Elastic Load Balancing, Amazon CloudWatch, and Auto Scaling

EXERCISE - 4.1

Create an Elastic Load Balancing Load Balancer

In this exercise, you will use the AWS Management Console to create an Elastic Load Balancing load balancer.

1. Launch an Amazon EC2 instance using an AMI with a web server on it, or install and configure a web server.
2. Create a static page to display and a health check page that returns HTTP 200. Configure the Amazon EC2 instance to accept traffic over port 80.
3. Register the Amazon EC2 instance with the Elastic Load Balancing load balancer, and configure it to use the health check page to evaluate the health of the instance.

EXERCISE - 4.2

Use an Amazon CloudWatch Metric

1. Launch an Amazon EC2 instance.
2. Use an existing Amazon CloudWatch metric to monitor a value.

EXERCISE - 4.3

Create a Custom Amazon CloudWatch Metric

1. Create a custom Amazon CloudWatch metric for memory consumption.
2. Use the CLI to `PUT` values into the metric.

EXERCISE - 4.4

Create a Launch Configuration and Auto Scaling Group

1. Using the AWS Management Console, create a launch configuration using an existing AMI.
2. Create an Auto Scaling group using this launch configuration with a group size of four and spanning two Availability Zones. Do not use a scaling policy. Keep the group at its initial size.
3. Manually terminate an Amazon EC2 instance, and observe Auto Scaling launch a new Amazon EC2 instance.

EXERCISE - 4.5

Create a Scaling Policy

1. Create an Amazon Cloud Watch metric and alarm for CPU utilization using the AWS Management Console.

2. Using the Auto Scaling group from Exercise 5.4, edit the Auto Scaling group to include a policy that uses the CPU utilization alarm.
3. Drive CPU utilization on the monitored Amazon EC2 instance(s) up to observe Auto Scaling.

EXERCISE - 4.6

Create a Web Application That Scales

1. Create a small web application architected with an Elastic Load Balancing load balancer, an Auto Scaling group spanning two Availability Zones that uses an Amazon CloudWatch metric, and an alarm attached to a scaling policy used by the Auto Scaling group.
2. Verify that Auto Scaling is operating correctly by removing instances and driving the metric up and down to force Auto Scaling.