

100 Days of DevOps — Day 31-Introduction to VPC Peering

Welcome to Day 31 of 100 Days of DevOps, Focus for today is VPC Peering

What is VPC Peering?

- Let say two VPC want to communicate with each other or share service between them, the best way to do that with the help of VPC Peering
- VPC Peering connection is a networking connection between two VPCs that allow us to route traffic between them using private IPv4 addresses.
- Instances in either VPC can communicate with each other as if they are part of the same network
- AWS uses the existing infrastructure of a VPC to create a VPC peering connection
- It's neither a gateway nor a VPN connection and doesn't rely on a separate piece of physical hardware
- There is no single point of failure or a bandwidth bottleneck i.e bandwidth between instances in peered VPC is no different than bandwidth between instances in the same VPC.



- VPC Peering doesn't support transitive peering i.e VPC1 can talk to VPC 2, VPC 2 can talk to VPC3 but VPC1 can't talk to VPC3. This is because of the security reason so if VPC1 want to communicate with VPC3 we need to establish one more peering connection between VPC1 and VPC3.
- Once VPC Peering is established instance in two VPC can communicate with each other using Private IP(no need to communicate via Internet Gateway)
- Inter-region VPC is supported
- VPC Peering is even supported between two different accounts
- Make sure there is no over-lapping IP between two VPC's

Go to your VPC Dashboard and look for Peering Connections → Create Peering Connection

Create Peering Connection

Peering connection name tag

vpc-peering-test

Select a local VPC to peer with

VPC (Requester)*

vpc-020a5a5f921c421

CIDRs	CIDR	Status	Status Reason
	172.16.0.0/16	associated	

Select another VPC to peer with

Account

My account

Another account

Region

This region (us-west-2)

Another Region

VPC (Acceptor)*

vpc-02d4f46c23024004a

CIDRs	CIDR	Status	Status Reason
	10.0.0.0/16	associated	

- Give some meaningful name to Peering connection name tag(eg: vpc-peering-test)
- Select Requester VPC
- As mentioned in the first part of the series, we can create VPC Peering between different account as well as between different region
- Select Acceptor VPC(As you can see Acceptor VPC has complete different CIDR region, as overlapping CIDR is not supported)

Even I am creating VPC Peering between the same account, I still need to accept peering connection

Name	Peering Connection	Status	Requester VPC	Acceptor VPC	Requester CIDRs	Acceptor CIDRs	Requester Owner	Acceptor Owner
vpc-peering-...	pcc-0945943db12217081	Pending Acceptance	vpc-020a5a5f921c421	vpc-02d4f46c23024004a	172.16.0.0/16		348934551430	348934551430

Create Peering Connection

Requester VPC

Acceptor VPC

Requester CIDRs

Acceptor CIDRs

Requester Owner

Acceptor Owner

- The final step is to update the individual VPC route table with the peering connection

Route Tables > Edit routes

Edit routes

Destination	Target	Status	Propagated
172.16.0.0/16	local	active	No
0.0.0.0/0	igw-0a693a3b01222951a	active	No
10.0.0.0/16	pcc-0945943db12217081	active	No

* Required

Cancel Save routes

Route Tables > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	active	No
0.0.0.0/0	igw-0a693a3b01222951a	active	No
172.16.0.0/16	pcc-0945943db12217081	active	No

* Required

Cancel Save routes

Terraform Code

```
1 resource "aws_vpc_peering_connection" "test-vpc" {
2   peer_vpc_id = "${var.secondary_vpc_id}"
3   vpc_id      = "${var.primary_vpc_id}"
4   auto_accept = true
5
6   tags {
7     Name = "my-vpc-peering"
8   }
9 }
10
11 resource "aws_route" "vpc1-to-vpc2" {
12   route_table_id = "${var.vpc1_route_id}"
13   destination_cidr_block = "${var.vpc2_cidr_block}"
14   vpc_peering_connection_id = "${aws_vpc_peering_connection.test-vpc.id}"
15 }
16
17 resource "aws_route" "vpc2-to-vpc1" {
18   route_table_id = "${var.vpc2_route_id}"
19   destination_cidr_block = "${var.vpc1_cidr_block}"
20   vpc_peering_connection_id = "${aws_vpc_peering_connection.test-vpc.id}"
21 }
```

vpc_peering.tf hosted with ❤ by GitHub

view raw

Looking forward from you guys to join this journey and spend a minimum an hour every day for the next 100 days on DevOps work and post your progress using any of the below medium.

- Twitter: [@100daysofdevops](#) OR [@lakhera2015](#)
- Facebook: <https://www.facebook.com/groups/795382630808645/>
- Medium: <https://medium.com/@devopslearning>
- Slack: <https://devops-myworld.slack.com/messages/CF41EFG49/>
- GitHub Link:<https://github.com/100daysofdevops>

Reference

100 Days of DevOps — Day 0

D-day is just one day away and finally, this is a continuation of the post(I posted a month earlier)


medium.com

 12 

Get an email whenever Prashant Lakhera publishes.

Your email

 Subscribe

By signing up, you will create a Medium account if you don't already have one. Review our [Privacy Policy](#) for more information about our privacy practices.

More from Prashant Lakhera

AWS Community Builder, Ex-Redhat, Author, Blogger, YouTuber, RHCA, RHCHDS, RHCE, Docker Certified,4XAWS, CCNA, MCP, Certified Jenkins, Terraform Certified, 1XGCP

 Follow 

Mar 12, 2019

100 Days of DevOps — Day 30-Introduction to AWS CLI

Welcome to Day 30 of 100 Days of DevOps, Focus for today is AWS CLI

What is AWS CLI? The AWS Command Line Interface (AWS CLI) is a unified tool that provides a consistent interface for interacting with all parts of...

AWS 9 min read



Share your ideas with millions of readers. [Write on Medium](#)

Mar 11, 2019

100 Days of DevOps — Day 29- Introduction to RDS — MySQL

Welcome to Day 29 of 100 Days of DevOps, Focus for today is RDS What is AWS RDS? Amazon Relational Database Service (Amazon RDS) is a web...

AWS 4 min read



Mar 10, 2019

100 Days of DevOps — Day 28- Introduction to VPC Flow Logs

Welcome to Day 28 of 100 Days of DevOps, Focus for today is VPC Flow logs What are VPC Flow logs? It comprised of IP traffic information These...

AWS 4 min read



Mar 9, 2019

100 Days of DevOps — Day 27- Introduction to Packer

Welcome to Day 27 of 100 Days of DevOps, In the last few weeks we focussed on terraform. Let continue our journey and one thing I want to stress in building any Cloud Infrastructure which is really critical is AMI and to automate the process of AMI creation we can...

Docker 9 min read



Mar 8, 2019

100 Days of DevOps — Day 26-Introduction to IAM

Welcome to Day 26 of 100 Days of DevOps. Let's continue our journey with terraform and see how we can terraform IAM. What is IAM? Identity and Access Management(IAM) is used to manage AWS Users Groups Roles Ap...

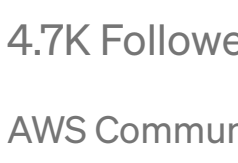
AWS 7 min read



Love podcasts or audiobooks? Learn on the go with our new app. [Try Knowable](#)

[Get started](#) [Sign In](#)

 Search



Prashant Lakhera
4.7K Followers


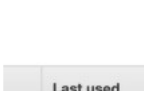
AWS Community Builder, Ex-Redhat, Author, Blogger, YouTuber, RHCA, RHCHDS, RHCE, Docker Certified,4XAWS, CCNA, MCP, Certified Jenkins, Terraform Certified, 1XGCP

 Follow 

More from Medium


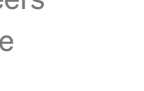
 Apoti.. in Dev G... 

Let's talk about K8S Certifications

 Narendra Reddy 

 Sujit Patel 

Easy Steps to Create AWS VPC with Terraform

 Yamaguchi Hisa.. 

Let's create an EKS resource with Terraform

Help Status Writers Blog Careers Privacy Terms About Knowable