Decentralized Virtual CDN

Cloud System Administration

Carlos Santos - 201607406
Rui Silva - 202005661

Supervisor:
Rolando Martins

University of Porto
Faculty of Science
Computer Science Department
May 2024

1 Introduction

The advent of the General Data Protection Regulation (GDPR) has profoundly transformed the landscape of data management and privacy, imposing stringent responsibilities on system architects, system administrators, and programmers. They are now tasked with developing innovative tools and frameworks that enable Data Protection Officers (DPOs) to ensure compliance with the legal requirements. A growing focus on data sovereignty is a huge challenge in an evolving regulatory environment, with regulators demanding privacy first solutions and stances that are often at odds with the business models of major technology companies. This lab assignment aims to explore viable solutions aimed at aligning with GDPR's stringent requirements. It proposes a decentralized virtual Content Delivery Network (CDN) incorporating opportunistic offloading strategies. This approach not only seeks to enhance data privacy and sovereignty but also aims to optimize performance and resource utilization within the constraints set by GDPR. And our goal during this project was to develop a system capable of meeting the demands of regulatory compliance, that prioritizes performance and cost-effectiveness.

2 Solution Architecture

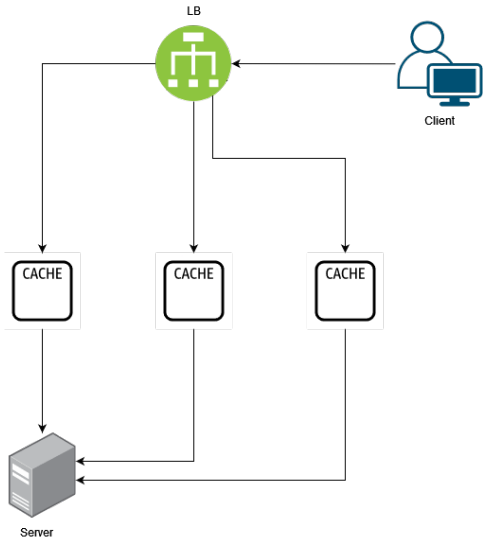

Figure 1: System Architecture

Our decentralized CDN system is composed of a Load Balancer, a central Server, and multiple Caches distributed across various regions, as showed in Figure 1:

- The **Load Balancer** serves as the entry point for all client requests. It is responsible for routing client requests to the nearest cache based on the client's geographic location, ensuring that clients receive the fastest possible response.
- The **Regional Caches** are strategically located to store frequently requested content in different regions. The Load Balancer reach each one of the regional Caches through the **instance groups**.
- The **Central Server** acts as the origin server and the authoritative source of content.

To optimise the delivery of content, the architecture is designed to minimise the latency and to efficiently manage resources.
The load balancer receive the request from the client when it's initiated, and then route that request to the nearest regional cache. If the cache has the content, it responds directly to the client. If not, the cache forwards the request to the server. Once the server retrieves the content and sends it to the cache, it stores the content locally and then responds to the client.
This approach ensures that future requests for the same content can be served directly from the cache, improving response times and reducing the workload of the central server.

3 Implementation

We configured our Content Delivery Network solution using Google Cloud Platform (GCP), however those actions can be automated, using a Python script designed to interact with GCP services using the **gcloud** command-line tool and the Google Cloud Python client library. The script has the following functions:

- **create_vm** creates a new virtual machine instance on the basis of the specified machine image.
- **create_unmanaged_instance_group** creates an unmanaged instance group, that allows the management a group of VM instances.
- **add_instance_to_unmanaged_group** adds a VM to an existing unmanaged instance group.
- **create_regional_health_check** creates a regional HTTP health check for monitoring the health of instances.
- **create_backend_service** creates a backend service, that defines how traffic is distributed among a group of VM instances.

- **add_backend_to_service** adds a backend to an existing backend service.
- **attach_instance_group_to_lb** attaches an instance group to a load balancer backend service.

4 Conclusion

The implementation of a decentralized virtual CDN is designed to respond strategicly to the challenges that strict data protection legislation poses, such as the General Data Protection Regulation. This solution is in line with the changing regulatory landscape and addresses compliance requirements by focusing on data protection and sovereignty, while optimising performance and resources.
We proposed an architecture, which consists of a load balancing server, regional caches, and a central server, expecting to build a system that ensures low-latency access to content while minimizes the workload on the central server by using intelligent caching strategies. However, due to time constraints, we were not able to carry out performance tests for evaluating our CDN's scalability,throughput and latency. Nevertheless, we can conclude that our solution achieved the main goal.