About using GitHub-hosted runners in your Azure Virtual Network

In this article

You can create GitHub-hosted runners in your Azure Virtual Network(s) (VNET).

About using GitHub-hosted runners in your Azure Virtual Network (VNET)

Notes:

- Using GitHub-hosted larger runners with an Azure Virtual Network (VNET) is in private beta and subject to change. This feature may not be available to all users.
- Only larger runners are supported with Azure VNET. For more information about larger runners, see "About larger runners."

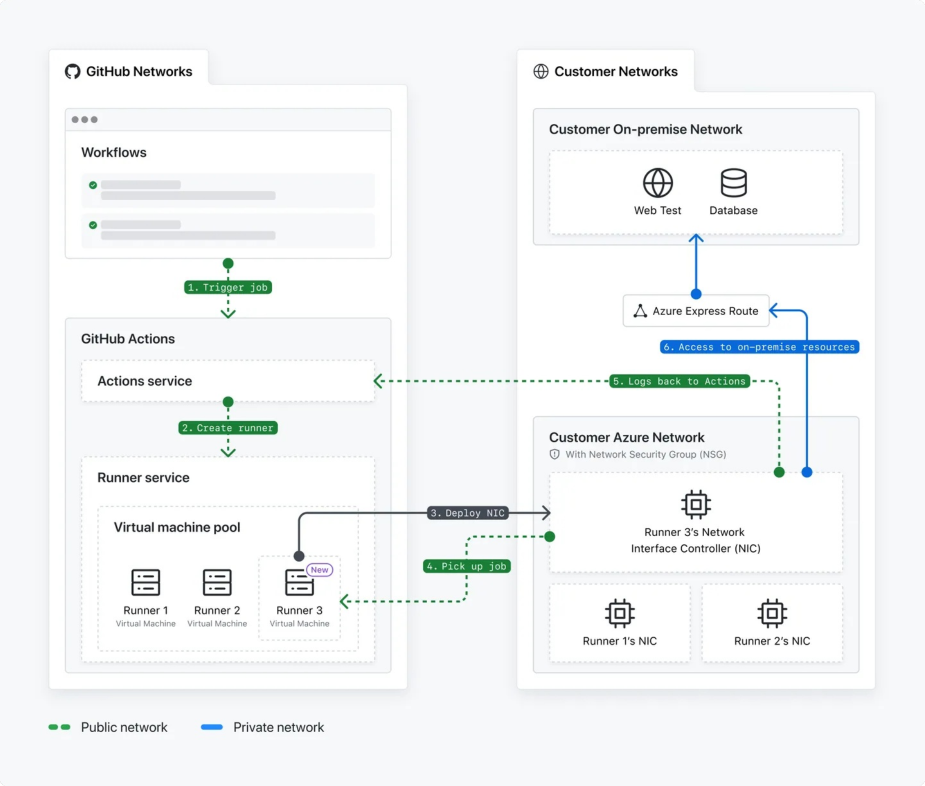
If you are using Azure and GitHub Enterprise Cloud, you can create GitHub-hosted runners in your Azure VNET(s). This enables you to take advantage of GitHub-managed infrastructure for your CI/CD while providing you with full control over the networking policies of your runners. For more information about Azure VNET, see What is Azure Virtual Network? in the Azure documentation.

Using GitHub-hosted runners within Azure VNET allows you to perform the following actions.

- Privately connect a runner to resources inside an Azure VNET without opening internet ports, including on-premises resources accessible from the Azure VNET.
- Restrict what GitHub-hosted runners can access or connect to with full control over outbound network policies.
- Monitor network logs for GitHub-hosted runners and view all connectivity to and from a runner.

About network communication

To facilitate communication between GitHub networks and your VNET, GitHub-hosted runner's network interface card (NIC) deploys into your Azure VNET. This way, all communication is kept private within the network boundaries, and networking policies applied to the VNET also apply to the runner.



1. A GitHub Actions workflow is triggered.
2. The GitHub Actions service creates a runner.
3. The runner service deploys the GitHub-hosted runner's network interface card (NIC) into your Azure VNET.
4. The runner agent picks up the workflow job. The GitHub Actions service queues the job.
5. The runner sends logs back to the GitHub Actions service.
6. The NIC accesses on-premise resources.

Using your VNET's network policies

Because the GitHub-hosted runner's NIC is deployed into your Azure VNET, networking policies applied to the VNET also apply to the runner.

For example, if your VNET is configured with an Azure ExpressRoute to provide access to on-premises resources (e.g. Artifactory) or connected to a VPN tunnel to provide access to other cloud-based resources, those access policies also apply to your runners. Additionally, any outbound rules applied to your VNET's network security group (NSG) also apply, giving you the ability to control outbound access for your runners.

If you have enabled any network logs monitoring for your VNET, you can also monitor network traffic for your runners.

Using GitHub-hosted runners with an Azure VNET

To use GitHub-hosted runners with Azure VNET, you must configure Azure and configure your GitHub settings to use GitHub-hosted runners with a VNET.

For more information about configuring Azure, see "Configuring Azure resources for private networking with GitHub-hosted runners."

For more information about configuring your GitHub settings to use GitHub-hosted runners with a VNET, see "Configuring your GitHub settings for use with Azure Virtual Network."