



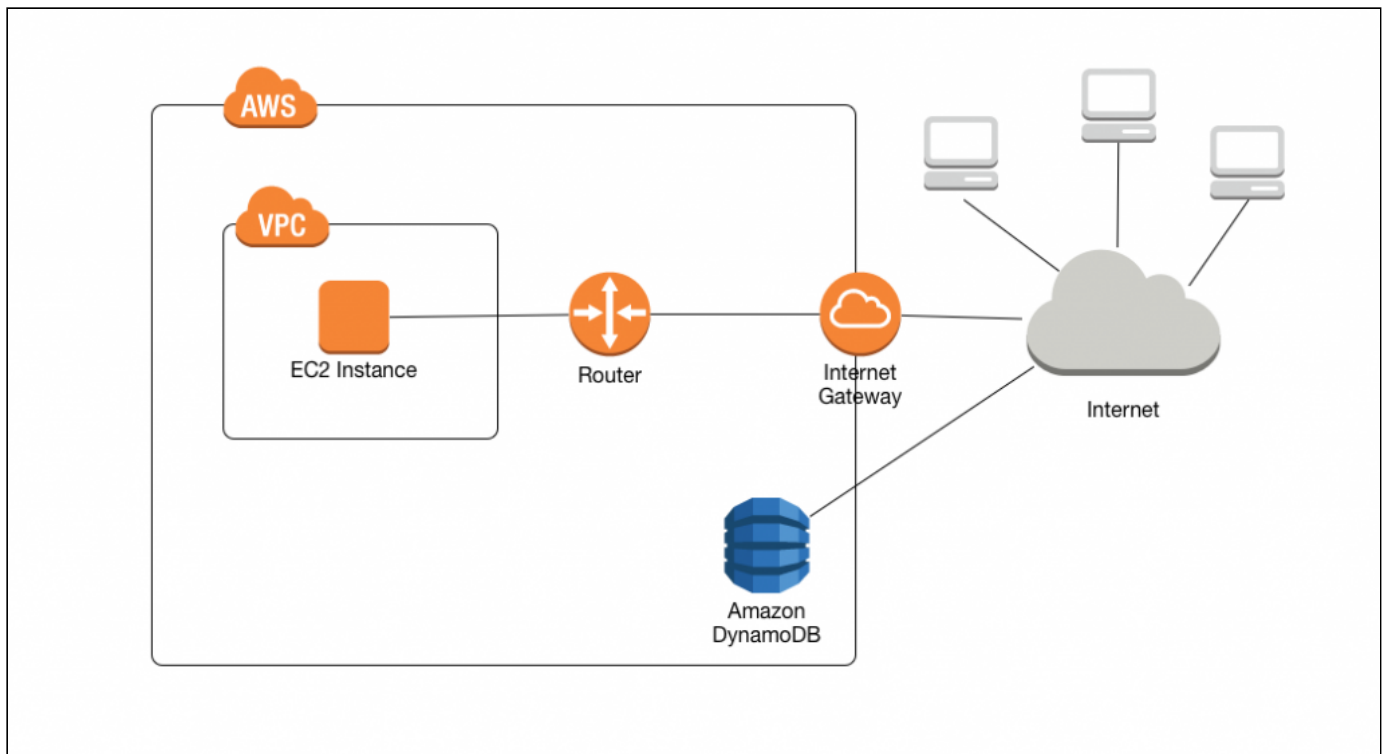
AWS News Blog

New – VPC Endpoints for DynamoDB

by [Randall Hunt](#) | on 16 AUG 2017 | in [Amazon DynamoDB](#), [Amazon VPC](#), [Launch](#), [News](#) | [Permalink](#) | [Share](#)

Starting today [Amazon Virtual Private Cloud](#) (VPC) Endpoints for [Amazon DynamoDB](#) are available in all public AWS regions. You can provision an endpoint right away using the [AWS Management Console](#) or the [AWS Command Line Interface \(CLI\)](#). There are no additional costs for a VPC Endpoint for DynamoDB.

Many AWS customers run their applications within a [Amazon Virtual Private Cloud](#) (VPC) for security or isolation reasons. Previously, if you wanted your EC2 instances in your VPC to be able to access DynamoDB, you had two options. You could use an Internet Gateway (with a NAT Gateway or assigning your instances public IPs) or you could route all of your traffic to your local infrastructure via VPN or [AWS Direct Connect](#) and then back to DynamoDB. Both of these solutions had security and throughput implications and it could be difficult to configure NACLs or security groups to restrict access to just DynamoDB. Here is a picture of the old infrastructure.



Creating an Endpoint

Let's create a VPC Endpoint for DynamoDB. We can make sure our region supports the endpoint with the [DescribeVpcEndpointServices](#) API call.

Bash

```
aws ec2 describe-vpc-endpoint-services --region us-east-1
{
  "ServiceNames": [
    "com.amazonaws.us-east-1.dynamodb",
    "com.amazonaws.us-east-1.s3"
  ]
}
```

Great, so I know my region supports these endpoints and I know what my regional endpoint is. I can grab one of my VPCs and provision an endpoint with a quick call to the CLI or through the console. Let me show you how to use the console.

First I'll navigate to the VPC console and select "Endpoints" in the sidebar. From there I'll click "Create Endpoint" which brings me to this handy console.

Create Endpoint

Step 1: Configure Endpoint
Step 2: Configure Route Tables

Step 1: Configure Endpoint

A VPC Endpoint allows you to securely connect your Amazon VPC to another AWS service.

VPC* vpc-bc0ebbd9 | ranman ⓘ

Service com.amazonaws.us-east-1.dynamodb ⓘ

Policy*
☒ Full Access - Allow access by any user or service within the VPC using credentials from any AWS accounts to any resources in this AWS service. All policies — IAM user policies, VPC endpoint policies, and AWS service-specific policies (e.g. Amazon S3 bucket policies, any S3 ACL policies) — must grant the necessary permissions for access to succeed.
☐ Custom

Use the [policy creation tool](#) to generate a policy, then paste the generated policy below.

```
{
  "Statement": [
    {
      "Action": "*",
      "Effect": "Allow",
      "Resource": "*",
      "Principal": "*"
    }
  ]
}
```

ⓘ

[Cancel and Exit](#) [Next Step](#)

You'll notice the [AWS Identity and Access Management \(IAM\)](#) policy section for the endpoint. This supports all of the [fine grained access control](#) that DynamoDB supports in regular IAM policies and you can restrict access based on IAM policy conditions.

For now I'll give full access to my instances within this VPC and click "Next Step".

Create Endpoint

[Step 1: Configure Endpoint](#)



Step 2: Configure Route Tables

Step 2: Configure Route Tables

A rule with destination **pl-02cd2c6b (com.amazonaws.us-east-1.dynamodb)** and a target with this endpoints' ID (e.g. vpce-12345678) will be added to the route tables you select below.

Subnets associated with selected route tables will be able to access this endpoint.

	Route Table ID	N	Main	Associated With
<input type="checkbox"/>	rtb-2d8a2048		Yes	0 subnets
<input checked="" type="checkbox"/>	rtb-2c8a2049		No	2 subnets

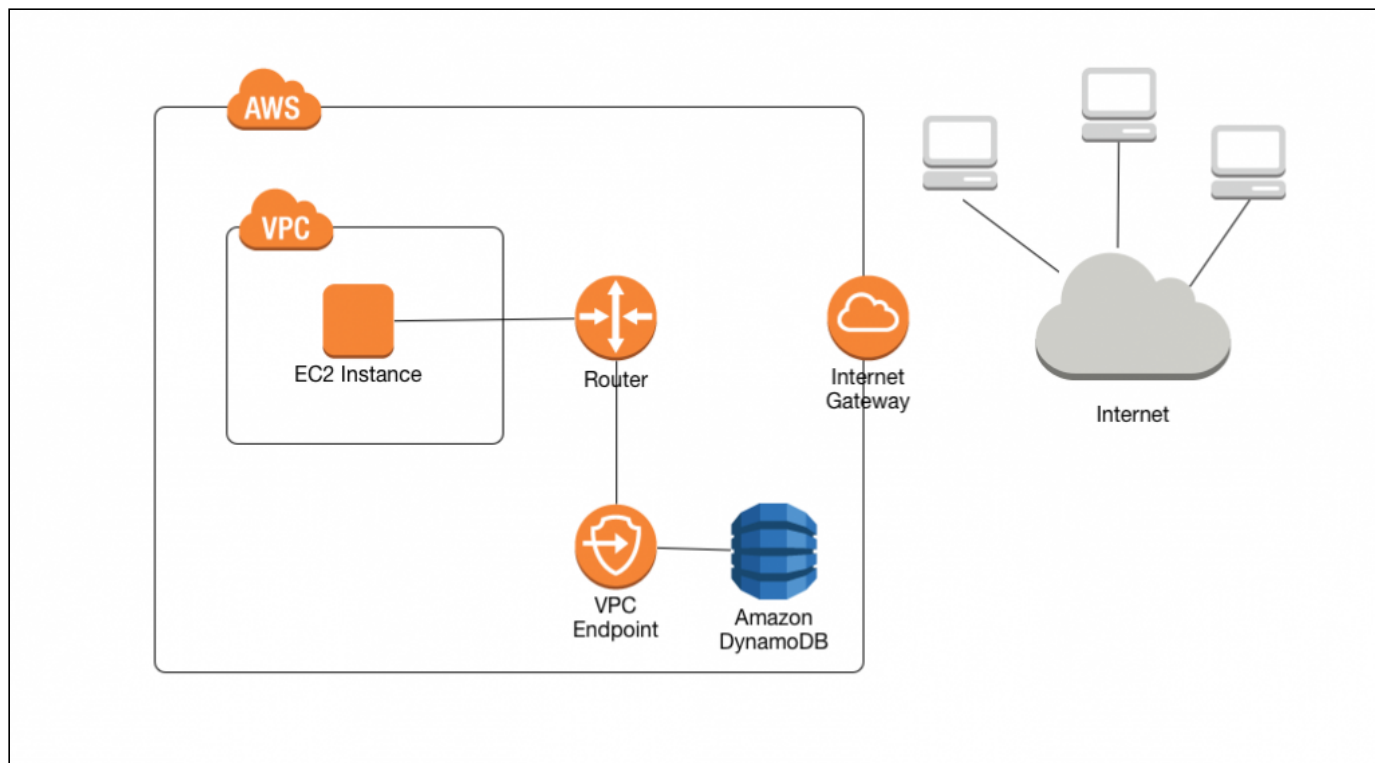
 When you use an endpoint, the source IP addresses from your instances in your affected subnets for accessing the AWS service in the same region will be private IP addresses, not public IP addresses. Existing connections from your affected subnets to the AWS service that use public IP addresses may be dropped. Ensure that you don't have critical tasks running when you create or modify an endpoint

[Cancel and Exit](#)[Previous Step](#)[Create Endpoint](#)

This brings me to a list of route tables in my VPC and asks me which of these route tables I want to assign my endpoint to. I'll select one of them and click "Create Endpoint".

Keep in mind the note of warning in the console: if you have source restrictions to DynamoDB based on public IP addresses the source IP of your instances accessing DynamoDB will now be their private IP addresses.

After adding the VPC Endpoint for DynamoDB to our VPC our infrastructure looks like this.



That's it folks! It's that easy. It's provided at no cost. Go ahead and start using it today. If you need more details you can read the [docs here](#).

TAGS: [Amazon DynamoDB](#)