Xgrid

Aug 11, 2022

The Journey From Crashing on Black Fridays to 90% Revenue Spike In 21 Days

Industry: **Ecommerce**

Project: **Ecommerce app | DevOps**

An e-commerce store with crashes, zero infrastructure control, and lax security protections

The application frequently crashed during peak Black Friday traffic hours. In addition, the website architecture had multiple single points of failure.

Xgrid helped the client to make its digital store highly available to avoid critical peak hour crashes. Also, we developed a cost-effective solution involving multiple AWS-managed services to enable a robust e-commerce platform that is flexible and scalable.

After the solution was implemented and production traffic routed to the new infrastructure, the application could seamlessly handle high traffic load during peak hours with considerable cost reduction, thereby tapping the previously missed revenue potential.

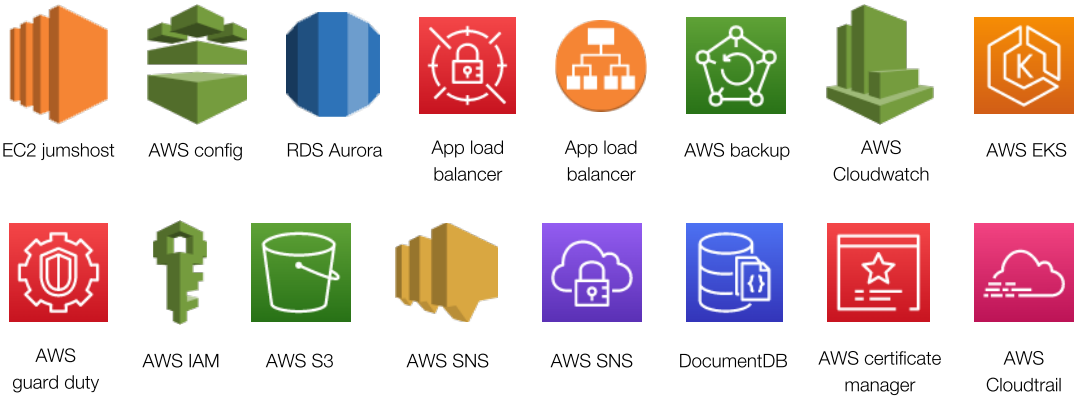
Outcomes achieved by embracing an elastic distributed architecture

- Achieved high traffic load on peak hours, around **10k+ Requests per minute.**
- Successfully achieved near **0% net downtime.**
- Achieved high availability status of the application using auto-scaling. **At least 2+ healthy instances running at a time.**
- Successfully avoided single-point of failure by **decoupling infrastructure** to multiple managed services.

Let's learn about the blueprints of building such a network for yourself.

Designing and implementing a scalable architecture with zero downtime

To implement a scalable and future-looking technical architecture, Xgrid used the following AWS Services:



- We migrated the application from a single instance to **auto-scaling with load balancing. EFS** was used to enable application state sharing between computes.
- We used **Route 53 to send automated requests** over the internet for application discovery. **S3 was used with CloudFront** for static content delivery.
- We added **multi-regional architecture** to provide disaster recovery. The website architecture had multiple single points of failure. We used a multi-region active/passive strategy to implement a **hot standby strategy for Disaster Recovery (DR).**
- To further bulletproof the application against downtime, we used Redis for in-memory caching to enhance across-region application performance-enabled **CloudWatch metrics** and **AWS's predictive scaling. This helped in providing** insights on traffic flow to identify and fix critical performance bottlenecks. **AWS Patch Manager** was used to auto-detect and fix critical security issues on the compute.

This project was successfully delivered in 3 weeks.

Approach and the best practices to follow

- Customers don't really know this, but for a **highly available architecture**, the answer is usually a **distributed architecture**. It decreases the **load per instance** and removes **single points of failure**.
- Always apply the **80:20 rule.** 20% of the calls are the ones that are always hot. 80% are usually not. Identify that 20% of the calls and then build a cache around it.
- The **introduction of a multi-cache layer** can elevate a website's performance and user experience significantly.

Did You Know?

This future-proof cloud infrastructure empowered our client to rapidly deploy newer products, campaigns, and services. It enabled them to manage peak traffic hours with the confidence of ~zero e-commerce outages while obviating dependency on external IT vendors for consistent scalability support.

About Us

Established in 2012, Xgrid has a history of delivering a wide range of intelligent and secure cloud infrastructure, user interface and user experience solutions. Our strength lies in our team and its ability to deliver end-to-end solutions using cutting edge technologies.

Schedule an Assessment

Reach out to us at: **letstalk@xgrid.co**