Sukhumi load shifting

Sukhumi load shifting

To start using zonal shift for ALB or NLB, you must set the load balancer attribute zonal_shift nfig.enabled to true. For NLBs using cross-zone load balancing, you must also ensure that target_health_state.unhealthy nnection_termination.enabled is set to false. With the feature enabled, you can start a zonal shift to mitigate impact when you identify an impairment in a single Availability Zone (AZ).

You may choose to also perform a zonal shift on your ASG behind the load balancer during an AZ impairment. If you"ve configured the ASG to replace unhealthy instances during a zonal shift, this may result in instances being terminated in the impaired AZ and new instances being launched in the other AZs. It"s also possible EC2 Auto Scaling scales out your application during the zonal shift and launches those new instances in the unaffected AZs. This can create a capacity imbalance among your AZs.

When you determine the AZ impairment has ended, you can cancel the shift and rebalance traffic into the AZ. Cross-zone load balancing helps make rebalancing safer when you have a capacity imbalance because the overall traffic percentage received per target will decrease when you end the load balancer zonal shift. This happens because the load balancer distributes traffic evenly across each target in your target group, as shown in the Figure 2.

In contrast, cross-zone disabled load balancing distributes traffic evenly to each AZ. The load balancer then distributes requests across available targets in that zone. A capacity imbalance among AZs can cause certain instances to receive more load than others after you end the load balancer zonal shift. This could lead to overload and impact to your application. For example, Figure 3 shows how the instance in AZ 2 is receiving approximately twice as much traffic as the targets in AZ 1 and AZ 3. In this configuration, it"s important to use target_group_health.dns_failover.minimum_healthy_targets unt to prevent the AZ from accepting traffic until enough healthy hosts are available.

Cross-zone enabled load balancing is the default for ALBs and can optionally be enabled for NLBs. This allows you to take advantage of zonal shift without having to make large-scale changes to the configuration of your ALB target groups. You can also opt-in to zonal autoshift for your ALBs in their default configuration. AWS starts an autoshift when internal telemetry indicates that there is an AZ impairment that could potentially impact customers. You can use zonal autoshift in conjunction with the weighted random routing algorithm. This helps you minimize recovery time during an event, and reduces the additional observability you need to take advantage of zonal shift.

While zonal autoshift and Automatic Target Weights (ATW) anomaly mitigation are the preferred ways to react to single-AZ impacts, these tools may not detect certain infrastructure gray failures or single-AZ application impairments. For example, an application deployment containing a bug that was deployed to a single AZ, or a small amount of packet loss impacting a handful of instances that starts causing application

errors. You may need to develop additional observability to detect these situations. In the next section, I examine how to detect single-AZ impairments with cross-zone load balancing enabled.

Monitoring metrics such as request count, fault rate, and latency per AZ are a prerequisite to determining when an AZ may be experiencing an impairment, and allow you to safely mitigate potential impact. The following three signals can help you know when to use zonal shift.

Let"s review how you can start collecting metrics about the health of your application in each AZ.

One of the observability best practices for resilience is to monitor your customer experience with synthetic canaries. These act as an early-warning indicator so you can notify yourself of a problem before your customers do. In the post Rapidly recover from application failures in a single AZ, we used Amazon CloudWatch synthetics to monitor the zonal endpoints of your ALBs and NLBs to produce per-AZ metrics, as shown in Figure 4.

Synthetics are still a best practice with cross-zone load balancing enabled. However, it"s not as useful to test each zonal endpoint for an ALB or NLB because the response could come from a target in any AZ. Instead, for ALBs, you can use the ALB load balancer Amazon CloudWatch metrics to identify when targets in a specific AZ show elevated fault rates or latency. ALB target metrics provide 2XX, 3XX, 4XX, and 5XX counts as well as a metric for TargetResponseTime. All of these metrics have AvailabilityZone as a metric dimension, which represents the AZ of the target that produced the response.

For NLBs it can be more difficult to determine changes in application health because its target metrics are mostly layer 4 information. You could monitor the TCP_Target_Reset_Count metric as a possible proxy to application health, but this may still be insufficient. When cross-zone load balancing is enabled on your NLB or its target groups, you should utilize custom server-side metrics that provide the target"s AZ as a metric dimension. Refer to Publishing custom metrics and the CloudWatch embedded metric format for more details on how to achieve this.

You can also monitor the UnHealthyHostCount target metric for your load balancers. If the AZ impairment is causing targets to fail their health checks, this is a direct signal of that impact. To automatically respond to this metric, you can use the target_group_health.dns_failover.minimum_healthy_targets unt attribute for your NLB or ALB target groups. This ensure the load balancer automatically shifts away from an AZ when there are too few healthy hosts.

Sukhumi load shifting