2. Secured and monitored web infrastructure

Firewalls:

What are firewalls for: Firewalls control the flow of network traffic based on predetermined security rules. They filter traffic, block unauthorized access, and protect against various cyber threats, including unauthorized access attempts, malware, and denial-of-service attacks.

SSL Certificate (HTTPS):

Why traffic is served over HTTPS: HTTPS encrypts data in transit, ensuring that sensitive information is secure during communication. It authenticates the server, prevents tampering, and provides a secure connection, crucial for protecting user privacy and maintaining the integrity of data.

Monitoring:

What monitoring is used for: Monitoring is used to track the performance, availability, and security of the infrastructure. It helps identify and address issues proactively, optimize resource usage, and ensure a reliable and responsive system.

How the monitoring tool collects data: Monitoring tools collect data through agents or APIs deployed on servers. They capture metrics such as CPU usage, memory, disk I/O, network traffic, and application-specific metrics. The collected data is then sent to a central repository for analysis and reporting.

Monitoring web server QPS: To monitor the web server QPS (Queries Per Second), you can set up monitoring agents or use built-in server metrics. Collect data on incoming HTTP requests or transactions over time, analyze trends, and set up alerts for abnormal QPS levels.

Issues with the Infrastructure:

Terminating SSL at the Load Balancer Level:

Why it's an issue: Terminating SSL at the load balancer means that decrypted traffic is forwarded to the backend servers. While this improves server performance, it raises security concerns as the communication between the load balancer and backend servers is not encrypted. This exposes data to potential interception on the internal network.

Having Only One MySQL Server Capable of Accepting Writes:

Why it's an issue: A single MySQL server for write operations is a single point of failure. If this server goes down, it disrupts write operations, leading to potential data loss or unavailability. A high-availability setup with at least one replica capable of handling writes is recommended for redundancy and fault tolerance.

Servers with All the Same Components (Database, Web Server, and Application Server):

Why it might be a problem: Homogeneous server components can be a problem if they all share the same vulnerabilities. A single exploit could impact the entire infrastructure. It's advisable to diversify technologies and versions to minimize the risk of a widespread failure due to a common weakness.