Cloud Computing
Winter Term 2021/2022

Practical Assignment No. 1

Due: 19.11.2021 11:55 pm

For this assignment, you will explore a cloud computing system from user perspective. The primary goal is to gain insight into performance characteristics of virtualization by benchmarking different cloud providers and comparing and evaluating the benchmark results. The secondary goal is to familiarize yourself with Amazon EC2 and Google Cloud, and the respective tools as an example of today's Infrastructure-as-a-Service clouds. Furthermore, this assignment serves as preparation for practical assignment no. 2.

1. Prerequisites

1.1. Student groups
We assume you already formed groups of 5 students on the course website in ISIS.

1.2. Unix Platform
Work on a Unix-like platform to solve the assignments. Linux is recommended, Mac OS works as well. If you use Windows privately, work in a virtual machine (either locally, such as VirtualBox, or use one of the Cloud platforms).

1.3. Amazon AWS Free Tier
Create an Amazon AWS free tier account to work with the cloud platform and perform further tasks.

- Go to aws.amazon.com/de/free and go through the registration process.
  Important:
  - You will be asked to enter credit card information. Amazon will block 1$ for a few days to verify the credit card validity. The card will not be charged if you use free tier instances. For details, please check the tutorial slides.
  - Nobody in your group is has a credit card available, please write me an email (alexander.acker@tu-berlin.de). We will try to resolve the issue.
- AWS Documentation: Using the AWS CLI documentation and the Amazon EC2 online documentation you can find out how to setup and use the API and AMI tools.
- Configure your CLI tools so that they can access your AWS account, i.e., commands like the following should work:
  - `aws ec2 describe-instances`

Note: It is ESPECIALLY IMPORTANT to **shut down your VMs** when you are done working with them, otherwise your credit card might be charged! You will receive several emails before actual charging happens.

1.4. Google Cloud

- Visit this link to retrieve your GCP credit coupon:
  - Enter your TUB Email address
  - Each student can retrieve one coupon
  - Each coupon is worth 50$ and valid until 17.10.2022
  - We recommend to not redeem all coupons at once. Use one up and redeem next if required.
  - **Do not spend all google cloud credits on this assignment. You will need them for the following assignments as well.**
- You will receive an email with further instructions on how to create an account and redeem your coupon
- You can optionally check if you are eligible for additional free trial credit at https://cloud.google.com/free/
- Study the documentation of the Google Cloud SDK and learn how to set up and use the command line tools
- Configure your Google Cloud SDK that it can access your GCP account, i.e. commands like the following should work:
  - `gcloud compute instances list`

Note: You might run out of credits. To prevent this situation, do not redeem all credits at once and remember to **shut down your VMs when you do not use them!**

Do not spend all google cloud credits on this assignment. You will need them for the following assignments as well.

2. Virtual Machines

Write two well-commented <u>Shell scripts</u> that prepare and start a virtual machine on the AWS and GCP platforms, respectively.

- You can first manually experiment with the `aws` and `gcloud` commands and make notes, before finally combining the commands into a script.
- Each command in the scripts should be commented, make sure the scripts are clean and not cluttered with unnecessary text.
- After writing the scripts, test them: delete all running VMs, run the script, and check that the result is satisfactory.
- Do **not** include <u>any private information</u> such as your access key or secret key in your submission (if necessary, replace them with dummy strings).
- All requirements listed below should be covered by at least one command.

General requirements (apply for both platforms):
1. Generate a local SSH key pair and name the output files `id_rsa` and `id_rsa.pub`. Make sure to specify a valid username as the comment in the public key (required for GCP).

Requirements for AWS:
1. Ensure that you have a default VPC network in your AWS project.
2. Upload the generated public key as a key pair.
3. Create a security group and allow incoming ICMP and SSH traffic.
4. Run an instance with the following parameters:
   - The created key pair and security group
   - Instance type t2.micro (**free tier eligible**)
   - Image for "Ubuntu Server 18.04" (**free tier eligible**)
5. Resize the VM disk volume size to 100 GB

Requirements for GCP:
1. Prepare a modified copy of the public key as described in the GCP documentation. Use shell commands like `echo`, `cat` and file redirection ("`>`"). Note: the username in the prepared key must match the username specified with `ssh-keygen`.
2. Upload the public key into your project metadata
3. Create a Firewall rule that allows incoming ICMP and SSH traffic. The rule must apply only for VMs with the tag "cloud-computing".
4. Launch an instance with the following parameters:
   - Machine type e2-standard-2
   - Add the tag "cloud-computing"
   - Image for "Ubuntu Server 18.04".
5. Resize the VM disk volume size to 100 GB

<u>**Outputs:**</u>
- Scripts **create-aws.sh** and **create-gcp.sh**
  - Containing all commands for preparing and starting your VMs, including comments explaining what the commands do.

3. Prepare Performance Benchmarks

We will use the `sysbench` tool to run rudimentary benchmarks on our VMs. Manually install the tool on both VMs:

```
sudo apt update && sudo apt install -y sysbench
```

Write a benchmarking Shell script called `bench.sh` that uses the `sysbench` command to perform the 4 benchmarks described below.

The script must be executed without parameters and output a *single line* on the standard output that contains all measurements in CSV format, prefixed by the Unix epoch timestamp when the measurements were started:

1573143062,1058.52,9890.61,132.06,414.23

On the standard error output, the script can print arbitrary output.
The benchmarked resources are:

- CPU speed (events/s)
- Memory access (MiB/s)
  - Use block size of 4KB and a total size of 100TB (tera bytes)
- Random-access disk read speed (MiB/s)
- Sequential disk read speed (MiB/s)
  - For both disk tests, use 1 file with size 1GB, and make sure sysbench uses ***direct*** disk access to avoid caching effects.

Each benchmark must be executed for 60 seconds. The results of all 4 benchmarks must be collected in the script, then printed as one line, as shown above. Except for `sysbench`, the script is only allowed to use commands that are available in the default Ubuntu images used (don't install any additional packages with `apt` or similar tools).
Make sure that every command in your script is commented.

Hints:
- The sysbench tool outputs a lot of information that we do not need. Use typical Shell tools such as `grep` and `awk` to obtain only the necessary piece of output from each `sysbench` invocation.

Outputs:
- Script `bench.sh`

4. Execute Performance Benchmarks

Execute the prepared benchmarks on the following two Cloud platforms. Each benchmark must be executed once every 30 minutes for two days. The results must be collected in one CSV per VM. In the end, each CSV file should have at least 2*48=96 entries.

Example of such a CSV file:

```
time,cpu,mem,diskRand,diskSeq
1573143062,1058.52,9890.61,132.06,414.23
1573143092,1057.51,9891.62,133.07,415.24
1573143122,1056.50,9892.63,134.08,416.25
...
```

Before or after starting the benchmarks, make sure to manually add the header line.

Use cron to automate your benchmarks. Use the `crontab` and `echo` commands to add a single entry to the Crontab of your user. The entry should execute the run_bench.sh script and append its output to the respective CSV file.

Additional tasks:
- Add the two full `crontab` commands to the end of your `create-*.sh` scripts.
- After taking all benchmarks, use the `plot.py` Python script provided on ISIS to generate plots for your benchmarks. If the scripts complete without errors, your files have the correct format.
- Answer the questions below in a well-formatted, readable text.

Hints:
- When executing long-running experiments in a public cloud, remember to keep an eye on your credits. You should have more than enough credit to run these benchmarks, but make sure to shut down your VMs when you don't use them.

Outputs:
- Two CSV files **gcp_results.csv** and **aws_results.csv**
- 4 plots generated by `plot.py`:
    - **[cpu|mem|diskRand|diskSeq]-plot.png**
- Text with answers to questions
    - Please write the answers into the text field within the submission of assignment 1 or submit a .txt or .pdf file with the answers. Other file formats will not we accepted.
    - Please use max. 200 words per question

CPU benchmark questions:
1. Shortly describe, how sysbench performs CPU benchmark. What does the resulting events/s value represent?
2. Look at the plots of your long-term measurements. Do you see any seasonal changes?

Memory benchmark questions:
1. Shortly describe, how `sysbench` measures memory performance.
2. How would you expect virtualization to affect the memory benchmark? Why?

Disk benchmark questions:
1. Shortly describe, how `sysbench` performs the disk benchmarks.
2. Compare the results for the two operations (sequential, random). What are reasons for the differences?

General question:
1. Compare the overall long-term measurement plots for the two platforms AWS and GCP. Name one type of application that you would expect to perform better on AWS, and one that would perform better on GCP, respectively. Shortly explain your decisions.

5. Submission Deliverables

Submit your solution on the ISIS platform as individual files. Please submit ONLY the necessary files and use exactly the given file names!
Submit the text with answers to the questions directly in the submission text field or upload a .txt or .pdf file.

Expected submission files:

- `create-aws.sh`
- `create-gcp.sh`
  - **2 script files, 10 points each**
- `bench.sh`
  - **1 script file, 8 points**
- `gcp_results.csv`
- `aws_results.csv`
  - **2 CSV files, 1 points each**
- `[cpu|mem|diskRand|diskSeq]-plot.png`
  - **4 image files, 1 point each**
- Text / files with answers to questions
  - **1 point per question**

Total points: 41

> **Final Warning: Please make sure your VMs are shut down when you are not using them! Make good use of your credits :-)**
>
> **Do not spend all google cloud credits on this assignment. You will need them for the following assignments as well.**