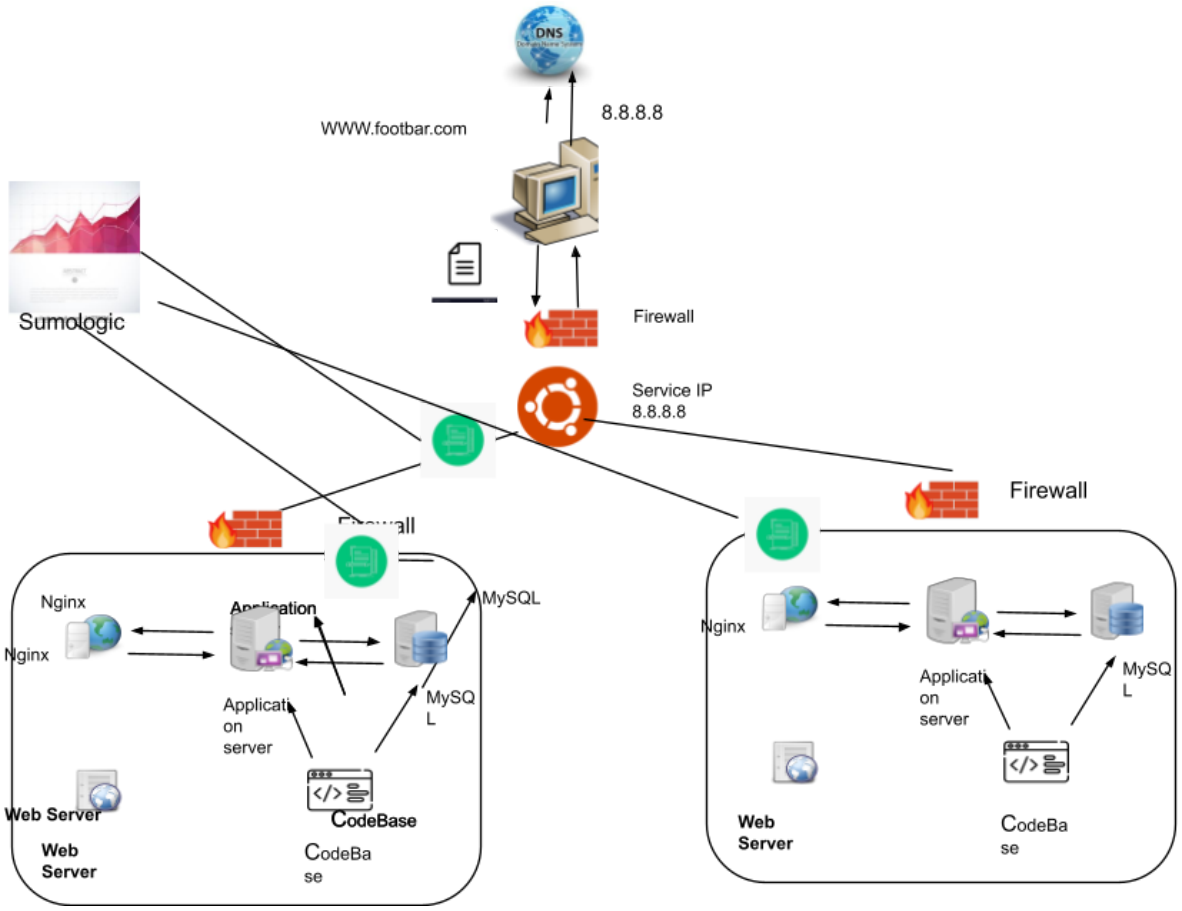
Jks



Three-Server Web Infrastructure for www.foobar.com

This document outlines the design of a three-server web infrastructure to host the website www.foobar.com. The requirements, the purpose of each element, specific configurations, and potential issues with this setup will be covered in details.

Elements

1. **Three Servers:**

High Availability: Having three servers ensures redundancy and fault tolerance. If one server fails, the others can continue serving traffic.
Load Distribution: Distributing the load across multiple servers prevents overload and ensures better performance.
Scalability: Additional servers can be added to handle increased traffic.

2. **Firewalls:**

Security: Firewalls protect the servers from unauthorized access, malicious traffic, and potential attacks.
Access Control: Firewalls allow or deny traffic based on predefined rules.

3. **SSL Certificate (HTTPS):**

Encryption: HTTPS encrypts data transmitted between clients and servers, ensuring confidentiality and integrity.
Authentication: SSL certificates verify the authenticity of the website, preventing man-in-the-middle attacks.

4. **Monitoring Clients (Data Collectors):**

Performance Monitoring: Collecting metrics helps identify bottlenecks, resource utilization, and performance issues.
Alerts: Monitoring tools notify administrators of anomalies or failures.
Sumologic or Other Services: Sumologic collects logs and metrics, providing insights into server health and application performance.

Load Balancer Distribution Algorithm
Use a load balancer to distribute incoming requests across the three servers.
Algorithms like round-robin, least connections, or weighted round-robin can be used.

Active-Active vs. Active-Passive Setup
Active-Active: All servers handle traffic simultaneously.
Active-Passive: One server serves traffic while others remain idle as backups. Failover occurs if the active server fails.

Database Primary-Replica (Master-Slave) Cluster
Set up a MySQL primary-replica cluster:
Primary (Master): Accepts writes and updates.
Replicas (Slaves): Serve read requests and replicate data from the primary.

Difference Between Primary and Replica Nodes
Primary Node:
Accepts writes.
Critical for data consistency.
Requires robust backup and recovery mechanisms.
Replica Nodes:
Serve read requests.
Improve read scalability.
Can be used for reporting or analytics.

Issues with the Infrastructure
Terminating SSL at the Load Balancer Level:
Issue: SSL termination at the load balancer exposes decrypted traffic within the internal network.
Solution: Implement end-to-end encryption by terminating SSL at the application servers.
Single MySQL Server Capable of Accepting Writes:
Issue: Single point of failure. If the MySQL server fails, writes are blocked.
Solution: Use a primary-replica cluster for redundancy and failover.
Servers with Identical Components:
Issue: Uniform vulnerabilities. If one server is compromised, others may be too.
Solution: Diversify components (e.g., different web server technologies) to reduce risk.