

Deploy 9 - Terraform Challenge

Kenneth Tan

Part 1:

1. Create new VPC with:

- 5 subnets (2 public, 1 private, 2 internal)
- 2 route tables (public & private)
- an Internet Gateway
- and 1 NAT Gateway (in 1 of the private subnets)

Your VPCs (4) Info					
<input type="text" value="Filter VPCs"/>					
<input type="checkbox"/>	Name	VPC ID	State	IPv4 CIDR	
<input type="checkbox"/>	Deploy9 VPC	vpc-08e5690e028dff4f2	Available	10.0.0.0/18	

Subnets (5) Info					
<input type="text" value="Filter subnets"/>					
search: deploy <input type="button" value="X"/> <input type="button" value="Clear filters"/>					
<input type="checkbox"/>	Name	Subnet ID	State	VPC	
<input type="checkbox"/>	Internal02	subnet-0531674c1541b6071	Available	vpc-08e5690e028dff4f2 Deploy9 VPC	
<input type="checkbox"/>	Internal01	subnet-096b6e8a22812a368	Available	vpc-08e5690e028dff4f2 Deploy9 VPC	
<input type="checkbox"/>	Deploy9-pub2	subnet-0a7c635c230ab13fa	Available	vpc-08e5690e028dff4f2 Deploy9 VPC	
<input type="checkbox"/>	Deploy9-pub1	subnet-044e1e71e46efe14a	Available	vpc-08e5690e028dff4f2 Deploy9 VPC	
<input type="checkbox"/>	Deploy9-priv1	subnet-0777009cde821463d	Available	vpc-08e5690e028dff4f2 Deploy9 VPC	

Route tables (3) Info			
<input type="text" value="Filter route tables"/>			
search: deploy9 <input type="button" value="X"/> <input type="button" value="Clear filters"/>			
<input type="checkbox"/>	Name	Route table ID	Explicit subnet associations
<input type="checkbox"/>	Deploy9-priv-rt	rtb-04f021020359a9a11	subnet-0777009cde821463d / Deploy9-priv1
<input type="checkbox"/>	Deploy9-pub-rt	rtb-0a1c1cbcf21a892e5	2 subnets

Internet gateways (1/1) Info			
<input type="text" value="Filter internet gateways"/>			
search: deploy9 X Clear filters			
<input checked="" type="checkbox"/>	Name ▾	Internet gateway ID ▾	State ▾
<input checked="" type="checkbox"/>	Deploy9-Internet-Gateway	igw-0b50cf273655b90a2	Attached
vpc-08e5690e028dff4f2 Deploy9 VPC			

NAT gateways (1/1) Info			
<input type="text" value="Filter NAT gateways"/>			
<input type="radio"/>	Name ▾	NAT gateway ID ▾	Connectivit... ▾
<input checked="" type="radio"/>	Deploy9-NAT-Gateway	nat-01f54ec818ce2f290	Public
Available			

This NAT is in the public subnet and traffic in the private subnet is routed to this. This will allow for the private EC2 to receive updates and maintenance as needed. An elastic IP was also needed for this NAT as shown below.

Elastic IP addresses (1/1)		
<input type="text" value="Filter Elastic IP addresses"/>		
<input checked="" type="checkbox"/>	Name ▾	Allocated IPv4 add... ▾
<input checked="" type="checkbox"/>	–	44.194.81.118
Public IP		

Part 2:

1. Create 1 EC2 instance in the private subnet with:
 - An Ubuntu AMI (version of your choosing)
 - Instance type/size, tags, and other settings of your choosing

EC2 > Instances > i-0edc72497bfea7cb6

Instance summary for i-0edc72497bfea7cb6 (App EC2) [Info](#)

Updated less than a minute ago

Refresh
Connect
Instance state ▼
Actions ▼

Instance ID i-0edc72497bfea7cb6 (App EC2)	Public IPv4 address -	Private IPv4 addresses 10.0.2.147
IPv6 address -	Instance state Running	Public IPv4 DNS -
Hostname type IP name: ip-10-0-2-147.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-2-147.ec2.internal	Answer private resource DNS name -
Instance type t2.micro	Elastic IP addresses -	VPC ID vpc-08e5690e028dff4f2 (Deploy9 VPC) Deploy VPC
AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more	IAM Role -	Subnet ID subnet-0777009cde821463d (Deploy9-priv1) Deploy Subnet

Details
Security
Networking
Storage
Status checks
Monitoring
Tags

▼ Instance details [Info](#)

Platform Ubuntu (Inferred)	AMI ID ami-083654bd07b5da81d	Monitoring disabled
Platform details Linux/UNIX	AMI name ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-20211021	Termination protection Disabled
Launch time Sat Dec 11 2021 08:29:59 GMT-0500 (Eastern Standard Time) (3 minutes)	AMI location 099720109477/ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-20211021	Lifecycle normal

- Create a security group for the EC2 with the following rules:
 - Ingress: allow port 80 traffic from the ALB security group
 - Egress: allow all outbound traffic to any ipv4 address

EC2 > Security Groups > sg-034eeafde37f644cd - Deploy9-EC2-sg

sg-034eeafde37f644cd - Deploy9-EC2-sg [Actions](#)

Details

Security group name Deploy9-EC2-sg	Security group ID sg-034eeafde37f644cd	Description Managed by Terraform	VPC ID vpc-08e5690e028dff4f2 Deploy VPC
Owner 377340475530	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

Inbound rules
Outbound rules
Tags

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

Inbound rules (1/1)

Refresh Manage tags Edit inbound rules

< 1 > [Filter](#)

<input checked="" type="checkbox"/>	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
<input checked="" type="checkbox"/>	-	sg-03476d5fa5b09d5e9	-	HTTP	TCP	80	sg-09162aea2f51efd99 / alb-sg	-

EC2 > Security Groups > sg-034eeafde37f644cd - Deploy9-EC2-sg

sg-034eeafde37f644cd - Deploy9-EC2-sg Actions

Details

Security group name Deploy9-EC2-sg	Security group ID sg-034eeafde37f644cd	Description Managed by Terraform	VPC ID vpc-08e5690e028dff4f2
Owner 377340475530	Inbound rules count 1 Permission entry	Outbound rules count 2 Permission entries	

Inbound rules
Outbound rules
Tags

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

Outbound rules (2)
Manage tags
Edit outbound rules

< 1 >

<input type="checkbox"/>	Name	Security group rule...	IP version	Type	Protocol	Port range	Destination	Description
<input type="checkbox"/>	-	sgr-0c04e4d5fb4f40fba	-	PostgreSQL	TCP	5432	sg-09eb741dc87ef3f09 / postgres-sg	-
<input type="checkbox"/>	-	sgr-046e2afcc540f0e17	IPv4	All traffic	All	All	0.0.0.0/0	-

Part 3:

1. Create 1 ALB in the 2 public subnets

<input type="checkbox"/>	Name	DNS name	State	VPC ID	Availability Zones	Type
<input type="checkbox"/>	deploy9-lb	deploy9-lb-2136201945.us-e...	Active	vpc-08e5690e028dff4f2	us-east-1a, us-east-1b	application

2. Create a security group for the ALB with the following rules:

- Ingress: allows only port 80 inbound traffic from any ipv4 address
- Egress: allow only port 80 outbound traffic to the EC2 security group

EC2 > Security Groups > sg-09162aea2f51efd99 - alb-sg

sg-09162aea2f51efd99 - alb-sg Actions

Details

Security group name alb-sg	Security group ID sg-09162aea2f51efd99	Description Allow web traffic into app ec2	VPC ID vpc-08e5690e028dff4f2
Owner 377340475530	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

Inbound rules
Outbound rules
Tags

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

Inbound rules (1/1)
Manage tags
Edit inbound rules

< 1 >

<input checked="" type="checkbox"/>	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
<input checked="" type="checkbox"/>	-	sgr-057d3a2a5b5dcb60c	IPv4	HTTP	TCP	80	0.0.0.0/0	-

EC2 > Security Groups > sg-09162aea2f51efd99 - alb-sg

sg-09162aea2f51efd99 - alb-sg Actions ▾

Details

Security group name alb-sg	Security group ID sg-09162aea2f51efd99	Description Allow web traffic into app ec2	VPC ID vpc-08e5690e028dff4f2
Owner 377340475530	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

Inbound rules
Outbound rules
Tags

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

Outbound rules (1/1) Manage tags Edit outbound rules

< 1 >

<input checked="" type="checkbox"/>	Name	Security group rule...	IP version	Type	Protocol	Port range	Destination	Description
<input checked="" type="checkbox"/>	-	sg-0ab453290fd610a9a	-	HTTP	TCP	80	sg-034eeafde37f644c...	-

3. Create a target group and add the EC2 instance to the group

EC2 > Target groups

Target groups (1) Info Create target group

< 1 >

<input type="checkbox"/>	Name	ARN	Port	Protocol	Target type	Load balancer	VPC ID
<input type="checkbox"/>	deploy9-target-group	arn:aws:elasticloadbalancing:us...	80	HTTP	Instance	deploy9-lb	vpc-08e5690e028dff4f2

4. Create an ALB listener that forwards traffic to the target group

Load balancer: **deploy9-lb**

Description
Listeners
Monitoring
Integrated services
Tags

Listeners listen for connection requests using their protocol and port. You can add, remove, or update listeners and listener rules.

To view and edit listener attributes, select the listener and choose Edit.



Add listener
Edit
Delete

<input type="checkbox"/>	Listener ID	Security policy	SSL Certificate	Rules
<input type="checkbox"/>	HTTP : 80 arn...8d97879287232f33 ▾	N/A	N/A	Default: forwarding to deploy9-target-group View/edit rules

Part 4:

1. Create 1 PostgreSQL RDS instance
 - Make it multi-az

- Name, instance type/size, tags, db username/password, and other settings of your choosing

Instance		
Configuration DB instance ID terraform-20211211132959922700000001 Engine version 9.6.20 DB name deploy9db License model Postgresql License Option groups default:postgres-9-6  In sync Amazon Resource Name (ARN) arn:aws:rds:us-east-1:377340475530:db:terraform-20211211132959922700000001 Resource ID db-IIBVYZEXQY2W7267TKCQRBL4JE Created time Sat Dec 11 2021 08:34:30 GMT-0500 (Eastern Standard Time) Parameter group default.postgres9.6  In sync Deletion protection Disabled	Instance class Instance class db.t2.micro vCPU 1 RAM 1 GB Availability Master username deploy9 IAM DB authentication Not enabled Multi-AZ Yes Secondary Zone us-east-1b	Storage Encryption Not enabled Storage type General Purpose SSD (gp2) Provisioned IOPS - Storage 20 GiB Storage autoscaling Disabled

2. Create a security group for the RDS with the following rule:
 - Ingress: allow traffic to its port from the EC2 security group

EC2 > Security Groups > sg-09eb741dc87ef3f09 - postgres-sg

sg-09eb741dc87ef3f09 - postgres-sg Actions ▾

Details

Security group name postgres-sg	Security group ID sg-09eb741dc87ef3f09	Description Managed by Terraform	VPC ID vpc-08e5690e028dff4f2 🔗
Owner 377340475530	Inbound rules count 1 Permission entry	Outbound rules count 0 Permission entries	

Inbound rules
Outbound rules
Tags

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

Inbound rules (1/1)
🔄
Manage tags
Edit inbound rules

< 1 > ⚙️

<input checked="" type="checkbox"/>	Name ▾	Security group rule... ▾	IP version ▾	Type ▾	Protocol ▾	Port range ▾	Source ▾	Description
<input checked="" type="checkbox"/>	-	sgr-0b36c5d2a44719b...	-	PostgreSQL	TCP	5432	sg-034eeafde37f64cd / Deploy9-EC2-sg	-

3. Create a DB subnet group for the RDS consisting of the 2 internal subnets

RDS > Subnet groups > deploy9-subnet-group

deploy9-subnet-group

Subnet group details

VPC ID vpc-08e5690e028dff4f2
ARN arn:aws:rds:us-east-1:377340475530:subgrp:deploy9-subnet-group
Description Managed by Terraform

Subnets (2)

Availability zone	Subnet ID	CIDR block
us-east-1b	subnet-0531674c1541b6071	10.0.6.0/24
us-east-1a	subnet-096b6e8a22812a368	10.0.5.0/24

Tags (3)
Manage tags

Key	Value
Deployment	DEPLOYMENT_09_TERRAFORM
Team	Kura Labs
Name	Deploy9 DB subnet group

Challenges:

One of the interesting challenges I ran into when creating the infrastructure with terraform was needing to work around cycling issues when creating security groups. Ultimately, the security groups were to only take traffic from each other but when doing so, I ran into a cycling error. This cycling error came up when trying to create 2 resources that depended on the existence of the other. The work around I found was using the `aws_security_group_rule` resource that was able to add on a rule for either inbound or outbound traffic.

Another challenge I faced was creating routing tables. Association of the tables was a completely separate function that needed to be called in the form of `aws_subnet_association`.