Subnets

1. Now we are going to create our subnets!
   1. First up, we create the public subnet:

resource “aws\_subnet” “public” {

vpc\_id = “${aws\_vpc.vpc.id}”

cidr\_block = “10.1.1.0/24”

map\_public\_ip\_on\_launch = true

availability\_zone = “us-east-1d”

tags {

Name = “public”

}

}

* 1. Here we have created the subnet, given it the id of “public” and attached it to our vpc. After that, we assigned it a CIDR block of 10.1.1.0/24. This is a subdivision of our VPC CIDR block of 10.1.0.0/16. This block provides 252 addresses to be used by public instances. After that, we instruct the subnet to map public IPs to instances assigned to it and specify the us-east-1d availability zone. You can also use a variable here, if you wish, to specify your availability zones. In multi-region environments, it is important to do so. After that, I tag the subnet with a Name of “public”.
  2. Now we will create the private subnets. I’m only going to walk through one of them to save time.

resource “aws\_subnet” “private1” {

vpc\_id = “${aws\_vpc.vpc.id}”

cidr\_block = “10.1.2.0/24”

map\_public\_ip\_on\_launch = false

availability\_zone = “us-east-1a”

tags {

Name = “private1”

}

}

* 1. As you can see, this is mostly the same, with a few differences. One, we have obviously changed the id and the tag to “private1”. Also, you will notice the map public ip on launch is false. We definitely don’t want our instances receiving public IP addresses. Also, we have assigned it a different availability zone.
  2. Ok, I have now added the other private subnets and the RDS subnets. You can see how I have distributed the Availability zones to ensure the resources we deploy are distributed properly to ensure maximum resiliency in case of an outage.

1. Ok! Now that our subnets are created, join me in the next lesson and we will continue with the VPC!