering connection**

⌚ A VPC peering connection pcx-07d2ba6f209f7802e / finalproject-luka-peering has been requested.

VPC > Peering connections > pcx-07d2ba6f209f7802e

### pcx-07d2ba6f209f7802e / finalproject-luka-peering

Actions ▾

**Pending acceptance**  
You can accept or reject this peering connection request using the 'Actions' menu. You have until Wednesday, December 15, 2021, 19:33:42 EST to accept or reject the request, otherwise it expires.

Details	Info
Requester owner ID <a href="#">305139984171</a>	Acceptor owner ID <a href="#">305139984171</a>
Requester VPC <a href="#">vpc-017241b598cf55ca2 / VPC-Shared</a>	Acceptor VPC <a href="#">vpc-00b37bf94e3aff2c4 / VPC2-test</a>
Requester CIDRs <a href="#">10.0.0.0/16</a>	Acceptor CIDRs –
Requester Region N. Virginia (us-east-1)	Acceptor Region N. Virginia (us-east-1)
Peering connection ID <a href="#">pcx-07d2ba6f209f7802e</a>	
Status ⌚ Pending Acceptance by 305139984171	
Expiration time Wednesday, December 15, 2021, 19:33:42 EST	

ClassicLink DNS Route tables Tags

ClassicLink settings Edit ClassicLink settings

Accepting the connection.

⌚ A VPC peering connection pcx-07d2ba6f209f7802e / finalproject-luka-peering has been requested.

Peering connections N. Virginia v vocabs/user1598691=Necajev,Luka @ 3051-3998-4171 ▾

VPC > Peering connections > pcx-07d2ba6f209f7802e

### pcx-07d2ba6f209f7802e / finalproject-luka-peering

Actions ▾

**Pending acceptance**  
You can accept or reject this peering connection request using the 'Actions' menu. You have until Wednesday, December 15, 2021, 19:33:42 EST to accept or reject the request, otherwise it expires.

Accept request Reject request Edit DNS settings

Allowing DNS to work in our private subnets over peering

The screenshot shows the 'DNS' tab selected in the navigation bar of the AWS VPC ClassicLink interface. Under the 'DNS settings' section, there are two main sections: 'Requester VPC' and 'Acceptor VPC'. Each section contains a status indicator and a 'Disabled' link. A 'Edit DNS settings' button is located in the top right corner of the 'DNS settings' section.

DNS settings	
<b>Requester VPC</b> ( <a href="#">vpc-017241b598cf55ca2 / VPC-Shared</a> ) <a href="#">Info</a>	<b>Acceptor VPC</b> ( <a href="#">vpc-00b37bf94e3aff2c4 / VPC2-test</a> ) <a href="#">Info</a>
Allow accepter VPC to resolve DNS of hosts in requester VPC to private IP addresses	Allow requester VPC to resolve DNS of hosts in accepter VPC to private IP addresses
<a href="#">Disabled</a>	<a href="#">Disabled</a>

**Edit DNS settings**

Accepting both the requester and acceptor DNS name resolution as we do not mind if both users are able to recognize each other by an ip/hostname.

The screenshot shows the AWS VPC Peering Connections DNS settings editor. At the top, there's a navigation bar with 'Services' and a search bar for 'peering connections'. Below it, the breadcrumb trail shows 'VPC > Peering connections > pcx-07d2ba6f209f7802e > Edit DNS settings'. The main title is 'Edit DNS settings' with an 'Info' link. A summary table provides details about the peering connection:

Peering connection ID	Name	Requester VPC	Acceptor VPC
pcx-07d2ba6f209f7802e	finalproject-luka-peering	vpc-017241b598cf55ca2	vpc-00b37bf94e3aff2c4

The 'Edit DNS settings' section contains two main sections: 'Requester DNS resolution' and 'Acceptor DNS resolution'. Under 'Requester DNS resolution', there's a note that if enabled, the DNS hostname of an instance in the requester VPC resolves to its private IP address when queried from instances in the accepter VPC. A checked checkbox allows the accepter VPC to resolve DNS of requester VPC hosts to private IP. Under 'Acceptor DNS resolution', there's a note that if enabled, the DNS hostname of an instance in the accepter VPC resolves to its private IP address when queried from instances in the requester VPC. A checked checkbox allows the requester VPC to resolve DNS of accepter VPC hosts to private IP. A callout box at the bottom left says: 'To use DNS resolution over peering you must enable 'DNS Hostname' on the VPCs involved in peering'. At the bottom right, there are 'Cancel' and 'Save changes' buttons.

**Updating the traffic routes.**

ng connections X | N. Virginia ▾ | vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171 ▾

⌚ Updated routes for rtb-097c417df5da53c42 / VPC-Shared-Public-RT successfully  
▶ Details X ⓘ

VPC > Route tables > rtb-097c417df5da53c42 Actions ▾

## rtb-097c417df5da53c42 / VPC-Shared-Public-RT

You can now check network connectivity with Reachability Analyzer Run Reachability Analyzer X

**Details** Info

Route table ID <a href="#">rtb-097c417df5da53c42</a>	Main <input checked="" type="checkbox"/> No	Explicit subnet associations 2 subnets	Edge associations –
VPC <a href="#">vpc-017241b598cf55ca2</a>   VPC-Shared	Owner ID <a href="#">305139984171</a>		

**Routes** Subnet associations Edge associations Route propagation Tags

**Routes (3)** Edit routes Filter routes Both < 1 > ⚙️

Destination	Target	Status	Propagated
192.168.0.0/16	<a href="#">pcx-07d2ba6f209f780e</a>	Active	No
10.0.0.0/16	local	Active	No
0.0.0.0/0	<a href="#">igw-039b92ab75c733fc9</a>	Active	No

Changing Public VPC1 to allow our data from VPC2

Creating connections X N. Virginia vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171

Updated routes for rtb-06530e7f10dcc398a / VPC-Shared-Private-RT successfully

VPC > Route tables > rtb-06530e7f10dcc398a

## rtb-06530e7f10dcc398a / VPC-Shared-Private-RT

You can now check network connectivity with Reachability Analyzer

Run Reachability Analyzer

**Details** Info

Route table ID rtb-06530e7f10dcc398a	Main No	Explicit subnet associations 2 subnets	Edge associations -
VPC vpc-017241b598cf55ca2   VPC-Shared	Owner ID 305139984171		

Routes Subnet associations Edge associations Route propagation Tags

### Routes (3)

Edit routes

Destination	Target	Status	Propagated
192.168.0.0/16	pxc-07d2ba6f209f7802e	Active	No
10.0.0.0/16	local	Active	No
0.0.0.0/0	nat-04227b2c8a2e153f3	Active	No

Allowing our private routing table access to the other VPC using peering

connections X

Updated routes for rtb-073c35d18c24d7a41 / VPC2-Public-RT successfully

Details X

VPC > Route tables > rtb-073c35d18c24d7a41

rtb-073c35d18c24d7a41 / VPC2-Public-RT Actions ▾

You can now check network connectivity with Reachability Analyzer Run Reachability Analyzer X

Details Info

Route table ID rtb-073c35d18c24d7a41	Main No	Explicit subnet associations 2 subnets	Edge associations -
VPC vpc-00b37bf94e3aff2c4   VPC2-test	Owner ID 305139984171		

Routes Subnet associations Edge associations Route propagation Tags

Routes (3) Edit routes

Filter routes Both < 1 > ⌂

Destination	Target	Status	Propagated
192.168.0.0/16	local	Active	No
10.0.0.0/16	pcx-07d2ba6f209f7802e	Active	No
0.0.0.0/0	igw-078a3382faf8b77f8	Active	No

Allowing our public routing table to send messages back to the Sharing VPC using peering

ing connections X | N. Virginia | vodlabs/user1598691=Necajev,Luka @ 3051-3998-4171 ▾

Updated routes for rtb-0a9ef5223c2585368 / VPC2-Private-RT successfully  
► Details

VPC > Route tables > rtb-0a9ef5223c2585368

### rtb-0a9ef5223c2585368 / VPC2-Private-RT

Actions ▾

You can now check network connectivity with Reachability Analyzer Run Reachability Analyzer X

Details Info

Route table ID rtb-0a9ef5223c2585368	Main No	Explicit subnet associations 2 subnets	Edge associations -
VPC vpc-00b37bf94e3aff2c4   VPC2-test	Owner ID 305139984171		

Routes Subnet associations Edge associations Route propagation Tags

Routes (3)

Filter routes Both < 1 > ⚙

Destination	Target	Status	Propagated
192.168.0.0/16	local	Active	No
10.0.0.0/16	pxx-07d2ba6f209f7802e	Active	No
0.0.0.0/0	nat-04e77b4917428b7b2	Active	No

Allowing our private routing table to send messages back to the Sharing VPC using peering

## Updating Security Groups to allow ICMP messages to VPC2 From VM1 in VPC1

### Security-Group-VPC1

What we had to do here was change the Routing table to allow all items that are coming from the second VPC's(192.168.0.0/16) subnet to our first VPC's subnet(10.0.0.0/16) into our VPC network. To do this we have to add the rule to allow all ICMP packets coming from VPC2's subnet being 192.168.0.0/16.

VPC > Security Groups > sg-06a01f91379849eb7 - VPC-Shared-VM-SG-20211208195138190100000004 > Edit inbound rules

### Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules <small>Info</small>						
Security group rule ID	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>	
sgr-0ac593e46b288915b	All ICMP - IPv4	ICMP	All	Custom	Ping from VM1 to VM2	
sgr-0ac3281c7a9c4e6fc	SSH	TCP	22	Custom	Access from bastion to VMs	
sgr-0cde206b305a1bda2	All ICMP - IPv4	ICMP	All	Custom	Allowing Access from VM2 CIDR	
<small>Add rule</small>						

Add rule

Cancel Preview changes **Save rules**

### Security-Group-VPC2

What we had to do here was change the Routing table to allow all items that are coming from the first VPC's(10.0.0.0/16) subnet to our second VPC's subnet(192.168.0.0/16). To do this we have to add the rule to allow all ICMP packets coming from VPC1's subnet being 10.0.0.0/16.

The screenshot shows the AWS VPC Security Groups interface for editing inbound rules. The top navigation bar includes 'Services', a search bar, and account information ('N. Virginia' and 'voclabs/user1598691=Necajev,Luka @ 3051-3998-4171'). The current path is '/PC > Security Groups > sg-0bdb907c7c2d885e1 - VPC2-Dev-VM-SG-20211208200557384400000001 > Edit inbound rules'. The main section is titled 'Edit inbound rules' with a 'Info' link. A note states: 'Inbound rules control the incoming traffic that's allowed to reach the instance.' Below this is a table titled 'Inbound rules' with an 'Info' link. The table columns are: Security group rule ID, Type (Info), Protocol (Info), Port range (Info), Source (Info), and Description - optional (Info). There are two rows of rules:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0fdbd8e691de50ff4d	SSH	TCP	22	Custom	Access from bastion to VMs 0.0.0.0/0
sgr-04dba6de68456e691	All ICMP - IPv4	ICMP	All	Custom	Allowing Access from VM1 CIDR 10.0.0.0/16

Buttons at the bottom include 'Add rule', 'Cancel', 'Preview changes', and a prominent orange 'Save rules' button.

## Testing ping from VPC1 VM2 to VPC2 VM1

Getting Bastion Host IP

## 8d (VPC-Shared-Bastion)

Networking | Storage | Status checks | Monitoring | Tags

Shared-Bastion)	Public IPv4 address	Private IPv4 addresses
	<a href="#">44.200.115.102   open address</a>	<a href="#">10.0.1.183</a>

Getting VPC1 VM1 IP

## 9c (VPC-Shared-VM1)

Networking | Storage | Status checks | Monitoring | Tags

!d-VM1)	Public IPv4 address	Private IPv4 addresses
	<a href="#">3.231.161.237   open address</a>	<a href="#">10.0.3.42</a>

Getting VPC2 VM1 IP

## 10c (VPC2-Dev-VM2)

Networking | Storage | Status checks | Monitoring | Tags

v-VM2)	Public IPv4 address	Private IPv4 addresses
	<a href="#">34.201.128.159   open address</a>	<a href="#">192.168.4.129</a>

## Getting VPC1 VM2 IP

Instance: i-08ca23b6e418e6729 (VPC-Shared-VM2)

X

The screenshot shows the AWS CloudWatch Metrics interface. A single metric named "AWS Lambda Function Invocations" is displayed. The value is currently at 1000. The chart shows a significant spike starting around January 15, reaching its peak of approximately 1000 on January 20, and then gradually decreasing. The X-axis represents time from January 1 to January 25.

Networking	Storage	Status checks	Monitoring	Tags
<p>You can now check network connectivity with Reachability Analyzer.</p> <p>Run Reachability Analyzer</p>				

**Networking details** Info

Public IPv4 address	Private IPv4 addresses	VPC ID
35.171.17.43   <a href="#">open address</a>	10.0.4.179	vpc-017241b598cf55ca2 (VPC-Shared)

Signing into bastion VM -> Then Signing into VM1 in VPC1 Shared -> Then trying to ping EC2 instance in other VPC

```
ddd_v1_k_MbIT_882554@runweb44416:~$ ssh -i ~/.ssh/labsuser.pem ec2-user@44.208.115.102
The authenticity of host '44.208.115.102 (44.208.115.102)' can't be established.
ECDSA key fingerprint is SHA256:YwCYv/Y2MnCCOfiJLMciCTXxlmDt4v58ZmCSY6Dd1+Y.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '44.208.115.102' (ECDSA) to the list of known hosts.
Last login: wed Dec  8 20:12:08 2021 from ec2-34-228-39-163.us-west-2.compute.amazonaws.com
```



```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-1-183 ~]$ 
[ec2-user@ip-10-0-1-183 ~]$ 
[ec2-user@ip-10-0-1-183 ~]$ 
[ec2-user@ip-10-0-1-183 ~]$ ls
key.pem
[ec2-user@ip-10-0-1-183 ~]$ ssh -i key.pem ec2-user@10.0.3.42
```

```
[ec2-user@ip-10-0-1-183 ~]$ ssh -i key.pem ec2-user@10.0.4.179
The authenticity of host '10.0.4.179 (10.0.4.179)' can't be established.
ECDSA key fingerprint is SHA256:DnxEeC9KwcauZh/z9Nzr2dNH3mrz8ffr2RnhuZ5vA.
ECDSA key fingerprint is MD5:92:5c:4d:aa:ca:fd:ac:cc:2a:01:0c:a4:a5:ac:27:88.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.4.179' (ECDSA) to the list of known hosts.

 _|_ _L_
_| ( / Amazon Linux 2 AMI
__\_\_\_\_|  
  
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-10-0-4-179 ~]$  
[ec2-user@ip-10-0-4-179 ~]$  
[ec2-user@ip-10-0-4-179 ~]$  
[ec2-user@ip-10-0-4-179 ~]$ ping 192.168.4.129  
PING 192.168.4.129 (192.168.4.129) 56(84) bytes of data.  
64 bytes from 192.168.4.129: icmp_seq=1 ttl=255 time=0.486 ms  
64 bytes from 192.168.4.129: icmp_seq=2 ttl=255 time=0.558 ms  
64 bytes from 192.168.4.129: icmp_seq=3 ttl=255 time=0.536 ms  
64 bytes from 192.168.4.129: icmp_seq=4 ttl=255 time=0.551 ms  
^C  
--- 192.168.4.129 ping statistics ---
```

Now we can isolate the other ip's in the NACL to allow and deny specific items in and add the

Logging into the bastion host on VPC1 shared -> logging into VM2 in VPC1 Shared -> pinging VPC2-Dev-VM1

```
ddd_v1_w_MvhI_882554@runweb44437:~$ ssh -i ~/.ssh/labuser.pem ec2-user@44.208.115.182
Last login: Thu Dec  9 01:01:19 2021 from ec2-35-164-126-87.us-west-2.compute.amazonaws.com

  _\|_ _\|_
  \|(_ / Amazon Linux 2 AMI
  __\_\_\_|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-1-183 ~]$ ssh -i key.pem ec2-user@10.0.4.179
Last login: Thu Dec  9 01:26:42 2021 from ip-10-0-1-183.ec2.internal

  _\|_ _\|_
  \|(_ / Amazon Linux 2 AMI
  __\_\_\_|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-4-179 ~]$ ping 192.168.4.129
PING 192.168.4.129 (192.168.4.129) 56(84) bytes of data.
64 bytes from 192.168.4.129: icmp_seq=1 ttl=255 time=0.498 ms
64 bytes from 192.168.4.129: icmp_seq=2 ttl=255 time=0.500 ms
64 bytes from 192.168.4.129: icmp_seq=3 ttl=255 time=0.497 ms
64 bytes from 192.168.4.129: icmp_seq=4 ttl=255 time=0.537 ms
^C
--- 192.168.4.129 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3859ms
rtt min/avg/max/mdev = 0.497/0.500/0.537/0.016 ms
[ec2-user@ip-10-0-4-179 ~]$
```

## Testing ping from VPC1 VM1 to VPC2 VM1

Edit the Inbound NACL (VPC2-DEV-ACL) to not allow anything into the Subnet that is not VPC1 VM2

The screenshot shows the AWS Management Console interface for managing Network ACLs. The top navigation bar includes 'Services' (selected), a search bar, and account information ('N. Virginia' and 'vocabs/user1598691=Necajev,Luka @ 3051-3998-4171'). The current path is 'VPC > Network ACLs > acl-Odb873555381d40e8 / VPC2-DEV-ACL > Edit inbound rules'. The main content area is titled 'Edit inbound rules' with a 'Info' link. A sub-instruction says 'Inbound rules control the incoming traffic that's allowed to reach the VPC.' Below this is a table with columns: Rule number (Info), Type (Info), Protocol (Info), Port range (Info), Source (Info), and Allow/Deny (Info). There are two rows in the table:

Rule number	Type	Protocol	Port range	Source	Allow/Deny
100	All ICMP - IPv4	ICMP (1)	All	10.0.4.178/31	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

Buttons at the bottom include 'Add new rule', 'Sort by rule number', 'Cancel', 'Preview changes', and a prominent orange 'Save changes' button.

Logging out of that instance, logging into VPC VM1 to test if I can ping VPC2 VM1

```
[ec2-user@ip-10-0-1-183 ~]$ ssh -i key.pem ec2-user@10.0.3.42
Last login: Thu Dec  9 01:01:36 2021 from ip-10-0-1-183.ec2.internal
[ec2-user@ip-10-0-1-183 ~]$

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-3-42 ~]$
[ec2-user@ip-10-0-3-42 ~]$
[ec2-user@ip-10-0-3-42 ~]$ ping 192.168.3.155
PING 192.168.3.155 (192.168.3.155) 56(84) bytes of data.
```

There is no response from the VM1 and there will be no response when pinging other VM's

- e. **Bonus!** VPC-Shared-VM1 and VPC-Shared-VM2 should **not** be able to ping VPC-Dev-VM2

Since we do not want them to be able to ping VM2 and we want VM2 to be able to ping VM1, we can just add an outbound rule that would stop the packet from ever exiting VPC1 subnet.

The screenshot shows the AWS Management Console interface for managing Network ACLs. The top navigation bar includes the AWS logo, Services, a search bar, and user information. Below the navigation, the breadcrumb trail indicates the current path: VPC > Network ACLs > acl-014c5644f87bd09b8 / VPC-Shared-ACL > Edit outbound rules. A help icon is located in the top right corner.

**Edit outbound rules** Info

Outbound rules control the outgoing traffic that's allowed to leave the VPC.

Rule number <small>Info</small>	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Destination <small>Info</small>	Allow/Deny <small>Info</small>	Action
90	Custom ICMP - IPv4	All	N/A	192.168.4.129/31	Deny	<button>Remove</button>
100	All traffic	All	All	0.0.0.0/0	Allow	<button>Remove</button>
*	All traffic	All	All	0.0.0.0/0	Deny	<button>Remove</button>

Add new rule Sort by rule number

Cancel Preview changes Save changes

I am making sure that this is denying only the IP I specify which is VM2's IP

```
ddd_v1_w_MvbI_88255@runweb4443:~/~$  
ddd_v1_w_MvbI_88255@runweb44437:~$ ssh -i ~/ssh/labsuser.pem ec2-user@44.208.115.182  
Last login: Thu Dec  9 01:47:33 2021 from ec2-34-228-6-188.us-west-2.compute.amazonaws.com
```

```
 _|_ _L )  
_|( / Amazon Linux 2 AMI  
_|\_\_\_|
```

```
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-10-0-1-183 ~]$ ssh -i key.pem ec2-user@10.0.3.42  
Last login: Thu Dec  9 01:57:01 2021 from ip-10-0-1-183.ec2.internal
```

```
 _|_ _L )  
_|( / Amazon Linux 2 AMI  
_|\_\_\_|
```

```
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-10-0-3-42 ~]$ ping 192.168.4.129  
PING 192.168.4.129 (192.168.4.129) 56(84) bytes of data.  
^C  
--- 192.168.4.129 ping statistics ---  
6 packets transmitted, 0 received, 100% packet loss, time 5103ms
```

```
[ec2-user@ip-10-0-3-42 ~]$  
[ec2-user@ip-10-0-3-42 ~]$ ping 192.168.4.129  
PING 192.168.4.129 (192.168.4.129) 56(84) bytes of data.  
^C  
--- 192.168.4.129 ping statistics ---  
5 packets transmitted, 0 received, 100% packet loss, time 4894ms
```

```
[ec2-user@ip-10-0-3-42 ~]$
```

## Functionality Validation

Please find the table below with the functionality validation artifacts that should be part of your final project's delivery.

Functionality	Artifacts to provide	Comments
---------------	----------------------	----------

TF code building out VPCs, subnets, route tables, GWs, SGs and bastion hosts	Terraform code is deployable, well structured, modular and uses variables  Terraform code should be submitted as a zip file  The Terraform code should use a remote state stored in S3.  There is a clear README.md that explains all the steps for successful deployment	You can use external TF modules  You can deploy part of the required infrastructure via TF for a reduced mark. TF code that cannot be deployed will incur zero mark for this part of the assignment.  README.md <b>must</b> be added to the root of the zip file.
Building out the rest of the infrastructure on the diagram: S3 bucket to store TF remote state  Peering connection  Updated route tables (if needed)  Updated SGs (if needed)	AWS CLI command screenshots (if done with AWS CLI) and AWS Console screenshots clearly showing the infrastructure components have been deployed in <b>your</b> account.	This part can be fully or partially implemented with TF for extra marks.  All the deployed infrastructure should be properly tagged where possible.
Create S3 bucket with to store remote Terraform state.	AWS CLI command screenshots (if done with AWS CLI) and AWS Console screenshots clearly showing the infrastructure components have been deployed in <b>your</b> account.  Screenshot of the S3 bucket resource policy and of the permissions tab clearly showing that the bucket is not publicly accessible. Any other screenshots that prove the bucket is private.	In case you are struggling with the private bucket you can make it public. Mark will be reduced.

<p>Admin users should be able to ssh to PC-Shared-VM1 and PC-Shared-VM2 in the private subnets</p>	<p>Explain the process of logging into Linux EC2 instances running in a private subnet.  Support your explanation with the relevant screenshots from all the steps required to log into PC-Shared-VM1. Compliment the screenshots with a brief explanation.</p>	<p>You can ssh from your laptop or AWS Academy Terminal.  You can use your own ssh key or AWS Academy provided ssh key.</p>
<p>VPC-Shared-VM1 should be able to ping VPC-Shared-VM2 and vice versa</p>	<p>Screenshots of relevant SGs and ping command</p>	
<p><b>Bonus!</b>  VPC-Shared-VM1 should be able to copy an image from the S3 bucket you created</p>	<p>Screenshots of VPC-Shared-VM1 IAM role and AWS CLI command performed on VPC-Shared-VM1 that copies the image  Explanation of:  <ul style="list-style-type: none"> <li>- What kind of permissions were used to access S3 bucket? Why we cannot access this image via browser but we can do it from the VPC-Shared-VM1?</li> <li>- Detailed explanation or diagram of the route that the request made to reach S3 bucket.</li> </ul> </p>	<p>Picture is better than a thousand words.</p>
<p>VPC-Shared-VM2 should be able to ping VPC-Dev-VM1  VPC-Shared-VM2 should <b>not</b> be able to ping VPC-Dev-VM1</p>	<p>Screenshots of relevant SGs, route tables and ping commands</p>	

<b>Bonus!</b>  VPC-Shared-VM1 and VPC-Shared-VM2 should <b>not</b> be able to ping VPC-Dev-VM2	Screenshots of relevant SGs, route tables and ping commands.	Blocking the ping with SG is only a partial solution.  The route tables should not have routing rules that allow traffic flow between VPC-Shared-VM2 and VPC-Dev-VM1
--	--	--

## Grading Guidelines

Deliverable	Points
TF code building out VPCs, subnets, route tables, GWs, SGs and bastion hosts	50

Building out the rest of the infrastructure on the diagram: S3 bucket to store TF remote state  Linux VMs in private subnets  Peering connection  Updated route tables (if needed)  Updated SGs (if needed)	<b>25</b>
Create S3 bucket with an image.	<b>5</b>
Admin users should be able to ssh to PC-Shared-VM1 and PC-Shared-VM2 in the private subnets	<b>5</b>
VPC-Shared-VM1 should be able to ping VPC-Shared-VM2 and vice versa	<b>5</b>
<b>Bonus!</b>  VPC-Shared-VM1 should be able to copy an image from the S3 bucket you created	<b>10</b>
VPC-Shared-VM2 should be able to ping VPC-Dev-VM1  VPC-Shared-VM1 should <b>not</b> be able to ping VPC-Dev-VM1	<b>10</b>
<b>Bonus!</b>  VPC-Shared-VM1 and VPC-Shared-VM2 should <b>not</b> be able to ping VPC-Dev-VM2	<b>10</b>

