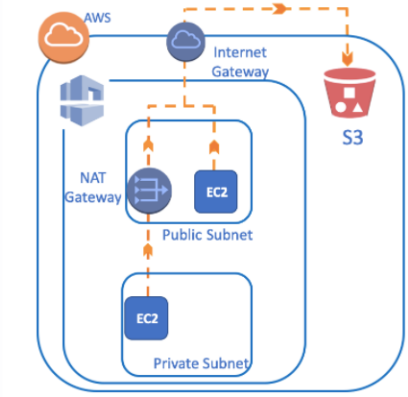
Creating Endpoint

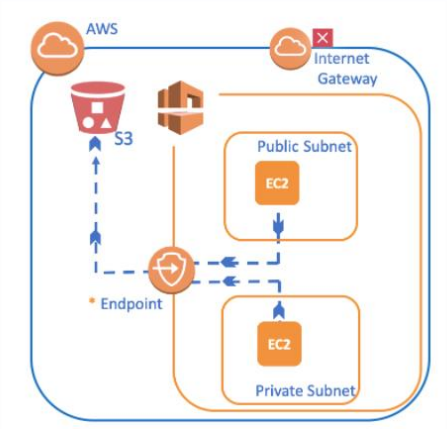
Connecting from Instance to AWS services via IGW



Connecting from Instance to AWS services via IGW

- As shown in the figure above, normally, when our instances in VPC need to connect to any AWS service, for example, S3 ;
  - An instance in a public subnet first reaches the Internet via the **Internet Gateway(IGW)** and then reaches S3 by connecting AWS environment from the outside.
  - As for an instance in the Private Subnet, it should first connect to **NAT Gateway/Instance** in the Public Subnet, from there to the Internet, and then back to S3 from the outside.
- In fact, both S3 and instance may be sharing the same physical environment in the AWS environment. Nevertheless, instance first establish an internet connectivity then reach S3 from the public internet.

Connecting from Instance to AWS services via Endpoint



Connecting from Instance to AWS services via Endpoint

Thanks to Endpoint, an instance in AWS can no longer be required to use Internet Gateway or NAT Instance. It allows you to make the connection **via the network of AWS**, not through the internet.

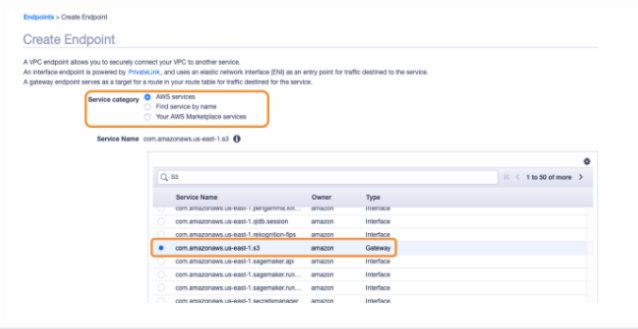
In this way, **both Public Instance and Private Instance** can directly connect to the S3 via Endpoint as seen in the figure above.

So, it's faster, more easy and secure. Let's create a new Endpoint on AWS console.

Creating an Endpoint-1

Endpoint

First, click the **Endpoint** section from the left-hand menu on the VPC dashboard as seen above and Then click **Create Endpoint**.



Create Endpoint

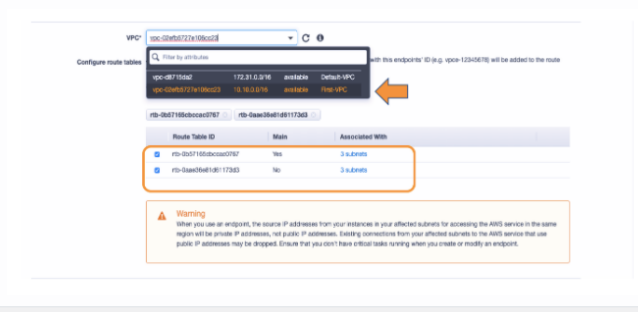
- **Service Category:** we have 3 options here;
  - AWS Services: It means AWSs own services. For example S3.
  - Find Service by Name: If you want to connect the services from **another AWS account** you'll choose this option
  - Your AWS Marketplace Services: Here, you can select third-party applications.

We'll continue with AWS Services.

- **Service Name:**

Now we'll select the AWS service that endpoint will connect to. Since we will use AWS services for S3, we select com.amazonaws.us-east-1.s3 as a gateway from the list.

Creating an Endpoint-2



Creating an Endpoint

- **VPC:**

We select **First-VPC** as target a VPC that Endpoint will connect.

- **Configure Route Tables:**

We determine Route Tables here. Thus, subnets associated with these selected route tables will be able to access this endpoint. In fact, we select indirectly the target subnets. And also these selected Route Tables automatically will be updated for the Endpoint.

Since we want our private and public instances to access S3, we choose Route Tables of **both public and private subnets**.

Creating an Endpoint-3

Creating an Endpoint

- **Policy:**

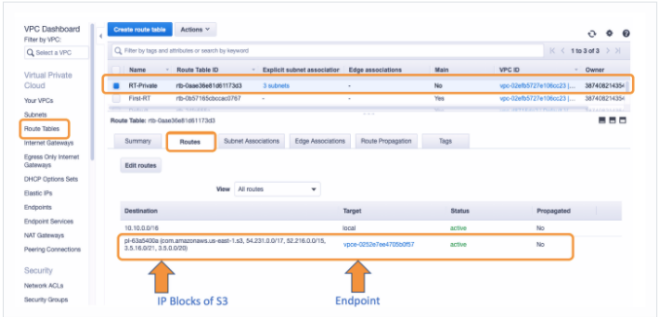
We can allow **Full Access**, or create a **Custom** policy here. But, we don't need to create a new policy, so let's go on with **Full Access** option for now.

- **Add Tags:**

Here we can enter **Name** for Key and **First-Endpoint** for value.

Then, click **Create Endpoint** tab. Your Endpoint is ready.

Checking the Route Tables



Checking the Route Tables

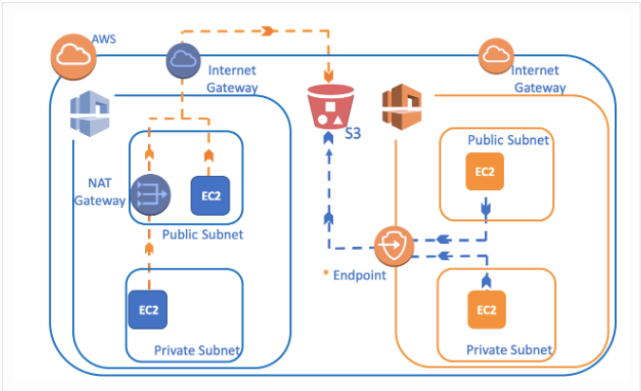
Let's check what happened to our selected Route Tables.

- So, click the **Route Tables** section from the left-hand menu on the VPC dashboard as seen above
- Then click **Routes** while Route Table of First-VPC's Private Subnets is selected,
- As you see in the picture below, a **new route** has been created in the Routes section.
- In the created new route,
  - **IP blocks of S3** are seen as **Destination**,
  - **Endpoint** that we created is seen as a **Target**.

This means that packages that will go to S3 will be delivered to Endpoint, not to the Intenet gateway anymore.

You'll see the same new route if you check the public subnet's Route Table also.

Conclusion: Endpoint vs. Internet Gateway



Endpoint vs. Internet Gateway