Kubernetes and Docker based Content Delivery Network Application (CDN_App)

Submitted to: Prof. Dr. Christian Baun

Presented by: TEAM - 14
Syed Ahmed Zaki (ID: 1322363), syed.zaki@stud.fra-uas.de
Daniel von Barany(ID: 1323045), daniel.vonbarany@stud.fra-uas.de

Cloud Computing (WS2021) Project Presentation
M.Sc. in High Integrity Systems
Date: Thursday 18th February, 2021

Faculty 2: Computer Science and Engineering
Frankfurt University of Applied Sciences
Frankfurt, Germany

OVERVIEW

- MOTIVATION
- USE CASES OF CDN
- POPULAR CDN PLATFORMS
- PROBLEM DESCRIPTION
- FINAL VERSION OF ARCHITECTURE DIAGRAM
- IMPLEMENTATION IN SHORT
- FINAL VIEW
- CONCLUSION and FUTURE GOALS
- REFERENCES
- LIVE PROJECT DEMONSTRATIONS

MOTIVATION

"I don't need a hard disk in my computer if I can get to the server faster... carrying around these non-connected computers is byzantine by comparison." - Steve Jobs, late chairman of Apple

1. We wanted to build a system on cloud that can make an easy way for users for delivering content in the shortest possible time based on location.
2. There are always some issues with downloading and uploading speed alongside the streaming and bandwidth limit.
3. To find out the solution for best performance relative to network limits and other connectivity issues.
4. Content Delivery Network (CDN) running on virtual machines can solve these problems.

USE CASES OF CDN

Why we need it:

1. We need to find content as little time as possible.
2. When several users try to reach the same website because of limited bandwidth.
3. To adjust the volume of requests according to the traffic.

When not to use it:

1. When the system has a smaller number of users then it has a smaller number of requests.
2. So, in that case it is not a good idea to use a content delivery network.

POPULAR CDN PLATFORMS

1. There are various paid cloud deployment platform available online.
2. Some popular ones name are mentioned here, for example: Amazon cloudfront [2], CloudFlare [3], Akamai technologies[1] etc.
3. Most of them offer several common services, some of them are: Firewall, Latency distribution, DDoS attacks protection, Load balancing, Faster content delivery which help to speed up websites, etc [1].

PROBLEM DESCRIPTION

The prototype in figure 1, was our first idea but in later implementation we slightly changed this.

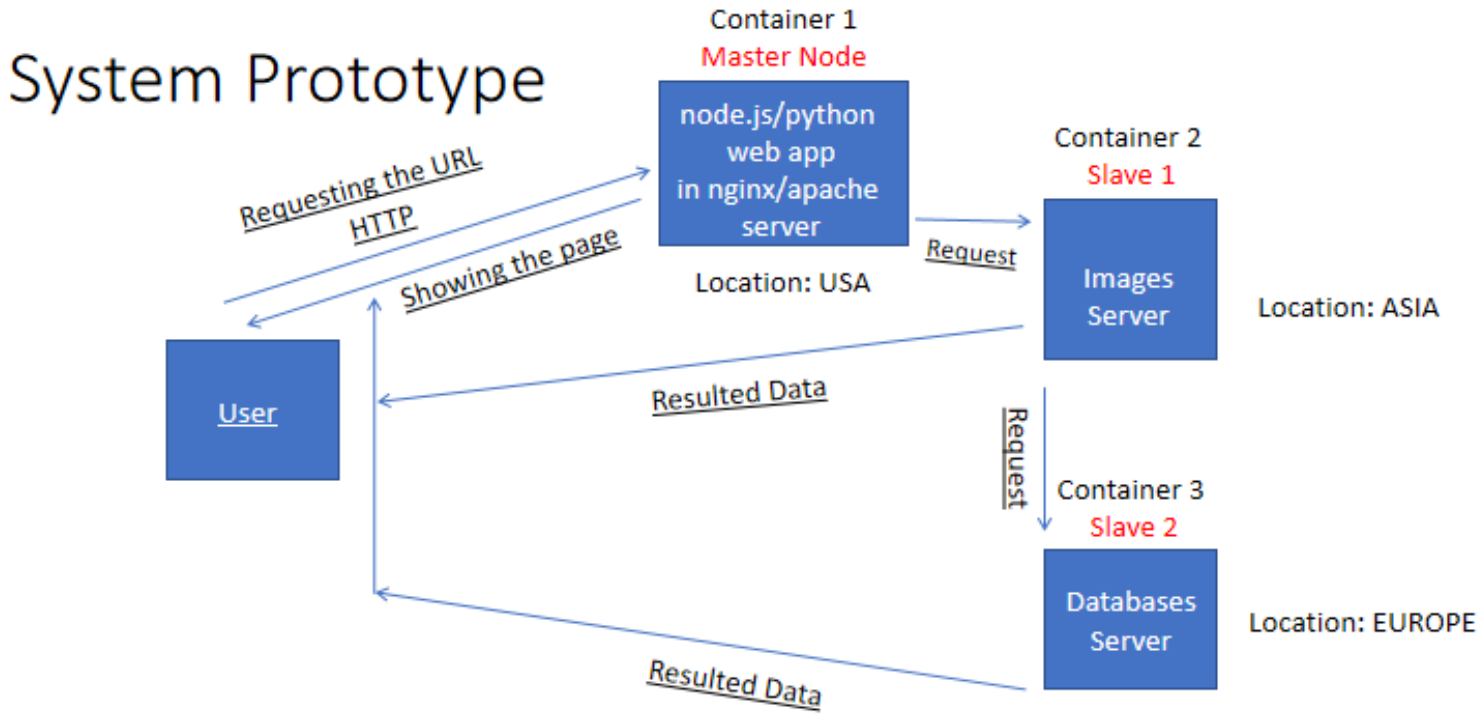

Fig: Multinode Cloud Architecture

Figure 1: Multinode Cloud Architecture Prototype

FINAL VERSION OF ARCHITECTURE DIAGRAM

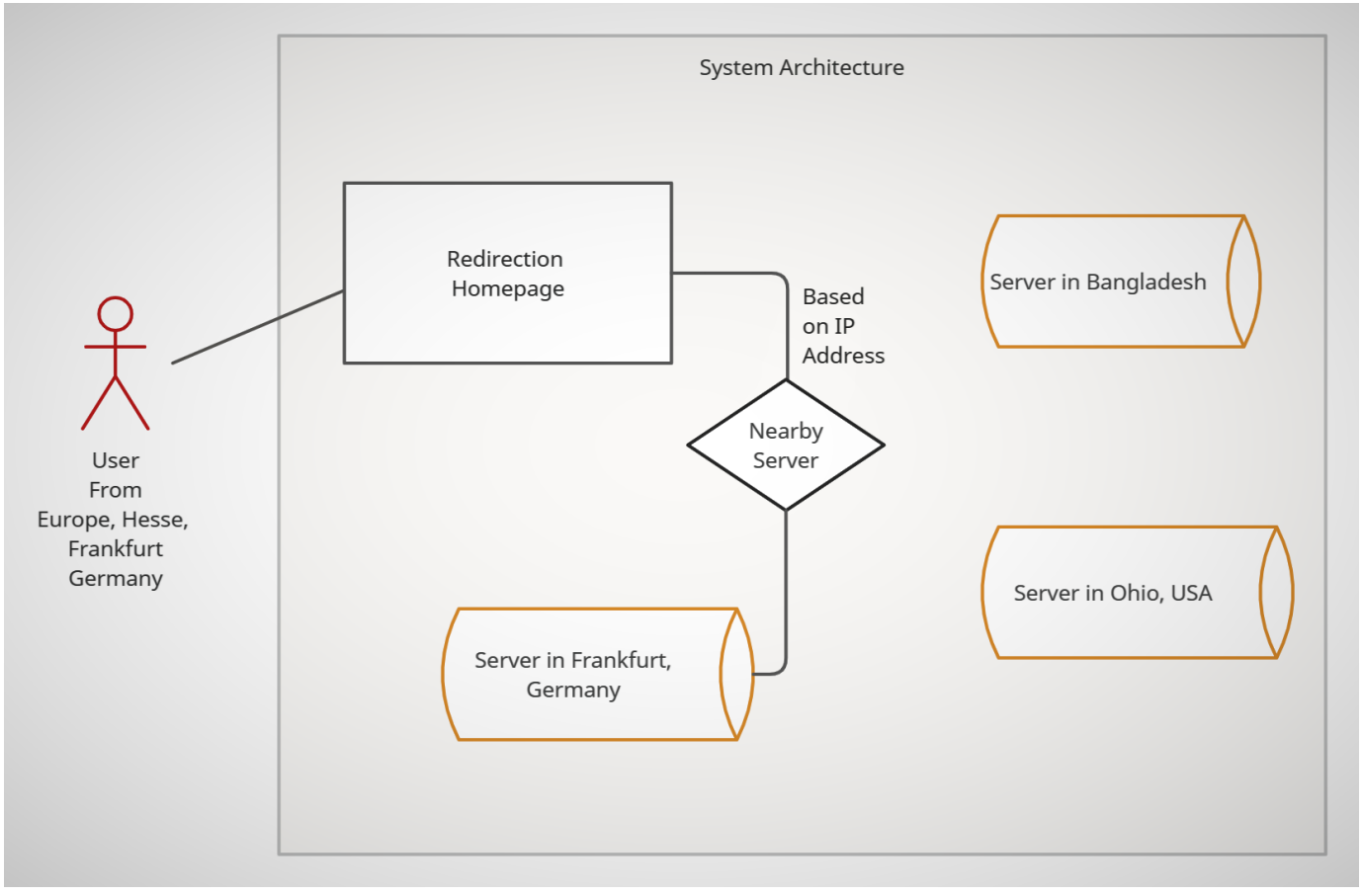

Figure 2: Updated Multinode Cloud Architecture Diagram

IMPLEMENTATION IN SHORT

1. We customized the Apache webserver and then virtualized with Docker.
2. Tested in AWS, VPS and local.
3. Kubernetes with multi node scalable cluster and pods.
4. Docker, kubernetes settings, project files available opensource in github: `https://github.com/enttty/cdn_project`
5. Docker image available publicly in docker hub:`https://hub.docker.com/r/entty/cloud_project`
6. For cloud deployment we used Amazon Web Services (AWS)
7. Different AWS services have been used: Elastic Compute Cloud (EC2), Elastic Block Store (EBS), Virtual Private Cloud (VPC), Elastic Kubernetes Service (EKS), Autoscaling and Load Balancer
8. Used ngrok in system to forward local machine to expose on internet

FINAL VIEW

1. In figure 3, the location of the user is taking from IP address then after 30 seconds it will go to certain page for redirection.
2. This webpage is built on JS, PHP, HTML, CSS, and IP API. Location detection and animation details are shared in our report.

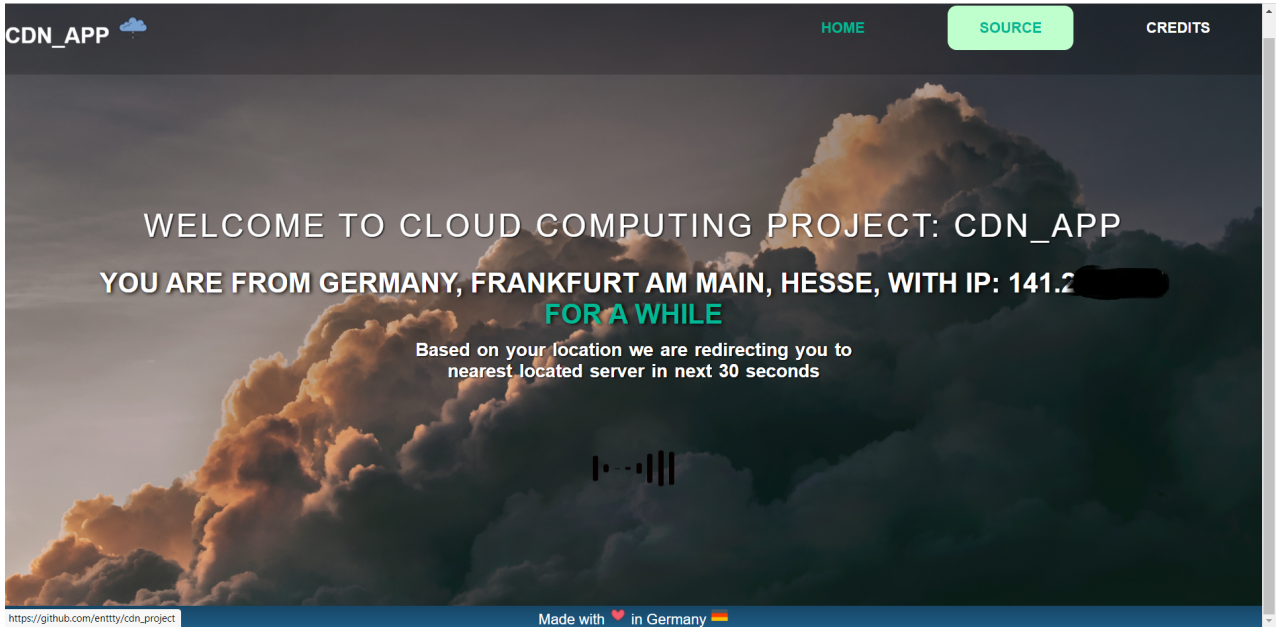

Figure 3: The homepage of IP redirection(IP has blurred for privacy)

CONCLUSION and FUTURE GOALS

CONCLUSION:

1. Building something different is not an easy task but from our perspective, we have tried our best to build it and learned this new technology of Docker, Kubernetes and AWS.
2. We have faced lots of trouble to build this project. But we enjoyed. Especially the teamwork.
3. Lots of sleepless nights to figure out where is the problem and how we can solve it.

FUTURE GOALS:

1. Plan to increase the nodes in different locations of the world though it is costly but it will make things faster and more accurate to user.
2. Plan to continue this project and interested to release updates.
3. You are welcome to contribute!

REFERENCES

[1] "Akamai technologies." [Online]. Available from: https://en.wikipedia.org/wiki/Akamai_Technologies. **Accessed: 03 February 2021.**

[2] "Content delivery network (cdn) — low latency, high transfer speeds, video streaming — amazon cloudfront." [Online]. Available from: https://aws.amazon.com/cloudfront/. **Accessed: 03 February 2021.**

[3] "Cloudflare - the web performance & security company—cloudflare." [Online].Available from:https://www.cloudflare.com/. **Accessed: 03 February 2021.**

LIVE PROJECT DEMONSTRATIONS

LIVE DEMO