Secured and monitored web infrastructure

Added features:

1. 3 firewalls
2. 1 SSL certificate to serve `www.foobar.com` over HTTPS
3. 3 monitoring clients (data collector for Sumologic or other monitoring services)

Why add 3 fierwalls?

- The addition of firewalls over each of our servers strengthens our infrastructure's security posture by enforcing access control policies, filtering incoming and outgoing traffic, and protecting against unauthorized access, malicious attacks, and security breaches, such as hackers, malware, and denial-of-service (DoS) attacks.
- By deploying firewalls strategically, we can control traffic flow, segment network resources, and create secure zones to isolate sensitive assets and minimize the impact of security incidents.

Why have an SSL certificate?

- Implementing an SSL certificate ensures a secure communication between clients and our web servers by encrypting data transmitted over the HTTPS Protocol.
- SSL/TLS encryption protects sensitive information, such as usernames, passwords, and financial transactions, from eavesdropping, interception, and tampering by unauthorized parties.
- SSL certificates establish trust and authenticity, assuring users that they are connecting to a legitimate and secure website, which helps build confidence and credibility in our online presence.

Why have 3 monitoring clients and a data collector such as Sumo Logic?

- Installing monitoring clients, such as data collectors for Sumo Logic or other monitoring services, enables real-time monitoring, analysis, and visualization

of infrastructure metrics, logs, and events. By establishing these monitoring systems we can proactively detect and respond to performance degradation, security incidents, and compliance violations, helping ensure the reliability, scalability, and compliance of your web services.

How is the monitoring tool collecting data?

- Monitoring clients collect and transmit telemetry data to monitoring platforms, allowing us to gain insights into our system performance, availability, and health, identify anomalies, troubleshoot issues, and optimize resource usage.
- Monitoring tools collect data from monitored systems using various methods, including:
    1. **Agent-Based Monitoring:** Lightweight software agents installed on systems gather metrics, logs, and other data locally and transmit it to a central monitoring server or service.
    2. **Remote Monitoring:** Tools query networked devices, servers, and services using protocols like SNMP, SSH, or WMI to collect data remotely.
    3. **API-Based Monitoring:** Data is collected from APIs exposed by applications, services, and cloud platforms, enabling access to performance metrics, logs, and configuration settings.
    4. **Log Parsing and Analysis:** Tools parse log files generated by applications and network devices to extract relevant information, such as timestamps, log levels, and error codes, for analysis.
    5. **Metric Scraping:** Metrics are collected by querying endpoints exposed by monitoring agents or exporters, retrieving data in standardized formats (e.g., Prometheus metrics format).
    6. **Telemetry and Instrumentation:** Modern applications expose metrics, traces, and events for monitoring and observability purposes, allowing monitoring tools to collect data directly from instrumented

applications using standardized protocols and libraries (e.g., OpenTelemetry, StatsD).

What to do if you want to monitor your web server QPS?

- We are using the Log management and Analytics system Sumo Logic to monitor the Query Per Second (QPS) metric for our web servers.

Issues:

While adding firewalls, SSL certificates, and monitoring systems over our web infrastructure significantly increase the security, reliability, and overall infrastructure integrity, we still may face some technical issues. Following are some of these issues:

1. **Terminating SSL at the load balancer level**:

   Terminating SSL at the load balancer level can introduce potential drawbacks, including increased network traffic, loss of end-to-end encryption, limited visibility into encrypted traffic, performance overhead, management complexity, and compliance concerns.

   To mitigate these issues, we can consider the following approaches:

   a. **End-to-End Encryption:** Maintain end-to-end encryption by terminating SSL/TLS connections at the backend servers rather than at the load balancer.
   b. **Dedicated Hardware or SSL Offload Appliances:** Deploy dedicated hardware or specialized SSL offload appliances to alleviate the performance overhead associated with SSL termination at the load balancer.

c. **Careful Evaluation:** Evaluate the specific requirements, security considerations, and performance implications of SSL termination at the load balancer level.

2. **Having only one MySQL server capable of accepting writes**:

   Having only one MySQL server capable of accepting writes introduces a single point of failure, limited scalability, and lack of redundancy. In the event of server failure, there's a risk of downtime and potential data loss.

3. **Having servers with all the same components**:

   Without monitoring, it's challenging to detect and respond to issues promptly, leading to potential service disruptions, performance degradation, or security incidents going unnoticed.

   Monitoring helps track the health, performance, and availability of infrastructure components, identify bottlenecks or failures, and proactively address issues before they impact users.