CSC 3322: Computer Architecture

AWS PROJECT: SPRING 2018

Amazon Web Services (AWS) is one of the major and most widely used cloud computing provider. AWS allows subscribers to have at their disposal a full-fledged virtual cluster of computers, available all the time, through the Internet. In this individual project, you are going to learn about cloud computing and will be exposed to Amazon EC2 Web Services. The objectives of this project are as follows:

- Setup an Amazon Web Services account.
- Create two EC2 instances with different software platforms.
- Benchmark the two instances with two different benchmarking software.
- Report your findings of the benchmark.

Project Milestones and Deliverables:

Part 1 (10%):

Milestone:

1. Create an AWS account.

Deliverable:

1. A screenshot of your AWS EC2 dashboard.

Useful links:

1. Tutorial slides.
2. Setting up AWS account: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/get-set-up-for-amazon-ec2.html#sign-up-for-aws
3. First 7 minutes of https://www.youtube.com/watch?v=ubCNZRNjhyo

Part 2 (50%):

Milestone:

1. Create two Amazon Linux AMI micro instance using the EC2 dashboard.

Deliverable:

1. A screenshot of your AWS EC2 dashboard after two instances were created.

Useful links:

1. Tutorial slides.
2. Tutorial video.
3. AWS: https://aws.amazon.com
4. AWS getting started guide: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html
5. First 15 minutes of https://www.youtube.com/watch?v=ubCNZRNjhyo

Part 3 (40%):

Milestone:

1. Run both benchmarking software in both instances and gather data (more information later).

Deliverable:

1. A report that will list what data has been collected; how they are compared and what the data reveals about the two instances. More details bellow.

2. A screenshot of your AWS EC2 dashboard's "Monitoring" tab for each of the instances.

Useful links:

1. Tutorial slides.
2. Tutorial video.
3. Systester: http://systester.sourceforge.net/
4. Phoronix: http://www.phoronix-test-suite.com/

For Systester:

Systester is a CPU-bound program that calculates the n^{th} digit of PI. In your benchmarking, you're going to use the Use Gauss-Legendre algorithm for different *n* and record the time it takes to calculate. You are going to run the same test in both of your AWS instances and then compare them to see which AWS instance is faster.

To run Systerster for Gauss-Legendre algorithm with different n values, use the following command:

```
./systester-cli  -gausslg  n  -bench
```

The value for n should be 128K, 256K, 512K, 1M, 2M, 4M, 8M, 16M. So, for instance, to run Systerster with n=1M, use the following command:

```
./systester-cli  -gausslg  1M  -bench
```

You will see something like the following:

```
Loop Digits State   Time
 ---------------------------------
   0     1  N/A   0h  0m  0.308s
   1     2  N/A   0h  0m  0.524s
   2     4  N/A   0h  0m  0.805s
   3     8  N/A   0h  0m  1.086s
   4    16  N/A   0h  0m  1.367s
   5    32  N/A   0h  0m  1.649s
   6    64  N/A   0h  0m  1.929s
   7   128  N/A   0h  0m  2.208s
   8   256  N/A   0h  0m  2.489s
   9   512  N/A   0h  0m  2.770s
  10    1K  N/A   0h  0m  3.051s
  11    2K  N/A   0h  0m  3.332s
  12    4K  N/A   0h  0m  3.613s
  13    8K  N/A   0h  0m  3.894s
  14   16K  N/A   0h  0m  4.177s
  15   32K  N/A   0h  0m  4.457s
  16   64K  N/A   0h  0m  4.733s
  17  128K  N/A   0h  0m  5.011s
  18  256K  N/A   0h  0m  5.286s
  19  512K  N/A   0h  0m  5.547s
  20    1M  N/A   0h  0m  5.728s ← This is the time it took for n=1M, record that
 Checksum validation...Gauss PASSED!!!
```

Run each Systester benchmark for an individual n, 5 times and then calculate their average. Once you run Systester for different n values and recorded the average time, create a line graph of that, where the x-axis will show the different n. Create two such line graphs for two instances and compare them in your report and see whether you can conclude anything looking at the graphs.

For Phoronix

Phoronix is a benchmarking suit that has a list of benchmarking tools for different parts of the computer. In this project, we are going to use the *iozone* benchmarking tool to benchmark computer's IO read/write operations. You are going to run Phoronix for different record size (4KB and 64KB) and file size (for 512 MB, 2 GB and 8 GB). Additionally, each of these tests should be done for both write and read performance.

When you run Phoronix using the following command:

```
./phoronix-test-suite   benchmark   iozone
```

You will see the following:

```
Phoronix Test Suite v7.8.0

  Installed:    pts/iozone-1.9.3

IOzone 3.465:
  pts/iozone-1.9.3
  Disk Test Configuration
    1: 4Kb
    2: 64Kb
    3: 1MB
    4: Test All Options
    Record Size: 1  ← My input

    1: 512MB
    2: 2GB
    3: 4GB
    4: 8GB
    5: Test All Options
    File Size: 1 ← My input

    1: Write Performance
    2: Read Performance
    3: Test All Options
    Disk Test: 2 ← My input

System Information

 PROCESSOR:       Intel Xeon E5-2676 v3 @ 2.40GHz
  Core Count:     1
  Extensions:     SSE 4.2 + AVX2 + AVX + RDRAND + FSGSBASE
  Cache Size:     30720 KB
  Microcode:      0x3c

 GRAPHICS:        Cirrus Logic GD 5446

 MOTHERBOARD:       Xen HVM domU
  BIOS Version:    4.2.amazon
  Chipset:       Intel 440FX- 82441FX PMC
 MEMORY:          1024MB
 DISK:           8GB
  File-System:    xfs
  Mount Options:   attr2 inode64 noatime noquota rw
  Disk Scheduler:  NOOP

 OPERATING SYSTEM:   Amazon Linux 2.0
  Kernel:        4.9.76-38.79.amzn2.x86_64 (x86_64)
  Compiler:       GCC 7.3.1 20180303
  System Layer:    Xen HVM domU 4.2.amazon
  Security:       KPTI + Full generic retpoline Protection

  Would you like to save these test results (Y/n): n ← My input

IOzone 3.465:
  pts/iozone-1.9.3 [Record Size: 4Kb - File Size: 512MB - Disk Test: Read Performance]
  Test 1 of 2
  Estimated Trial Run Count:    3
  Estimated Test Run-Time:      3 Minutes
  Estimated Time To Completion: 5 Minutes [17:52 UTC]
    Started Run 1 @ 17:48:06
    Started Run 2 @ 17:48:23
    Started Run 3 @ 17:48:41

  Record Size: 4Kb - File Size: 512MB - Disk Test: Read Performance: ← This shows what we just benchmarked
    4472.509765625
    4467.0107421875
    4403.4248046875

  Average: 4447.65 MB/s ← This is what we need to record
  Deviation: 0.86%

  OpenBenchmarking.org Dynamic Comparison:
  MB/s > Higher Is Better
  miami-iozone .. 2935.96 |======================================
  Virtual disk .. 3149.37 |========================================
  MIAMI-IOZONE .. 3226.26 |=========================================
  Result ........ 4447.65 |=========================================================
  Result Perspective: https://openbenchmarking.org/result/1706290-TR-MIAMIIOZO22
```

Run each benchmark for a record size/file size, 5 times and then calculate their average. You need to create a graph for reading performance that will show the effect of record size with file size. You need to create a similar graph for writing performance as well. Once you get those done for one instance; repeat the same benchmarking for the other instances. Finally compare graphs from both instances and draw some conclusions.

CSC 3322: Computer Architecture

AWS Project: Spring 2018

Each Group's work distribution:

Task	Member 1	Member 2	Important Notes
Create an AWS account	Together as a team		Designate 1 member for using the credit card information for verification purposes. **Both members should know the user name and password of the account so that they can work individually.**
Install EC2 instances	Instance 1	Instance 2	Each member should install 1 instance and then install the two benchmarking software in their instance. They should then run those benchmarking software in their instance and gather data as described above.
Install Systester	X	X	
Install Phoronix	X	X	
Run and gather data using Systester	X	X	
Run and gather data using Phoronix	X	X	
Write the report with screenshots, graphs and description	Together as a team		Put a coversheet and list all group member's name.

Groups:

Group Number	Members	Group Number	Members
1	Bodrick, Miles	9	Harris, Roy
	Cooper, Leah		Nwangwu, Christian
2	Abdur-Raheem, Ameenah	10	Sutton, Courtney
	Thompson, Martine'		Phillips, Christen
3	Weanquoi, Patrickson	11	Walker, Jordan
	Kouame, Raymond		Norman, Shammond
4	Vanhook, Brandon	12	Caines, Daniel
	Clavijo, Carlos		Crump, Jasmine
5	Colquhoun, Eric	13	Deese, Alexus
	Chance, Samuel		Douglas, Kenneth
6	Byers, Jalen	14	Hairston, Haven
	Cousins, Sebastian		Kennedy, Michael
7	Cornwell, Trevon	15	Hughes, Gregory
	Milton, John		Martin, Jonathan
8	Huff, Sean	16	Holmes, Brandon
	Hodge, Austin		Senegal, John