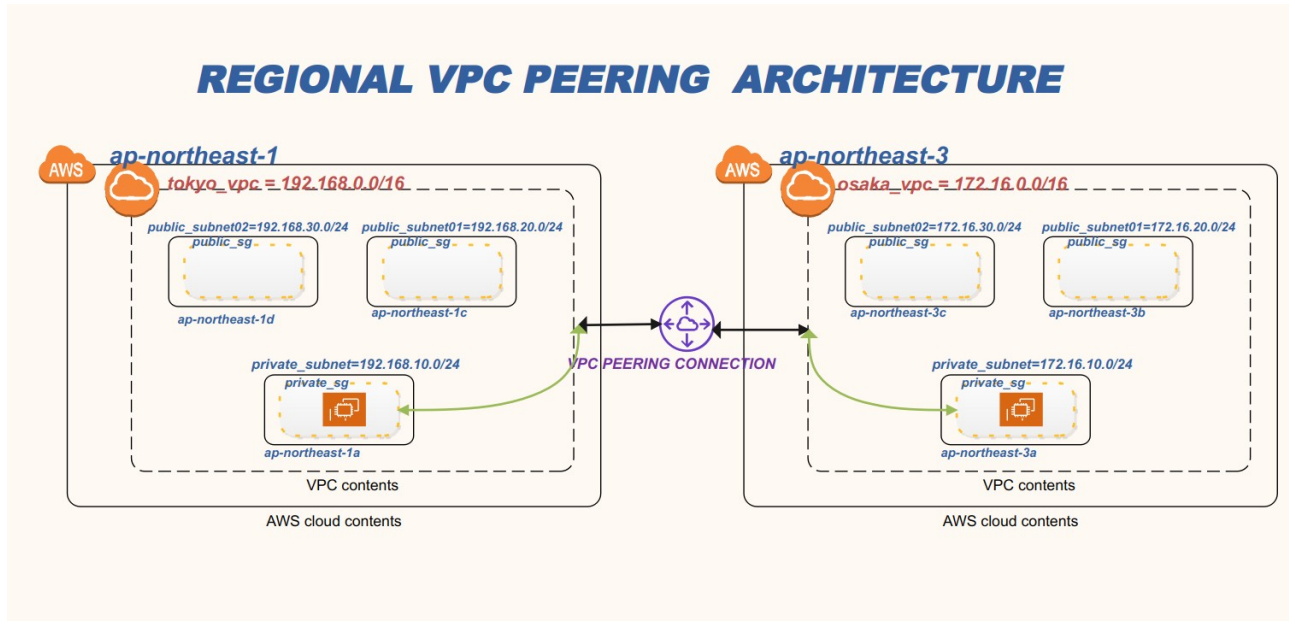


Cross-Regional VPC Peering: A Secure and Efficient Cloud Network

Infrastructure with Terraform

Project Architecture:



Project Descriptions:

- This project will create the architecture above using **Terraform**

Prerequisites:

1. AWS Account
2. AWS ACCESS KEY and SECRET ACCESS KEY with administrator ACCESS
3. Terraform Must be Installed
4. GIT Must be Installed

Instructions:

1. clone project repository :
git clone -b terraform <https://github.com/robudexIT/awsdevopsproject.git>
2. cd to terraform project directory:
cd awsdevopsproject/terraform/vpcpeering/
3. **Run these commands**
 - terraform init
 - terraform validate
 - terraform plan
 - terraform apply -auto-approve

```
(base) robudex@robudex-Dell-System-Vostro-3360:~/SBTPHPROJECTS/awsdevopsproject/terraform/vpcpeering$ terraform init
```

Initializing the backend...

Initializing modules...

- osaka_vpc in vpc
- tokyo_osaka_vpc_peering in vpc_peering
- tokyo_vpc in vpc

Initializing provider plugins...

- Finding hashicorp/aws versions matching "~> 5.0"...
- Installing hashicorp/aws v5.31.0...
- Installed hashicorp/aws v5.31.0 (signed by HashiCorp)

Terraform has created a lock file `.terraform.lock.hcl` to record the provider selections it made above. Include this file in your version control repository so that Terraform can guarantee to make the same selections by default when you run "terraform init" in the future.

```
(base) robudex@robudex-Dell-System-Vostro-3360:~/SBTPHPROJECTS/awsdevopsproject/terraform/vpcpeering$ terraform validate
```

Warning: Reference to undefined provider

on main.tf line 13, in module "osaka_vpc":
13: `aws` = aws.osaka

There is no explicit declaration for local provider name "aws" in module.osaka_vpc, so Terraform is assuming you mean to pass a configuration for "hashicorp/aws".

If you also control the child module, add a `required_providers` entry named "aws" with the source address "hashicorp/aws".

Success! The configuration is valid, but there were some validation warnings as shown above.

```
(base) robudex@robudex-Dell-System-Vostro-3360:~/SBTPHPROJECTS/awsdevopsproject/terraform/vpcpeering$ terraform plan
```

```
module.osaka_vpc.data.aws_availability_zones.available: Reading...
module.tokyo_vpc.data.aws_availability_zones.available: Reading...
module.osaka_vpc.data.aws_availability_zones.available: Read complete after 1s [id=ap-northeast-3]
module.tokyo_vpc.data.aws_availability_zones.available: Read complete after 1s [id=ap-northeast-1]
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create
<= read (data resources)

Terraform will perform the following actions:

```
(base) robudex@robudex-Dell-System-Vostro-3360:~/SBTPHPROJECTS/awsdevopsproject/terraform/vpcpeering$ terraform apply -auto-approve
```

```
module.tokyo_vpc.data.aws_availability_zones.available: Reading...
module.tokyo_vpc.data.aws_availability_zones.available: Read complete after 1s [id=ap-northeast-1]
module.osaka_vpc.data.aws_availability_zones.available: Reading...
module.osaka_vpc.data.aws_availability_zones.available: Read complete after 2s [id=ap-northeast-3]
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create
<= read (data resources)

Terraform will perform the following actions:

Apply complete! Resources: 30 added, 0 changed, 0 destroyed.

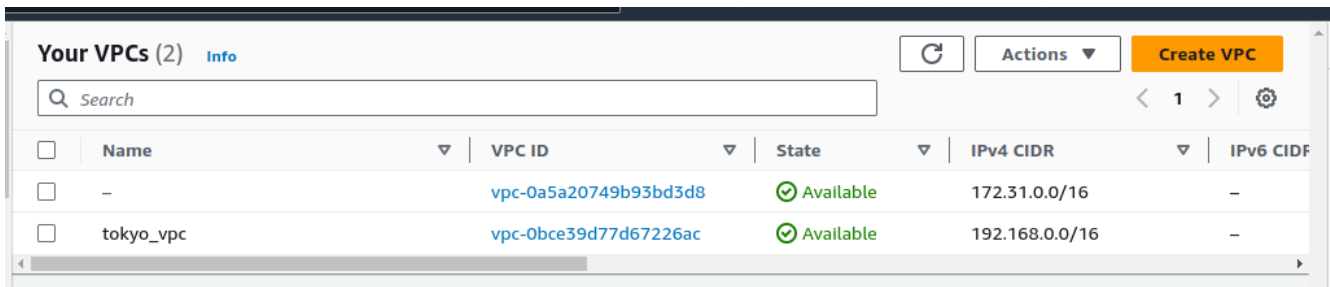
Outputs:

```
osaka_private_route_table_id = "rtb-0f6e1ec3eba283e95"
osaka_private_security_group_id = "sg-04963afd6ae560c35"
osaka_private_subnet_id = "subnet-031a7f11d8a479ea1"
osaka_vpc_id = "vpc-08fa9571f99350bf7"
tokyo_private_route_table_id = "rtb-0e8fd86f1a3938104"
tokyo_private_security_group_id = "sg-0e629a997d839378f"
tokyo_private_subnet_id = "subnet-0d7621ada68e30350"
tokyo_vpc_id = "vpc-0bce39d77d67226ac"
```

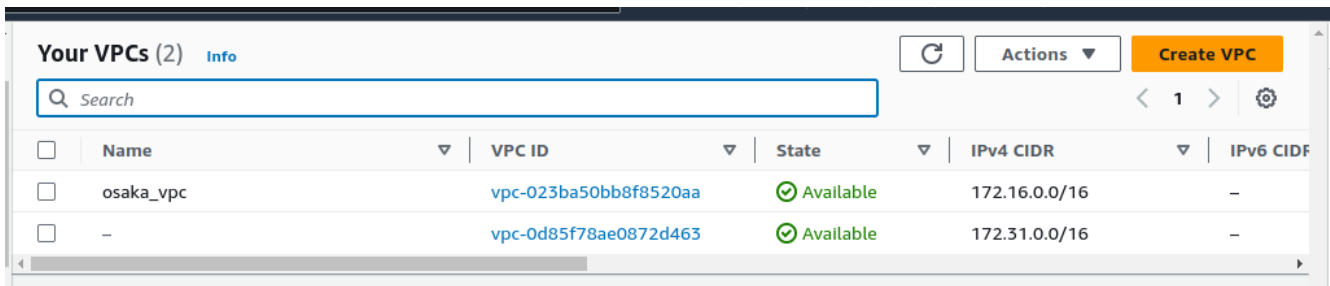
4. When Terraform completes its execution, it creates the following AWS services and resources based on the configurations defined in your Terraform files

- 2 VPCS one for Tokyo Region and one for Osaka Region
- **Each VPC Contains:**
 1. 3 Subnets (Two Public and One Private)
 2. One public routing table with associations to two public subnets.
 3. One public routing table with associations to one subnet.
 4. 2 Security Groups (One Public and One Private)
 5. VPC Connection between two VPCs
 6. Internet Gateway

5. Open the AWS console in two separate tabs. In one tab, navigate to the **Tokyo region**, and in the other tab, navigate to the **Osaka region**. Once there, go to the VPC section in each region and verify if the VPCs have been successfully created.



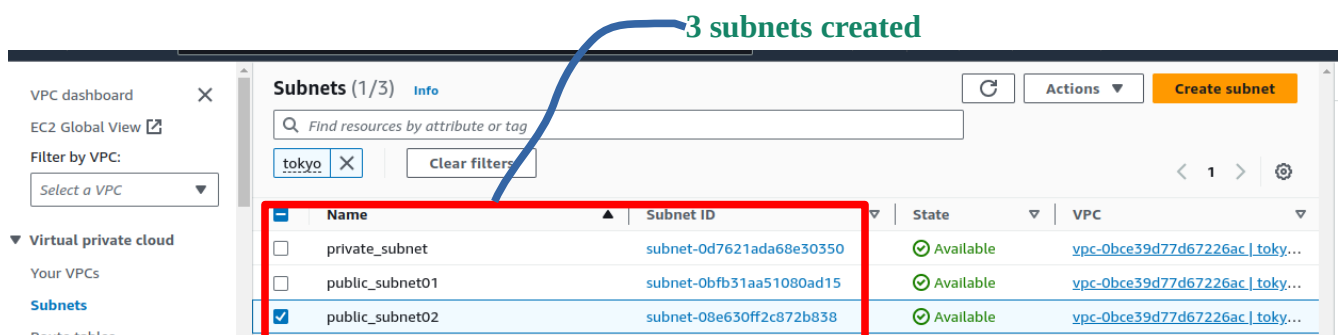
| <input type="checkbox"/> | Name | VPC ID | State | IPv4 CIDR | IPv6 CIDR |
|--------------------------|-----------|-----------------------|-----------|----------------|-----------|
| <input type="checkbox"/> | - | vpc-0a5a20749b93bd3d8 | Available | 172.31.0.0/16 | - |
| <input type="checkbox"/> | tokyo_vpc | vpc-0bce39d77d67226ac | Available | 192.168.0.0/16 | - |



| <input type="checkbox"/> | Name | VPC ID | State | IPv4 CIDR | IPv6 CIDR |
|--------------------------|-----------|-----------------------|-----------|---------------|-----------|
| <input type="checkbox"/> | osaka_vpc | vpc-023ba50bb8f8520aa | Available | 172.16.0.0/16 | - |
| <input type="checkbox"/> | - | vpc-0d85f78ae0872d463 | Available | 172.31.0.0/16 | - |

6. Verify the Following resources in each region.

FOR TOKYO REGION



3 subnets created

| <input checked="" type="checkbox"/> | Name | Subnet ID | State | VPC |
|-------------------------------------|-----------------|--------------------------|-----------|---------------------------------|
| <input type="checkbox"/> | private_subnet | subnet-0d7621ada68e30350 | Available | vpc-0bce39d77d67226ac toky... |
| <input type="checkbox"/> | public_subnet01 | subnet-0bfb31aa51080ad15 | Available | vpc-0bce39d77d67226ac toky... |
| <input checked="" type="checkbox"/> | public_subnet02 | subnet-08e630ff2c872b838 | Available | vpc-0bce39d77d67226ac toky... |

Public and Private routing
table created

Virtual private cloud

Route tables

Route tables (3)

Find resources by attribute or tag

tokyo X Clear filters

| Name | Route table ID | Explicit subnet associati... | Edge associations | Main |
|---|-----------------------|--|-------------------|------|
| <input type="checkbox"/> tokyo_vpc-private_rt | rtb-0e8fd86f1a3938104 | subnet-0d7621ada68e30... | - | No |
| <input type="checkbox"/> - | rtb-032b97fe1280bb5e | - | - | Yes |
| <input type="checkbox"/> tokyo_vpc-public_rt | rtb-025497a7c32874bc | 2 subnets | - | No |

tokyo_vpc-private_rt

rtb-0e8fd86f1a3938104 / tokyo_vpc-private_rt

Details Routes Subnet associations Edge associations Route propagation Tags

Routes (2)

Filter routes

| Destination | Target | Status | Propagated |
|----------------|---------------------------------------|--------|------------|
| 172.16.10.0/24 | pcx-0f966f10693ab3926 | Active | No |
| 192.168.0.0/16 | local | Active | No |

Private Routing Table has Destination route of 172.16.10.0/24 with VPC Peering Connection ID as Target

tokyo_vpc-private_rt

rtb-0e8fd86f1a3938104 / tokyo_vpc-private_rt

Details Routes Subnet associations Edge associations Route propagation Tags

Explicit subnet associations (1)

Find subnet association

| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR |
|----------------|--|-----------------|-----------|
| private subnet | subnet-0d7621ada68e30350 | 192.168.10.0/24 | - |

Subnets without explicit associations (0)

| <input checked="" type="checkbox"/> | Name | Route table ID | Explicit subnet associati... | Edge associations | Main |
|-------------------------------------|---------------------|-----------------------|------------------------------|-------------------|------|
| <input checked="" type="checkbox"/> | tokyo_vpc-public_rt | rtb-025497a7c328474bc | 2 subnets | - | No |

rtb-025497a7c328474bc / tokyo_vpc-public_rt

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

Routes (2)

< 1 >

| Destination | Target | Status | Propagated |
|----------------|---------------------------------------|--------|------------|
| 0.0.0.0/0 | lgw-065060b4e3a648f99 | Active | No |
| 192.168.0.0/16 | local | Active | No |

Public Routing Table with a route to the internet

| <input checked="" type="checkbox"/> | Name | Route table ID | Explicit subnet associati... | Edge associations | Main |
|-------------------------------------|---------------------|-----------------------|------------------------------|-------------------|------|
| <input checked="" type="checkbox"/> | tokyo_vpc-public_rt | rtb-025497a7c328474bc | 2 subnets | - | No |

rtb-025497a7c328474bc / tokyo_vpc-public_rt

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

Explicit subnet associations (2)

< 1 >

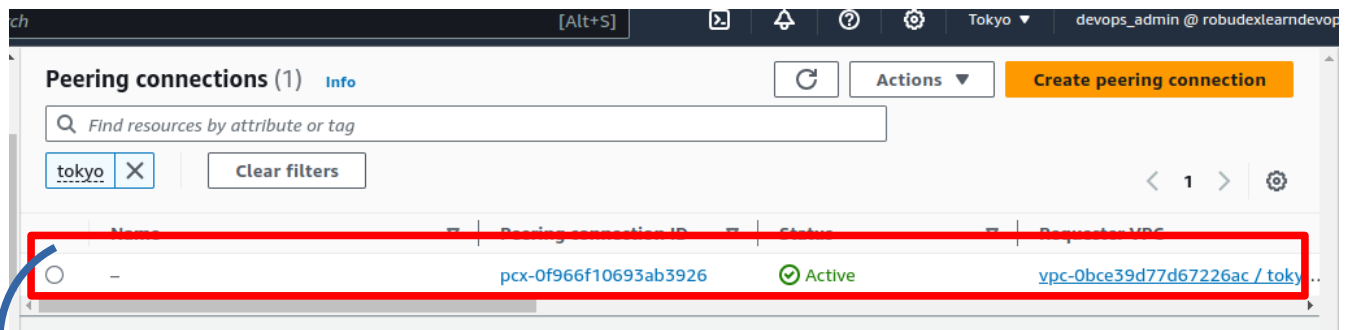
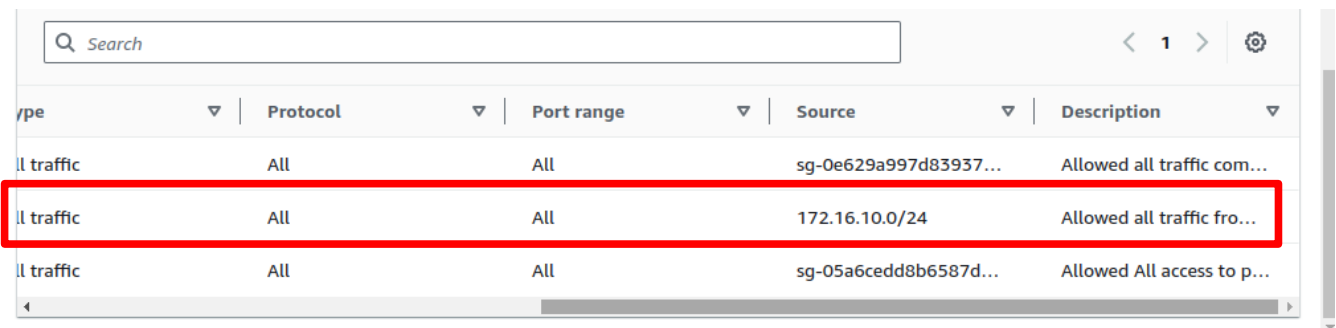
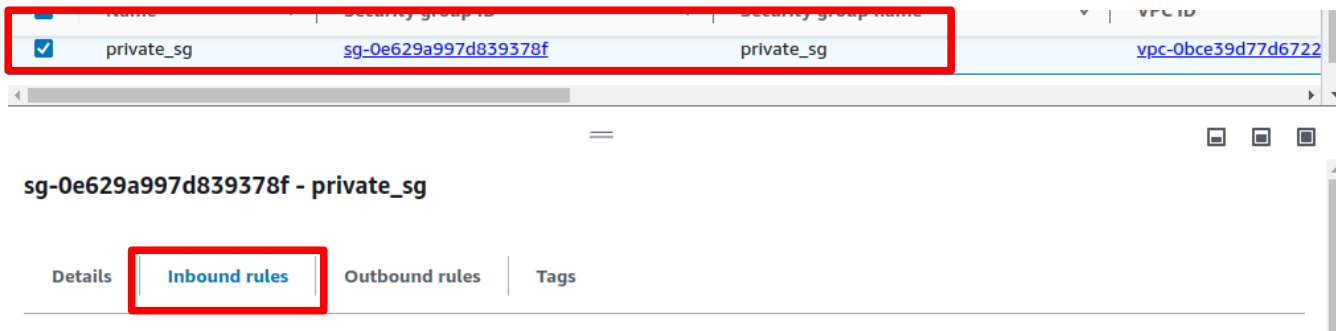
| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR |
|-----------------|--|-----------------|-----------|
| public_subnet01 | subnet-0bfb31aa51080ad15 | 192.168.20.0/24 | - |
| public_subnet02 | subnet-08e630ff2c872b838 | 192.168.30.0/24 | - |

Security Groups (4) [Info](#)

< 1 >

| <input type="checkbox"/> | Name | Security group ID | Security group name | VPC ID |
|--------------------------|------------|--------------------------------------|---------------------|------------------------------------|
| <input type="checkbox"/> | - | sg-0af786ddd6d14eda8 | default | vpc-0bce39d77d6722 |
| <input type="checkbox"/> | public_sg | sg-05a6cedd8b6587d5f | public_sg | vpc-0bce39d77d6722 |
| <input type="checkbox"/> | private_sg | sg-0e629a997d839378f | private_sg | vpc-0bce39d77d6722 |

Select the '**private_sg**,' then click on '**Inbound rules**' to ensure that a rule allowing traffic from the **subnet** in the **OSAKA** region has been added.

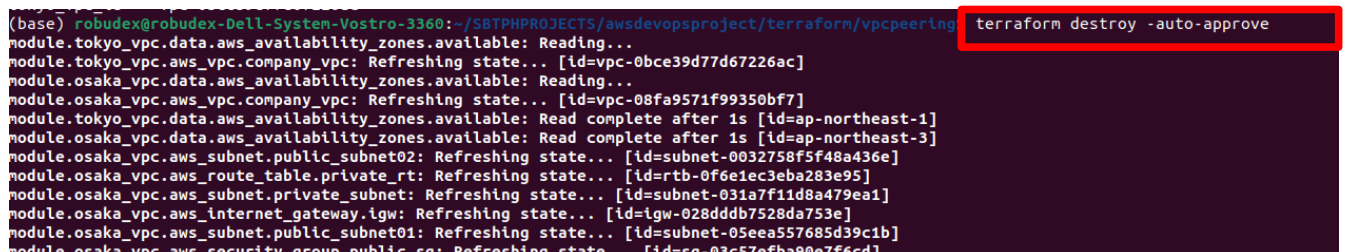


VPC PEERING Successfully created

7. Do the same on the **OSAKA REGION**

8. Once you've done, dont forget to delete your project. Goto the terminal and type the command:

- **terraform destroy -auto-approve**



Warning: Reference to undefined provider

on main.tf line 13, in module "osaka_vpc":
13: aws = aws.osaka

There is no explicit declaration for local provider name "aws" in module.osaka_vpc, so Terraform is assuming you mean to pass a configuration for "hashicorp/aws".

If you also control the child module, add a required_providers entry named "aws" with the source address "hashicorp/aws".

Destroy complete! Resources: 30 destroyed.