What are firewalls for?

Firewalls act as a barrier between a trusted internal network and untrusted external networks, controlling incoming and outgoing network traffic based on predetermined security rules. In the context of www.foobar.com, a firewall helps protect the server from unauthorized access and malicious attacks by filtering traffic and blocking potentially harmful connections.

Why is the traffic served over HTTPS?

Traffic is served over HTTPS to ensure secure communication between the server and the user's browser. HTTPS encrypts data transmitted between the client and the server, preventing eavesdropping and tampering by malicious third parties. This is crucial for protecting sensitive information such as login credentials, payment details, and personal data on www.foobar.com.

What monitoring is used for?

Monitoring is used to track the performance, availability, and health of the infrastructure and applications running on www.foobar.com. It helps identify and address issues proactively, optimize resource utilization, and ensure smooth operation of the website. Monitoring tools provide insights into system metrics, application performance, and user experience.

How is the monitoring tool collecting data?

The monitoring tool collects data by periodically querying various metrics and logs from the server, application, and network components of www.foobar.com. It uses agents or probes installed on the servers to gather real-time data on CPU usage, memory usage, disk I/O, network traffic, error logs, and other relevant parameters. This data is then analyzed and visualized to provide insights into the system's health and performance.

Explain what to do if you want to monitor your web server QPS.

To monitor the web server QPS, you can configure the monitoring tool to collect metrics related to incoming HTTP requests. This can include tracking the number of requests received by the web server within a specified time frame. By analyzing this data, you can monitor the server's workload, identify traffic patterns, and ensure that it can handle the expected load without becoming overloaded or slowing down.

Issues with this infrastructure:

Why terminating SSL at the load balancer level is an issue?

Terminating SSL at the load balancer level means that HTTPS decryption occurs before requests reach the web servers. While this offloads the decryption workload from the servers, it also introduces a potential security vulnerability. If the SSL termination point at the load balancer is compromised, it exposes decrypted data, including sensitive information, to potential attackers.

Why having only one MySQL server capable of accepting writes is an issue?

Having only one MySQL server capable of accepting writes creates a single point of failure for the database. If this server fails, write operations are disrupted, leading to downtime and potential data loss. Additionally, it limits scalability and performance, as a single server may struggle to handle high write loads or spikes in traffic.

Why having servers with all the same components (database, web server, and application server) might be a problem?

Having identical servers with all the same components increases the risk of widespread failure. If a critical vulnerability or software bug affects one component, it can impact all servers simultaneously, leading to widespread downtime or compromised security. Introducing diversity in server configurations and software versions can mitigate this risk by reducing the likelihood of uniform vulnerabilities across the entire infrastructure.