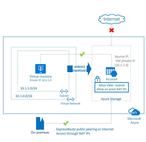
Cloud Network Security 101: Azure Service Endpoints vs. Private Endpoints

Becki Lee | October 8, 2020



Azure offers two similar but distinct services to allow virtual network (VNet) resources to privately connect to other Azure services. Azure VNet Service Endpoints and Azure Private Endpoints (powered by Azure Private Link) both promote network security by allowing VNet traffic to communicate with service resources without going over the internet, but there are some differences. This three-part blog series goes into detail about both services.

- In part 1 of this series, we looked at service endpoints.
- In part 2, we went over Private Link and private endpoints.
- In part 3 (**this part!**), we'll compare and contrast the two and explain when to use which.

Ready to learn more about service endpoints and private endpoints? Let's jump in!

What are service endpoints and private endpoints?

First, a quick recap if you haven't read the previous two blog posts:

A service endpoint allows virtual network resources to use private IP addresses to connect to an Azure service's public endpoint, extending the identity of the virtual network to the target resource. This means traffic flows to the service resource over the Azure backbone network instead of over the internet.

Below, the virtual machine has the private IP address 10.1.1.4 but can access the storage account over a service endpoint. Note that on-premises traffic cannot use service endpoints, and must go over the internet to access the storage account.

Image from https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-service-endpoints-overview#secure-azure-services-to-virtual-networks

A Private Link private endpoint allows virtual network resources to privately connect to other resources as if they were part of the same network, effectively bringing the target resources into the VNet and carrying traffic across the Microsoft Azure backbone instead of the internet. Below, virtual machines in a VNet can use an Azure Private Link private endpoint to connect to a specific SQL database as if it were part of the VNet, even with an NSG denying outbound traffic. The private endpoint makes it possible for traffic to flow from a private IP address to another private IP in the same VNet -- no internet traversal necessary.

Image from https://azure.microsoft.com/en-us/services/private-link/#how-it-works

How are they similar?

As we mentioned, both types of endpoint enable VNet resources to reach other Azure resources without going over the internet. This brings a number of benefits, including improved security. In both cases, you can avoid some of the risks involved in exposing your VNet resources to the internet, such as data exfiltration and denial of service attacks. And you can have a network security group (NSG) lock down the VNet so outbound traffic is denied except to the target resource.

You also benefit from optimized routing. Since service traffic isn't competing with internet traffic, your service traffic can go through fewer hops and potentially reach its destination faster.

And with both service endpoints and private endpoints, you can simplify network architecture and maintenance -- no need to configure NAT or gateway devices, because source traffic never traverses the internet.

How are they different?

Service endpoints and private endpoints have a lot in common, but there are some important differences.

- **Public vs. private destination IP address:** With service endpoints, you're still connecting to the target resource's public endpoint. This effectively extends the identity of the VNet to the target resource. With private endpoints, you're assigning the target resource a private IP address from the VNet, essentially bringing it into the network. The target resource's public IP address doesn't go away, but you can lock it down so all traffic over the internet is denied.
- **Cost:** Service endpoints don't cost extra. Private endpoints are charged according to resource hours used and inbound/outbound data processed. (If you're using a private link service, where you use a private endpoint to connect consumer resources to your own service running behind a standard load balancer, there's no charge for the private link service -- just the private endpoint itself.)
- **Maintenance:** Having a private endpoint at the destination comes with the added requirement of configuring DNS. Since Azure services have fully qualified domain names (FQDN) that automatically resolve to public IP addresses, DNS must be configured such that the public address resolves to the private IP of its private endpoint. If you have a custom DNS setup, you'll need to do some manual work to make everything work together.
- **On-premises support:** Service endpoints do not support on-premises traffic because they can only be secured to virtual networks. However, private endpoints support traffic from on-premises via ExpressRoute, private peering, and VPN tunnels.
- **Granularity:** Service resources are scoped to an entire service. For example, if you have a subnet with a service endpoint enabling access to a storage account, all resources in that subnet can access other storage accounts, too. To get around this, you can use a service endpoint policy to scope access to a single storage account (or all accounts in a region or subscription), but this is only supported for the Azure Storage service. In contrast, private endpoints can be scoped to a specific resource or sub-resource, and it's not limited to Azure Storage -- you'll find a full list of supported services here.
- **Availability:** Private endpoints support more Azure services than service endpoints do, so the decision may come down to the service you're trying to secure. While Azure services such as Cosmos DB, Key Vault, and others work with both types of endpoints, some services can only implement private endpoints. So if you're using Azure Kubernetes Service or Azure Monitor, for example, the only available option is private endpoints. For a full list of what's supported, see the service endpoint and private endpoint documentation.

When should you use which?

Whether you use a service endpoint or private endpoint depends largely on the particulars of your use case.

- If you want to be able to block all internet traffic to a target resource, use a private endpoint.
- If you're dealing with traffic from on-premises, use a private endpoint.
- If you want to secure a specific sub-resource to your VNet resources, use a private endpoint.
- If you want to secure a specific storage account to your VNet resources, you can use a private endpoint, or a service endpoint with a service endpoint policy.
- If you don't need a private IP address at the destination, service endpoints are considerably easier to create and maintain, and they don't require special DNS configuration.
- And if cost is a concern, note that service endpoints are free.

Ultimately, as with all things network security, the endpoint type you choose is up to you and your specific use case.

Conclusion

Service endpoints and private endpoints can both reduce the risks associated with exposing virtual network resources and target resources to the internet, such as malicious actors bringing down your service or stealing sensitive data. The two types of endpoints have differences, but each should be considered an important part of a cloud security engineer's toolkit and an effective way of implementing defense in depth.

CATEGORIZED UNDER

Cloud | cloud security | Azure | network security

SEARCH

Search

RELATED POSTS

Something as a Service
July 29, 2014

Cloud Network Security 101: AWS VPC Endpoints
September 12, 2019

Securing Microsoft Azure Virtual Networks and Network Security Groups
November 7, 2019

Cloud Network Security 101: Azure Virtual Network Service Endpoints
September 17, 2020

Cloud Network Security 101: Azure Private Link & Private Endpoints
September 25, 2020

Securing Infrastructure as Code and Cloud Environments with the Same Policies
September 29, 2021

POPULAR CATEGORIES

Security & Compliance | Cloud | AWS | Fugue | cloud security

Featured Posts

3 Big Amazon S3 Vulnerabilities You May Be Missing
Drew Wright
May 21, 2020

Cloud Security for Newly Distributed Engineering Teams
Drew Wright
March 19, 2020

Cloud Infrastructure Drift: The Good, the Bad, and The Ugly
Drew Wright
February 6, 2019

Trending Topics | Popular Blogs | Events | News | Resources

- Cloud Security Posture Management
- Infrastructure as Code and Security
- AWS Cloud Security
- Azure Cloud Security
- Cloud Security for Google Cloud Platform
- DevSecOps for Cloud Infrastructure Security
- CIS AWS Foundations Benchmark
- CIS Azure Foundations Benchmark
- Fugue Best Practices Policy Framework
- GDPR
- HIPAA
- ISO 27001
- NIST 800-53
- PCI
- SOC 2 Cloud Compliance