VMware Cloud on AWS Web Application Load Balancing with Amazon Route53 and Amazon Application Load Balancer

VMware Integrations

Table of contents

VMware Cloud on AWS Web Application Load Balancing with Amazon Route53 and Amazon Application Load Balancer

Introduction

As a highly reliable platform, VMware Cloud on AWS offers a 99.9% SLA for single-AZ SDDCs and an enhanced 99.99% SLA for Multi-AZ SDDCs. Leveraging Amazon Route 53 and Amazon Elastic Load Balancer can further enhance resilience and minimize service interruptions for business-critical applications, safeguarding against the rare instances of Availability Zone (AZ) or regional failures. This document offers considerations and recommendations designed to assist you in architecting and constructing robust, fault-tolerant web applications utilizing Amazon Route 53 and Amazon Application Load Balancer (ALB).

Summary and Considerations

	VMware Cloud on AWS: SDDC Network Architecture Designlet: VMware Transit Connect for VMware Cloud on AWS User Guide for Application Load Balancers Amazon Route 53

Planning and Implementation

The following sections provide the guideline for planning and implementing a highly resilient web application using Amazon Application Load Balancer and Amazon Route 53.

ALB Scheme

The ALB scheme can be set to either Internet-facing or Internal. An Internet-facing ALB is designed to serve clients from the Internet. When the ALB is configured with an Internal scheme, its clients can be located within VMware Cloud on AWS SDDCs, on-premises networks, or Amazon VPCs.

ALB Deployment and Network Connectivity

As an AWS regional service, a single ALB can provide service across multiple single-AZ VMware Cloud on AWS SDDCs within the same region, as well as a single multi-AZ VMware Cloud on AWS SDDC.

To protect against regional failures, one ALB is required in each region where a VMware Cloud AWS SDDC resides.

An ALB can be deployed either in a VMware Cloud on AWS connected VPC or in an alternative customer owned VPC.

Each VMware Cloud on AWS SDDC is linked to a connected VPC. This VPC is connected to the NSX edge appliance of VMware Cloud on AWS SDDC using cross-account Elastic Network Interfaces (X-ENI). Deploying an ALB in a connected VPC is a common deployment pattern when the ALB is used to provide load balancing service to the workload within the linked VMware Cloud on AWS SDDC. Also, customers may choose this design as data transfer from the ALB in the connected VPC to the SDDC backend servers within the same AZ is free of charge. However, this deployment may not be suitable for scenarios requiring a centralized ingress VPC to terminate incoming service requests using the ALB, and deliver application flows to multiple SDDCs, with each SDDC linked to its own VPC.

When an ALB is deployed in a customer-owned VPC, connectivity from the ALB to workloads in VMware Cloud on AWS can be established through AWS Transit Gateway (TGW) or VMware Transit Connect. This method is often seen in a multi-SDDC environment, where AWS TGW or VMware Transit Connect has been used to provide high-bandwidth, low-latency, and resilient connectivity among different VPCs and VMware Cloud on AWS SDDCs. Figures 1 and 2 illustrate the network topologies for two scenarios using this option:

- two SDDCs located in different AZs but within the same AWS region.
- two SDDCs located in two different AWS regions.

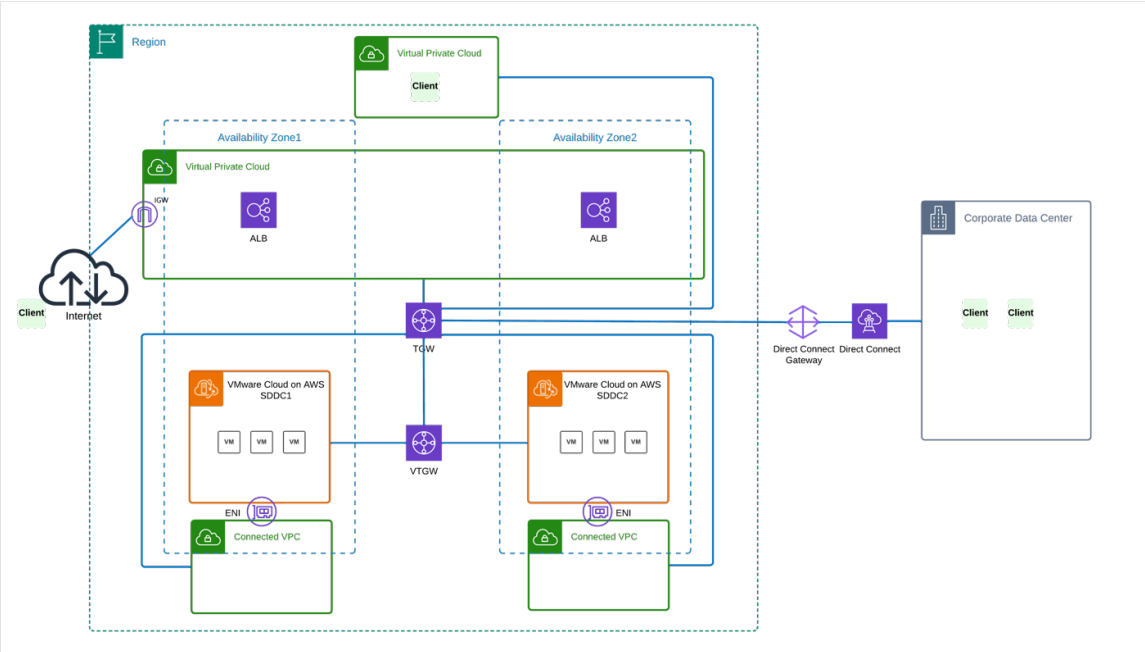

Figure 1

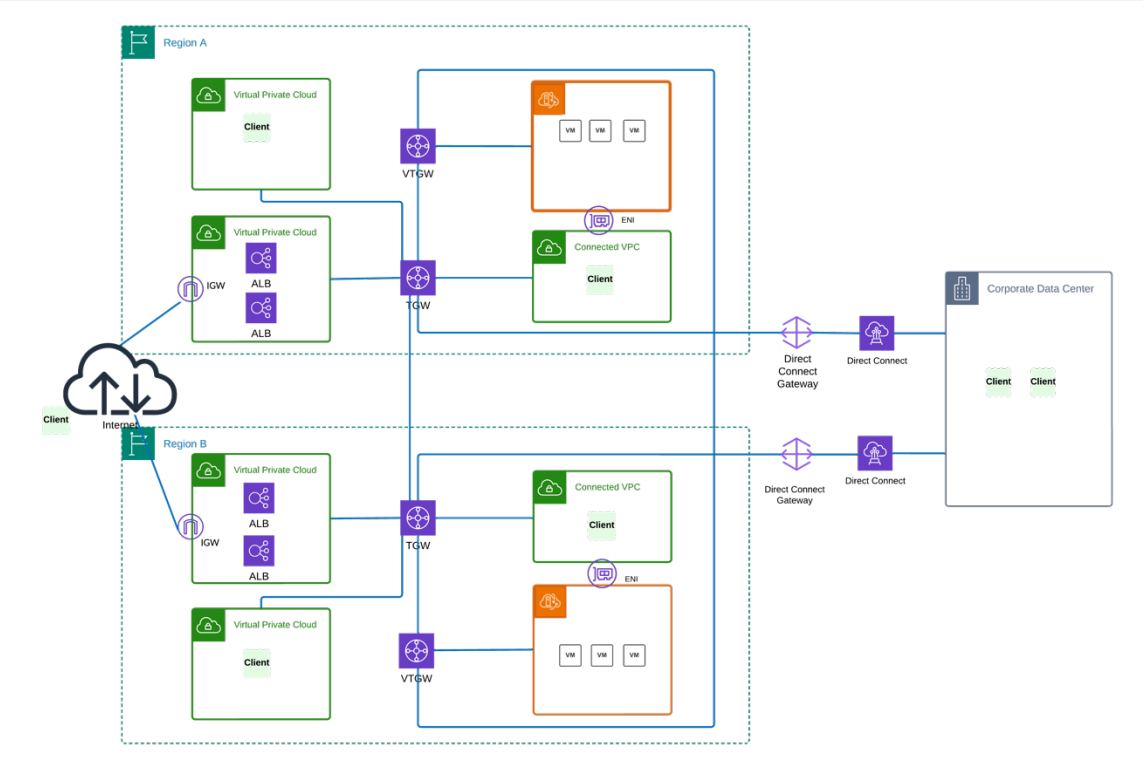

Figure 2

As shown in Figure 1 and 2, the design supports Internet-facing and internal ALB deployment. Clients can access your web application from the Internet, on-prem networks, Amazon VPCs or VMware Cloud on AWS SDDCs.

Through this design, the ALB can offer load-balancing services to any SDDCs within the same region, and workloads in native Amazon VPCs. While this architecture enhances flexibility, it also introduces additional costs associated with AWS Transit Gateway and VMware Transit Connect. In the first case, where two SDDCs are located in different AZs but within the same AWS region, it is noteworthy that all inter-AZ data transfer charges from ALB to backend servers via Transit Gateway are now waived.

As the ALB is deployed outside of VMware Cloud on AWS, it may not be suitable for East-West web load balancing within VMware Cloud on AWS. This is because the same traffic must exit the SDDCs and then re-enter the same SDDC again before reaching the target servers. Such a routing pattern increases latency and places additional workload on the NSX. In such cases, the VMware NSX Advanced load balancer is recommended.

ALB HTTPs Listener

When configuring an HTTPS listener for your ALB, it is imperative to utilize a valid SSL/TLS certificate issued by a trusted Certificate Authority (CA). This ensures the authenticity of your website and the security of the data in transit. For scenarios involving multiple domain names, consider using a wildcard certificate, which covers all subdomains of a single domain, or a certificate with Subject Alternative Names (SANs) that lists each additional domain.

Please note that ALB does not support custom security policies. It is highly recommended to apply a robust security policy from AWS's predefined security policies to your HTTPS listener. The applied policy should exclude insecure protocols, such as TLS 1.0, and weak cipher suites. You can find the AWS recommended security policies here.

In a highly secure environment, you may choose to enable TLS mutual authentication on the ALB to ensure that a client has a valid certificate.

ALB Target Group

An ALB target group is a collection of service targets. Only IP Address type of target is supported when using ALB for VMware Cloud on AWS workloads.

When there are two or more SDDCs in the same region, you can create a single target group that includes all target servers across

the different SDDCs. This approach simplifies the ALB and Route 53 configuration but also reduces service interruption in the event of an AZ failure.

If session persistence is required, a single target group should be used for each ALB as the session persistence is configured at target group level.

ALB Health Check

Health checks are critical when utilizing a load-balancing service like AWS ALB. However, crafting effective health checks for an enterprise application on any load balancer, including an ALB, can be very challenging. Simply verifying that the ALB receives an HTTP 2xx response is often insufficient, especially for multi-tier applications requiring all services across all tiers to be healthy. A noteworthy practice we have observed is the inclusion of built-in, regular, and thorough health checks that cover all service tiers in well-designed enterprise applications. A health status page is then updated to mirror the results of these checks. We consider this approach as a recommended best practice, as it allows for the easy verification of an application's health by searching for a specific string on the health status page.

The default ALB health check interval, timeout, healthy and unhealthy threshold values may not suit your application. Adjust them as necessary to better meet your specific requirements.

Please note that AWS ALB uses the private primary IPs of ALB ENI interfaces as source IP when performing health checks.

ALB Session Stickiness

The ALB supports cookie-based session stickiness. Both cookies generated by applications and cookies generated by the ALB are encrypted by ALB.

Please note that session persistence may prevent efficient load sharing between application servers.

ALB Reserve Client IP

ALB uses its own ENI interfaces' IP to send the traffic to the target servers. To forward the client IP information to target servers, ALB utilizes HTTP X-Forwarded-For request header. If the original request doesn't include X-Forwarded-For request header, the ALB generates one, assigning the client's IP address as the header value. If the header already exists, the ALB appends the client's IP address to it before passing the header to your server. The X-Forwarded-For request header may contain multiple comma-separated IP addresses. The left-most address represents the client's IP address where the request originated.

Security Consideration for ALB

At the network layer, security controls can be implemented from multiple points:

- AWS security groups and/or network access control lists can be used to manage connections to/from ALBs.
- Within VMware Cloud on AWS SDDC, NSX firewalls (either compute gateway or distributed firewall) can be set to allow only application traffic and health checks initiated from the primary private IP of ALB ENIs.

At the application layer, the NSX distributed IDS/IPS can be configured to detect or prevent intrusions and suspicious activities. Additionally, AWS Web Application Firewall (WAF) and AWS Shield Advanced can be enabled alongside your ALB to protect your web applications against common exploits and DDoS attacks.

ALB Access and Connection Logs

ALB access logs capture detailed information about requests sent to your load balancer. Each log contains information such as the time the request was received, the client's IP address, latencies, request paths, and server responses.

ALB connection logs capture detailed information about requests sent to your load balancer. Each log contains information such as the client's IP address and port, listener port, the TLS cipher and protocol used, TLS handshake latency, connection status, and client certificate details.

ALB access logs and connection logs are optional features. These log events are extremely helpful for troubleshooting issues with the load balancing service.

Private Hosted Zone

A private hosted zone is likely required to host your internal domain. In such a scenario, a Route 53 DNS resolver inbound endpoint is needed in a VPC that is accessible by on-prem networks and VMware Cloud on AWS. This setup enables both on-prem networks and VMware Cloud on AWS SDDC to use the Route 53 DNS resolver for DNS resolution.

Amazon Route 53 Alias Record

DNS CNAME records are a way to point one domain name to another. This type of DNS record is widely used in global load balancing solution/service. However, DNS resolvers must make more queries to answer CNAMEs, which increases latency and costs.

Amazon Route 53 alias records provide a Route 53–specific extension to DNS functionality. These records allow for the direct routing of traffic to selected AWS resources, such as ALB and Amazon S3 buckets, enabling Route 53 to provide a direct answer for AWS resources (e.g., an ALB).

Also, unlike a CNAME record, alias records can be created at the top node of a DNS namespace, also known as the zone apex. For example, if you register the DNS name example.com, the zone apex is example.com. You can't create a CNAME record for example.com, but you can create an alias record for example.com that routes traffic to www.example.com (as long as the record type for www.example.com is not of type CNAME).

Given these considerations, alias records are generally the preferred choice in most scenarios.

Amazon Route 53 Routing Policy

When you create a Route 53 record, you select a routing policy that determines how Amazon Route 53 responds to DNS queries, tailored to your application's needs. For instance, a failover routing policy would be utilized to configure DNS records for applications operating in active-passive mode, whether between two (one primary SDDC and one secondary) or among multi SDDCs (several primary and secondary). With this policy, Amazon Route 53 will answer DNS queries with the IP address of primary SDDCs as long as the applications hosted there are healthy. In the event that all primary SDDCs fail, AWS Route 53 will redirect clients to the secondary SDDC.

Similarly, if you plan to provide your application service from multiple SDDCs in different regions concurrently, you can choose a routing policy from the follow options: geolocation, geoproximity, latency-based or weighted routing policy. These policies allow all healthy SDDCs to provide service. Additionally, it's crucial to configure a default route when using geolocation, geoproximity, or latency-based routing policies to prevent a 'no answer' response.

Amazon Route 53 Health Check

When creating an alias record with Amazon Route 53, the endpoint is point to an ALB. Evaluate target health should be used to check the health of the resource specified by ALB Endpoint.

For an ALB to be considered healthy, a target group that contains targets must contain at least one healthy target. If any target group contains only unhealthy targets, the ALB is considered unhealthy, and Route 53 routes queries to other resources.

When alias records cannot be used, a Route 53 health check becomes essential. To perform a health check for an internal ALB, consider utilizing an Amazon CloudWatch-based health check. For more information, please refer to the AWS blog post: Performing Route 53 health checks on private resources in a VPC with AWS Lambda and Amazon CloudWatch. In the case of the Route 53 health check for an Internet-facing ALB, it is imperative to configure the NSX firewall to allow Route 53 to conduct health checks. The source IPs utilized by Route 53 can be found here.

Amazon Route 53 DNS Failover

Amazon Route 53 is a DNS-based global load-balancing solution that is bound by the operational mechanisms of DNS. Therefore, even after a DNS record has been updated in Amazon Route 53 to redirect to healthy endpoints in another SDDC during a failure, clients may still attempt to connect to the application in the failed SDDC if the DNS cache has not yet expired on the client devices or the DNS resolver is still caching the old information. To expedite the failover process, it's advisable to set a lower DNS record Time to Live (TTL) on your local DNS resolvers, with 60 or 120 seconds are commonly chosen TTL values for these situations. Additionally, it's important to note that when using Alias records in the Route 53, the TTL is not configurable, and Route 53 will apply the default 60 seconds TTL for these records.

Amazon Route 53 and Session Stickiness

Some applications require session stickiness between a client and a server. This means all requests from a client involved in a long-lived transaction must be directed to the same server. Failing to do so may disrupt the application session, negatively impacting the client experience. The requirement persists in an active-active application deployment scenario. When using latency-based, weighted, or failover with multiple primary SDDCs routing policy, AWS Route 53 may resolve the domain name of an application to IP Addresses of another ALB for a client during a prolonged session. In such scenarios, session stickiness can’t be maintained because the new ALB will not recognize the session stickiness cookies assigned by the original ALB. As a result, it will treat the request as a new session and direct the traffic to its backend servers.

Author and Contributors

David Zhang, Staff Technical Product Manager - VMware Cloud on AWS, Broadcom

Sheng Chen, Senior Migration Solutions Architect - VMware Cloud on AWS, Amazon Web Services

Osama Masfary, Staff Technical Marketing Cloud Solutions Architect - VMware Cloud on AWS, Broadcom