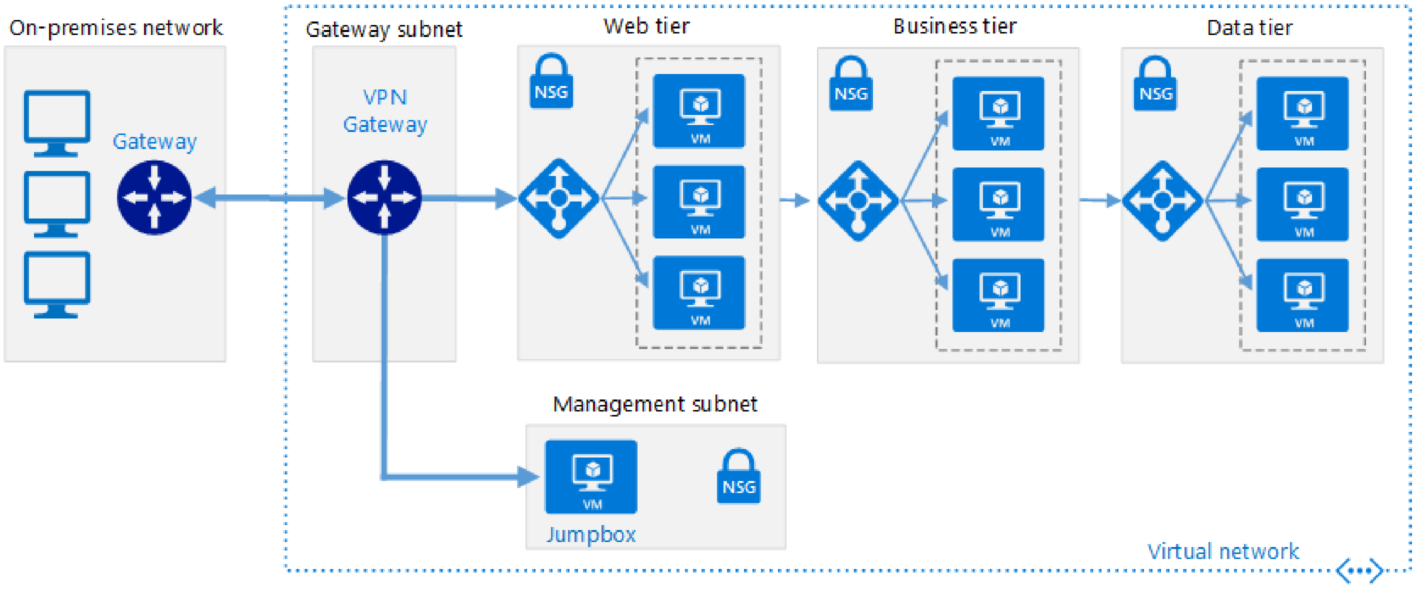
Connect an on-premises network to Azure using a VPN gateway

10/22/2018 • 9 minutes to read • Contributors all

In this article

This reference architecture shows how to extend an on-premises network to Azure, using a site-to-site virtual private network (VPN). Traffic flows between the on-premises network and an Azure Virtual Network (VNet) through an IPSec VPN tunnel. Deploy this solution.



Download a Visio file of this architecture.

Architecture

The architecture consists of the following components.

- **On-premises network**. A private local-area network running within an organization.
- **VPN appliance**. A device or service that provides external connectivity to the on-premises network. The VPN appliance may be a hardware device, or it can be a software solution such as the Routing and Remote Access Service (RRAS) in Windows Server 2012. For a list of supported VPN appliances and information on configuring them to connect to an Azure VPN gateway, see the instructions for the selected device in the article About VPN devices for Site-to-Site VPN Gateway connections.
- **Virtual network (VNet)**. The cloud application and the components for the Azure VPN gateway reside in the same VNet.

- **Azure VPN gateway**. The VPN gateway service enables you to connect the VNet to the on-premises network through a VPN appliance. For more information, see Connect an on-premises network to a Microsoft Azure virtual network. The VPN gateway includes the following elements:
  - **Virtual network gateway**. A resource that provides a virtual VPN appliance for the VNet. It is responsible for routing traffic from the on-premises network to the VNet.
  - **Local network gateway**. An abstraction of the on-premises VPN appliance. Network traffic from the cloud application to the on-premises network is routed through this gateway.
  - **Connection**. The connection has properties that specify the connection type (IPSec) and the key shared with the on-premises VPN appliance to encrypt traffic.
  - **Gateway subnet**. The virtual network gateway is held in its own subnet, which is subject to various requirements, described in the Recommendations section below.

- **Cloud application**. The application hosted in Azure. It might include multiple tiers, with multiple subnets connected through Azure load balancers. For more information about the application infrastructure, see Running Windows VM workloads and Running Linux VM workloads.

- **Internal load balancer**. Network traffic from the VPN gateway is routed to the cloud application through an internal load balancer. The load balancer is located in the front-end subnet of the application.

Recommendations

The following recommendations apply for most scenarios. Follow these recommendations unless you have a specific requirement that overrides them.

VNet and gateway subnet

Create an Azure VNet with an address space large enough for all of your required resources. Ensure that the VNet address space has sufficient room for growth if additional VMs are likely to be needed in the future. The address space of the VNet must not overlap with the on-premises network. For example, the diagram above uses the address space 10.20.0.0/16 for the VNet.

Create a subnet named *GatewaySubnet*, with an address range of /27. This subnet is required by the virtual network gateway. Allocating 32 addresses to this subnet will help to prevent reaching gateway size limitations in the future. Also, avoid placing this subnet in the middle of the address space. A good practice is to set the address space for the gateway subnet at the upper end of the VNet address space. The example shown in the diagram uses 10.20.255.224/27. Here is a quick procedure to calculate the CIDR:

1. Set the variable bits in the address space of the VNet to 1, up to the bits being used by the gateway subnet, then set the remaining bits to 0.
2. Convert the resulting bits to decimal and express it as an address space with the prefix length set to the size of the gateway subnet.

For example, for a VNet with an IP address range of 10.20.0.0/16, applying step #1 above becomes 10.20.0b11111111.0b11100000. Converting that to decimal and expressing it as an address space yields 10.20.255.224/27.

> ⚠ Warning
>
> Do not deploy any VMs to the gateway subnet. Also, do not assign an NSG to this subnet, as it will cause the gateway to stop functioning.

Virtual network gateway

Allocate a public IP address for the virtual network gateway.

Create the virtual network gateway in the gateway subnet and assign it the newly allocated public IP address. Use the gateway type that most closely matches your requirements and that is enabled by your VPN appliance:

- Create a policy-based gateway if you need to closely control how requests are routed based on policy criteria such as address prefixes. Policy-based gateways use static routing, and only work with site-to-site connections.
- Create a route-based gateway if you connect to the on-premises network using RRAS, support multi-site or cross-region connections, or implement VNet-to-VNet connections (including routes that traverse multiple VNets). Route-based gateways use dynamic routing to direct traffic between networks. They can tolerate failures in the network path better than static routes because they can try alternative routes. Route-based gateways can also reduce the management overhead because routes might not need to be updated manually when network addresses change.

For a list of supported VPN appliances, see About VPN devices for Site-to-Site VPN Gateway connections.

ⓘ Note

After the gateway has been created, you cannot change between gateway types without deleting and re-creating the gateway.

Select the Azure VPN gateway SKU that most closely matches your throughput requirements. For more information, see Gateway SKUs

ⓘ Note

The Basic SKU is not compatible with Azure ExpressRoute. You can **change the SKU** after the gateway has been created.

You are charged based on the amount of time that the gateway is provisioned and available. See VPN Gateway Pricing.

Create routing rules for the gateway subnet that direct incoming application traffic from the gateway to the internal load balancer, rather than allowing requests to pass directly to the application VMs.

On-premises network connection

Create a local network gateway. Specify the public IP address of the on-premises VPN appliance, and the address space of the on-premises network. Note that the on-premises VPN appliance must have a public IP address that can be accessed by the local network gateway in Azure VPN Gateway. The VPN device cannot be located behind a network address translation (NAT) device.

Create a site-to-site connection for the virtual network gateway and the local network gateway. Select the site-to-site (IPSec) connection type, and specify the shared key. Site-to-site encryption with the Azure VPN gateway is based on the IPSec protocol, using preshared keys for authentication. You specify the key when you create the Azure VPN gateway. You must configure the VPN appliance running on-premises with the same key. Other authentication mechanisms are not currently supported.

Ensure that the on-premises routing infrastructure is configured to forward requests intended for addresses in the Azure VNet to the VPN device.

Open any ports required by the cloud application in the on-premises network.

Test the connection to verify that:

- The on-premises VPN appliance correctly routes traffic to the cloud application through the Azure VPN gateway.
- The VNet correctly routes traffic back to the on-premises network.

- Prohibited traffic in both directions is blocked correctly.

Scalability considerations

You can achieve limited vertical scalability by moving from the Basic or Standard VPN Gateway SKUs to the High Performance VPN SKU.

For VNets that expect a large volume of VPN traffic, consider distributing the different workloads into separate smaller VNets and configuring a VPN gateway for each of them.

You can partition the VNet either horizontally or vertically. To partition horizontally, move some VM instances from each tier into subnets of the new VNet. The result is that each VNet has the same structure and functionality. To partition vertically, redesign each tier to divide the functionality into different logical areas (such as handling orders, invoicing, customer account management, and so on). Each functional area can then be placed in its own VNet.

Replicating an on-premises Active Directory domain controller in the VNet, and implementing DNS in the VNet, can help to reduce some of the security-related and administrative traffic flowing from on-premises to the cloud. For more information, see Extending Active Directory Domain Services (AD DS) to Azure.

Availability considerations

If you need to ensure that the on-premises network remains available to the Azure VPN gateway, implement a failover cluster for the on-premises VPN gateway.

If your organization has multiple on-premises sites, create multi-site connections to one or more Azure VNets. This approach requires dynamic (route-based) routing, so make sure that the on-premises VPN gateway supports this feature.

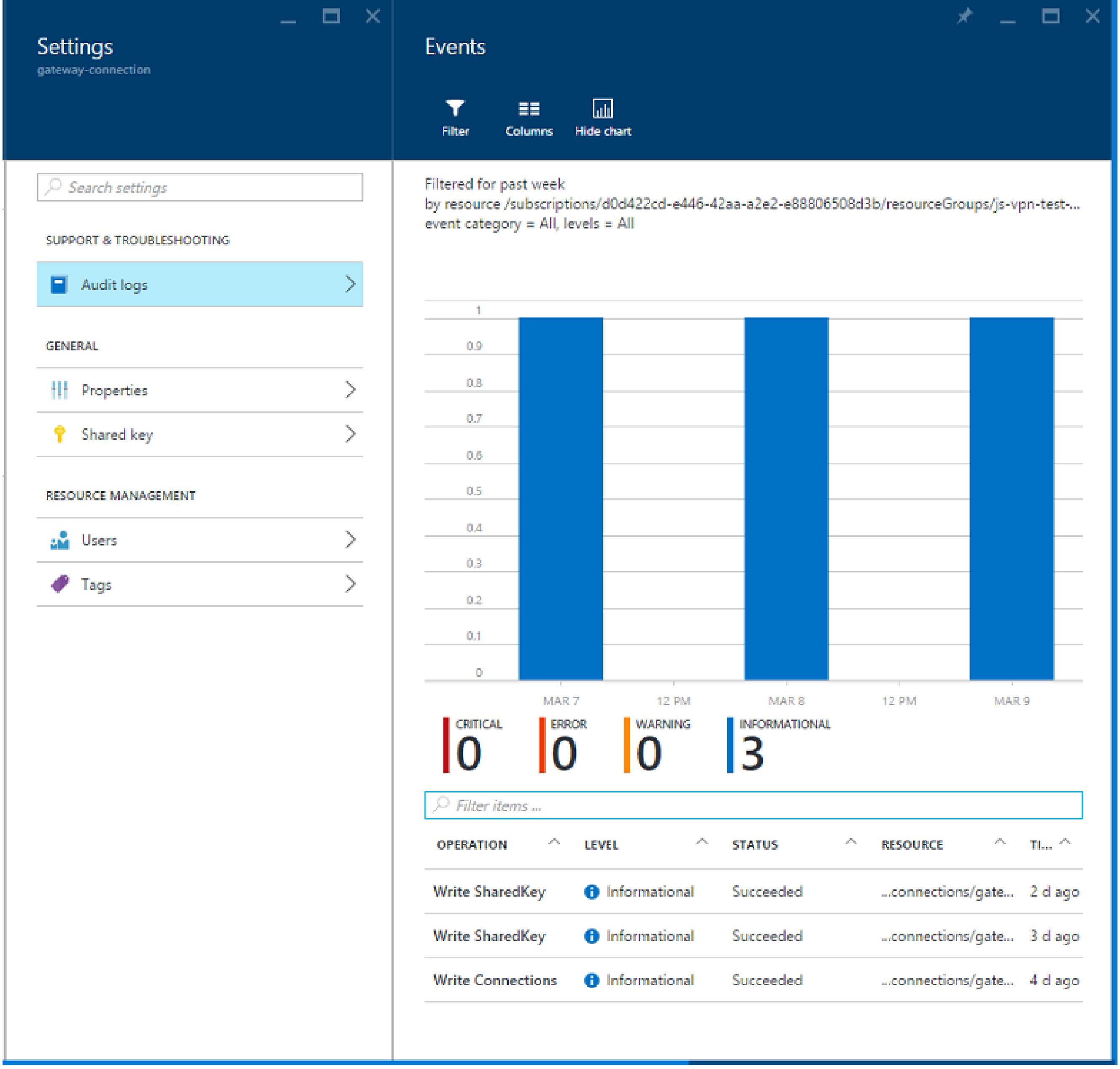
For details about service level agreements, see SLA for VPN Gateway.

Manageability considerations

Monitor diagnostic information from on-premises VPN appliances. This process depends on the features provided by the VPN appliance. For example, if you are using the Routing and Remote Access Service on Windows Server 2012, RRAS logging.

Use Azure VPN gateway diagnostics to capture information about connectivity issues. These logs can be used to track information such as the source and destinations of connection requests, which protocol was used, and how the connection was established (or why the attempt failed).

Monitor the operational logs of the Azure VPN gateway using the audit logs available in the Azure portal. Separate logs are available for the local network gateway, the Azure network gateway, and the connection. This information can be used to track any changes made to the gateway, and can be useful if a previously functioning gateway stops working for some reason.



Filtered for past week
by resource /subscriptions/d0d422cd-e446-42aa-a2e2-e88806508d3b/resourceGroups/js-vpn-test-...
event category = All, levels = All

CRITICAL 0 | ERROR 0 | WARNING 0 | INFORMATIONAL 3

OPERATION	LEVEL	STATUS	RESOURCE	TI...
Write SharedKey	Informational	Succeeded	...connections/gate...	2 d ago
Write SharedKey	Informational	Succeeded	...connections/gate...	3 d ago
Write Connections	Informational	Succeeded	...connections/gate...	4 d ago

Monitor connectivity, and track connectivity failure events. You can use a monitoring package such as Nagios to capture and report this information.

Security considerations

Generate a different shared key for each VPN gateway. Use a strong shared key to help resist brute-force attacks.

> ⓘ **Note**
>
> Currently, you cannot use Azure Key Vault to preshare keys for the Azure VPN gateway.

Ensure that the on-premises VPN appliance uses an encryption method that is compatible with the Azure VPN gateway. For policy-based routing, the Azure VPN gateway supports the AES256, AES128, and 3DES encryption algorithms. Route-based gateways support AES256 and 3DES.

If your on-premises VPN appliance is on a perimeter network (DMZ) that has a firewall between the perimeter network and the Internet, you might have to configure additional firewall rules to allow the site-to-site VPN connection.

If the application in the VNet sends data to the Internet, consider implementing forced tunneling to route all Internet-bound traffic through the on-premises network. This approach enables you to audit outgoing requests made by the application from the on-premises infrastructure.

> ⓘ **Note**
>
> Forced tunneling can impact connectivity to Azure services (the Storage Service, for example) and the Windows license manager.

Deploy the solution

Prerequisites. You must have an existing on-premises infrastructure already configured with a suitable network appliance.

To deploy the solution, perform the following steps.

1. Click the link below to deploy the solution.

   Deploy to Azure

2. Wait for the link to open in the Azure portal, then follow these steps:

   - The **Resource group** name is already defined in the parameter file, so select **Create New** and enter `ra-hybrid-vpn-rg` in the text box.
   - Select the region from the **Location** drop down box.
   - Do not edit the **Template Root Uri** or the **Parameter Root Uri** text boxes.
   - Review the terms and conditions, then click the **I agree to the terms and conditions stated above** checkbox.
   - Click the **Purchase** button.

3. Wait for the deployment to complete.

To troubleshoot the connection, see Troubleshoot a hybrid VPN connection.