Microsoft

Azure SQL Network Security

A primer on basic Azure network security constructs for your Modern Data Estate…

Marek Chmel

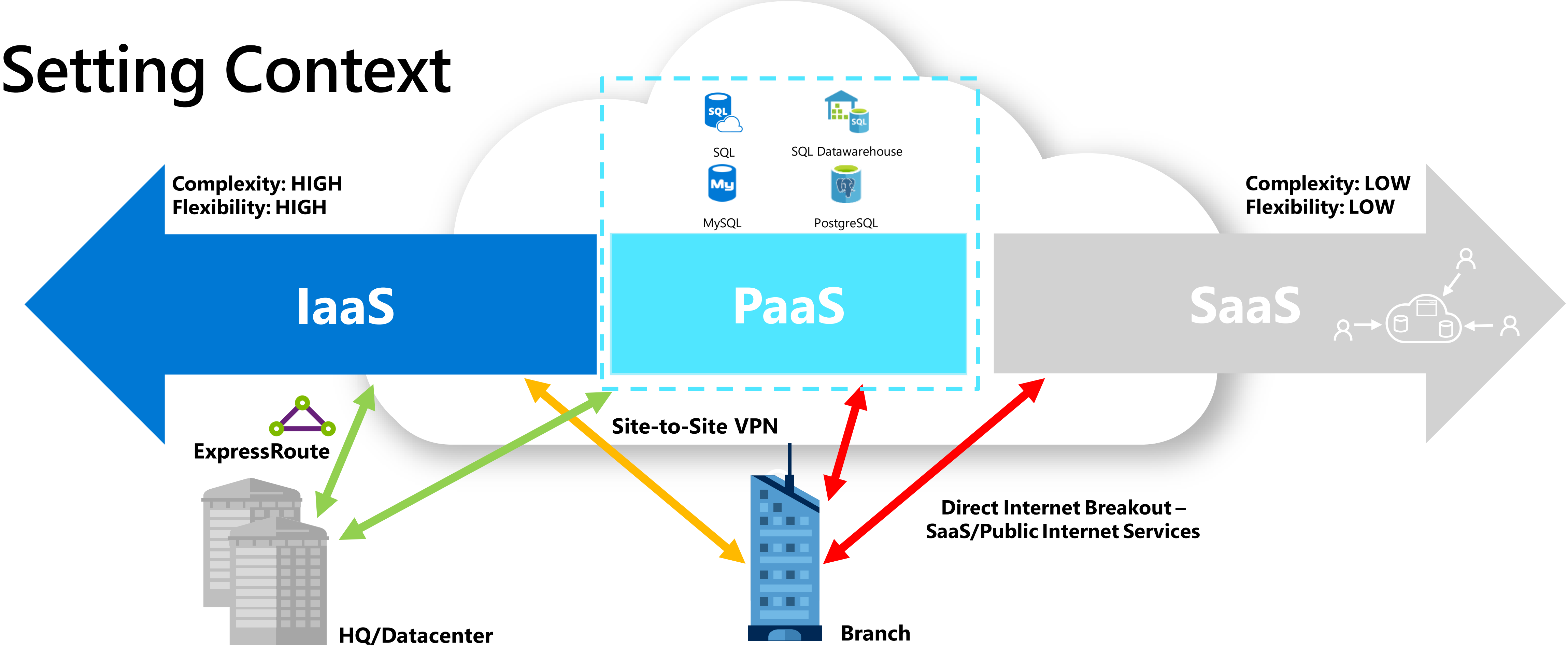
Microsoft

Concepts

Securing public endpoint access with ***PaaS Firewall****…*
Exposing private access through ***Service Endpoints****…*
Building "private PaaS" via ***VNet Injection****…*
Connecting PaaS services using ***Private Link****…*

DEMONSTRATION

Setting Context



- Optimized for private connectivity
- Extends on-premises networking into the cloud
- Extends your existing security boundary

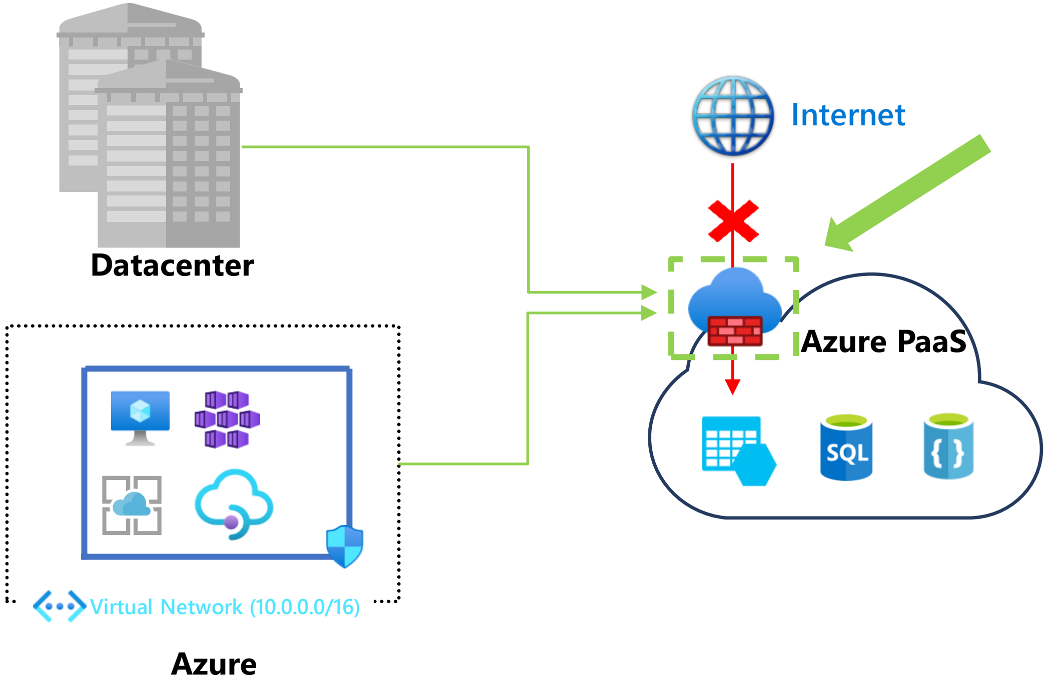
- Optimized public (Internet) connectivity
- Highly distributed, global application experience.
- Focus on scale out across the planet
- Considered external to your on-premises network

Azure PaaS Firewall

Protects public endpoint multi-tenant service

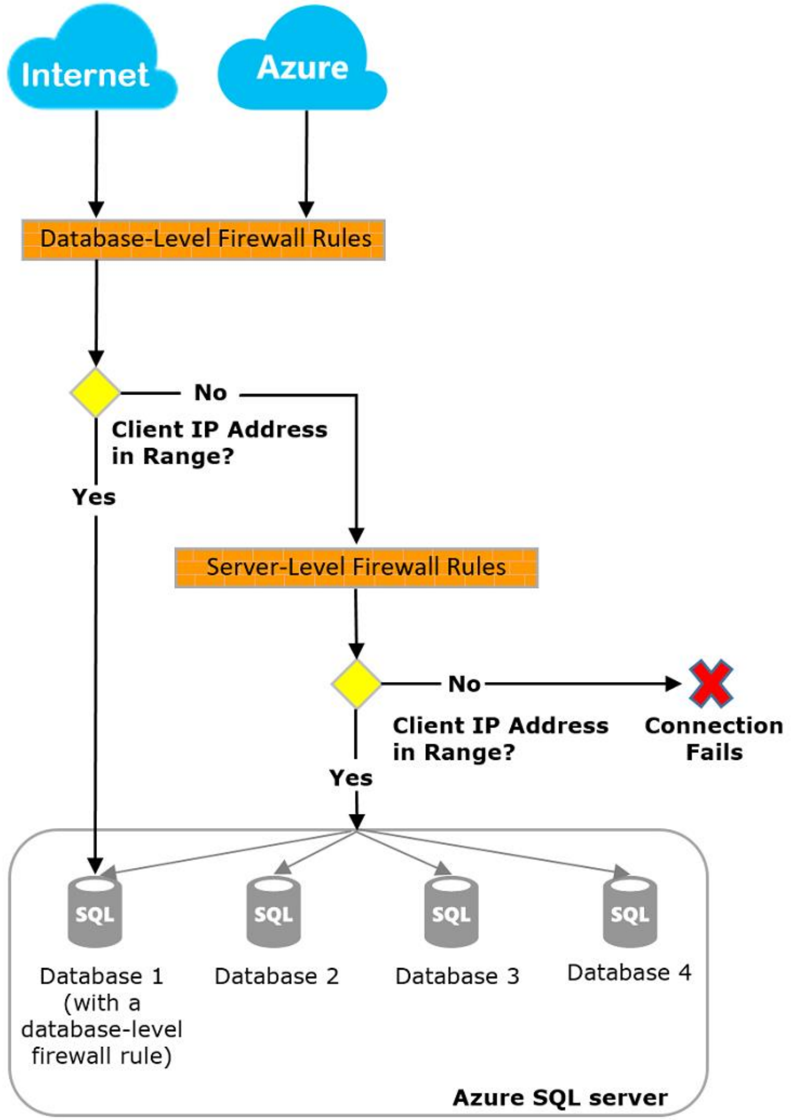
sqlctxpsrv01.database.windows.net

20.45.121.1

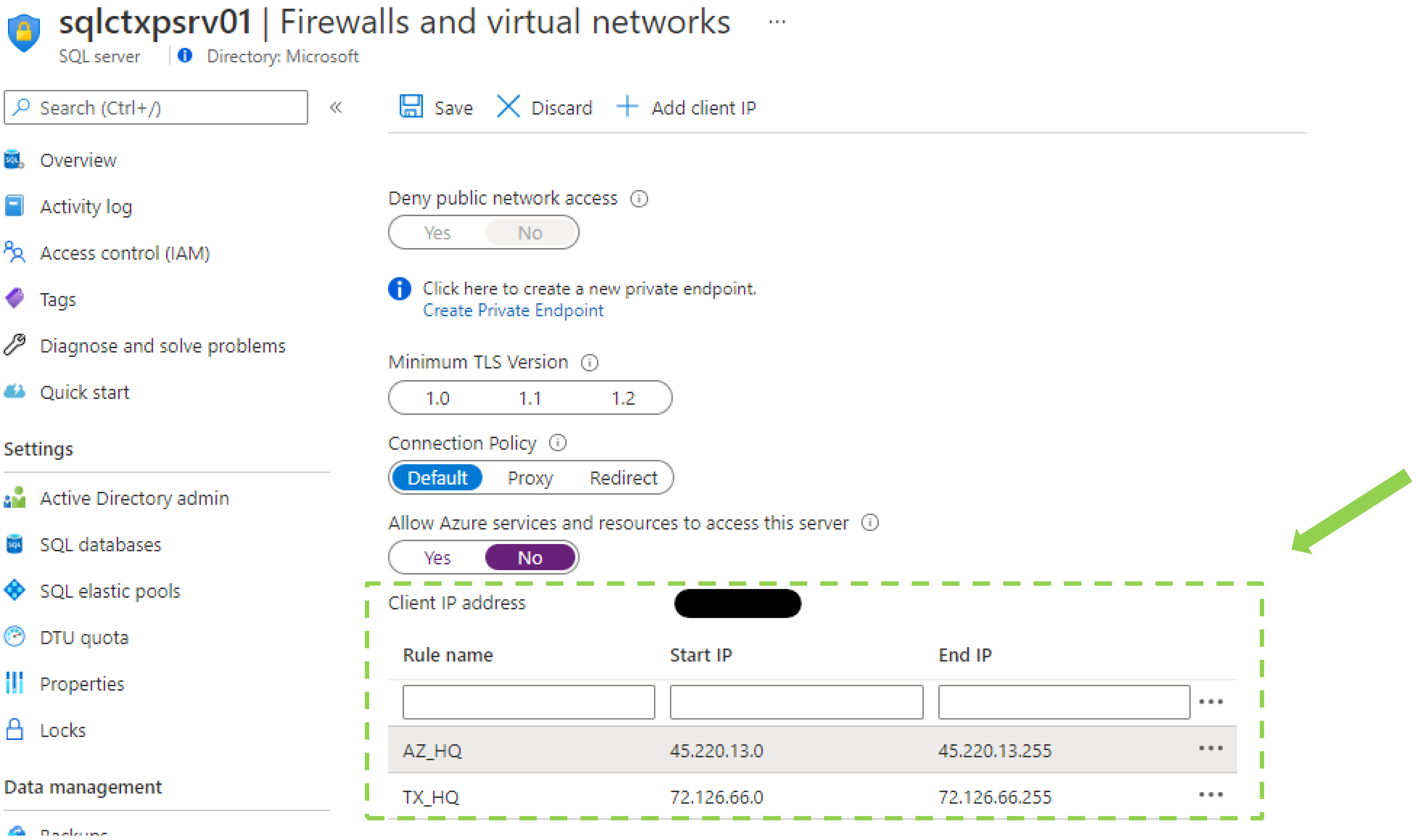


Inbound-only access to Azure Service via Public Endpoint

Multi Tenancy


Internet
Azure
Database-Level Firewall Rules
No
Client IP Address in Range?
Yes
Server-Level Firewall Rules
No
Client IP Address in Range?
Connection Fails
Yes
SQL
Database 1 (with a database-level firewall rule)
Database 2
Database 3
Database 4
Azure SQL server

Azure PaaS Firewall



sqlctxpsrv01 | Firewalls and virtual networks ...

SQL server | Directory: Microsoft

Deny public network access ⓘ
Yes [✓ No]

Click here to create a new private endpoint.
Create Private Endpoint

Minimum TLS Version ⓘ
1.0 1.1 1.2

Connection Policy ⓘ
[✓ Default] Proxy Redirect

Allow Azure services and resources to access this server ⓘ
Yes [✓ No]

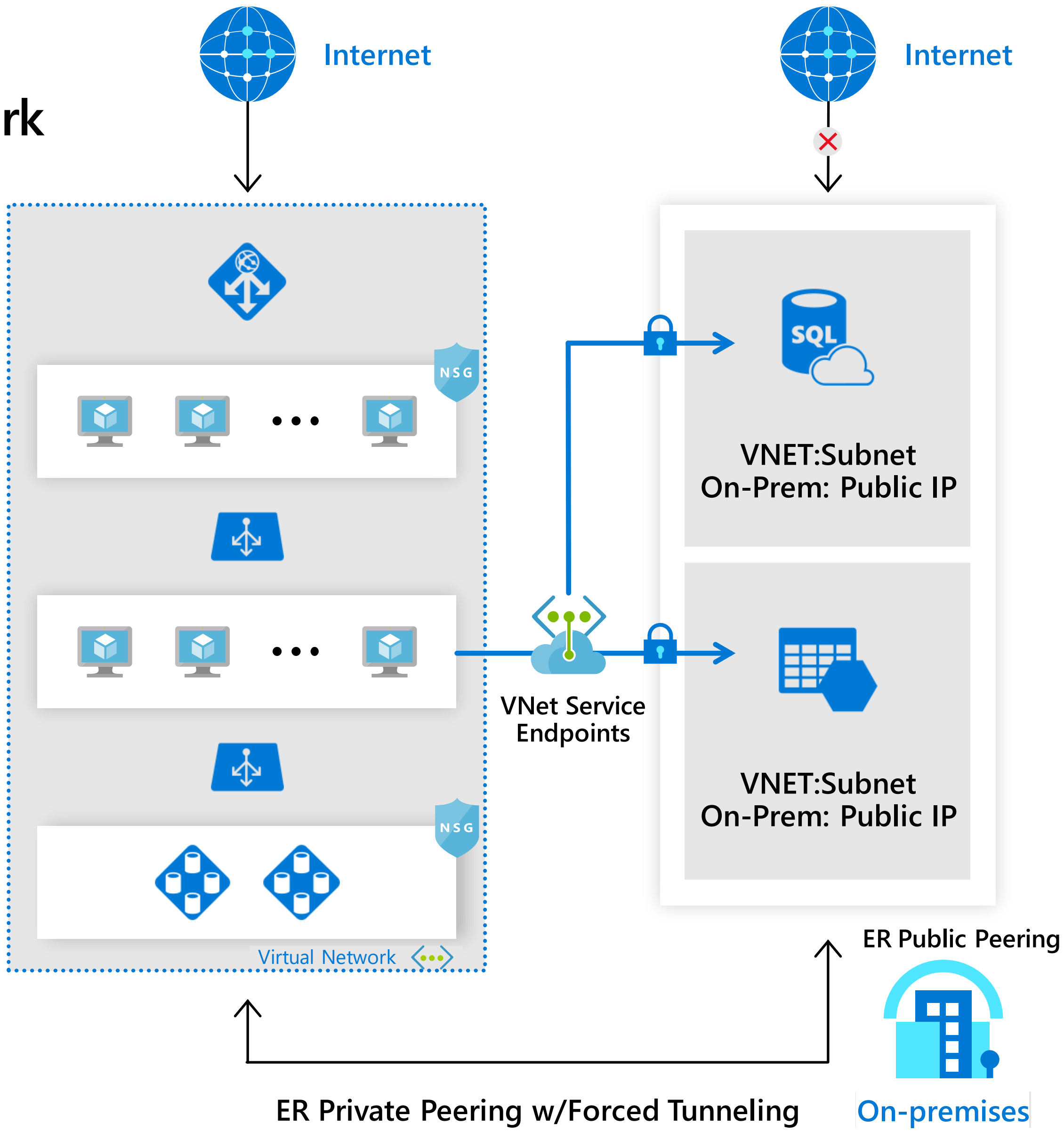
Client IP address [redacted]

Rule name	Start IP	End IP	
			...
AZ_HQ	45.220.13.0	45.220.13.255	...
TX_HQ	72.126.66.0	72.126.66.255	...

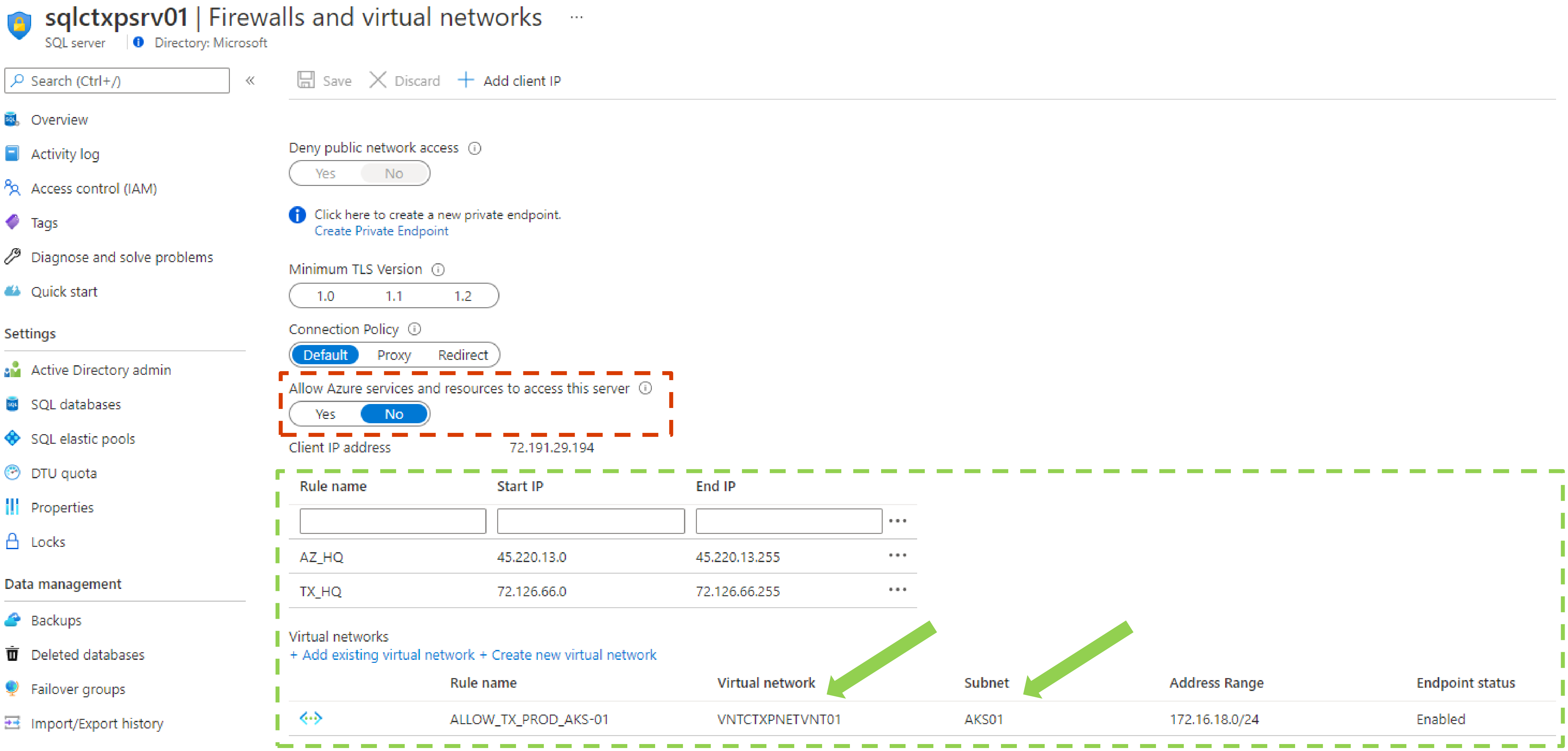
VNet Service Endpoints

Shared resources secured to customer's virtual network

- ✓ Directly extends VNet to the service
- ✓ Secure critical Azure resources to only your VNet
- ✓ Traffic remains on the Microsoft backbone
- ✓ On-premises access through ER public peering
- ✓ Forced Tunneling overridden



VNet Service Endpoints



sqlctxpsrv01 | Firewalls and virtual networks ...

SQL server | Directory: Microsoft

Deny public network access
Yes No

Click here to create a new private endpoint.
Create Private Endpoint

Minimum TLS Version
1.0 1.1 1.2

Connection Policy
[✓ Default] Proxy Redirect

Allow Azure services and resources to access this server
Yes [✓ No]

Client IP address 72.191.29.194

Rule name	Start IP	End IP
AZ_HQ	45.220.13.0	45.220.13.255
TX_HQ	72.126.66.0	72.126.66.255

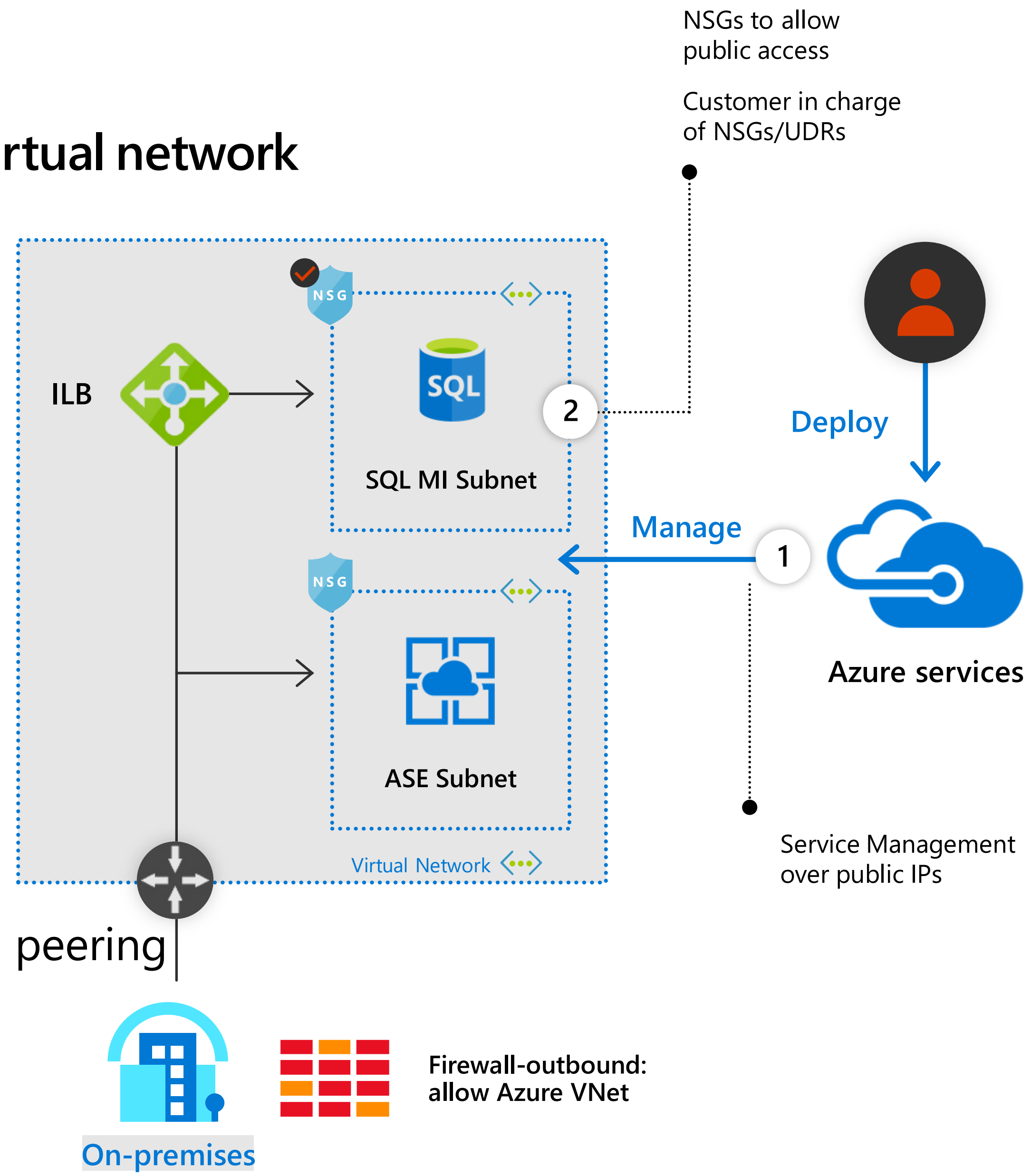
Virtual networks
+ Add existing virtual network + Create new virtual network

Rule name	Virtual network	Subnet	Address Range	Endpoint status
ALLOW_TX_PROD_AKS-01	VNTCTXPNETVNT01	AKS01	172.16.18.0/24	Enabled

VNet Injection

Service deploys dedicated instances into customer's virtual network

- ✓ Services in your VNet, managed by Azure!
- ✓ Single Tenancy; Private IPs for service resources
- ✓ Service data plane exposed privately, ILB
- ✓ Inbound and Outbound access to Azure Service
- ✓ On-premises access through Site-to-Site or ER private peering



Azure Private Link

Render or Consume Services Privately on Azure

Render a Service

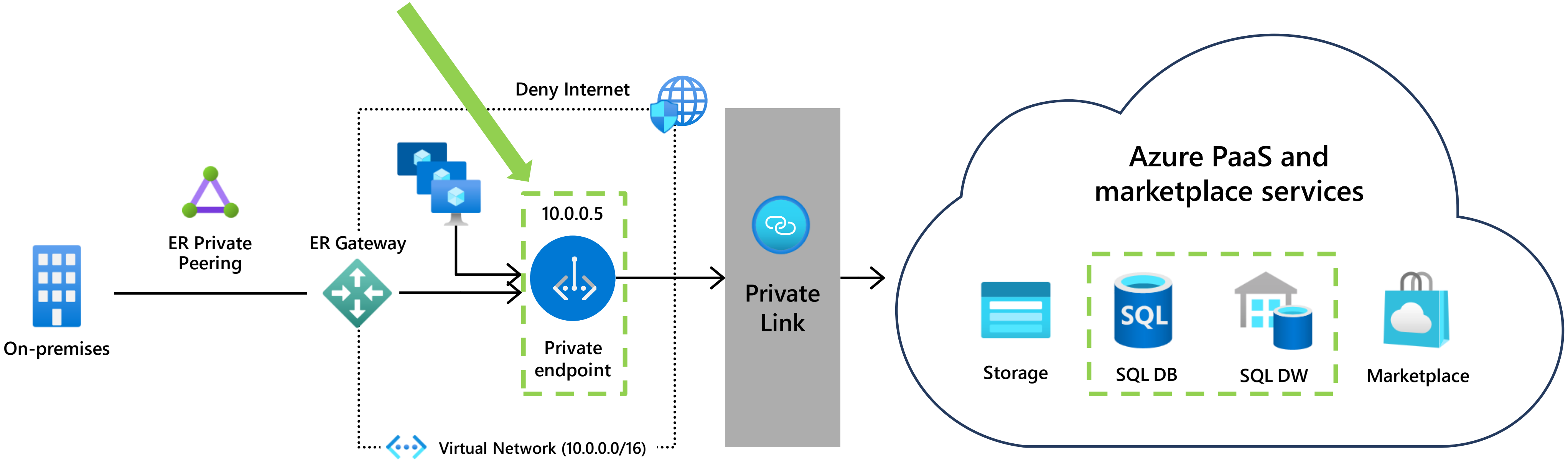
Persona: Service Provider
Resource: Private Link Service

Consume a Service

Persona: Service Consumer
Resource: Private Endpoint

Azure Private Link

Highly secure and private connectivity to Azure services

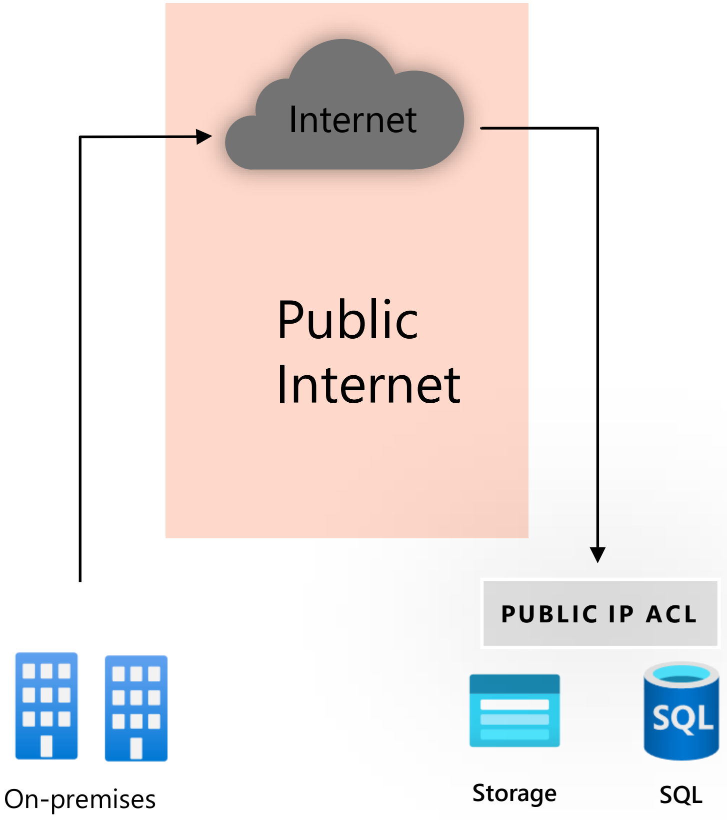


Private Link for Azure SQL DB (and other PaaS Services)

Private access from Virtual Network resources, peered networks and on-premise networks	In-built Data Exfiltration Protection	Predictable private IP addresses for PaaS resources	Unified experience across PaaS, Customer Owned and marketplace Services

Secure connectivity from on-premises

Good

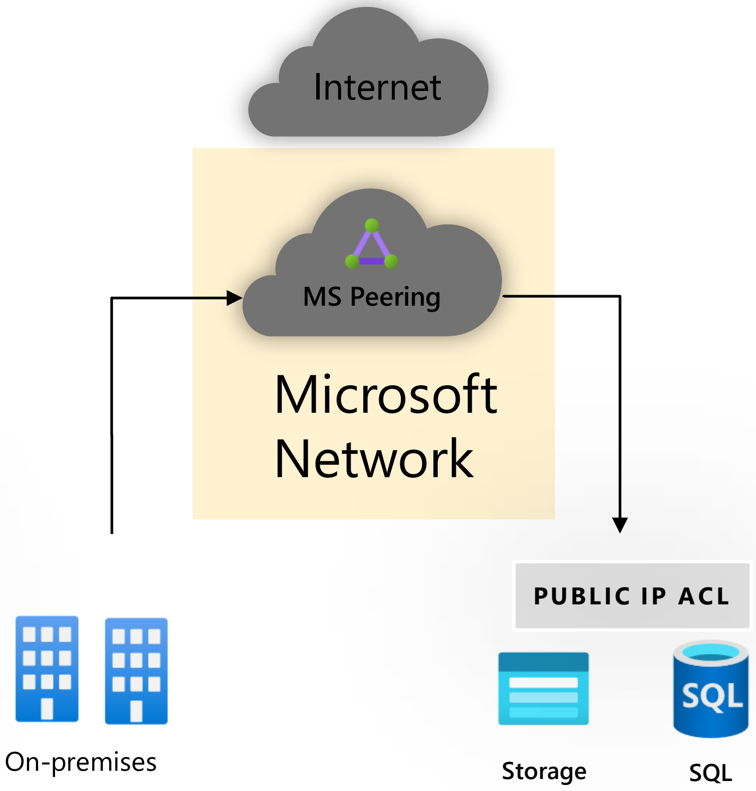


Traffic traverses the Internet

Secured using ACLs on Public Ips

Corporate firewall open to Azure Public IPs

Better

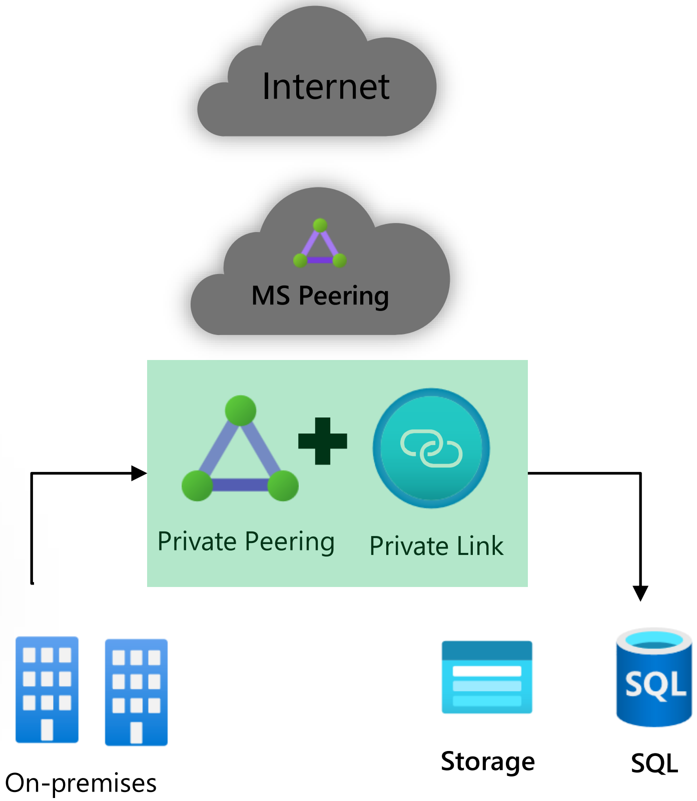


Traffic stays within Microsoft and partner network

MS Peering draws Microsoft Public IP traffic

Corporate Firewall open to Azure Public IPs

Best



Traffic is fully private traversing the Microsoft network

No exposure of public IPs on either side

Corporate Firewall open only to private

Okay so what?

Quick comparison of each service and when each make sense to use…

✓ Inbound access primarily from external sources, predictable ingress IP(s), no need for advanced NVA?

→ Basic PaaS Firewall

✓ Inbound access primarily from within Azure VNET, limited external access, no need for advanced NVA?

→ Service Endpoints

✓ Inbound access from VPN (or ER private peering) AND definite need for advanced NVA?

→ Private Link

✓ Inbound **AND outbound** access from VPN (or ER private peering) AND definite need for advanced NVA?

→ VNET Injection*