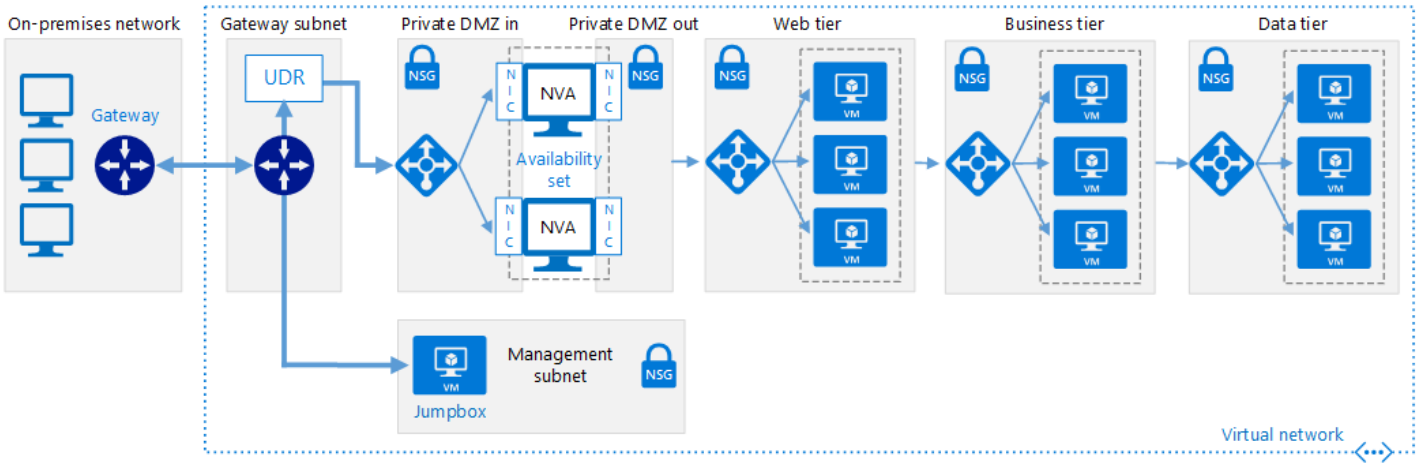
Implement a DMZ between Azure and your on-premises datacenter

10/22/2018 • 12 minutes to read • Contributors all

In this article

This reference architecture shows a secure hybrid network that extends an on-premises network to Azure. The architecture implements a DMZ, also called a *perimeter network*, between the on-premises network and an Azure virtual network (VNet). The DMZ includes network virtual appliances (NVAs) that implement security functionality such as firewalls and packet inspection. All outgoing traffic from the VNet is force-tunneled to the Internet through the on-premises network, so that it can be audited. Deploy this solution.

> **Note**
>
> This scenario can also be accomplished using Azure Firewall, a cloud-based network security service.



Download a Visio file of this architecture.

This architecture requires a connection to your on-premises datacenter, using either a VPN gateway or an ExpressRoute connection. Typical uses for this architecture include:

- Hybrid applications where workloads run partly on-premises and partly in Azure.
- Infrastructure that requires granular control over traffic entering an Azure VNet from an on-premises datacenter.
- Applications that must audit outgoing traffic. This is often a regulatory requirement of many commercial systems and can help to prevent public disclosure of private information.

Architecture

The architecture consists of the following components.

- **On-premises network**. A private local-area network implemented in an organization.
- **Azure virtual network (VNet)**. The VNet hosts the application and other resources running in Azure.
- **Gateway**. The gateway provides connectivity between the routers in the on-premises network and the VNet.
- **Network virtual appliance (NVA)**. NVA is a generic term that describes a VM performing tasks such as allowing or denying access as a firewall, optimizing wide area network (WAN) operations (including network compression), custom routing, or other network functionality.
- **Web tier, business tier, and data tier subnets**. Subnets hosting the VMs and services that implement an example 3-tier application running in the cloud. See Running Windows VMs for an N-tier architecture on Azure for more information.
- **User defined routes (UDR)**. User defined routes define the flow of IP traffic within Azure VNets.

  > **Note**
  >
  > Depending on the requirements of your VPN connection, you can configure Border Gateway Protocol (BGP) routes instead of using UDRs to implement the forwarding rules that direct traffic back through the on-premises network.

- **Management subnet**. This subnet contains VMs that implement management and monitoring capabilities for the components running in the VNet.

Recommendations

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

Access control recommendations

Use role-based access control (RBAC) to manage the resources in your application. Consider creating the following custom roles:

- A DevOps role with permissions to administer the infrastructure for the application, deploy the application components, and monitor and restart VMs.
- A centralized IT administrator role to manage and monitor network resources.
- A security IT administrator role to manage secure network resources such as the NVAs.

The DevOps and IT administrator roles should not have access to the NVA resources. This should be restricted to the security IT administrator role.

Resource group recommendations

Azure resources such as VMs, VNets, and load balancers can be easily managed by grouping them together into resource groups. Assign RBAC roles to each resource group to restrict access.

We recommend creating the following resource groups:

- A resource group containing the VNet (excluding the VMs), NSGs, and the gateway resources for connecting to the on-premises network. Assign the centralized IT administrator role to this resource group.

- A resource group containing the VMs for the NVAs (including the load balancer), the jumpbox and other management VMs, and the UDR for the gateway subnet that forces all traffic through the NVAs. Assign the security IT administrator role to this resource group.
- Separate resource groups for each application tier that contain the load balancer and VMs. Note that this resource group shouldn't include the subnets for each tier. Assign the DevOps role to this resource group.

Virtual network gateway recommendations

On-premises traffic passes to the VNet through a virtual network gateway. We recommend an Azure VPN gateway or an Azure ExpressRoute gateway.

NVA recommendations

NVAs provide different services for managing and monitoring network traffic. The Azure Marketplace offers several third-party vendor NVAs that you can use. If none of these third-party NVAs meet your requirements, you can create a custom NVA using VMs.

For example, the solution deployment for this reference architecture implements an NVA with the following functionality on a VM:

- Traffic is routed using IP forwarding on the NVA network interfaces (NICs).
- Traffic is permitted to pass through the NVA only if it is appropriate to do so. Each NVA VM in the reference architecture is a simple Linux router. Inbound traffic arrives on network interface *eth0*, and outbound traffic matches rules defined by custom scripts dispatched through network interface *eth1*.
- The NVAs can only be configured from the management subnet.
- Traffic routed to the management subnet does not pass through the NVAs. Otherwise, if the NVAs fail, there would be no route to the management subnet to fix them.
- The VMs for the NVA are placed in an availability set behind a load balancer. The UDR in the gateway subnet directs NVA requests to the load balancer.

Include a layer-7 NVA to terminate application connections at the NVA level and maintain affinity with the backend tiers. This guarantees symmetric connectivity, in which response traffic from the backend tiers returns through the NVA.

Another option to consider is connecting multiple NVAs in series, with each NVA performing a specialized security task. This allows each security function to be managed on a per-NVA basis. For example, an NVA implementing a firewall could be placed in series with an NVA running identity services. The tradeoff for ease of management is the addition of extra network hops that may increase latency, so ensure that this doesn't affect your application's performance.

NSG recommendations

The VPN gateway exposes a public IP address for the connection to the on-premises network. We recommend creating a network security group (NSG) for the inbound NVA subnet, with rules to block all traffic not originating from the on-premises network.

We also recommend NSGs for each subnet to provide a second level of protection against inbound traffic bypassing an incorrectly configured or disabled NVA. For example, the web tier subnet in the reference architecture implements an NSG with a rule to ignore all requests other than those received from the on-premises network (192.168.0.0/16) or the VNet, and another rule that ignores all requests not made on port 80.

Internet access recommendations

Force-tunnel all outbound Internet traffic through your on-premises network using the site-to-site VPN tunnel, and route to the Internet using network address translation (NAT). This prevents accidental leakage of any confidential information stored in your data tier and allows inspection and auditing of all outgoing traffic.

> **Note**
>
> Don't completely block Internet traffic from the application tiers, as this will prevent these tiers from using Azure PaaS services that rely on public IP addresses, such as VM diagnostics logging, downloading of VM extensions, and other functionality. Azure diagnostics also requires that components can read and write to an Azure Storage account.

Verify that outbound internet traffic is force-tunneled correctly. If you're using a VPN connection with the routing and remote access service on an on-premises server, use a tool such as WireShark or Microsoft Message Analyzer.

Management subnet recommendations

The management subnet contains a jumpbox that performs management and monitoring functionality. Restrict execution of all secure management tasks to the jumpbox.

Do not create a public IP address for the jumpbox. Instead, create one route to access the jumpbox through the incoming gateway. Create NSG rules so the management subnet only responds to requests from the allowed route.

Scalability considerations

The reference architecture uses a load balancer to direct on-premises network traffic to a pool of NVA devices, which route the traffic. The NVAs are placed in an availability set. This design allows you to monitor the throughput of the NVAs over time and add NVA devices in response to increases in load.

For details about the bandwidth limits of VPN Gateway, see Gateway SKUs. For higher bandwidths, consider upgrading to an ExpressRoute gateway. ExpressRoute provides up to 10 Gbps bandwidth with lower latency than a VPN connection.

For more information about the scalability of Azure gateways, see the scalability consideration section in Implementing a hybrid network architecture with Azure and on-premises VPN and Implementing a hybrid network architecture with Azure ExpressRoute.

Availability considerations

As mentioned, the reference architecture uses a pool of NVA devices behind a load balancer. The load balancer uses a health probe to monitor each NVA and will remove any unresponsive NVAs from the pool.

If you're using Azure ExpressRoute to provide connectivity between the VNet and on-premises network, configure a VPN gateway to provide failover if the ExpressRoute connection becomes unavailable.

For specific information on maintaining availability for VPN and ExpressRoute connections, see the availability considerations in Implementing a hybrid network architecture with Azure and on-premises VPN and Implementing a hybrid network architecture with Azure ExpressRoute.

Manageability considerations

All application and resource monitoring should be performed by the jumpbox in the management subnet. Depending on your application requirements, you may need additional monitoring resources in the management subnet. If so, these resources should be accessed through the jumpbox.

If gateway connectivity from your on-premises network to Azure is down, you can still reach the jumpbox by deploying a public IP address, adding it to the jumpbox, and remoting in from the internet.

Each tier's subnet in the reference architecture is protected by NSG rules. You may need to create a rule to open port 3389 for remote desktop protocol (RDP) access on Windows VMs or port 22 for secure shell (SSH) access on Linux VMs. Other management and monitoring tools may require rules to open additional ports.

If you're using ExpressRoute to provide the connectivity between your on-premises datacenter and Azure, use the Azure Connectivity Toolkit (AzureCT) to monitor and troubleshoot connection issues.

You can find additional information specifically aimed at monitoring and managing VPN and ExpressRoute connections in the articles Implementing a hybrid network architecture with Azure and on-premises VPN and Implementing a hybrid network architecture with Azure ExpressRoute.

Security considerations

This reference architecture implements multiple levels of security.

Routing all on-premises user requests through the NVA

The UDR in the gateway subnet blocks all user requests other than those received from on-premises. The UDR passes allowed requests to the NVAs in the private DMZ subnet, and these requests are passed on to the application if they are allowed by the NVA rules. You can add other routes to the UDR, but make sure they don't inadvertently bypass the NVAs or block administrative traffic intended for the management subnet.

The load balancer in front of the NVAs also acts as a security device by ignoring traffic on ports that are not open in the load balancing rules. The load balancers in the reference architecture only listen for HTTP requests on port 80 and HTTPS requests on port 443. Document any additional rules that you add to the load balancers, and monitor traffic to ensure there are no security issues.

Using NSGs to block/pass traffic between application tiers

Traffic between tiers is restricted by using NSGs. The business tier blocks all traffic that doesn't originate in the web tier, and the data tier blocks all traffic that doesn't originate in the business tier. If you have a requirement to expand the NSG rules to allow broader access to these tiers, weigh these requirements against the security risks. Each new inbound pathway represents an opportunity for accidental or purposeful data leakage or application damage.

DevOps access

Use RBAC to restrict the operations that DevOps can perform on each tier. When granting permissions, use the principle of least privilege. Log all administrative operations and perform regular audits to ensure any configuration changes were planned.

Deploy the solution

A deployment for a reference architecture that implements these recommendations is available on GitHub.

Prerequisites

1. Clone, fork, or download the zip file for the reference architectures GitHub repository.
2. Install Azure CLI 2.0.
3. Install Node and NPM
4. Install the Azure building blocks npm package.

```
bash
```

Copy

```
npm install -g @mspnp/azure-building-blocks
```

5. From a command prompt, bash prompt, or PowerShell prompt, sign into your Azure account as follows:

```bash
az login
```

Deploy resources

1. Navigate to the `/dmz/secure-vnet-hybrid` folder of the reference architectures GitHub repository.
2. Run the following command:

```bash
azbb -s <subscription_id> -g <resource_group_name> -l <region> -p onprem.json --deploy
```

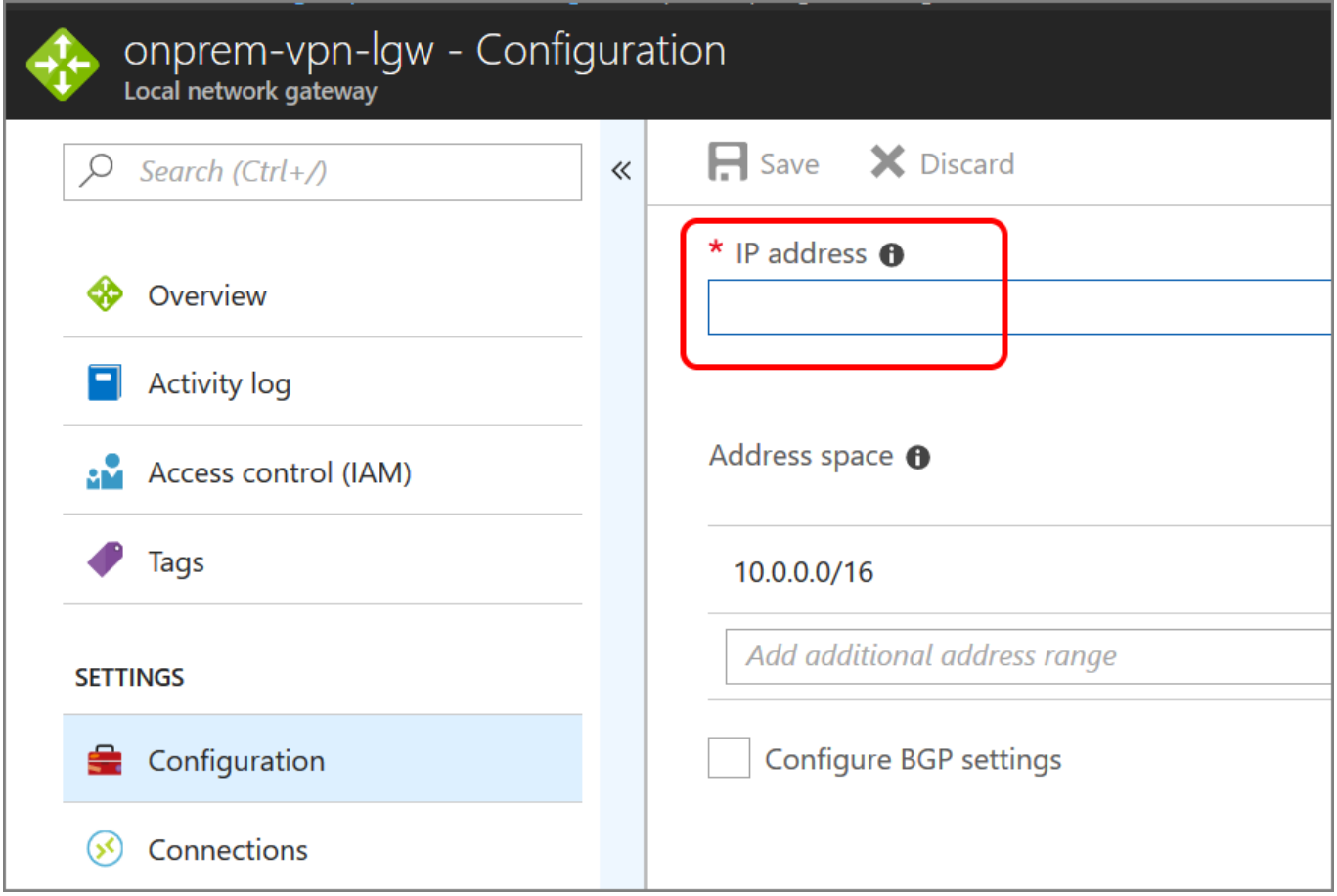
3. Run the following command:

```bash
azbb -s <subscription_id> -g <resource_group_name> -l <region> -p secure-vnet-hybrid.json --deploy
```

Connect the on-premises and Azure gateways

In this step, you will connect the two local network gateways.

1. In the Azure Portal, navigate to the resource group that you created.
2. Find the resource named `ra-vpn-vgw-pip` and copy the IP address shown in the **Overview** blade.
3. Find the resource named `onprem-vpn-lgw`.
4. Click the **Configuration** blade. Under **IP address**, paste in the IP address from step 2.



5. Click **Save** and wait for the operation to complete. It can take about 5 minutes.
6. Find the resource named `onprem-vpn-gateway1-pip`. Copy the IP address shown in the **Overview** blade.
7. Find the resource named `ra-vpn-lgw`.
8. Click the **Configuration** blade. Under **IP address**, paste in the IP address from step 6.
9. Click **Save** and wait for the operation to complete.
10. To verify the connection, go to the **Connections** blade for each gateway. The status should be **Connected**.

Verify that network traffic reaches the web tier

1. In the Azure Portal, navigate to the resource group that you created.
2. Find the resource named `int-dmz-lb`, which is the load balancer in front of the private DMZ. Copy the private IP address from the **Overview** blade.
3. Find the VM named `jb-vm1`. Click **Connect** and use Remote Desktop to connect to the VM. The user name and password are specified in the onprem.json file.
4. From the Remote Desktop Session, open a web browser and navigate to the IP address from step 2. You should see the default Apache2 server home page.

Next steps

- Learn how to implement a DMZ between Azure and the Internet.
- Learn how to implement a highly available hybrid network architecture.
- For more information about managing network security with Azure, see Microsoft cloud services and network security.
- For detailed information about protecting resources in Azure, see Getting started with Microsoft Azure security.

- For additional details on addressing security concerns across an Azure gateway connection, see Implementing a hybrid network architecture with Azure and on-premises VPN and Implementing a hybrid network architecture with Azure ExpressRoute.
- Troubleshoot network virtual appliance issues in Azure