Understanding & High Level Design Document

Version: 1.2

Document Revision History

Version	Date	Author/Team	Comments
1.0	22-11-2018	XXXX	Initial Version
1.1	28-10-2021	Siddharth Jagtiani	Adding details and structure for clarity to students

TABLE OF CONTENT

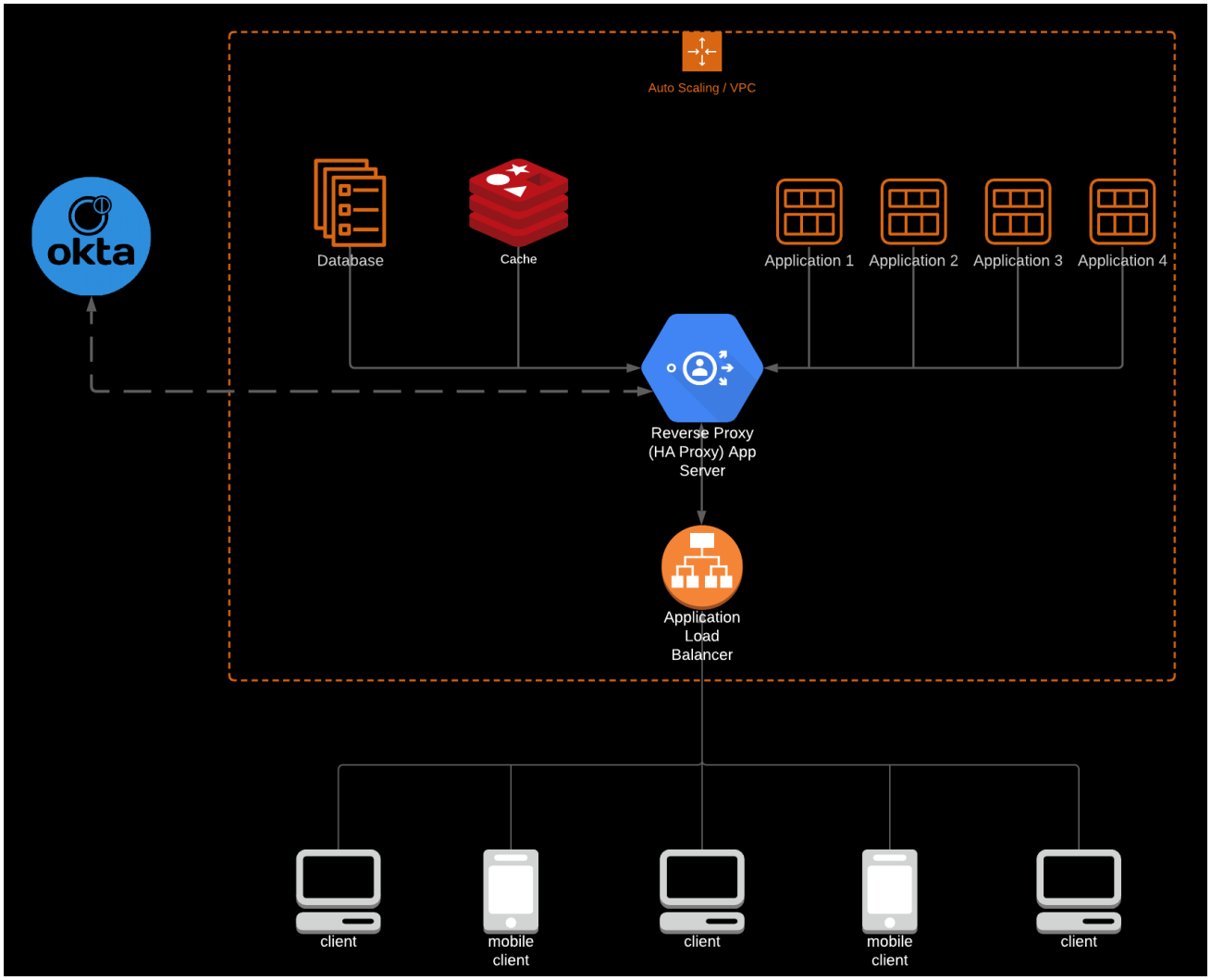
1 Introduction :

Our micro service will act as an intermediate layer between OKTA & other websites of Techier, This will surely increase scalability & efficiency of rolling out updates & changes to the Authentication system.

2 Understanding of Problem Statement :

The problem addressed in the problem statement is the requirement to create an authentication mechanism by developing a microservice which would successfully authenticate the applications deployed in different geographies built using different technologies(multiple web portals of the Techier organization) with the Authentication Service (OKTA).

3 Architecture Diagram :



4 Design / Implementation Details :

Implementing a **central authorization layer** for applications accessed by HTTPS, so you can use an application-level access control model instead of relying on network-level firewalls.

Our **Reverse proxy service** acts as a middleware between the organisation's applications and client, the proxy server checks the authentication (with the desired logic) with each request from client side. If the client is not authenticated it will take the client to the authentication page then to the application, and in case of authenticated client it will serve the application directly

This reverse proxy server will pass the auth info (like Auth token, user information, Unique User identification etc) to the Application services in headers, that can be used by the application for the desired operations.

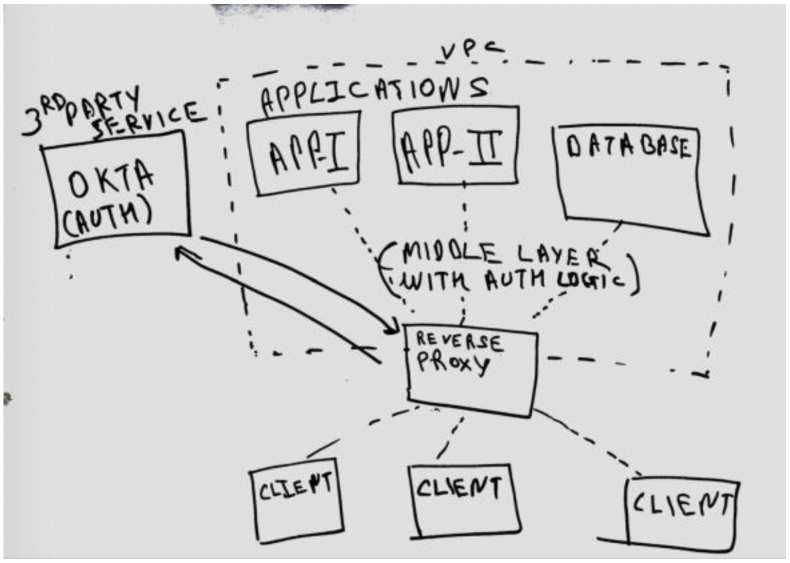
This can serve a common authentication across multiple services/Applications too.

Reverse proxy Works in layer 7 of OSI networking model, which gives a-lot of flexibility and features which can be implemented which can further enhance security also and prevent common attacks like DDOS.

Core Concepts : **reverse-proxy, Micro-services, containers**

configuration

- **domain1, domain2, domain3, domain4 pointing to the reverse proxy micro service**
- **domain 1's request is proxied to application 1**
- **domain 2's request is proxied to application 2**
- **domain 3's request is proxied to application 3**
- **domain 4's request is proxied to application 4**
- **and in case of unauthenticated request the request maybe redirected to authentication page (common for all application) or proxied to the auth page.**



5 Scope / Assumption :

5.1 In-scope

Microservice that provides a central authorization layer for applications accessed by HTTPS which acts as a middleware between organization's applications and clients. Considering 4 different applications (Not multiple instances of these applications) and internal database, with 3rd party api access to OKTA to the proxy server.

5.2 Out-of-scope

Our service does not include a configuration dashboard that can be built to define rules to configure customers with pre-built enterprise (Techier) applications.

Auto-scaling, caching and advanced attack protection. Regional Filters, Metric server and additional load balancers. And a caching mechanism for optimisation. and SSL termination. (Scaling of applications)

6 Tools & Technology Stack

- NodeJs
- ExpressJS / HttpServer
- HA Proxy
- Docker
- cloud
- mongo db
- Redis Cache
- Okta JavaScript SDK

7 Appendix

- **Reverse Proxy** : https://www.researchgate.net/publication/221034753_Reverse_Proxy_Patterns
- **Central authorization layer :** https://www.hindawi.com/journals/scn/2018/4351603/

8 References

Version	Name	File	Date Received
1.0			