Diagram (ASCII Representation)

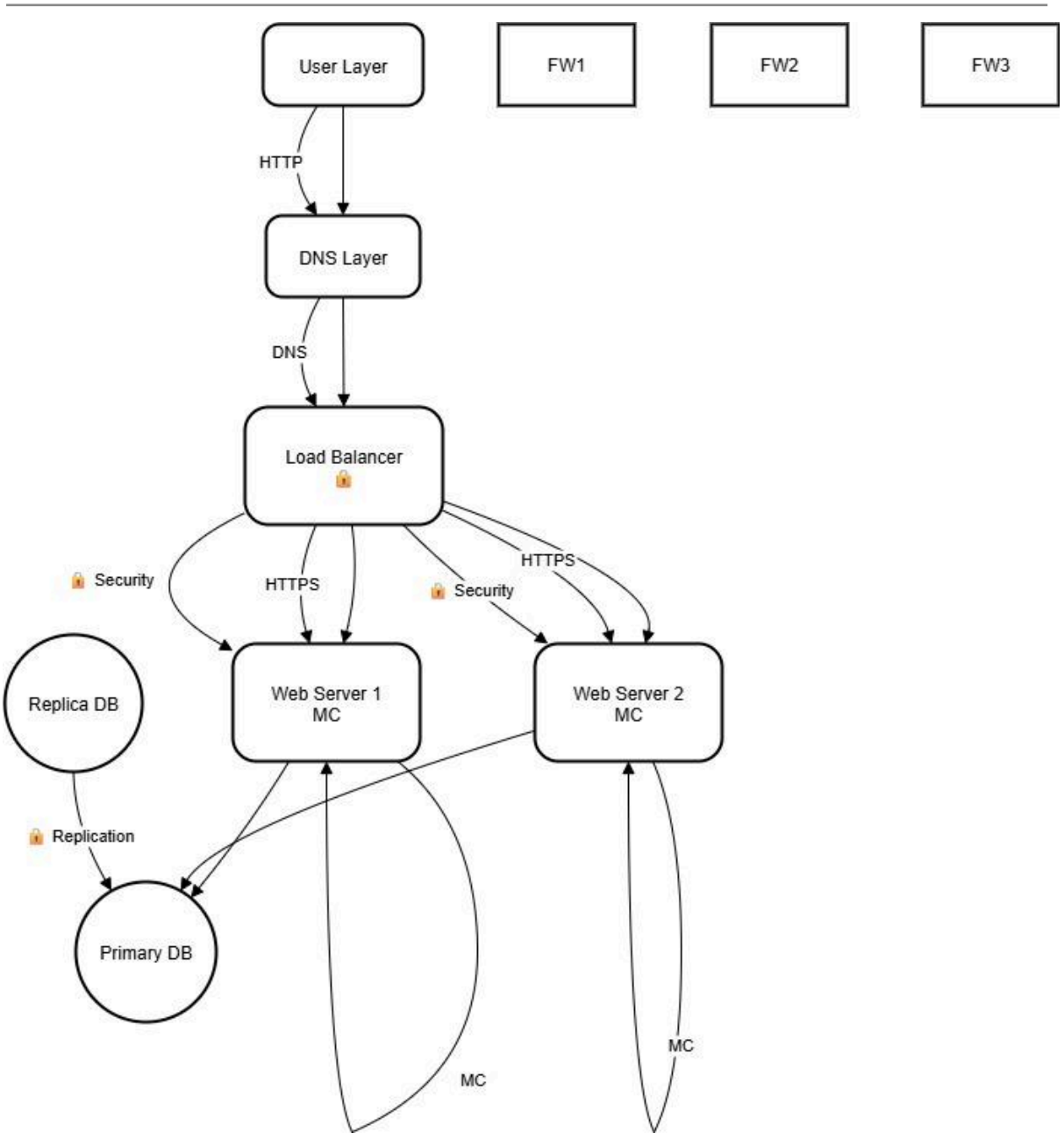
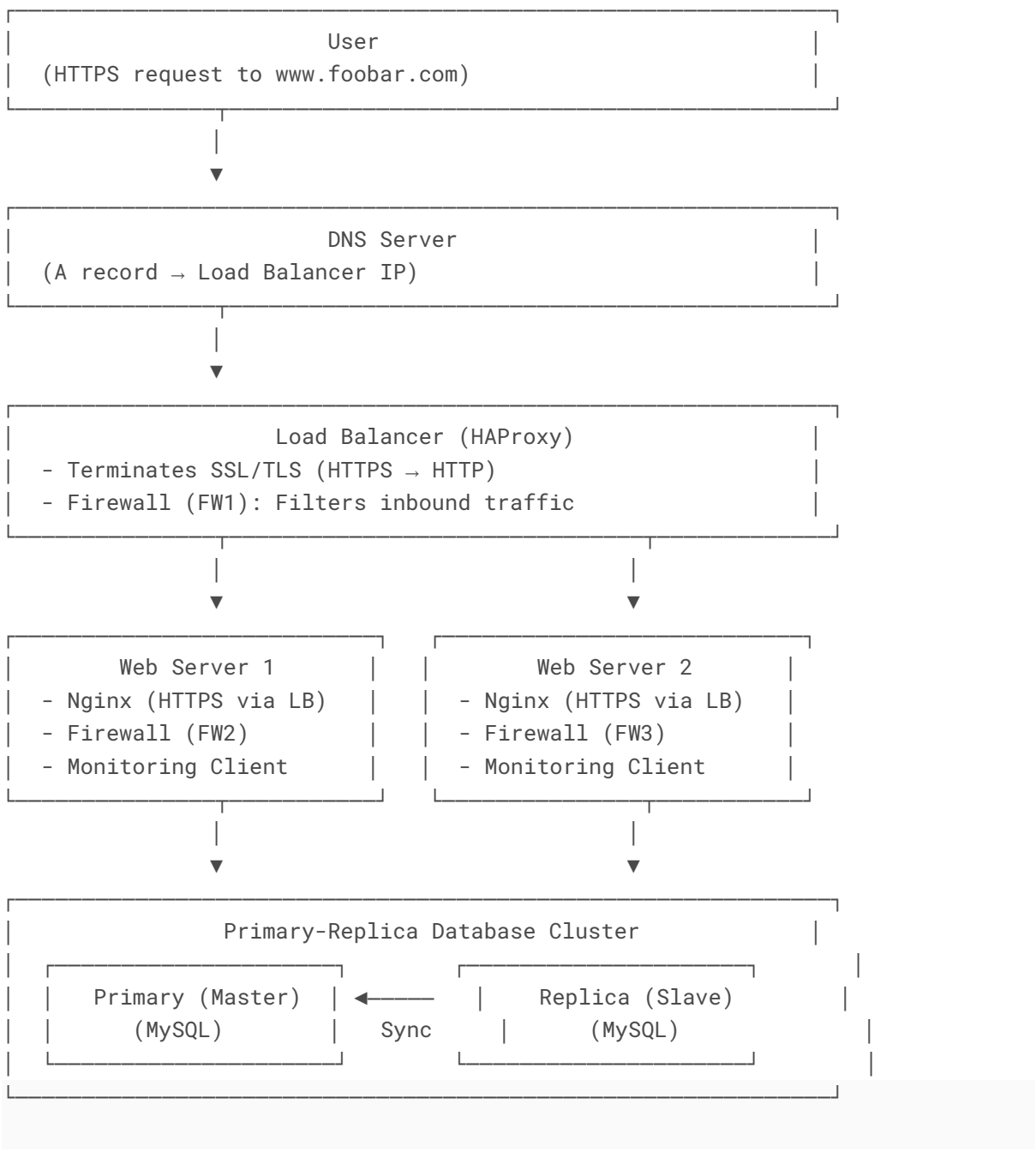


Diagram (Simplified Structure)

text



Added Components & Justifications

1. Firewalls (FW1, FW2, FW3)

- Purpose:

    - FW1 (Load Balancer): Blocks malicious traffic (DDoS, SQLi).
    - FW2/3 (Web Servers): Restrict access to only LB and monitoring tools.
- Why?: Defense against attacks (e.g., unauthorized access, brute force).

2. SSL Certificate (HTTPS)

- Purpose: Encrypts traffic between users and the load balancer.
- Why?:
    - Prevents eavesdropping (e.g., passwords, credit cards).
    - Required for SEO and browser trust (padlock icon).

3. Monitoring Clients (e.g., Sumologic)

- Purpose: Collect metrics (CPU, RAM, QPS, errors).
- How Data is Collected:
    - Agents on servers send logs/metrics to a central dashboard.
    - Alerts trigger if thresholds are breached (e.g., high latency).
- Monitor QPS (Queries Per Second):
    - Track Nginx logs or use tools like Prometheus + Grafana.
    - Set up alerts if QPS exceeds server capacity.

Key Issues in This Infrastructure

1. SSL Termination at Load Balancer

- Problem:
    - Traffic between LB and web servers is HTTP (unencrypted).
    - Risk: Internal network snooping (MITM attacks).
- Fix: Use end-to-end HTTPS (LB → Servers) with mutual TLS.

2. Single Write-Capable MySQL Server

- Problem:
    - Primary DB is a SPOF; crashes = no writes (e.g., user signups fail).
- Fix: Multi-primary replication or distributed SQL (e.g., Galera).

3. Identical Components on All Servers

- Problem:
    - Wasted resources: DB on web servers competes for CPU/RAM.

    - Complexity: Harder to debug (e.g., is the issue in Nginx or MySQL?).
- Fix: Separate servers by role (dedicated DB, app servers).