ALX - 0x09. Web Infrastructure Design

Task 1: Distributed Web Infrastructure

1. ***Why are we adding elements, specifically a new server*?** The purpose is to incorporate a load balancer that can manage heavy incoming traffic and eliminate the risk of a single point of failure, which could occur if we rely solely on one server.
2. ***Can you explain the load balancer's distribution algorithm and function*?** Our load balancer employs the Round Robin algorithm, which connects to servers sequentially unless a server is unavailable. It serves requests to each server one after the other. Once the last server is reached, it returns to the first server. This algorithm is suitable when servers have equal specifications and few persistent connections.
3. ***Could you clarify if your load balancer facilitates an Active-Active or Active-Passive setup and what sets them apart?*** The load balancer enables an Active-Active setup, where both nodes (servers) provide the same service simultaneously. In contrast, an Active-Passive setup involves some nodes remaining inactive; for instance, in a two-node setup, the second node remains on standby if the first node is active. The key distinction lies in performance, as Active-Active allows access to all server resources during regular operation, while Active-Passive only utilizes the backup server during failover.
4. ***How does a database Primary-Replica (Master-Slave) cluster operate?*** In a master-slave replication, data from one database server (the master) is replicated to another database server (the slaves). The master logs the updates, which are then propagated to the slaves. Depending on the timing of changes, they can be synchronous (made simultaneously on both master and slave) or asynchronous (queued and written later). This replication approach primarily distributes read access across multiple servers for scalability. Still, it can also serve purposes like failover or data analysis on the slave to avoid overloading the master.

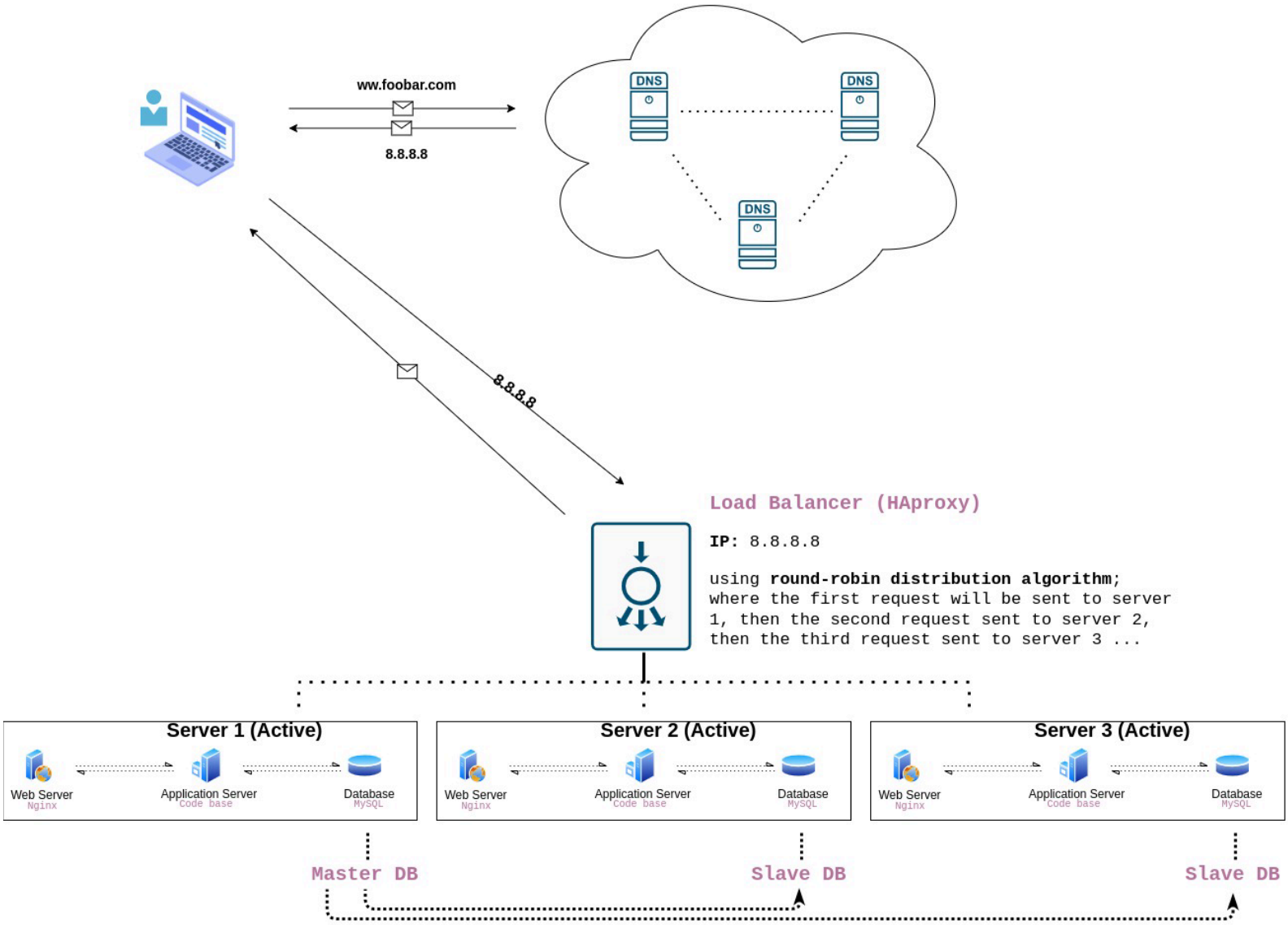
5. ***What distinguishes the Primary node from the Replica node regarding their role within the application?*** The Replica node duplicates the Primary node, providing redundant copies of the application codebase. This redundancy safeguards against hardware failures and enhances the capacity to handle read requests, such as searches or document retrievals.

Issues:

A. Single Point Of Failure (SPOF): The main concern in this infrastructure is having only one load balancer, which represents a single point of failure and may cause disruptions if it malfunctions.

B. Security issues (lack of firewall and HTTPS): Significant security concerns arise from using an insecure HTTP protocol for communication, enabling potential attackers to intercept sensitive information due to plain text transmissions. Additionally, the absence of a firewall leaves the system vulnerable to denial-of-service (DoS or DDoS) attacks and potential breaches through unidentified open ports, leading to data exfiltration.

C. Lack of monitoring: The absence of proper monitoring for servers, websites, or applications can hinder the identification and timely resolution of problems, downtime, or security threats. Implementing effective monitoring would improve productivity and user experience and help prevent potential issues from escalating and incurring additional IT support costs.


ww.foobar.com
8.8.8.8
DNS
DNS
DNS
8.8.8.8
Load Balancer (HAproxy)
IP: 8.8.8.8
using round-robin distribution algorithm;
where the first request will be sent to server
1, then the second request sent to server 2,
then the third request sent to server 3 ...
Server 1 (Active)
Web Server
Nginx
Application Server
Code base
Database
MySQL
Server 2 (Active)
Web Server
Nginx
Application Server
Code base
Database
MySQL
Server 3 (Active)
Web Server
Nginx
Application Server
Code base
Database
MySQL
Master DB
Slave DB
Slave DB