Checklist

Auto-Scaling Checklist for Traffic Spikes

1. Configure Auto-Scaling for Your Cloud Provider

Amazon Web Services (AWS)

□ **Create Launch Configuration/Template:**
Define the instance type, AMI (Amazon Machine Image), and configuration details.

□ **Set Up Auto Scaling Group (ASG):**
Define the minimum, maximum, and desired number of instances.

□ **Define Scaling Policies:**
Set up scaling triggers based on CloudWatch metrics such as CPU utilization or request count.

□ **Scheduled Scaling:**
Set scaling policies based on predictable schedules for known traffic surges.

Microsoft Azure

□ **Create Autoscale Setting:**
Configure autoscaling for virtual machines, app services, or other resources in the Azure portal.

□ **Set Thresholds:**
Define thresholds for scaling up or down based on CPU usage, memory, or request counts.

□ **Define Scaling Rules and Schedules:**
Create scaling rules for performance metrics and scheduled scaling for regular traffic peaks.

□ **Set Instance Limits:**
Set minimum and maximum instance limits to avoid over-commitment.

Google Cloud Platform (GCP)

□ **Enable Autoscaling:**
Set up autoscaling for your instance group (Compute Engine instances or Kubernetes clusters).

□ **Define Autoscaling Policies:**
Set target CPU utilization, request rates, or custom metrics to trigger autoscaling actions.

□ **Set Resource Limits:**
Define minimum and maximum instances the autoscaler can use to manage costs effectively..

2. Best Practices for Effective Auto-Scaling

□ **Monitor and Adjust Metrics Regularly:**
Continuously monitor key metrics (CPU usage, memory, request count) to adjust thresholds as needed.

□ **Set Limits on Scaling:**
Define sensible minimum and maximum resource limits to avoid overprovisioning.

□ **Test Auto-Scaling in Advance:**
Simulate high-traffic scenarios to ensure your setup responds correctly.

□ **Combine with Load Balancing:**
Use load balancing to evenly distribute traffic across instances and regions for optimal performance..

□ **Optimize for Cost and Efficiency:**
Review your auto-scaling configuration regularly to avoid over-scaling and reduce unnecessary costs.

□ **Test, Test, Test:**
Ensure your auto-scaling setup is aligned with real-time demand forecasts to optimize resource usage.

3. Plan for Steep Traffic Spikes (e.g., Product Releases, Black Friday)

□ **Pre-Allocate Resources for Initial Surge:**
Allocate additional servers or instances in advance to absorb the initial traffic surge, ensuring the auto-scaling mechanism isn't delayed.

□ **Analyze Historical Traffic Data:**
Review past traffic patterns (e.g., previous Black Friday, product launches) to forecast the expected surge and prepare your auto-scaling thresholds accordingly.

□ **Test Different Spike Scenarios:**
Simulate various spike scenarios (e.g., 600% increase in traffic over an hour or 200% in a minute) to evaluate how your system handles sudden, sharp traffic increases...

Have questions or need help?
Find us at Aknostic.com