TEAM LEAD VERSION (DevOps-Week-1)

TEAMWORK

CLARUSWAY
WAY TO REINVENT YOURSELF

Meeting Agenda

- ▶ Icebreaking
- ▶ Microlearning
- ▶ Questions
- ▶ Interview/Certification Questions
- ▶ Coding Challenge
- ▶ Article of the week
- ▶ Video of the week
- ▶ Retro meeting
- ▶ Case study / project

Teamwork Schedule

Ice-breaking 5m

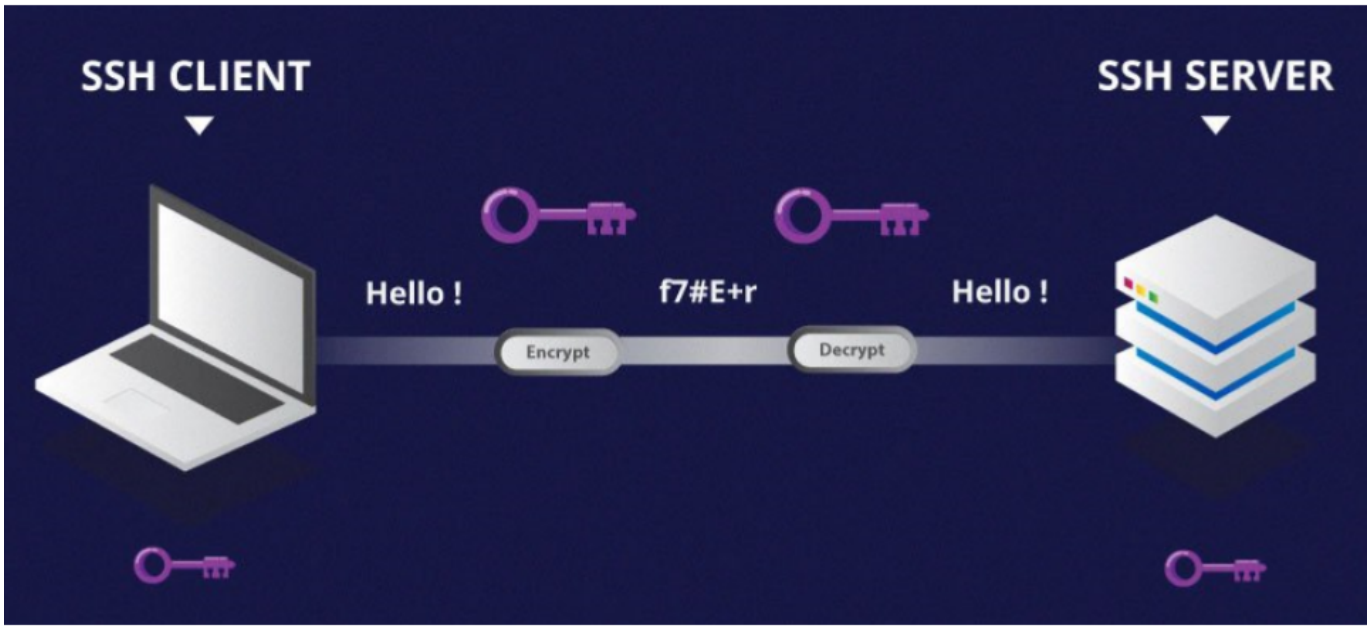
- Personal Questions (Stay at home & Corona, Study Environment, Kids etc.)
- Any challenges (Classes, Coding, AWS, studying, etc.)
- Ask how they're studying, give personal advice.
- Remind that practice makes perfect.

Team work 10m

- Ask what exactly each student does for the team, if they know each other, if they care for each other, if they follow and talk with each other etc.

Microlearning 15m

What is SSH command?



The **ssh** command provides a secure encrypted connection between two hosts over an insecure network. This connection can also be used for terminal access, file transfers, and for tunneling other applications.

Using the Linux client:

Linux typically uses the OpenSSH client. The ssh command to log into a remote machine is very simple. To log in to a remote computer which IP address is 172.31.7.57, type the following command at a shell prompt:

```
ssh 172.31.7.57
```

If this is the first time you use ssh to connect to this remote machine, you will see a message like:

```
The authenticity of host '172.31.7.57' cannot be established. DSA key fingerprint
is 04:48:30:31:b0:f3:5a:9b:01:9d:b3:a7:38:e2:b1:0c. Are you sure you want to
continue connecting (yes/no)?
```

Type **yes** to continue. This will add the server to your list of known hosts **(~/.ssh/known_hosts)** as seen in the following message:

```
Warning: Permanently added '172.31.7.57' (DSA) to the list of known hosts.
```

Each server has a host key *(A host key is a cryptographic key used for authenticating computers in the SSH protocol. Host keys are key pairs, typically using the RSA, DSA, or ECDSA algorithms. Public host keys are stored on and/or distributed to SSH clients, and private keys are stored on SSH servers.)*, and the above question related to verifying and saving the host key, so that next time you connect to the server, it can verify that it actually is the same server.

Once the server connection has been established, the user is authenticated. Typically, it asks for a password. For some servers, you may be required to type in a one-time password generated by a special hardware token.

Once authentication has been accepted, you will be at the shell prompt for the remote machine.

Specifying a different user name:

It is also possible to use a different username at the remote machine by entering the command as:

```
ssh alternative-username@172.31.7.57
```

The above can also be expressed with the syntax:

```
ssh -l alternative-username 172.31.7.57
```

Executing remote commands on the server:

The ssh command is often also used to remotely execute commands on the remote machine without logging in to a shell prompt. The syntax for this is:

```
ssh hostname command`
```

For example, to execute the command:

```
ls /tmp/doc
```

on host sample.ssh.com, type the following command at a shell prompt:

```
ssh 172.31.7.57  ls /tmp/doc
```

After authenticating to the remote server, the contents of the remote directory will be displayed, and you will return to your local shell prompt.

Note: The ssh command reads its configuration from the SSH client configuration file ``~/.ssh/config```.

Ask Questions 15m

1. How can we rename a branch ? (git)

A. git checkout -b new-branch-name
B. git branch checkout new-branch-name
C. git branch -m new-branch-name
D. git clone new-branch-name

Answer: *C*

2. Which command is used to terminate the Terraform-managed infrastructure?

A. terraform terminate
B. terraform erase
C. terraform delete
D. terraform destroy

Answer: *D*

3. Which command is used to list of the resources in state in Terraform?

A. terraform state --list
B. terraform show list
C. terraform state list
D. terraform ls state

Answer: *C*

4. Containers include the application and all of its dependencies, but share the kernel with other containers. They run as an isolated process in userspace on the host operating system. They're also not tied to any specific infrastructure – Docker containers run on any computer, on any infrastructure, and in any cloud.

A. True
B. False

Answer: *A*

5. What command should you run to see all running container in Docker?

A. docker run
B. docker ps
C. docker --help
D. docker build
E. docker pull

Answer: *B*

6. Which command is used to remove all the stopped containers, all the networks that are not used, all dangling images and all build caches?

A. docker system prune
B. docker login
C. docker pull
D. docker rm

Answer: *A*

7. What is this command used for? (Docker)

```
$ sudo docker run -i -t alpine /bin/bash
```

A. to stop the docker container
B. to see all running container in Docker
C. to run the image as a container
D. to copy the docker container

Answer: *C*

8. You can't create multiple containers from the same image?

A. True
B. False

Answer: *B*

9. How many containers can run per host?

A. 1
B. 100
C. 947
D. unlimited

Answer: *D*

10. Which of the following is not a state of Docker container?

A. Running
B. Freezed
C. Paused
D. Restarting
E. Exited

Answer: *B*

Interview/Certification Questions 20m

1. What are the different phases in DevOps?

The various phases of the DevOps lifecycle are as follows:

Plan: Initially, there should be a plan for the type of application that needs to be developed. Getting a rough picture of the development process is always a good idea.

Code: The application is coded as per the end-user requirements.

Build: Build the application by integrating various codes formed in the previous steps.

Test: This is the most crucial step of the application development. Test the application and rebuild, if necessary.

Integrate: Multiple codes from different programmers are integrated into one.

Deploy: Code is deployed into a cloud environment for further usage. It is ensured that any new changes do not affect the functioning of a high traffic website.

Operate: Operations are performed on the code if required. Monitor - Application performance is monitored. Changes are made to meet the end-user requirements.

2. Explain the concept behind Infrastructure as Code (IaC).

Answer:

Infrastructure as Code (IaC) is a process for managing and operating data servers, storage systems, system configurations, and network infrastructure.

In traditional configuration management practices, each minute configuration change required manual action by system administrators and the IT support team. But with IaC, all the configuration details are managed and stored in a standardized file system, wherein the system automatically manages infrastructure changes and deals with system configurations.

Therefore, we do not require most of the manual effort since everything is managed and automated by following the IaC approach. Tools such as Terraform or Ansible can be used to implement IaC approach.

3. How Terraform works?

Answer:

Terraform produce an execution plan delineate, what it will do to reach the desired state, and then executes it to build the described infrastructure. As the configuration changes, Terraform is able to determine what changed and create incremental execution plans which can be applied.

4. What is difference between virtualization and containerization?

Answer:

Containers provide an isolated environment for running the application. The entire user space is explicitly dedicated to the application. Any changes made inside the container is never reflected on the host or even other containers running on the same host. Containers are an abstraction of the application layer. Each container is a different application.

Whereas in Virtualization, hypervisors provide an entire virtual machine to the guest(including Kernal). Virtual machines are an abstraction of the hardware layer. Each VM is a physical machine.

5. What are Docker Images?

Answer:
Docker image is the source of Docker container. In other words, Docker images are used to create containers. When a user runs a Docker image, an instance of a container is created. These docker images can be deployed to any Docker environment.

Article of the Week 10m

- How to Use Git/GitHub without asking for authentication always: Passwordless Usage of Private Git Repositories

Video of the Week 10m

- Terraform Explained

Retro Meeting on a personal and team level 10m

Ask the questions below:

- What went well?
- What could be improved?
- What will we commit to do better in the next week?

Coding Challenge 5m

- Coding Challenge: Reverse Input Number

Case study/Project 10m

- Project-202: Phonebook Application (Python Flask) deployed on AWS Application Load Balancer with Auto Scaling and Relational Database Service using Terraform

Closing 5m

-Next week's plan

-QA Session