CS15-319 / 15-619
Cloud Computing

Recitation 6
Sept 30\textsuperscript{th} and Oct 2\textsuperscript{nd}, 2014
Announcements

• The team information form is open until midnight **October 4th** Saturday. You should form teams and submit their information prior to the deadline.

• Checkpoint Quizzes
  – The project checkpoint quizzes are not timed and you have **three** attempts.
  – The unit checkpoint quizzes are timed and you have **one** attempt.
  – Read the checkpoint quiz instructions carefully.
Announcements

• Protect your AWS account!
  – Protect your credentials
  – Do not give anyone access to your account
• Budget Control!
  – Do remember to **TERMINATE** instances, ELB and ASG when you are done
  – You will incur penalties if you spend more than the required budget
Announcements

• **Do not** cheat!
  – Some suspected cases were found
  – We are using various tools to identify cheating
  – You learn nothing when you cheat

• Manual Grading
  – Will be done **one week** after the deadline
Announcements

- Please check the pinned post on Piazza for any updates and changes to this project’s write-up
  - P2.3 new AMI

- Ask proper questions on Piazza
  - Search Piazza or the internet (Google) before asking
  - Duplicate questions will be deleted
  - Posting solutions on Piazza will be deleted
  - The TAs will not debug for you, even if a private post
  - AWS SDK doc will answer most of your programming questions
Last Week Review

- Elastic Load Balancing
  - Junior System Architect at the MSB
- Auto Scaling Group
  -- Why is auto scaling useful?
- Cloud Watch
Last Week Review

- Scale Amazon EC2 capacity automatically according to conditions you define.
Elasticity

- Horizontal Scaling
  - Scale out
  - Scale in

Launchpad

Benchmark Script

Elastic Load Balancer

Web Server Pool

Instance: 1
Instance: 2
Instance: 3
Instance: 4
Amazon’s CloudWatch Alarm

- Monitor CloudWatch metrics for some specified alarm conditions
- Take automated action when the condition is met

![Diagram of CloudWatch Alarm Process]

- Resources with CloudWatch Enabled
  - CPU Utilization
  - Other Metrics...
  - CloudWatch Metrics Repository
  - CloudWatch Alarm
  - User-Defined Action

Amazon CloudWatch
Piazza Questions

• Do not use t1.micro when running load test (blank results) and t2.micro (does not support our AMI, does not use paravirtualization)

• Some students use too many instances, which causes too much delay at the ELB while dividing traffic

• Choose the correct metrics
  • Network In versus CPU Util
ELB Needs Warming Up

• ELB has a starting point for its initial capacity, and it will scale up or down based on traffic
• It is recommended that the load is increased at a rate of no more than 50 percent every five minutes
• You need to figure out how long the warm up should be, so you can achieve a good performance
Calculate instance-hours

• An instance-hour is the unit of cost when one m3.medium instance is billed for a period of one hour.
• For the purposes of our tests, time-warping makes 1 hour of MSB time correspond to 5 minutes of real time.
• The instance-hours for m1/m3.small/medium/large are based on the on-demand prices.
Calculate instance-hours

Piazza@865

• Take the maximum number of instances running in a 5 minutes window and aggregate them
• Instance hours for the first 40 minutes in the example is: 4+3+3+4+4+4+5+5
Unit 3: Virtualizing Resources for the Cloud

• UNIT 3: Virtualizing Resources for the Cloud
  – Module 6: Introduction and Motivation
  – Module 7: Virtualization
  – Module 8: Resource Virtualization - CPU
    – Module 9: Resource Virtualization - Memory
    – Module 10: Resource Virtualization – I/O
  – Module 11: Case Study
  – Quiz 3: Virtualizing Resources for the Cloud
Module 8: Resource Virtualization - CPU

- The Conditions for Virtualizing ISAs
- Full Virtualization and Paravirtualization
- Emulation
- Virtual CPU
This week - Project

• Introduction and APIs
  – MSB Recruitment Exam

• Elastic Load Balancing
  – Junior System Architect at the MSB

• Auto Scaling on Amazon
  – MSB Management Test
Failure Optimization

Imagine you have a service that may fail, you have three reactions:
(1) Monitor the service, to learn when it will fail
(2) Diagnose the failure and find out the reason
(3) Optimize the system

Project 2.3 you will do (1) and (3)
Amazon’s SNS

- Simple Notification Service
- Fast and flexible messaging service
- **Publishers** push when certain events happen
- Messages belong to **topics**
- **Subscriber** will instantly receive messages from the topic after they subscribe to it
Your Task

- Write a program to create an Elastic Load Balancer (ELB) and an Auto-Scaling Group (ASG) linked to ELB.
- Create CloudWatch Alarms.
- Configure SNS and publish ASG messages to SNS.
- Run the load generator and observe changes.
Highlight of Project 2.3

• You need to figure out which is the best instance type for the test

• Run once with browser only and find out the number of instances first

• In 2.3, instances may automatically shut down, you need to be able to track this
Resources

• Amazon’s Auto Scaling Service
  – http://aws.amazon.com/autoscaling/

• Amazon’s CloudWatch Alarm
  – http://aws.amazon.com/cloudwatch/

• Amazon’s SNS (Simple Notification Service)
  – http://aws.amazon.com/sns/

• Amazon’s Scaling Developer
Upcoming Deadlines

• Project 2.3 (Due Oct 5 11:59PM Pittsburgh)

<table>
<thead>
<tr>
<th>Assignment</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutoScaling on Amazon</td>
<td>Opens on 9/29/14 12:01 AM</td>
</tr>
<tr>
<td>(Gradebook) (Learning Dashboard)</td>
<td></td>
</tr>
<tr>
<td>MSB Management Test</td>
<td>Checkpoint</td>
</tr>
<tr>
<td></td>
<td>Not yet available</td>
</tr>
</tbody>
</table>

• Unit 3

<table>
<thead>
<tr>
<th>Module 8: Resource Virtualization - CPU</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Gradebook) (Learning Dashboard)</td>
<td>Opens on 9/29/14 12:01 AM</td>
</tr>
</tbody>
</table>