



slalom

Slalom AWS Introduction Workshop

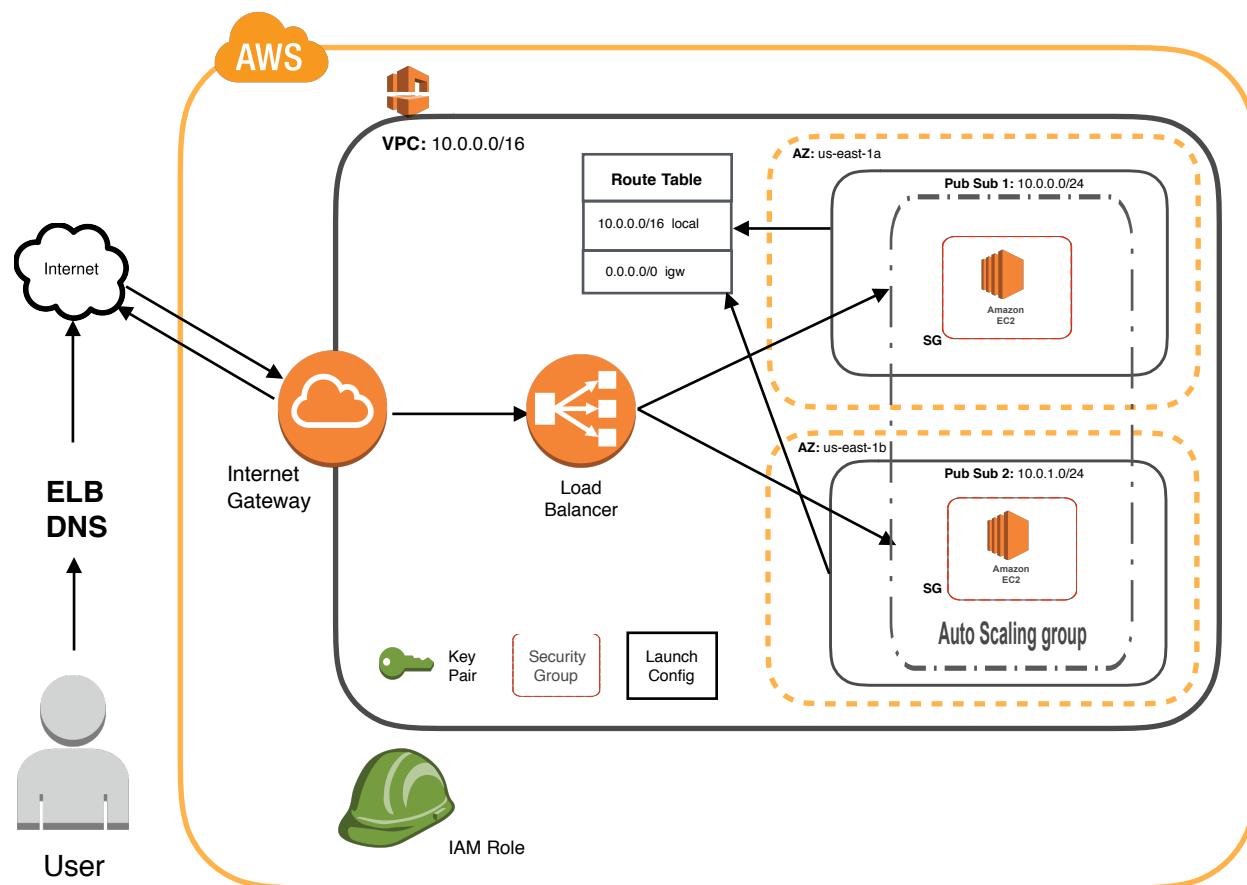
Introduction

Welcome to the lab section of the Slalom AWS Introduction Workshop! This document outlines 11 steps, walking you through the creation of the AWS resources required to setup a simple load-balanced autoscaling group serving a basic webpage.

At the end of each step (excluding Step 1) you will find an architectural summary of the AWS resources that you have built so far.

The next two pages contain a brief summary of what is achieved within each step of the process. In addition, please reach out to Bruce, Jeff or Todd for clarification on any of the content covered within this document!

The following image outlines the final architecture of what we are going to be building within this lab.



Step Overview

Step 1: Logging in to the AWS Console

By now you should have signed up for your own AWS account. This step simply directs you to log in to your account and select the appropriate AWS Region

Step 2: Create a VPC

Amazon Virtual Private Cloud (VPC) lets you provision a logically isolated section of the AWS Cloud. Think of it as your own private data center within the larger AWS cloud. You create your VPC using an IP address block, which describes how many IP addresses you wish your VPC to contain. In our example, we use 10.0.0.0/16, which provides 65531 IP addresses for use.

Step 3: Creation of Additional Public Subnet

Within a VPC, you can create subnets to logically divide the space within your VPC into smaller logical sections. Subnets can be both public (AWS resources created within are reachable over the public Internet) or private (AWS resources created within cannot be reached over the Internet). An important fact to know is that a subnet is tied to an Amazon availability zone such as us-east-1a or us-east-1b. **You cannot have a subnet that spans multiple availability zones.**

Step 4: Route Table Association

When you create a subnet, AWS also creates a route table for that subnet. The route table contains entries (routes) that define how to communicate with other AWS resources and resources within specific IP address blocks

Step 5: Create an IAM Role

AWS Identity and Access Management (IAM) is an AWS best practice that helps you to define what a resource can and cannot do in AWS. In our example, we will enable the virtual servers (EC2 instances) in our auto scale group to communicate with another popular AWS Service, Amazon S3. Communication with all other AWS Services will be denied. Unlike many of the other AWS Resources we create within this workshop, IAM is a Global service. This means any IAM resources that you create are visible and available for use in any AWS Region.

Step 6: Create a Key Pair

Generating a key pair allows you to log into an EC2 instance using ssh or RDP when the EC2 instance is created with an association to that key pair.

Step 7: Create a Security Group

Security groups provide an efficient way to assign access to AWS resources within your Virtual Private Cloud (VPC). Think of it as a firewall, where you can allow various network traffic types to pass through specific ports and IP address ranges. In our example, we open up port 80 to allow HTTP traffic from the Internet.

Step 8: Create a load balancer

An Elastic Load Balancer (ELB) is used to help distribute incoming traffic among any connected EC2 instances. In our example, we use it to distribute traffic among the EC2 instances in our auto scaling group. Our autoscaling group is spread across us-east-1a and us-east-1b, so this helps to introduce an element of high availability.

Step 9: Create a launch configuration

A launch configuration is a template that an Auto Scaling group uses to launch EC2 instances. Within our launch configuration we will include some of the features we have created earlier, such as the KeyPair and our Security Group.

Step 10: Create an AutoScaling Group

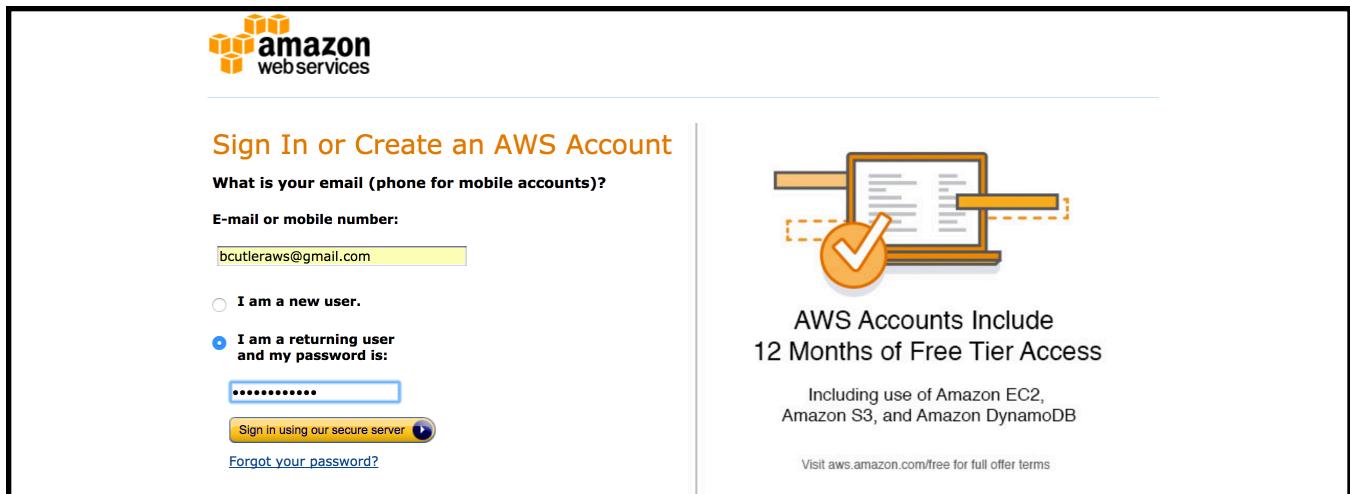
An Auto Scaling group is a collection of EC2 instances that share similar characteristics and are treated as a logical grouping. In our example, we create an Auto Scaling group with a size of two EC2 instances. Our autoscaling group will create instances in us-east-1a and us-east-1b availability zones. The load balancer ensures traffic that enters the load balancer is distributed between these two EC2 instances.

Step 11: Test it out!

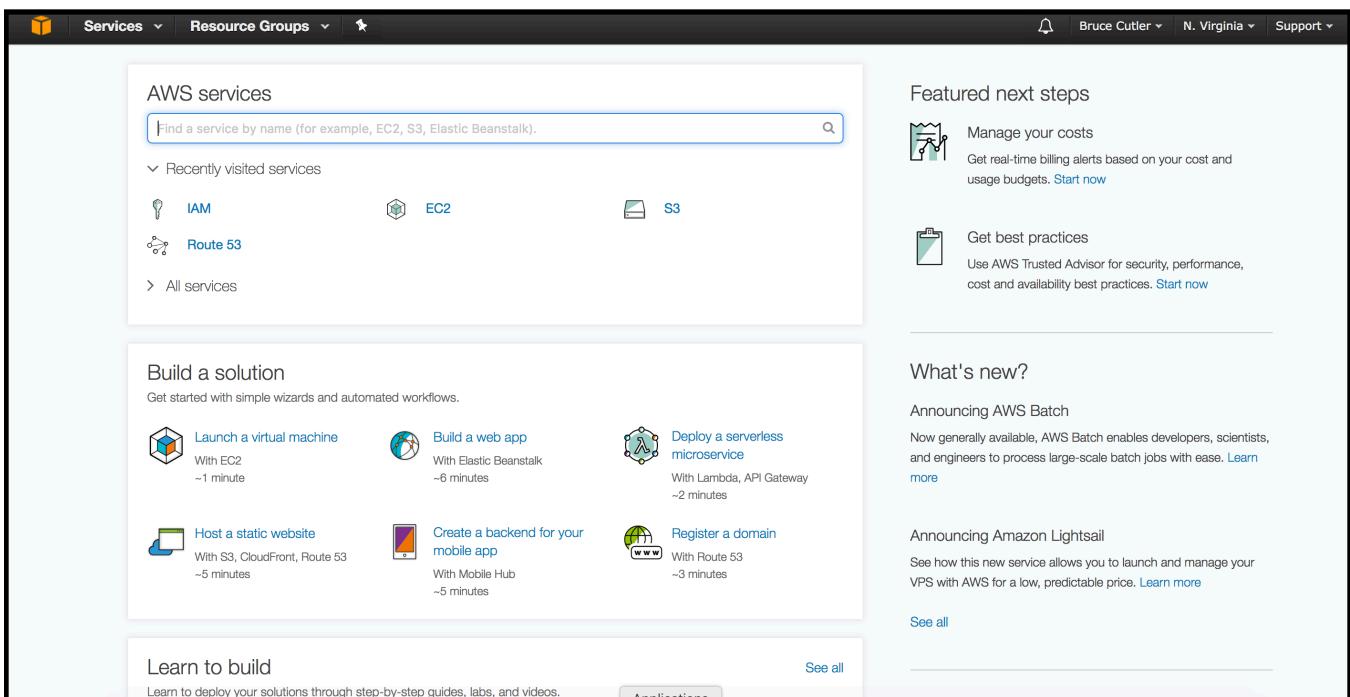
Once we've build all of the above components, we can use the DNS value from the load balancer in a browser to view our simple webpage.

Step 1: Logging in to the AWS Console

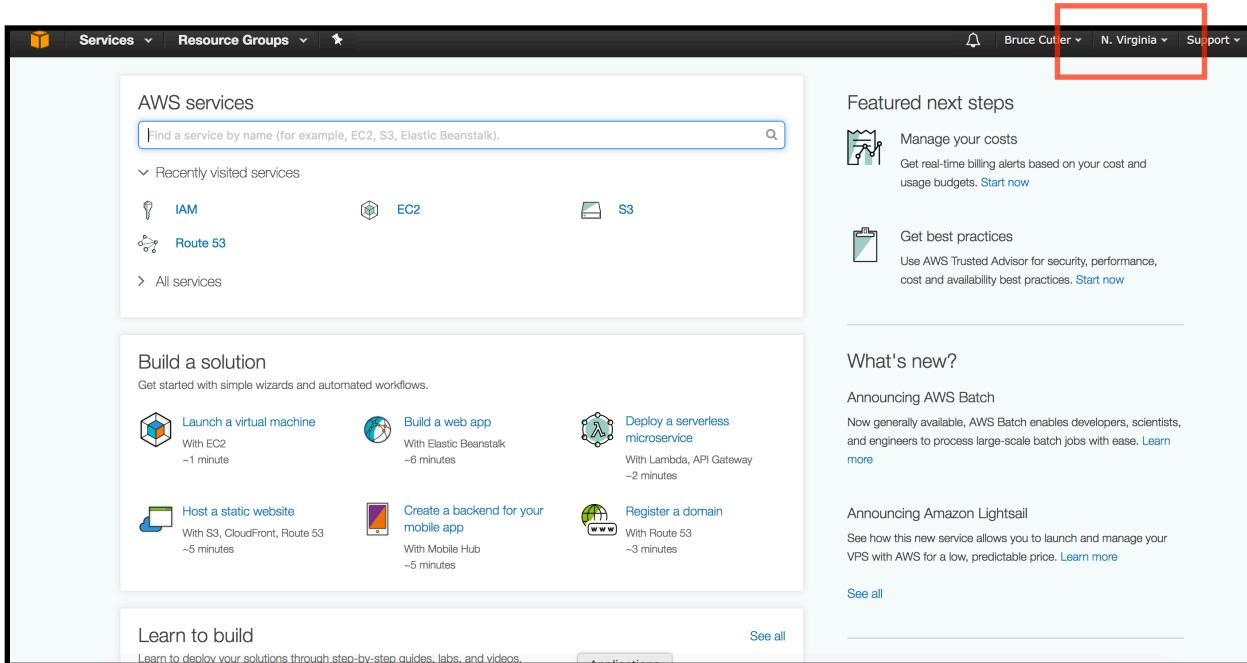
1. In a web browser, navigate to <https://console.aws.amazon.com/console/home>



2. Log in with the email / password combination you signed up for your account with. This will bring you to the AWS Console home page:



3. At the top right of the AWS Console screen, ensure that the **N. Virginia** is selected as your AWS Region



Step 2: Create a VPC (Virtual Private Cloud)

1. In the AWS Console, navigate to **Services** → **VPC**:

The screenshot shows the AWS VPC Dashboard. At the top left, there is a red box highlighting the 'Services' dropdown menu. Below it, the 'VPC Dashboard' section is visible. On the far right, there is a 'Service Health' panel with two entries: 'Amazon VPC - US East (N. Virginia)' and 'Amazon EC2 - US East (N. Virginia)', both marked as 'Service is operating normally'. A 'Start VPC Wizard' button is located at the top center of the dashboard.

2. Click on the blue **Start VPC Wizard** button towards the top left of the screen
3. In the first page of the VPC Wizard, select the top option, **VPC with a Single Public Subnet** and click the blue **Select** button:

The screenshot shows the 'Step 1: Select a VPC Configuration' page of the VPC Wizard. A red box highlights the 'VPC with a Single Public Subnet' option, which is currently selected. To its right, a detailed description of the configuration is provided, mentioning instances running in a private section with direct Internet access, and the creation of a /16 network with a /24 subnet. Below this, a diagram illustrates the setup: a cloud labeled 'Internet, S3, DynamoDB, SNS, SQS, etc.' connects to a 'Public Subnet' within the 'Amazon Virtual Private Cloud'. A blue 'Select' button is located at the bottom right of the description area.

4. Ensure that the following is entered when configuring your VPC and Public Subnet:

IPv4 CIDR Block: 10.0.0.0/16

VPC Name: slalom-aws-intro

Public subnet's IPv4 CIDR: 10.0.0.0/24

Availability Zone: us-east-1a

Subnet Name: slalom-aws-intro-pub1

Enable DNS Hostnames: Yes

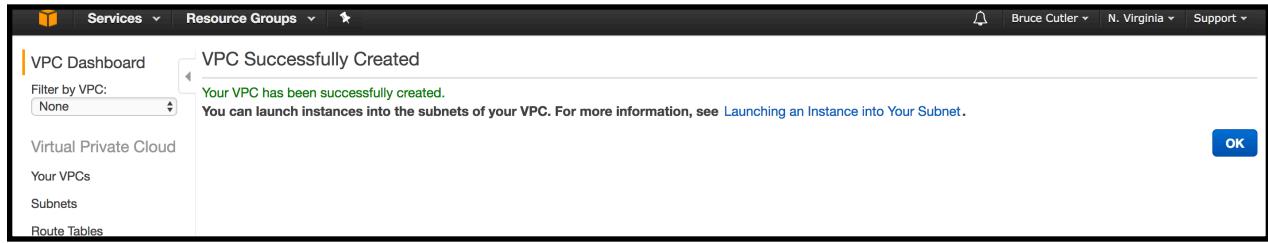
All other options can be left as default values, leaving you with the following setup:

The screenshot shows the 'Step 2: VPC with a Single Public Subnet' configuration page. The configuration includes:

- IPv4 CIDR block:** 10.0.0.0/16 (65531 IP addresses available)
- IPv6 CIDR block:** No IPv6 CIDR Block
 Amazon provided IPv6 CIDR block
- VPC name:** slalom-aws-intro
- Public subnet's IPv4 CIDR:** 10.0.0.0/24 (251 IP addresses available)
- Availability Zone:** us-east-1a
- Subnet name:** slalom-aws-intro-pub1
- A note: You can add more subnets after AWS creates the VPC.
- Service endpoints:** Add Endpoint button
- Enable DNS hostnames:** Yes No
- Hardware tenancy:** Default

5. Select the **Create VPC** button when you are ready

6. You should then see the following screen if VPC creation was successful



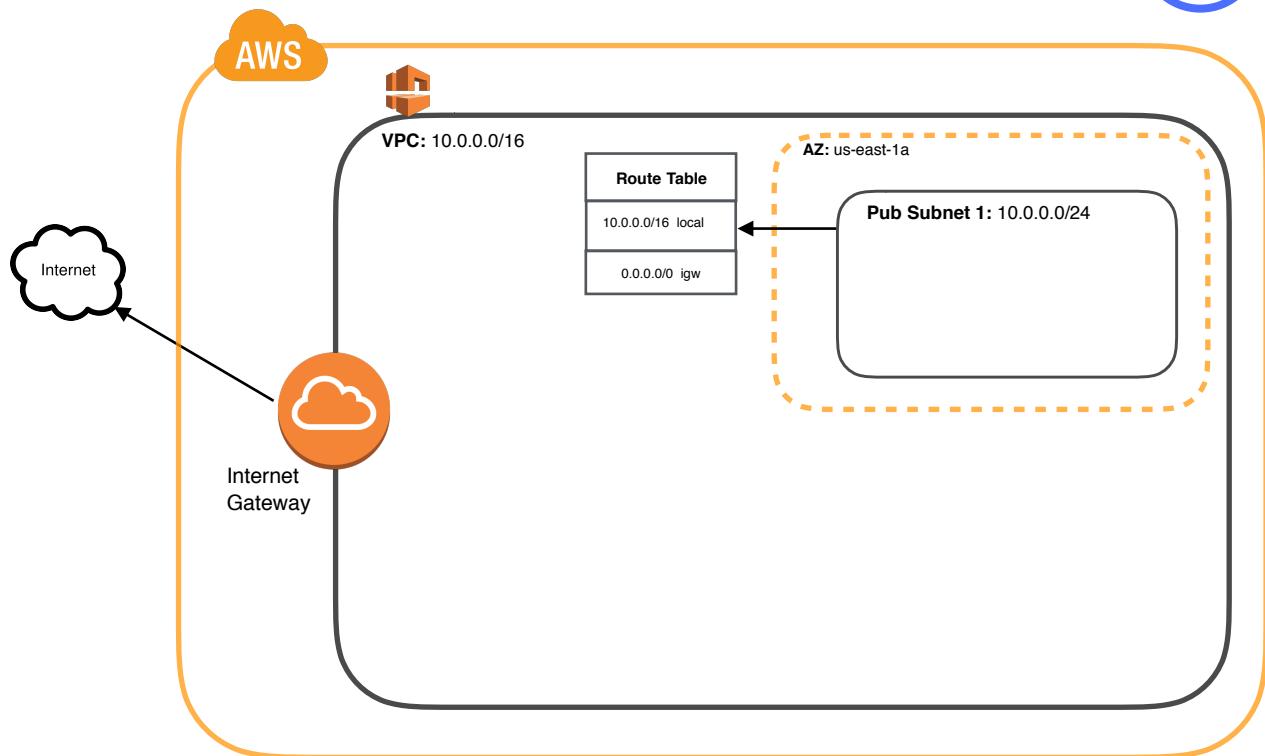
7. In addition to the VPC, the VPC Wizard has automatically created other components that are required for a successful networking setup.

Click in the following menu items on the left side to see what else has been created with the **slalom-aws-intro** VPC:

- **Subnets**
- **Route Tables**
- **Internet Gateways**

System Architecture: Step 2

2



Step 3: Creation of Additional Public Subnet

We created the **slalom-aws-intro-pub1** subnet as part of VPC creation in **Step 2**. To show the benefits of distributing resources across multiple AWS zones, we are going to create an additional subnet in a different availability zone

1. Navigate to **Services —> VPC** and click on the **Subnets** link in the left menu
2. Click on the blue **Create Subnet** button
3. Ensure that the following information is entered in the Create Subnet box that appears:

Name tag: slalom-aws-intro-pub2

VPC: slalom-aws-intro

Availability Zone: us-east-1b

IPv4 CIDR Block: 10.0.1.0/24

This should leave you with something like the screenshot below. Ignore the fact that CIDR values in the image differ to your configuration (This is because VPC's already exist in my account with the values you are using):

The screenshot shows the 'Create Subnet' dialog box. At the top, it says 'Create Subnet'. Below that is a note: 'Use the CIDR format to specify your subnet's IP address block (e.g., 10.0.0.0/24). Note that block sizes must be between a /16 netmask and /28 netmask. Also, note that a subnet can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.' The form fields are as follows:

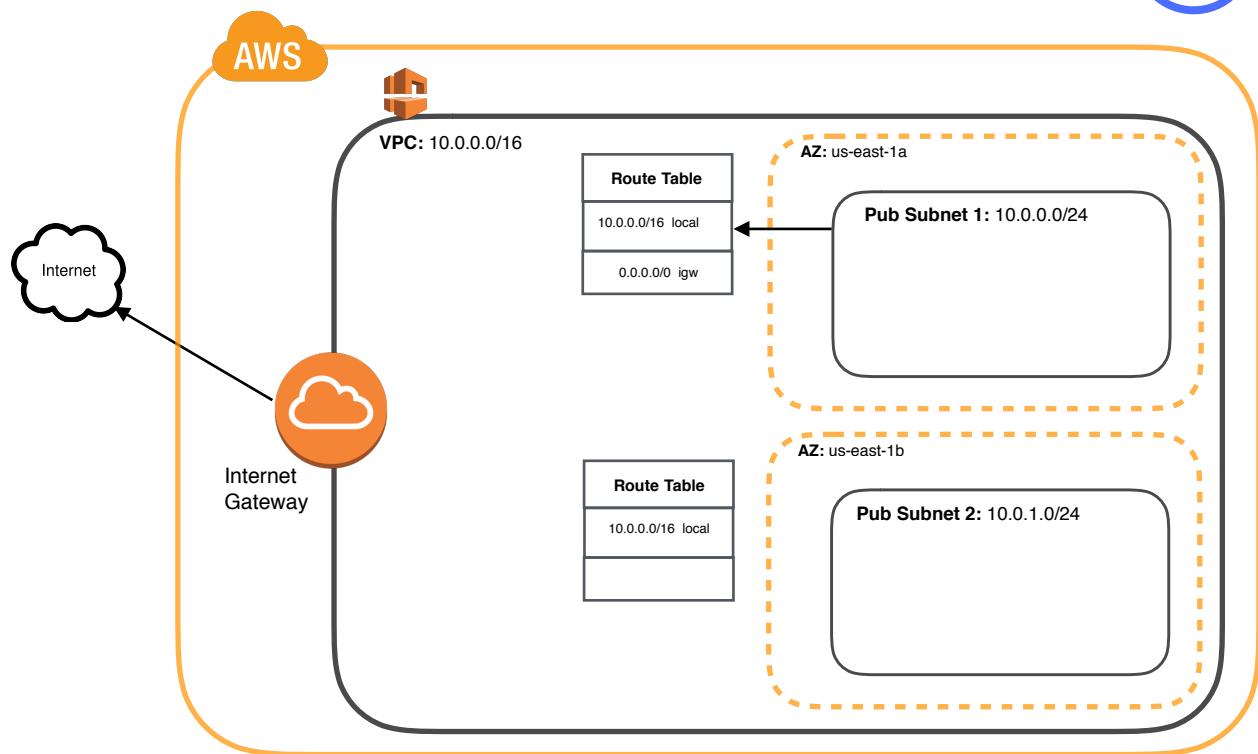
Name tag	slalom-aws-intro-pub2	i
VPC	vpc-59e4c43f slalom-aws-intro	i
VPC CIDRs		
CIDR	Status	Status Reason
10.1.0.0/16	associated	
Availability Zone	us-east-1b	i
IPv4 CIDR block	10.1.1.0/24	i

At the bottom right are two buttons: 'Cancel' and 'Yes, Create'.

4. Click the blue **Yes, Create** button when you are ready

System Architecture: Step 3

3



Step 4: Route Table Association

1. Navigate to **Services** → **VPC** and click on the **Subnets** link in the left hand menu
2. Select the **slalom-aws-intro-pub1** subnet and then navigate to the **Route Table** tab:

The screenshot shows the AWS VPC Dashboard. On the left sidebar, under the "Subnets" section, the "slalom-aws-intro-pub1" subnet is selected. The main content area displays a table of subnets. A red box highlights the "Route Table" tab for the selected subnet, which shows two route entries: one for local traffic and one pointing to an Internet Gateway.

Name	Subnet ID	State	VPC
subnet-10c4523d	available	vpc-a93ceacf	
slalom-aws-intro-pub1	available	vpc-59e4c43f slalom-aws-intro	
ChefPublic	available	vpc-8b8009ed ChefVPC	
subnet-575d6b1e	available	vpc-a93ceacf	
subnet-9db33bc6	available	vpc-a93ceacf	
subnet-ba0243f3	available	vpc-a93ceacf	
ChefPrivate	available	vpc-8b8009ed ChefVPC	
subnet-075d6b4e	available	vpc-a93ceacf	
subnet-a38b279f	available	vpc-a93ceacf	
subnet-6feb810a	available	vpc-a93ceacf	

subnet-f5c1faae | slalom-aws-intro-pub1

Destination	Target
10.1.0.0/16	local
0.0.0.0/0	igw-b3b6acd4

3. Within the Route Table tab above, notice that there are two route entries configured: one for local and one to the igw (Internet Gateway)
4. Now, select the slalom-aws-intro-pub2 subnet and navigate to the Route Table tab. You should notice that this subnet only contains a single route to local.

To ensure that traffic from the slalom-aws-intro-pub2 subnet can reach the Internet, we need to associate the route table from slalom-aws-intro-pub1 with slalom-aws-intro-pub2

- Click on the Route Tables link in the left menu
- Select the Route table with the slalom-aws-intro in the VPC column that is already associated with 1 Subnet. **For example:**

	Name	Route Table ID	Explicitly Associated	Main	VPC
<input type="checkbox"/>	rtb-e9e89b90	rtb-e9e89b90	1 Subnet	No	vpc-59e4c43f slalom-aws-intro
<input type="checkbox"/>	rtb-01939167	rtb-01939167	0 Subnets	Yes	vpc-a93ceacf
<input type="checkbox"/>	ChefPublic	rtb-8222ecfb	1 Subnet	No	vpc-8b8009ed ChefVPC
<input type="checkbox"/>		rtb-a825ebd1	0 Subnets	Yes	vpc-8b8009ed ChefVPC
<input type="checkbox"/>		rtb-0aee9d73	0 Subnets	Yes	vpc-59e4c43f slalom-aws-intro

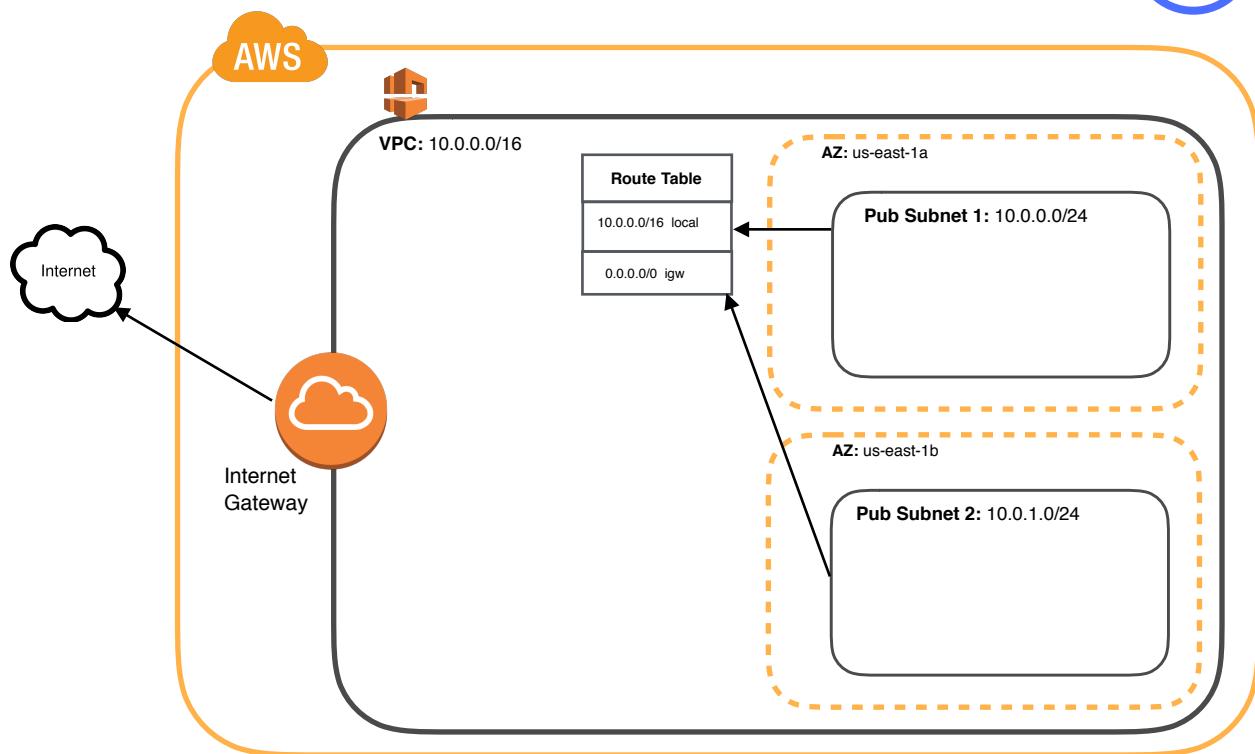
- With the correct route table selected, navigate to the **Subnet Associations** tab at the bottom (see screenshot below)
- Click **Edit**
- Select the checkbox next to the **slalom-aws-intro-pub2** subnet so that both slalom-aws-intro public subnets are selected and click **Save**

rtb-e9e89b90

Associate	Subnet	IPv4 CIDR	IPv6 CIDR	Current Route Table
<input checked="" type="checkbox"/>	subnet-f5c1faae slalom-aws-intro-pub1	10.1.0.0/24	-	rtb-e9e89b90
<input checked="" type="checkbox"/>	subnet-518bca34 slalom-aws-intro-pub2	10.1.1.0/24	-	Main

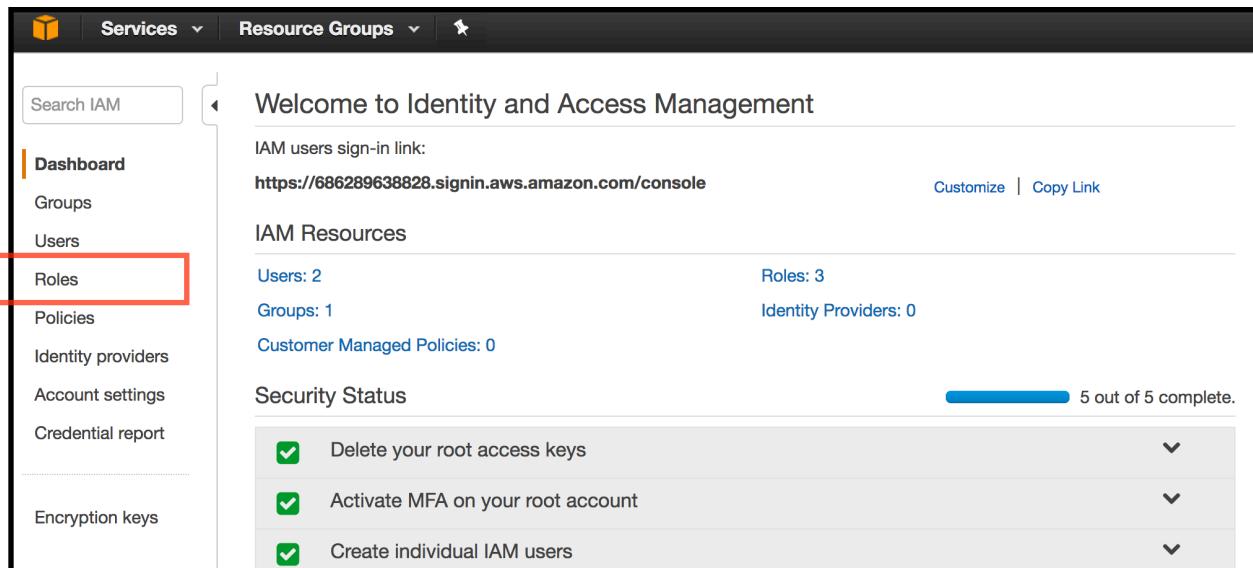
System Architecture: Step 4

4



Step 5: Create an IAM Role

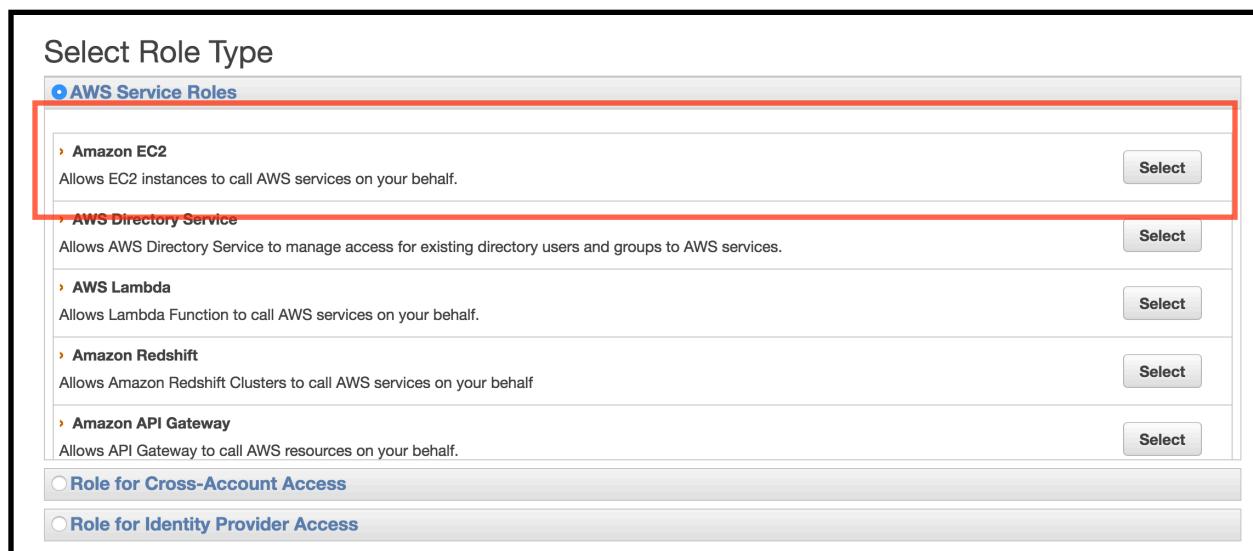
1. Navigate to **Services** → **IAM**
2. Select the **Roles** link in the left menu



The screenshot shows the AWS Identity and Access Management (IAM) console. The left sidebar has a 'Roles' link highlighted with a red box. The main area displays the following information:

- Welcome to Identity and Access Management**
- IAM users sign-in link:** <https://686289638828.signin.aws.amazon.com/console>
- Customize | Copy Link**
- IAM Resources**
 - Users: 2
 - Groups: 1
 - Customer Managed Policies: 0
 - Roles: 3
 - Identity Providers: 0
- Security Status**
 - 5 out of 5 complete.
 - Checklist items:
 - Delete your root access keys
 - Activate MFA on your root account
 - Create individual IAM users

3. Select the blue **Create New Role** button
4. Enter **slalom_web_server** as the Role Name and click **Next Step**
5. In the Select Role Type screen, click the Select button in the **Amazon EC2** section:



The screenshot shows the 'Select Role Type' screen. The 'AWS Service Roles' section is selected, and the 'Amazon EC2' role is highlighted with a red box. Its 'Select' button is also highlighted. Other service roles listed include AWS Directory Service, AWS Lambda, Amazon Redshift, and Amazon API Gateway, each with a 'Select' button. Below these, there are two additional sections: 'Role for Cross-Account Access' and 'Role for Identity Provider Access', both with unselected radio buttons.

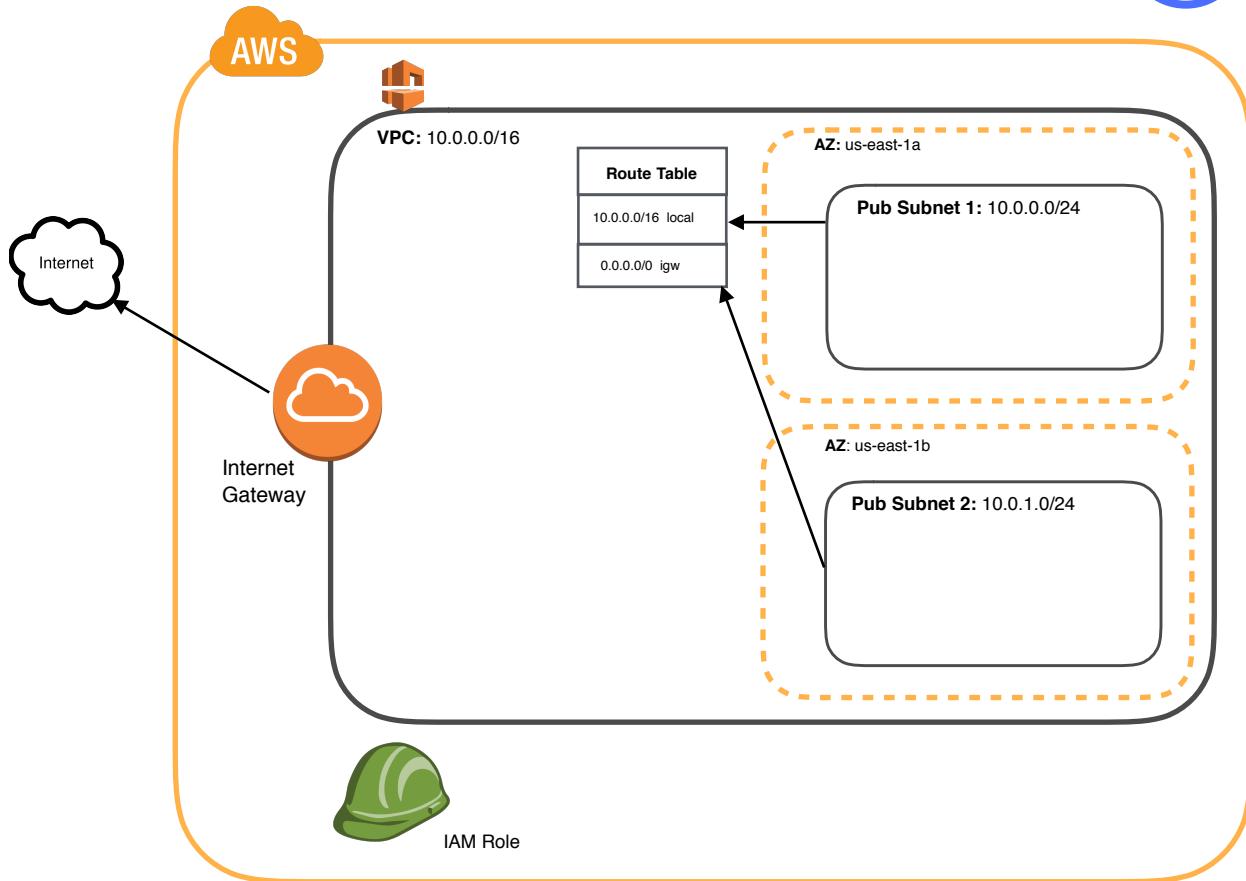
6. In the Attach Policy screen, select **AmazonS3FullAccess** and click Next Step:

Attach Policy				
Select one or more policies to attach. Each role can have up to 10 policies attached.				
Filter:	Policy Type	Filter	Showing 252 results	
	Policy Name	Attached Entities	Creation Time	Edited Time
<input type="checkbox"/>	AdministratorAccess	2	2015-02-06 13:39 EDT	2015-02-06 13:39 EDT
<input type="checkbox"/>	AmazonEC2FullAccess	2	2015-02-06 13:40 EDT	2015-02-06 13:40 EDT
<input checked="" type="checkbox"/>	AmazonS3FullAccess	2	2015-02-06 13:40 EDT	2015-02-06 13:40 EDT
<input type="checkbox"/>	CloudWatchFullAccess	1	2015-02-06 13:40 EDT	2015-02-06 13:40 EDT
<input type="checkbox"/>	IAMUserChangePassword	1	2016-11-14 19:25 EDT	2016-11-15 18:18 EDT
<input type="checkbox"/>	AmazonAPIGatewayAdministrator	0	2015-07-09 13:34 EDT	2015-07-09 13:34 EDT
<input type="checkbox"/>	AmazonAPIGatewayInvokeFullAc...	0	2015-07-09 13:36 EDT	2015-07-09 13:36 EDT
<input type="checkbox"/>	AmazonAPIGatewayPushToCloud...	0	2015-11-11 18:41 EDT	2015-11-11 18:41 EDT

7. Click the **Create Role** button to create the slalom_web_server IAM Role

System Architecture: Step 5

5



Step 6: Create A Key Pair

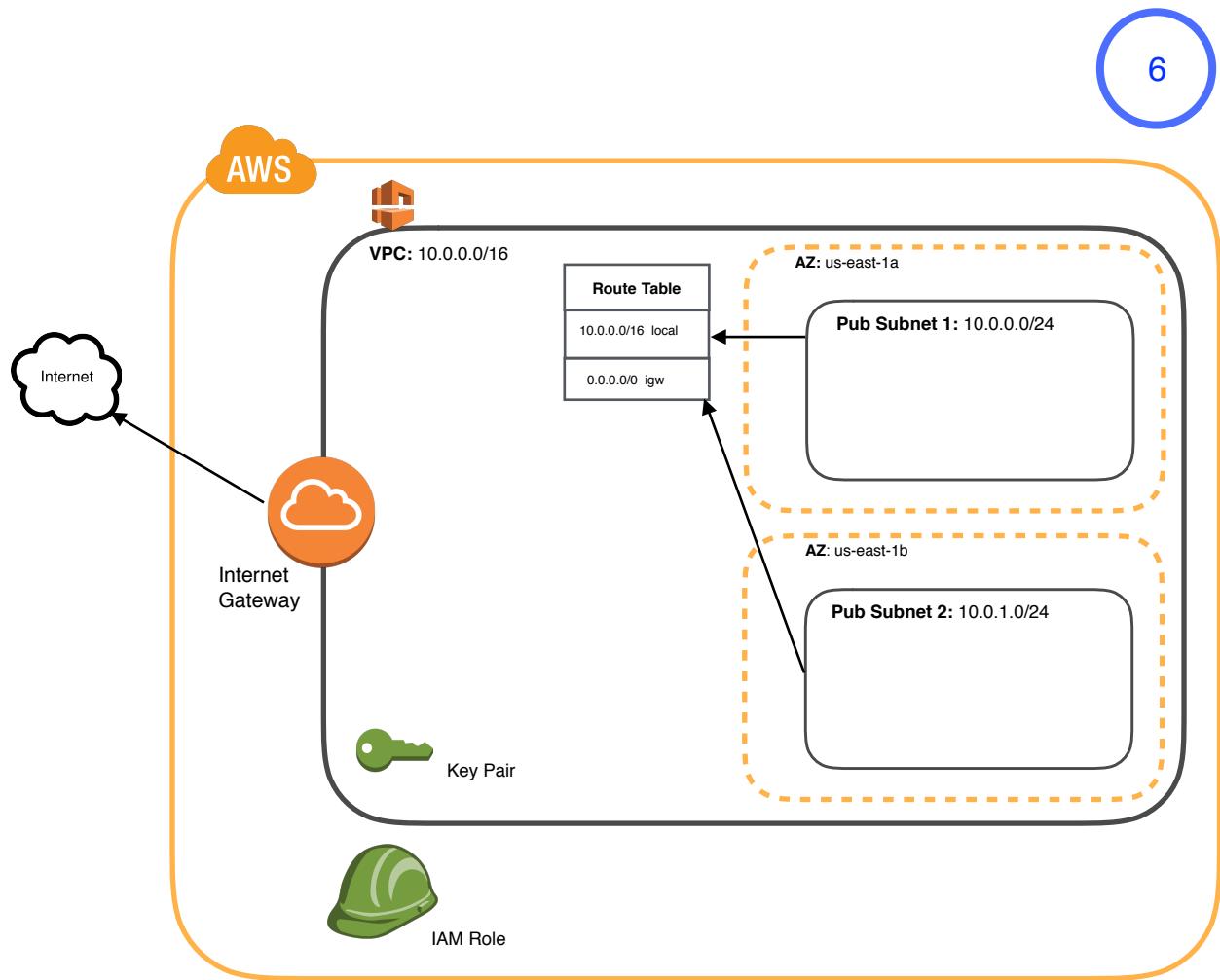
1. Navigate to **Services** → **EC2**
2. In the left menu, select **Key Pairs**:

The screenshot shows the AWS EC2 Dashboard. On the left, there's a navigation menu with links like EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES (with sub-links for Instances, Spot Requests, Reserved Instances, Scheduled Instances, Dedicated Hosts), IMAGES (with AMIs and Bundle Tasks), ELASTIC BLOCK STORE (with Volumes and Snapshots), and NETWORK & SECURITY (with Security Groups, Elastic IPs, Placement Groups, and Key Pairs). The 'Key Pairs' link is highlighted with a red box. The main content area is titled 'Resources' and shows resource counts: 0 Running Instances, 0 Dedicated Hosts, 2 Volumes, 1 Key Pairs, 0 Elastic IPs, 2 Snapshots, 0 Load Balancers, and 6 Security Groups. Below this is a promotional message about Amazon Lightsail. The 'Create Instance' section has a 'Launch Instance' button. The 'Service Health' section shows the US East (N. Virginia) region is operating normally. The 'Scheduled Events' section shows 'No events'.

3. Select the blue **Create Key Pair** button
4. For Key pair name, enter **slalom_aws_intro** and click the **Create** button
5. This will create a key pair for you in AWS and a file called slalom_aws_intro.pem file will download in your browser
6. Having access to this key file would be important for you if you wanted to log in using ssh (Linux Servers) or RDP (Windows Servers) to any virtual server you create in AWS. **However**, this is beyond the scope of this intro tutorial.

System Architecture: Step 6

6



Step 7: Create a security group

1. Navigate to **Services** → **EC2**
2. In the left menu, click the **Security Groups** link

The screenshot shows the AWS EC2 Dashboard. On the left, there's a navigation sidebar with the following sections:

- EC2 Dashboard** (highlighted with an orange border)
- Events
- Tags
- Reports
- Limits
- INSTANCES**
 - Instances
 - Spot Requests
 - Reserved Instances
 - Scheduled Instances
 - Dedicated Hosts
- IMAGES**
 - AMIs
 - Bundle Tasks
- ELASTIC BLOCK STORE**
 - Volumes
 - Snapshots
- NETWORK & SECURITY**
 - Security Groups** (highlighted with a red border)
 - Elastic IPs
 - Placement Groups
 - Key Pairs
 - Network Interfaces

The main content area is titled **Resources**. It displays the following resource counts in the US East (N. Virginia) region:

Resource Type	Count
Running Instances	0
Dedicated Hosts	0
Volumes	2
Key Pairs	1
Placement Groups	0
Elastic IPs	0
Snapshots	2
Load Balancers	0
Security Groups	6

A promotional message at the bottom of the main content area says: "Just need a simple virtual private server? Get everything you need to jumpstart your project - compute, storage, networking – for a low, predictable price. [Try Amazon Lightsail for free.](#)"

Below the main content area, there are two sections: **Create Instance** (with a "Launch Instance" button) and **Service Health** (showing "Service Status: US East (N. Virginia): This service is operating normally"). To the right of the Service Health section is a "Scheduled Events" section showing "US East (N. Virginia): No events".

3. Click the blue **Create Security Groups** button
4. In the Create Security Group screen, enter the following information:

Security group name: slalom_aws_intro_webserver
Description: Security group for web servers
VPC: slalom-aws-intro

5. On the Inbound tab, click **Add Rule** and add the following:

Type: Custom TCP Rule
Protocol: TCP
Port Range: 80
Source: Custom 0.0.0.0/0

This should look like the following:

Create Security Group

Security group name Description VPC

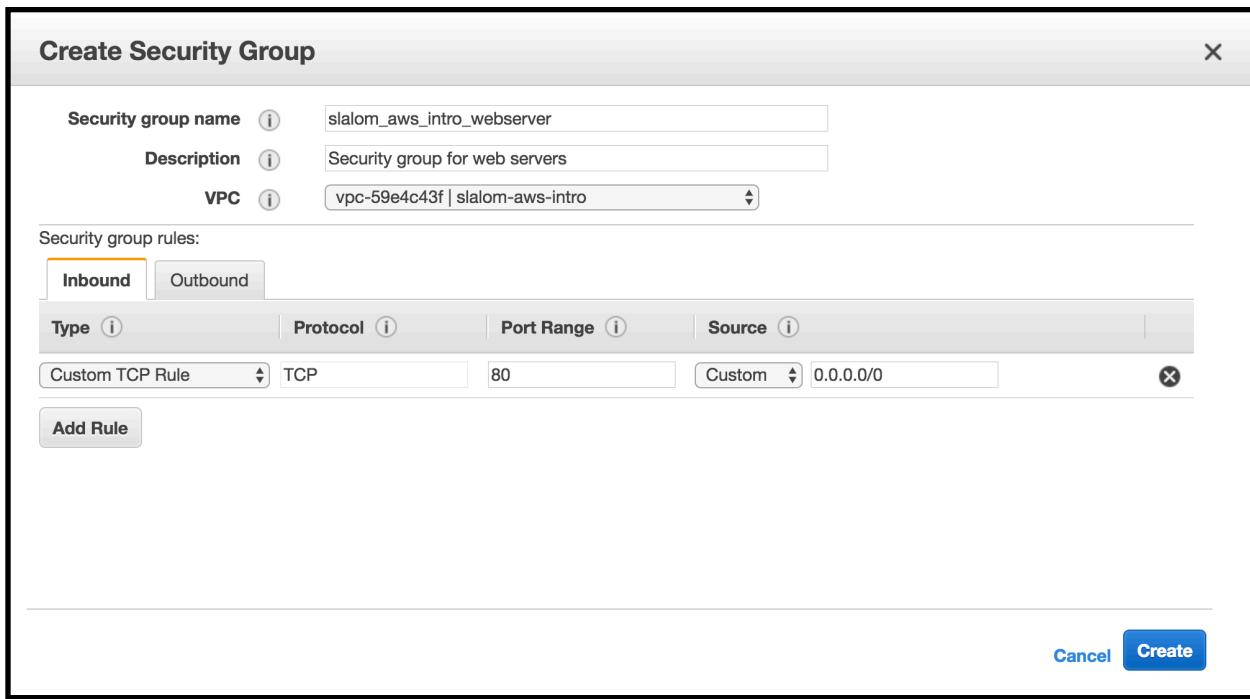
Security group rules:

Inbound Outbound

Type	Protocol	Port Range	Source
Custom TCP Rule	TCP	80	Custom 0.0.0.0/0

Add Rule

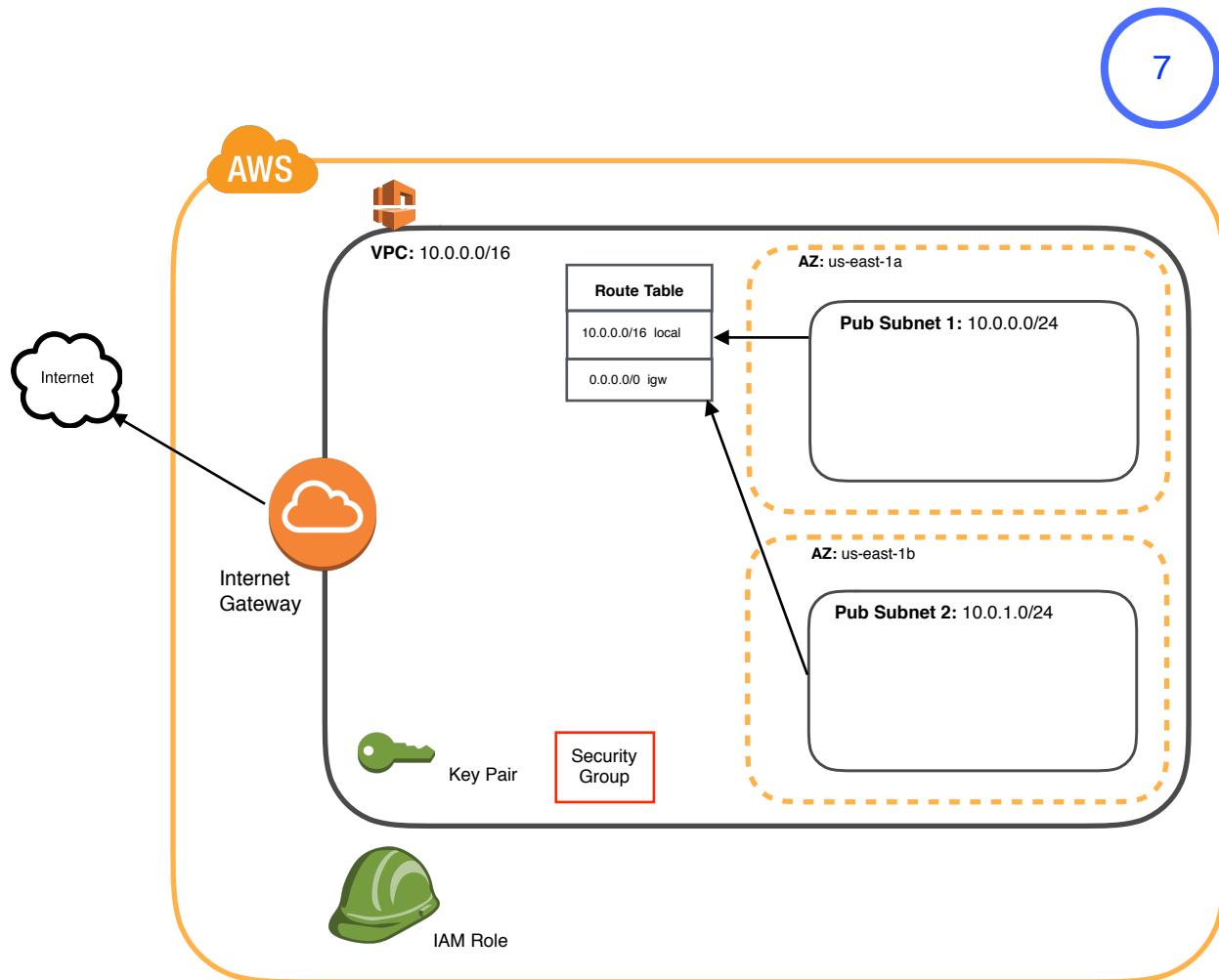
Cancel **Create**



6. Click the blue **Create** button when you are ready

System Architecture: Step 7

7

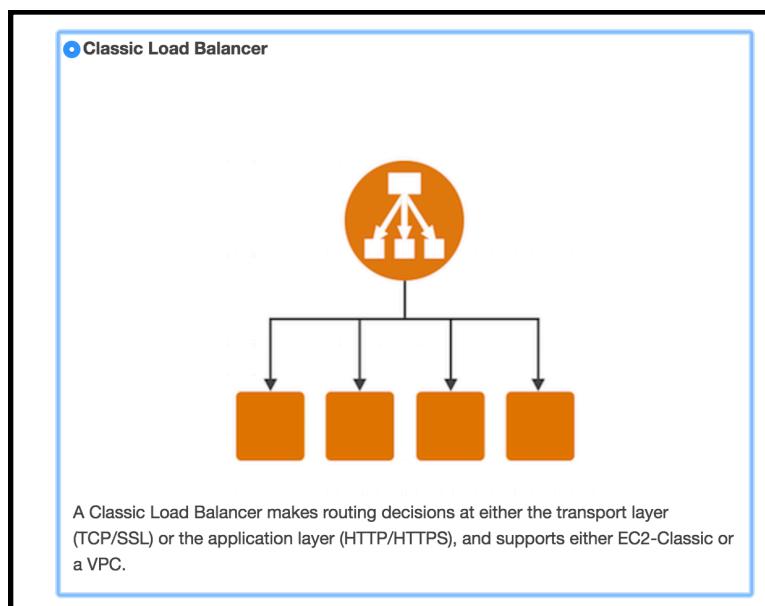


Step 8: Create a Load Balancer

1. Navigate to **Services** → **EC2**
2. In the left menu, select **Load Balancers**

The screenshot shows the AWS EC2 dashboard. On the left, there's a navigation sidebar with several categories: 'Bundle Tasks', 'ELASTIC BLOCK STORE' (Volumes, Snapshots), 'NETWORK & SECURITY' (Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces), 'LOAD BALANCING' (Load Balancers, Target Groups), 'AUTO SCALING' (Launch Configurations, Auto Scaling Groups), and 'SYSTEMS MANAGER SERVICES' (Run Command, State Manager). The 'LOAD BALANCING' section is highlighted with a red box around the 'Load Balancers' link. The main content area is titled 'Resources' and displays statistics for the US East (N. Virginia) region: 0 Running Instances, 0 Dedicated Hosts, 2 Volumes, 2 Key Pairs, 0 Placement Groups, 0 Elastic IPs, 2 Snapshots, 0 Load Balancers, and 7 Security Groups. Below this, a callout box suggests trying Amazon Lightsail for free. At the bottom, there are sections for 'Create Instance' (with a 'Launch Instance' button), 'Service Health', 'Scheduled Events', and 'Service Status: US East (N. Virginia)'.

3. Select the blue **Create Load Balancer** button
4. In the following screen, select the **Classic Load Balancer** option and click the blue **Continue** button



5. In the Define Load Balancer page, enter the following information:

Load Balancer Name: slalom-aws-intro-web-elb
Create LB Inside: slalom-aws-intro

6. In the Select Subnets section at the bottom, click the + symbols beside both **slalom-aws-intro-pub1** and **slalom-aws-intro-pub2**

This should leave you with the following setup:

Step 1: Define Load Balancer

Load Balancer name: slalom-aws-intro-web-elb
Create LB Inside: vpc-59e4c43f (10.1.0.0/16) | slalom-aws-intro
Create an internal load balancer: (what's this?)
Enable advanced VPC configuration:

Listener Configuration:

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port
HTTP	80	HTTP	80

Add

Select Subnets

You will need to select a Subnet for each Availability Zone where you wish traffic to be routed by your load balancer. If you have instances in only one Availability Zone, please select at least two Subnets in different Zones to provide higher availability for your load balancer.

VPC vpc-59e4c43f (10.1.0.0/16) | slalom-aws-intro

Available subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name

Selected subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
+	us-east-1a	subnet-f5c1faae	10.1.0.0/24	slalom-aws-intro-pub1
+	us-east-1b	subnet-518bca34	10.1.1.0/24	slalom-aws-intro-pub2

7. Click the **Next: Assign Security Groups** button at the bottom right
8. Use the **Select an existing security group** option and select **slalom_aws_intro_webserver** from the list of security groups
9. Click the **Next: Configure Security Settings** button
10. A warning message will appear, informing us that we can improve the load balancers security by using a secure listener. For the purposes of this workshop we will be using HTTP instead of HTTPS, so we can continue by clicking **Next: Configure Health Check**
11. We can leave the configured health check as it is, so click **Next: Add EC2 Instances**
12. We don't have any EC2 instances running currently, so continue to **Next: Add Tags**

13. Add a tag with the following values:

Key = Name

Value = slalom_aws_intro_web_elb

Step 6: Add Tags

Apply tags to your resources to help organize and identify them.

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. [Learn more](#) about tags.

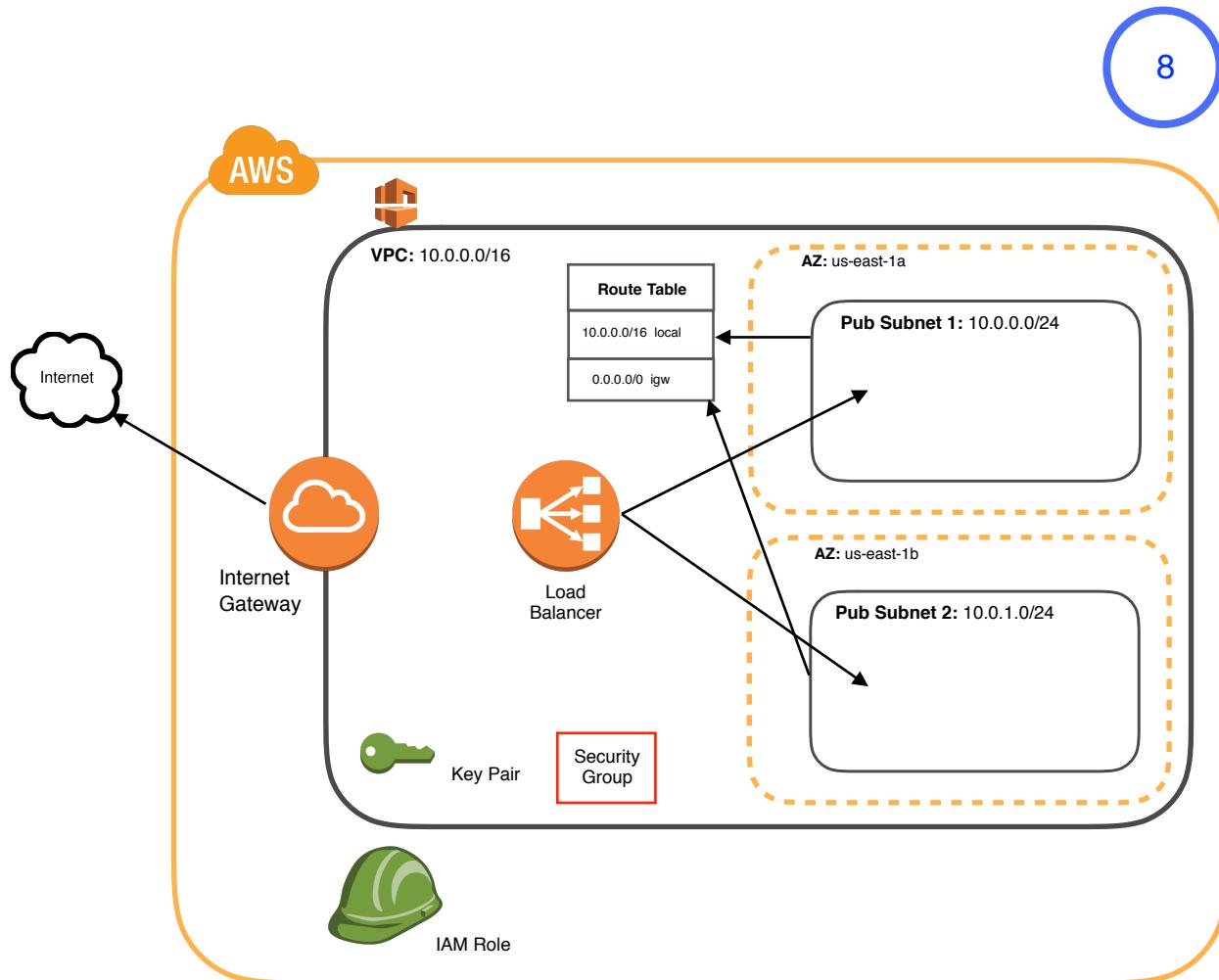
Key	Value
Name	slalom_aws_intro_web_elb

Create Tag

14. Click the blue **Review and Create** button at the bottom right

15. Click the blue **Create** button at the bottom right

System Architecture: Step 8



Step 9: Create A Launch Configuration for our Web Servers

1. Navigate to **Services** → **EC2**
2. In the left menu, scroll down and select **Launch Configurations**:

The screenshot shows the AWS EC2 Resources page. The left sidebar menu is open, and the 'LAUNCH CONFIGURATIONS' section is highlighted with a red box. The main content area displays various EC2 resources in the US East (N. Virginia) region, including 0 Running Instances, 0 Dedicated Hosts, 2 Volumes, 2 Key Pairs, 0 Placement Groups, 0 Elastic IPs, 2 Snapshots, 0 Load Balancers, and 7 Security Groups. A callout box points to the 'Launch Instance' button under the 'Create Instance' section.

3. Select the blue **Create Auto Scaling group** button:

The screenshot shows the AWS Auto Scaling Welcome page. The 'Create Auto Scaling group' button is highlighted with a red box. The page also features sections for 'Benefits of Auto Scaling', 'Reusable Instance Templates', 'Automated Provisioning', and 'Adjustable Capacity'.

4. At the bottom right of the screen, select the blue **Create launch configuration** button
5. Click the blue **Select** button for the **Amazon Linux AMI 2016.09.1**
6. For instance type, keep **t2.micro** selected and click **Next: Configure details**
7. On the Create Launch Configuration Page, enter the following:

Name: slalom_aws_intro_web_lc

IAM Role: slalom_web_server

7. Select the **Advanced Details** option and copy and paste the following script in to the white box using the **User data** (as text) option:

```
#!/bin/bash
sudo yum update -y
sudo yum install httpd -y
sudo service httpd start
sudo chkconfig httpd on
sudo groupadd www
sudo usermod -a -G www ec2-user
sudo chown -R ec2-user:ec2-user /var/www
sudo chmod 2775 /var/www
aws s3 cp s3://slalom-aws-intro/index.html /var/www/html/index.html
aws s3 cp s3://slalom-aws-intro/slalomlogo.png /var/www/html/slalomlogo.png
sudo find /var/www -type d -exec chmod 2775 {} +
sudo find /var/www -type f -exec chmod 664 {} +
```

IP Address Type - Assign a public IP address to every instance

This should leave you with the following configurations:

Create Launch Configuration

Name: slalom_aws_intro_web_lc

Purchasing option: Request Spot Instances

IAM role: slalom_web_server

Monitoring: Enable CloudWatch detailed monitoring

Advanced Details

Kernel ID: Use default

RAM Disk ID: Use default

User data:

```
#!/bin/bash
sudo yum update -y
sudo yum install httpd -y
sudo service httpd start
sudo chkconfig httpd on
sudo groupadd www
sudo usermod -a -G www ec2-user
sudo chown -R ec2-user:ec2-user /var/www
sudo chmod 2775 /var/www
aws s3 cp s3://slalom-aws-intro/index.html /var/www/html/index.html
aws s3 cp s3://slalom-aws-intro/slalomlogo.png /var/www/html/slalomlogo.png
sudo find /var/www -type d -exec chmod 2775 {} +
sudo find /var/www -type f -exec chmod 664 {} +
```

IP Address Type:

- Only assign a public IP address to instances launched in the default VPC and subnet. (default)
- Assign a public IP address to every instance.
- Do not assign a public IP address to any instances.

Note: this option only affects instances launched into an Amazon VPC

8. Once you have the above configuration, click **Next: Add Storage**
9. Leave the configured storage settings and click **Next: Configure Security Group**
10. Choose the option to **Select an existing security group** and then select the Security Group called **slalom_aws_intro_webserver** that was created earlier:

Create Launch Configuration

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below.

Assign a security group: Create a new security group Select an existing security group

Security Group ID	Name	VPC ID	Description
sg-8a36b0f6	ChefNode	vpc-8b8009ed	SG for Chef Nodes
sg-430fa63f	ChefServer	vpc-8b8009ed	Security group for the Chef server
sg-28369f54	default	vpc-8b8009ed	default VPC security group
sg-37911648	default	vpc-59e4c43f	default VPC security group
sg-d81d44a5	default	vpc-a93ceacf	default VPC security group
sg-0a786c7c	launch-wizard-2	vpc-a93ceacf	launch-wizard-2 created 2017-02-28T14:45:37.309-05:00
sg-944bcceb	slalom_aws_intro_webserver	vpc-59e4c43f	Security group for web servers

Inbound rules for sg-944bcceb Selected security groups: sg-944bcceb.

Type	Protocol	Port Range	Source
HTTP	TCP	80	0.0.0.0/0

11. Click the blue **Review** button at the bottom right
12. A warning message will appear telling you that you won't be able to access the server. This is ok for our purposes, so click **Continue** at the bottom right
13. Click the **Create launch configuration** button
14. In the key pair box that appears, select the **slalom_aws_intro** key that you created earlier and **also** select the checkbox acknowledging that you possess the key pair before clicking the blue **Create launch configuration** button

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Choose an existing key pair

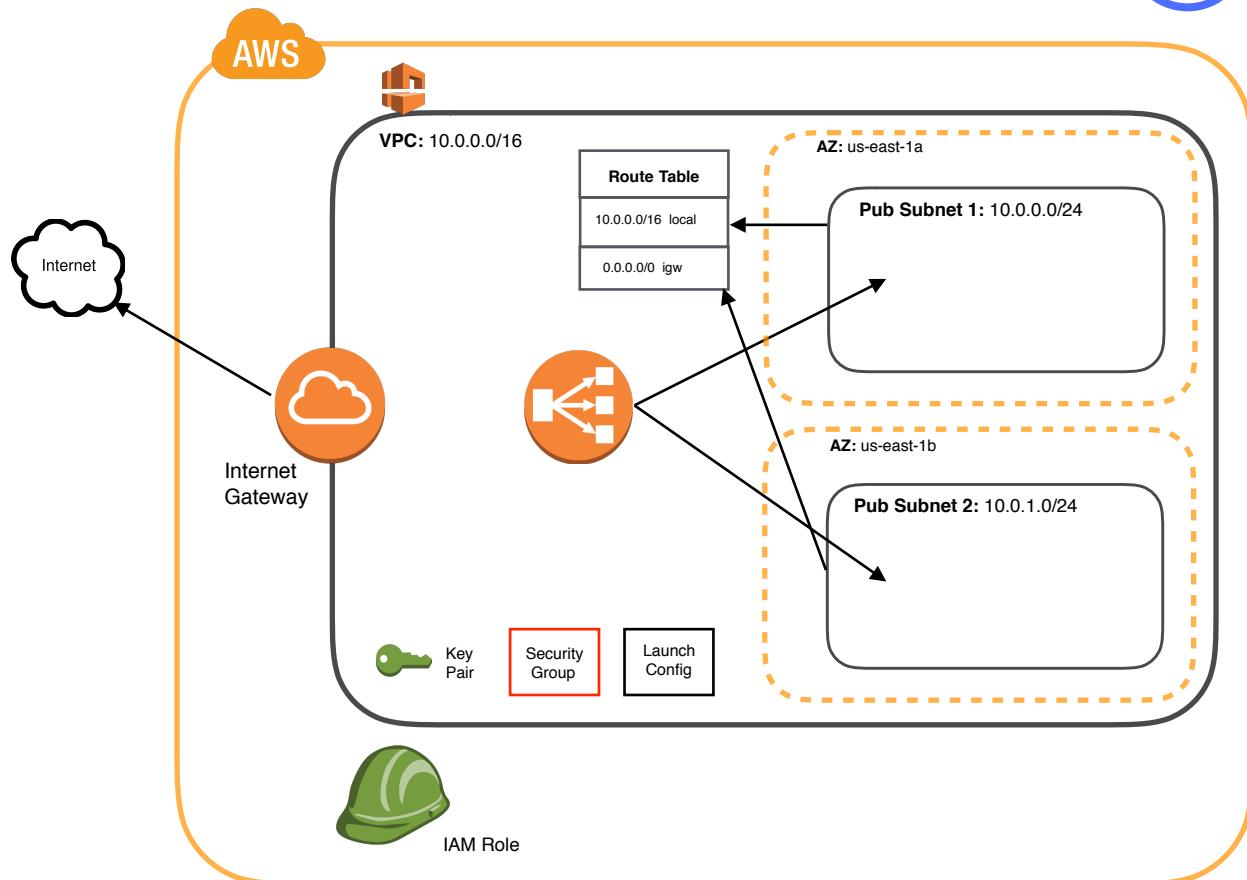
Select a key pair
slalom_aws_intro

I acknowledge that I have access to the selected private key file (slalom_aws_intro.pem), and that without this file, I won't be able to log into my instance.

Create launch configuration

System Architecture: Step 9

9



Step 10: Create an AutoScaling Group

1. Navigate to **Services** → **EC2**
2. In the left menu, scroll down and select **Auto Scaling Groups**:

The screenshot shows the AWS EC2 Resource Groups interface. On the left sidebar, under the 'AUTOSCALING' section, the 'Launch Configurations' and 'Auto Scaling Groups' options are listed. The 'Auto Scaling Groups' option is highlighted with a red box. The main content area displays various EC2 resources: 0 Running Instances, 0 Dedicated Hosts, 2 Volumes, 2 Key Pairs, 0 Placement Groups, 0 Elastic IPs, 2 Snapshots, 0 Load Balancers, and 7 Security Groups. A promotional message for Amazon Lightsail is visible. At the bottom right, there is a 'Launch Instance' button and a note about launching instances in the US East (N. Virginia) region.

3. Click the blue **Create Auto Scaling group** button at the bottom right
4. Using the **Create an Auto Scaling group from an existing launch configuration** option, select the Launch Configuration **slalom_aws_intro_web_lc** that you just created:

The screenshot shows the 'Create Auto Scaling Group' wizard. It starts with a brief description of what an Auto Scaling group is and how it uses launch configurations. Below this, two options are presented: 'Create a new launch configuration' (radio button unselected) and 'Create an Auto Scaling group from an existing launch configuration' (radio button selected). A search bar labeled 'Filter launch configurations...' is available. A table lists existing launch configurations: 'slalom_aws_intro_web_lc' (selected, indicated by a blue checkbox), which has an AMI ID of 'ami-0b33d91d' and an instance type of 't2.micro'; and 'slalom-simple-lc', which has an AMI ID of 'ami-9ffd0089' and an instance type of 't2.micro'.

5. In the Create AutoScaling Group box, enter the following options:

Group name: slalom_aws_intro_web_asg

Group Size: 2 instances

Network: slalom-aws-intro

Subnet: Click in the box and add both of the public subnets, **slalom-aws-intro-pub1** and **slalom-aws-intro-pub2**

6. On the same page, click the arrow beside the *Advanced Details* option and enter the following:

Load Balancing: Select the checkbox to Receive traffic from one or more load balancers

Classic Load Balancers: Click in the box and select **slalom-aws-intro-web-elb**

This should result in the following configurations:

1. Configure Auto Scaling group details 2. Configure scaling policies 3. Configure Notifications 4. Configure Tags 5. Review

Create Auto Scaling Group

Launch Configuration (i) slalom_aws_intro_web_lc

Group name (i) slalom_aws_intro_web_asg

Group size (i) Start with instances

Network (i) vpc-59e4c43f (10.1.0.0/16) | slalom-aws-intro Create new VPC

Subnet (i)
subnet-518bca34(10.1.1.0/24) | slalom-aws-intro-pub2
| us-east-1b

subnet-f5c1faae(10.1.0.0/24) | slalom-aws-intro-pub1 |
us-east-1a

Each instance in this Auto Scaling group will be assigned a public IP address. (i)

▼ Advanced Details

Load Balancing (i) Receive traffic from one or more load balancers [Learn about Elastic Load Balancing](#)

Classic Load Balancers (i) slalom-aws-intro-web-elb

Target Groups (i)

Health Check Type (i) ELB EC2

Health Check Grace Period (i) seconds

Monitoring (i) Amazon EC2 Detailed Monitoring metrics, which are provided at 1 minute frequency, are automatically enabled for the instances in this Auto Scaling group. You can change this setting in the [Monitoring tab](#).

7. Once you have the correct configuration, click **Next: Configure scaling policies**
8. Select the **Keep this group at its initial size** option. Click **Next: Configure Notifications**
9. We don't need to configure any notifications, so continue to click **Next: Configure Tags**
10. Add a tag with the following values:

Key = Name

Value = slalom_aws_intro_web_asg

Create Auto Scaling Group

A tag consists of a case sensitive key-value pair that you can use to identify your group. For example, you could define a tag with Key = Environment and Value = Production. You can optionally instances in the group when they launch. [Learn more](#).

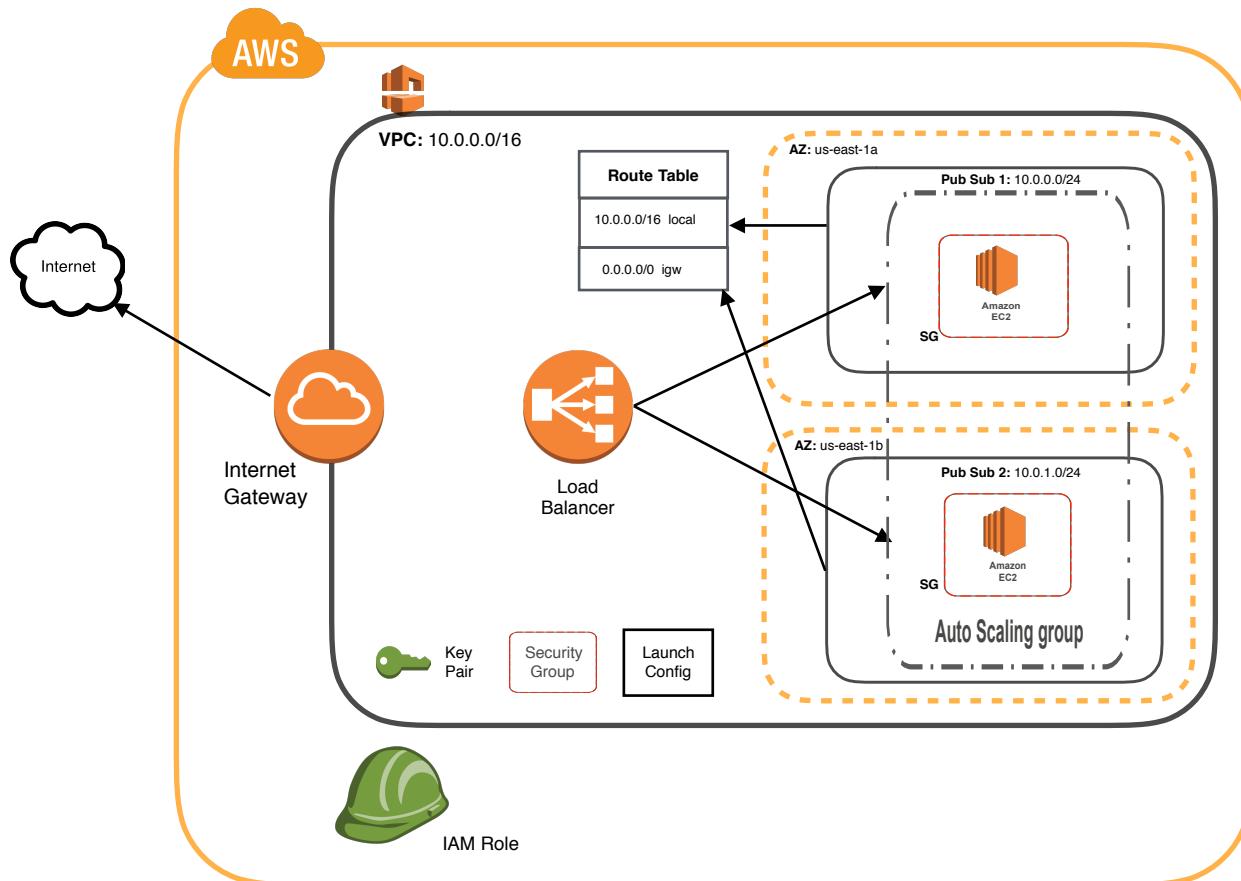
Key	Value	Tag New Instances
Name	slalom_aws_intro_web_asg	<input checked="" type="checkbox"/>

Add tag 49 remaining

11. Click the blue **Review** button once you have configured the Name Tag
12. Continue to click the blue **Create Auto Scaling group** button

System Architecture: Step 10

10



Step 11: Test it out!

1. Navigate to **Services** → **EC2**
2. In the left menu, select **Load Balancers**
3. Navigate to the **slalom-aws-intro-web-elb** load balancer
4. Copy and paste the value in the DNS Name column

The screenshot shows the AWS EC2 Load Balancers page. On the left, there's a sidebar with links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Instances, Spot Requests, Reserved Instances, Scheduled Instances, and Dedicated Hosts. The main area has a 'Create Load Balancer' button and an 'Actions' dropdown. A search bar labeled 'Filter: Search' is present. A table lists one load balancer:

Name	DNS name	State	VPC ID
slalom-aws-intro-web-elb	slalom-aws-intro-web-elb-334478801.us-east-1.elb.amazonaws.com	Active	vpc-59e4c43f

5. Navigate to a web browser (Chrome / Firefox etc.), paste the value in and hit enter
6. It may take a few minutes for the EC2 instances to finish creating, so the page may not load immediately within your chosen web browser
7. If all has gone to plan, you should see a Hello message!



System Architecture: Step 11

11

