AWS PrivateLink concepts

PDF (aws-privatelink.pdf#concepts) | RSS (privatelink-updates.rss)

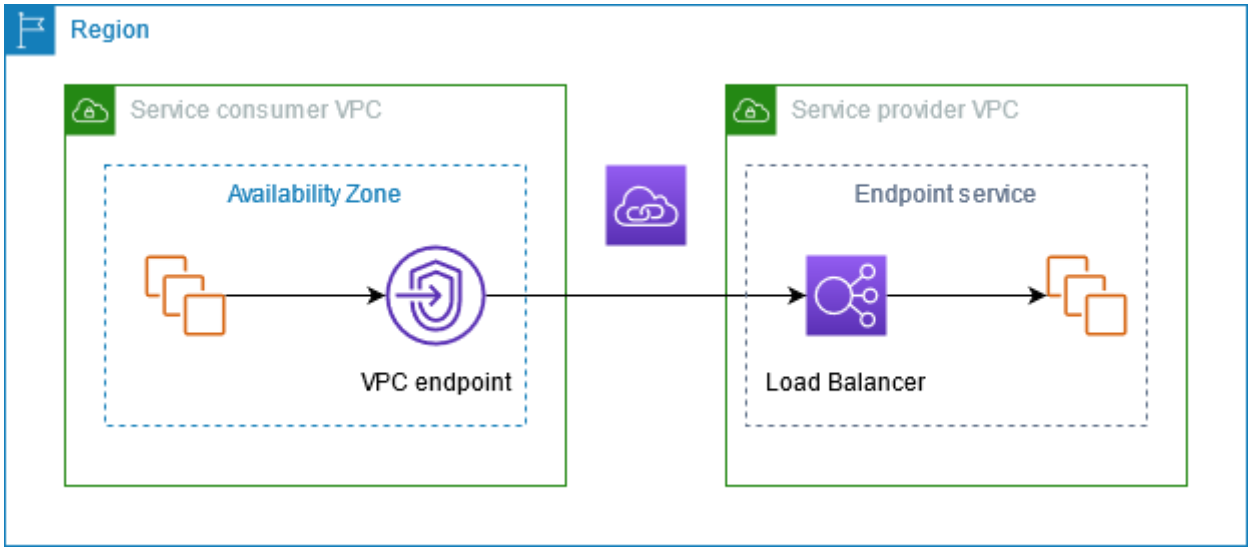
You can use Amazon VPC to define a virtual private cloud (VPC), which is a logically isolated virtual network. You can launch AWS resources in your VPC. You can provide connectivity to resources outside your VPC to the resources in your VPC using features such as internet gateways, NAT devices, and VPN connections. Alternatively, you can use AWS PrivateLink to connect the resources in your VPC to services using private IP addresses, as if those services were hosted directly in your VPC.

The following are important concepts to understand as you get started using AWS PrivateLink.

Concepts

Architecture diagram

The following diagram provides a high-level overview of how AWS PrivateLink works. Service consumers create interface VPC endpoints to connect to endpoint services that are hosted by service providers.



Service providers

The owner of a service is the *service provider*. Service providers include AWS, AWS Partners, and other AWS accounts. Service providers can host their services using AWS resources, such as EC2 instances, or using on-premises servers.

Endpoint services

A service provider creates an *endpoint service* to make their service available in a Region. A service provider must specify a load balancer when creating an endpoint service. The load balancer receives requests from service consumers and routes them to your service.

By default, your endpoint service is not available to service consumers. You must add permissions that allow specific AWS principals (AWS accounts, IAM users, and IAM roles) to connect to your endpoint service.

Service names

Each endpoint service is identified by a service name. A service consumer must specify the name of the service when creating a VPC endpoint. Service consumers can query the service names for AWS services. Service providers must share the names of their services with service consumers.

Service states

The following are the possible states for an endpoint service:

- `Pending` - The endpoint service is being created.

- `Available` - The endpoint service is available.
- `Failed` - The endpoint service could not be created.
- `Deleting` - The service provider deleted the endpoint service and deletion is in progress.
- `Deleted` - The endpoint service is deleted.

Service consumers

The user of a service is a *service consumer*. Service consumers can access endpoint services from AWS resources, such as EC2 instances, or from on-premises servers.

VPC endpoints

A service consumer creates a *VPC endpoint* to connect their VPC to an endpoint service. A service consumer must specify the service name of the endpoint service when creating a VPC endpoint. There are multiple types of VPC endpoints. You must create the type of VPC endpoint that's required by the endpoint service.

- `Interface` - Create an *interface endpoint* to send traffic to endpoint services that use a Network Load Balancer to distribute traffic. Traffic destined for the endpoint service is resolved using DNS.
- `GatewayLoadBalancer` - Create a *Gateway Load Balancer endpoint* to send traffic to a fleet of virtual appliances using private IP addresses. You route traffic from your VPC to the Gateway Load Balancer endpoint using route tables. The Gateway Load Balancer distributes traffic to the virtual appliances and can scale with demand.
- `Gateway` - Create a *gateway endpoint* to send traffic to Amazon S3 or DynamoDB using private IP addresses. You route traffic from your VPC to the gateway endpoint using route tables. Gateway endpoints do not enable AWS PrivateLink.

Endpoint network interfaces

An *endpoint network interface* is a requester-managed network interface that serves as an entry point for traffic destined to an endpoint service. For each subnet that you specify when you create a VPC endpoint, we create an endpoint network interface in the subnet.

If a VPC endpoint supports IPv4, the endpoint network interfaces have IPv4 addresses. If a VPC endpoint supports IPv6, the endpoint network interfaces have IPv6 addresses. The IPv6 address for an endpoint network interface is unreachable from the internet. If you describe an network interface with an IPv6 address, notice that `denyAllIgwTraffic` is enabled.

Endpoint policies

A *VPC endpoint policy* is an IAM resource policy that you attach to a VPC endpoint. It determines which principals can use the VPC endpoint to access the endpoint service. The default VPC endpoint policy allows all actions by all principals on all resources over the VPC endpoint.

Endpoint states

When you create a VPC endpoint, the endpoint service receives a connection request. The service provider can accept or reject the request. If the service provider accepts the request, the service consumer can use the VPC endpoint after it enters the `Available` state.

The following are the possible states for a VPC endpoint:

- `PendingAcceptance` - The connection request is pending. This is the initial state if requests are manually accepted.
- `Pending` - The service provider accepted the connection request. This is the initial state if requests are automatically accepted. The VPC endpoint returns to this state if the service consumer modifies the VPC endpoint.
- `Available` - The VPC endpoint is available for use.
- `Rejected` - The service provider rejected the connection request. The service provider can also reject a connection after it is available for use.
- `Expired` - The connection request expired.
- `Failed` - The VPC endpoint could not be made available.
- `Deleting` - The service consumer deleted the VPC endpoint and deletion is in progress.
- `Deleted` - The VPC endpoint is deleted.

Private hosted zones

A *hosted zone* is a container for DNS records that define how to route traffic for a domain or subdomain. With a *public hosted zone*, the records specify how to route traffic on the internet. With a *private hosted zone*, the records specify how to route traffic in your VPCs.

You can configure Amazon Route 53 to route domain traffic to a VPC endpoint. For more information, see Routing traffic to a VPC endpoint using your domain name (https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/routing-to-vpc-interface-endpoint.html) .

You can use Route 53 to configure split-horizon DNS, where you use the same domain name for both a public website and an endpoint service powered by AWS PrivateLink. DNS requests for the public hostname from the consumer VPC resolve to the private IP addresses of the endpoint network interfaces, but requests from outside the VPC continue to resolve to the public endpoints. For more information, see DNS Mechanisms for Routing Traffic and Enabling

Failover for AWS PrivateLink Deployments (http://aws.amazon.com/blogs/apn/reviewing-dns-mechanisms-for-routing-traffic-and-enabling-failover-for-aws-privatelink-deployments/) .
