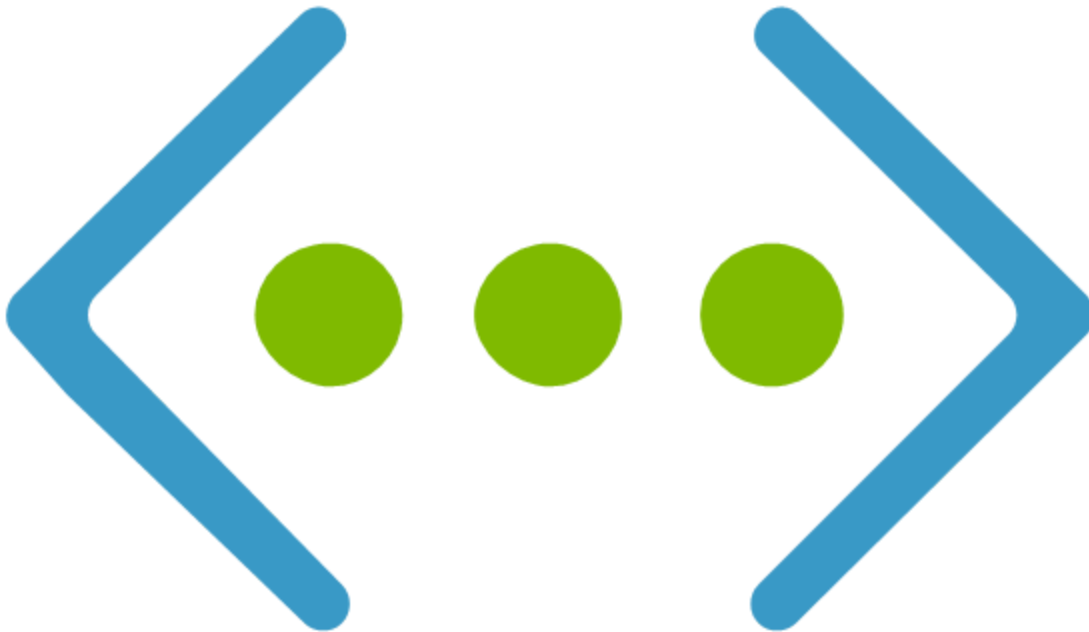


# Google Cloud VPN Interop Guide

Using Cloud VPN With Microsoft Azure™ VPN Gateway



Courtesy of Microsoft, Inc. Unauthorized use not permitted. Microsoft Azure® is a registered trademark or trademark of Microsoft, Inc. and/or its affiliates in the United States and certain other countries.

*Disclaimer: This interoperability guide is intended to be informational in nature and are examples only. Customers should verify this information via testing.*

# Contents

[Contents](#)

[Introduction](#)

[Topology](#)

[Preparation](#)

[Overview](#)

[IPsec Parameters](#)

[Configuration - Azure](#)

[Getting Started](#)

[Creating the Virtual and Local Networks](#)

[Configuration - GCP](#)

[Collecting the Required Details from Google Cloud Platform](#)

[Completing the Virtual and Local Network Configuration](#)

[Finalizing the Google Cloud Platform Configuration](#)

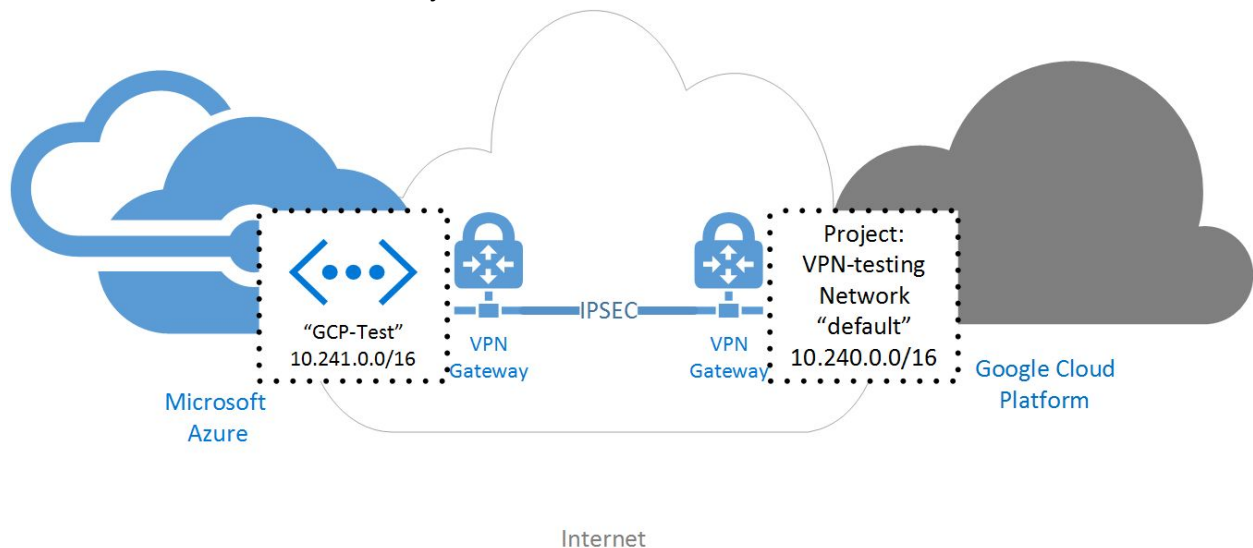
[Testing the Site-to-Site VPN](#)

# Introduction

This guide walks you through the process of configuring the Azure Virtual Network Gateway for integration with the [Google Cloud VPN service](#). This information is provided as an example only. Please note that this guide is not meant to be a comprehensive overview of IPsec and assumes basic familiarity with the IPsec protocol.

## Topology

The topology outlined by this guide is a basic site-to-site IPsec VPN tunnel configuration using the standard Azure VPN Gateway:



## Preparation

### Overview

The configuration samples which follow will include numerous value substitutions provided for the purposes of example only. Any references to IP addresses, device IDs, shared secrets or keys, account information or project names should be replaced with the appropriate values for your environment when following this guide.

This guide is intended to assist in the creation of IPsec connectivity to Google Compute Engine. The following is a high level overview of the configuration process which will be covered:

- Configuring the Azure Virtual Network Gateway
- Configuring the Google Cloud Platform VPN
- Connecting to GCP
- Testing the tunnel

The IPsec connectivity will utilize an **Azure generated pre-shared key** for authentication and will require the **dynamic routing** Azure gateway type.

## IPsec Parameters

For the Azure IPsec configuration, the following details will be used:

Parameter	Value
IPsec Mode	ESP+Auth Tunnel mode (Site-to-Site)
Auth Protocol	Pre-shared Key
Key Exchange	IKEv2
Start	auto
Perfect Forward Secrecy (PFS)	on
Dead Peer Detection (DPD)	aggressive
INITIAL_CONTACT (uniqueids)	on

The IPsec configuration used in this guide is specified below:

<i>Phase</i>	<i>Cipher Role</i>	<i>Cipher</i>
<i>Phase 1</i>	<i>Encryption</i>	<i>aes-256</i>
	<i>Integrity</i>	<i>sha-256</i>
	<i>prf</i>	<i>sha1-96</i>
	<i>Diffie-Hellman (DH)</i>	<i>Group 14 (modp_2048)</i>
	<i>Phase 1 lifetime</i>	<i>36,000 seconds (10 hours)</i>
<i>Phase 2</i>	<i>Encryption</i>	<i>aes-cbc-256</i>
	<i>Integrity</i>	<i>sha-256</i>

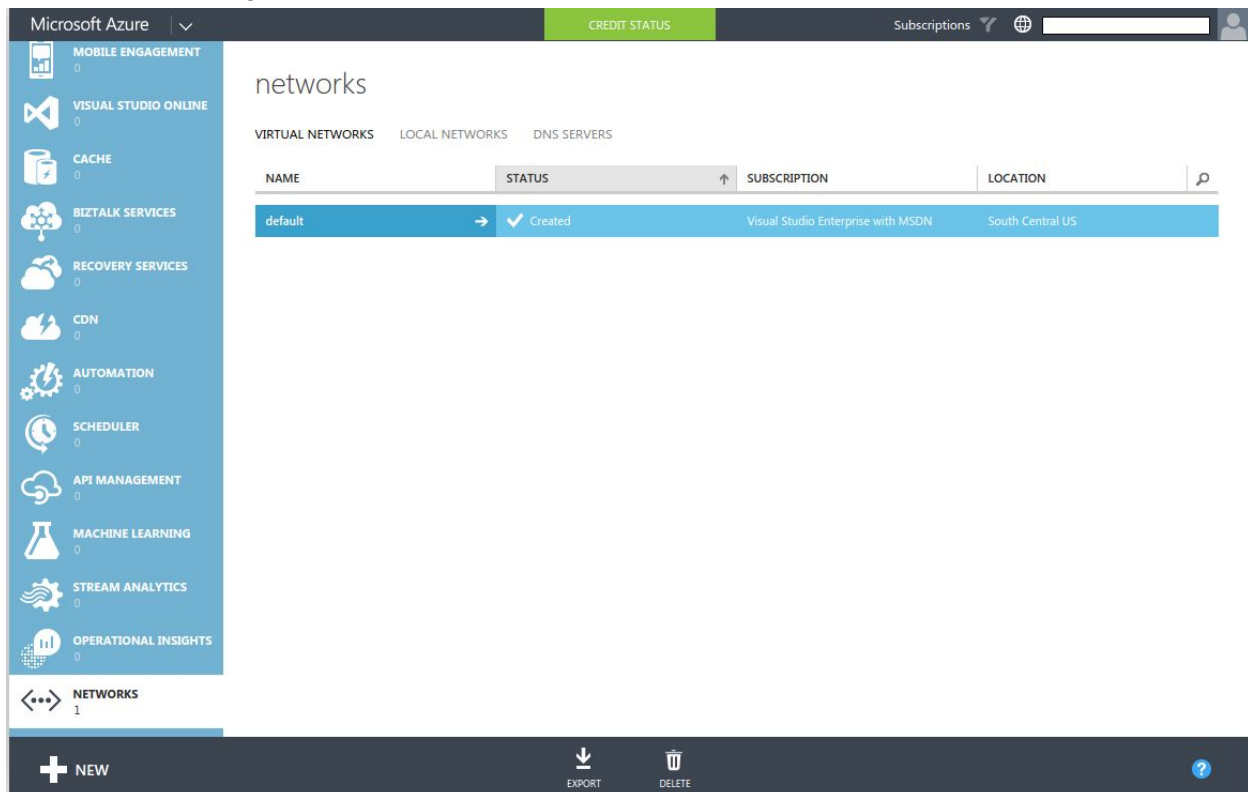
# Configuration - Azure

## Getting Started

The first step is to establish the base networking environment in Azure. Microsoft provides [documentation](#) for getting started with Azure networking. The basic concepts to understand are:

- **Virtual Networks** – these are private networks defined in the cloud service.
- **Gateway Network** – this is a subnet allocated from the “virtual network” IP space. This subnet will be the home network of the Azure IPsec gateway
- **Local Networks** – these are the on-premise networks that will be exposed to the Azure network via the IPsec tunnel. In the case of GCP integration, this will represent the GCP network into which the GCP VPN gateway was deployed

To get started, login to the Azure Management Console and select the *Networks* entry from the left hand services panel. Any existing networks will be displayed in the main panel. In our case we have an existing *virtual network* named “default”:



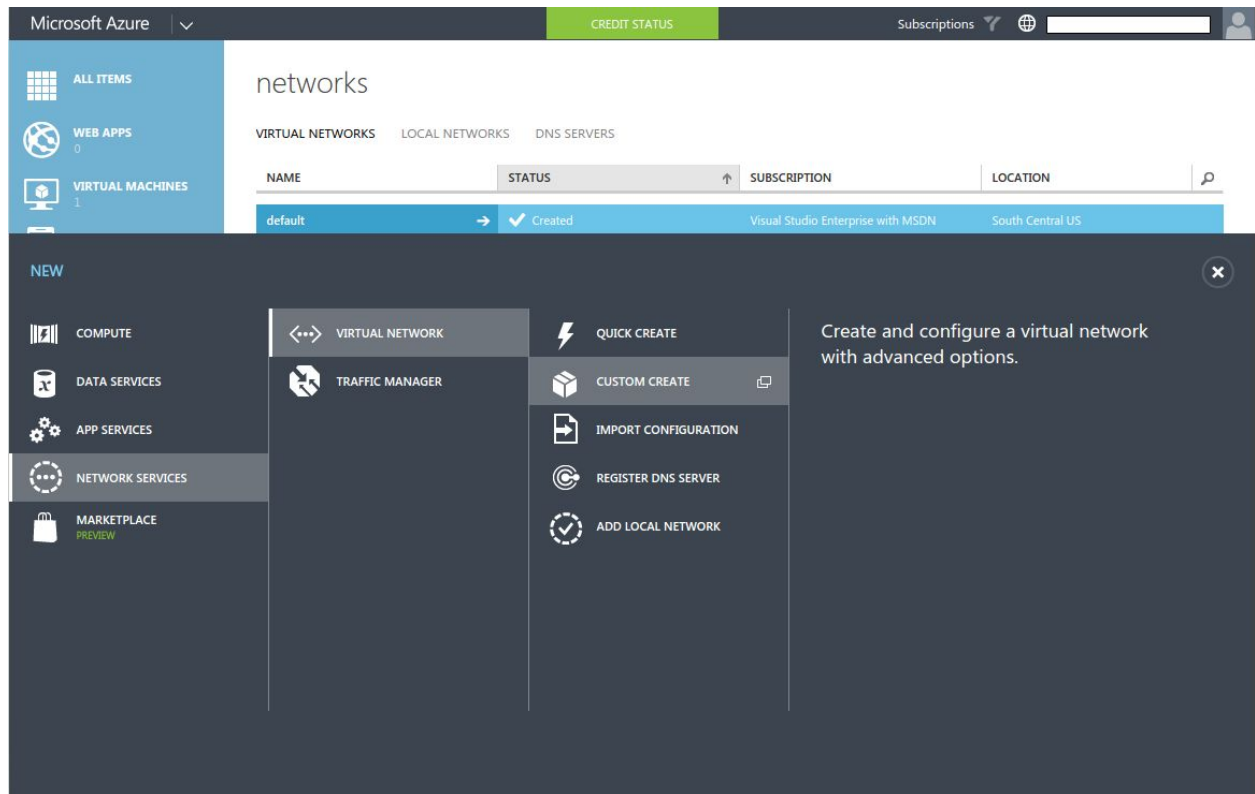
The screenshot shows the Microsoft Azure Management Console interface. The top navigation bar includes 'Microsoft Azure', 'CREDIT STATUS', and 'Subscriptions'. The left sidebar lists various services, with 'NETWORKS' selected and showing a count of 1. The main panel displays the 'networks' page, which includes tabs for 'VIRTUAL NETWORKS', 'LOCAL NETWORKS', and 'DNS SERVERS'. A table lists the virtual networks, with one entry named 'default'.

NAME	STATUS	SUBSCRIPTION	LOCATION
default	Created	Visual Studio Enterprise with MSDN	South Central US

## Creating the Virtual and Local Networks

For this guide, we will be creating a new virtual network named “GCP-Test” to use to connect to GCP. In addition, we will be assigning a new *local network* to the GCP-Test virtual network named “GCP”.

To get started, select NEW from the lower left of the action bar. This will invoke the Virtual Network creation workflow. Select CUSTOM CREATE:



The Virtual Network workflow is organized into pages. The first page is the basic Virtual Network Details. Enter the name of the new Virtual Network (“GCP-Test” in our case) and select both a *location* and the *associated Azure subscription*:

CREATE A VIRTUAL NETWORK ×

## Virtual Network Details

**NAME**  
GCP-Test

**LOCATION**  
South Central US

**SUBSCRIPTION**  
Visual Studio Enterprise with MSDN

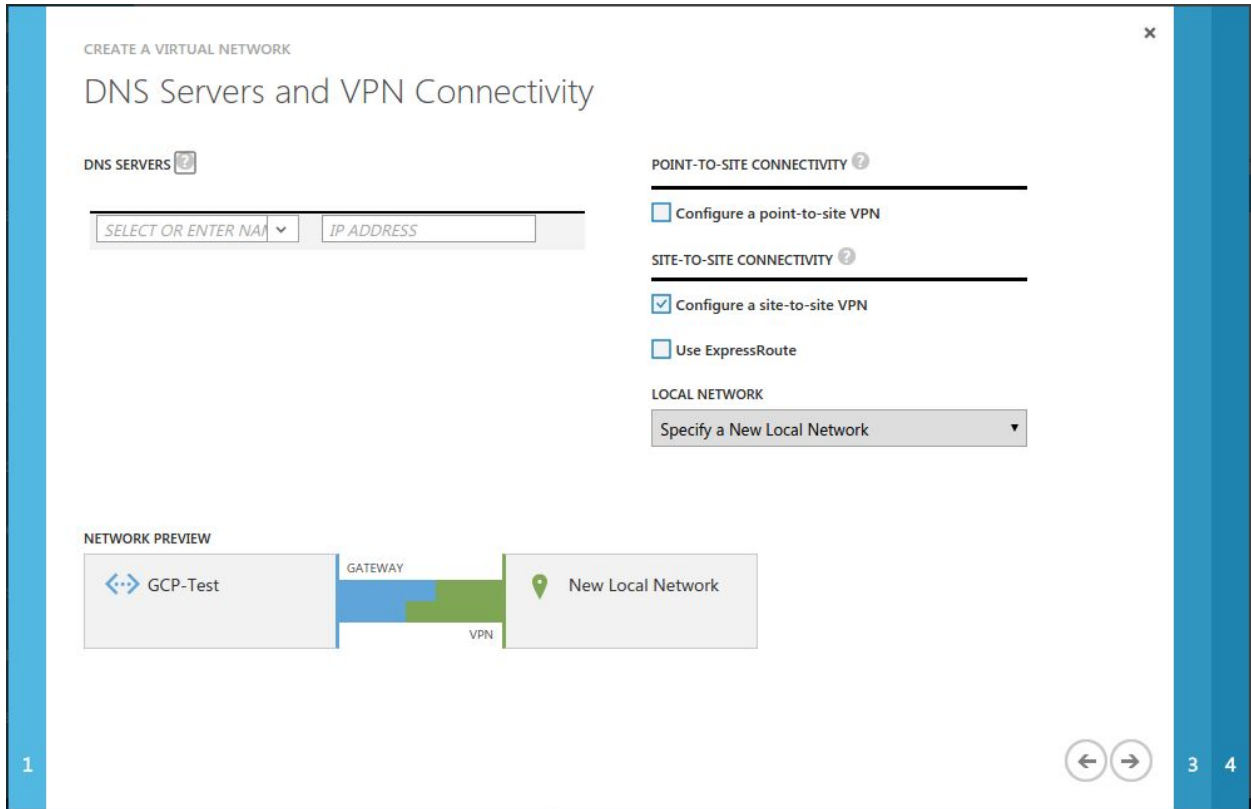
**NETWORK PREVIEW**  
↔ GCP-Test

→ 2 3

On the second page we enter the configuration for DNS and VPN Connectivity. The *optional* DNS SERVERS setting allows you to enter any *private* DNS servers you have deployed in the network you will be *connecting to Azure via VPN*, or any public DNS servers you want to explicitly specify. In our case we will *not* be specifying DNS.

SITE-TO-SITE CONNECTIVITY enables the IPsec VPN configuration workflow and should be checked. Checking this option will require selecting a *Local Network*. We will be *creating a New Local Network* for this exercise. Once all options have been entered, click the lower righthand arrow to move forward to the next step:





The SITE-TO-SITE CONNECTIVITY configuration panel is where the details of the Google Cloud VPN configuration are entered. At this point, enter a name we for the local network (in this case “GCP”), pause the Azure configuration process, and move over to GCP to collect the required Local Network details:

CREATE A VIRTUAL NETWORK

## Site-to-Site Connectivity

NAME

VPN DEVICE IP ADDRESS

ADDRESS SPACE

ADDRESS SPACE	STARTING IP	CIDR (ADDRESS COUNT)	USABLE ADDRESS RANGE
<i>ADDRESS SPACE</i>	<i>STARTING IP</i>	<i>/8 (16777...)</i>	<i>USABLE ADDRESS RANGE</i>

[add address space](#)

NETWORK PREVIEW

The network preview diagram shows a box labeled 'GCP-Test' on the left and a box labeled 'GCP' on the right. A blue line connects them, with 'GATEWAY' written above it and 'VPN' written below it. The 'GCP' box is highlighted with a yellow border.

1 2 ← → 4

## Completing the Virtual and Local Network Configuration

Once complete, the VPN properties form will display the newly allocated public IP address which will be used by the VPN. With this information we now have everything required to continue the Azure configuration. Return to where we left off with the Azure Virtual Network workflow and enter the public IP assigned to the GCP VPN gateway and the private IP space of the GCP network:

CREATE A VIRTUAL NETWORK

## Site-to-Site Connectivity

NAME:

VPN DEVICE IP ADDRESS:

ADDRESS SPACE

ADDRESS SPACE	STARTING IP	CIDR (ADDRESS COUNT)	USABLE ADDRESS RANGE
10.240.0.0/16	10.240.0.0	/16 (65536)	10.240.0.0 - 10.240.255.255

[add address space](#)

NETWORK PREVIEW

The network preview shows a box labeled 'GCP-Test' on the left and a box labeled 'GCP' on the right. A blue line labeled 'GATEWAY' connects the two boxes, and a green line labeled 'VPN' is positioned below it. The 'GCP' box is highlighted with a yellow border.

1 2 ← → 4

The next step is to configure the Virtual Network address space within Azure. In this case we will accept the default of 10.241.0.0/16 as this is compatible with our GCP address space of 10.240.0.0/16. The important thing is that these IP address spaces *must not overlap*. IPsec will be linking these two networks and will not tolerate an IP range conflict. The final step is to add a “Gateway Subnet” by clicking the “add a gateway subnet” button. With all required items completed we can click the check mark on the lower right to execute the configuration:

CREATE A VIRTUAL NETWORK

## Virtual Network Address Spaces

ADDRESS SPACE	STARTING IP	CIDR (ADDRESS COUNT)	USABLE ADDRESS RANGE
10.241.0.0/16	10.241.0.0	/16 (65536)	10.241.0.0 - 10.241.255.255
<b>SUBNETS</b>			
Subnet-1	10.241.0.0	/19 (8192)	10.241.0.0 - 10.241.31.255
Gateway	10.241.32.0	/29 (8)	10.241.32.0 - 10.241.32.7

add subnet    add gateway subnet

add address space

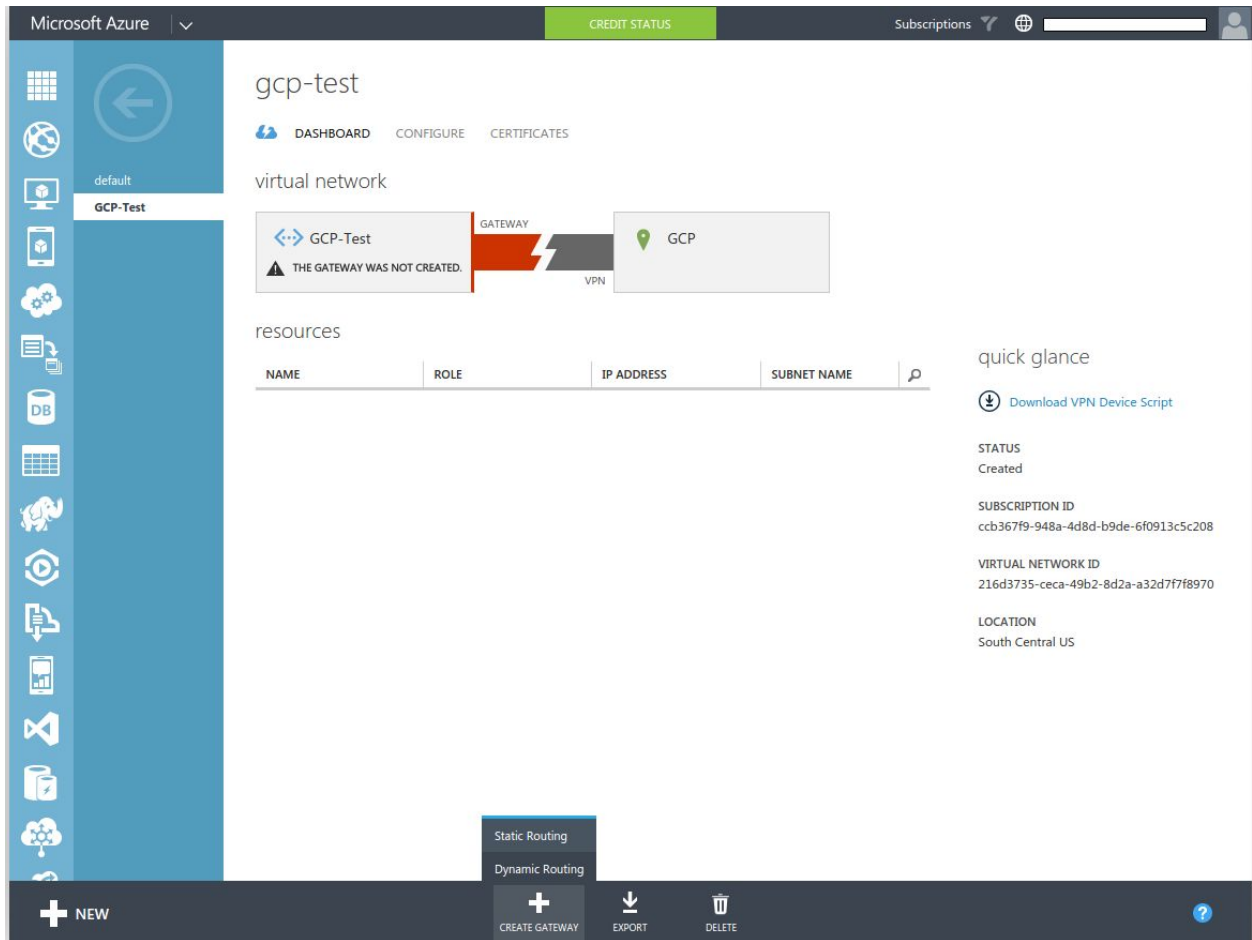
NETWORK PREVIEW

The diagram shows a box labeled 'GCP-Test' on the left, connected to a box labeled 'GCP' on the right. A blue line labeled 'GATEWAY' connects the two boxes, and a green line labeled 'VPN' also connects them.

1 2 3

← ✓

Once the Virtual Network has been created it will appear as a clickable entry in the Azure networks list. Clicking the Virtual Network will provide access to its dashboard which will display the current status of the Virtual Network. The next step is to create the VPN Gateway. To do this click the “Create Gateway” option in the action bar and be sure to select “**dynamic routing**” as the gateway type:



The Azure gateway may take some time to create. The expected behavior at this stage is for the creation to complete, but not successfully connect since the configuration on the GCP side is not yet complete. At this stage the IPsec pre-shared key which will be used for authentication, and the Azure VPN Gateway public IP address are required. When the Azure VPN gateway creation is complete the dashboard will update to show the public IP assigned (redacted in this example):

Microsoft Azure | CREDIT STATUS | Subscriptions | mlambert890@hotmail.com

# gcp-test

DASHBOARD | CONFIGURE | CERTIFICATES

## virtual network

**LAST GATEWAY EVENT** Unable to establish the cross-premise tunnel for site 'GCP'. Previous state: Initializing. Current state: Not Connected. 10/28/2015 2:33:02 PM

GCP-Test (GATEWAY) --- VPN --- GCP (DISCONNECTED)

DATA IN: 0B | DATA OUT: 0B | GATEWAY IP ADDRESS: PUBLIC IP TO USE IN GCP SETUP

NAME	ROLE	IP ADDRESS	SUBNET NAME
------	------	------------	-------------

**quick glance**

- Download VPN Device Script
- STATUS: Created
- SUBSCRIPTION ID: ccb367f9-948a-4d8d-b9de-6f0913c5c208
- VIRTUAL NETWORK ID: 216d3735-ceca-49b2-8d2a-a32d7f7f8970
- LOCATION: South Central US
- GATEWAY TYPE: Dynamic Routing

+ NEW | DELETE GATEWAY | CONNECT | EXPORT | MANAGE KEY | DELETE

The final step is to click on **MANAGE KEY** in the action bar to retrieve the IPsec pre-shared key automatically generated by Azure. Click the clipboard icon to copy the key:



# Manage Shared Key

Use this key to configure your local network VPN device to connect to the virtual network.

## MANAGE SHARED KEY

AUTO GENERATED PRE-SHARED KEY



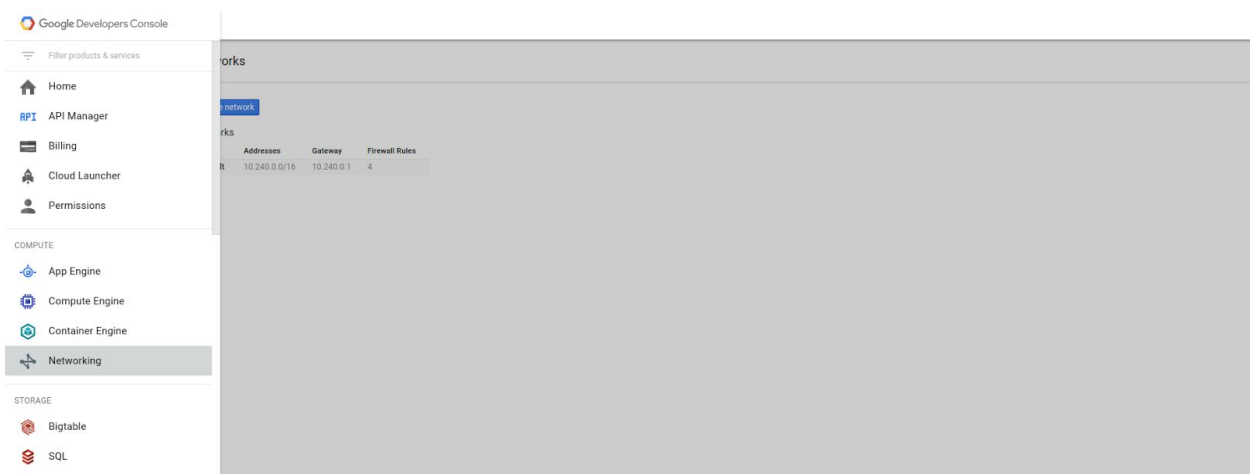
regenerate key



# Configuration - GCP

## Collecting the Required Details from Google Cloud Platform

In the Google Cloud Platform Developers Console, select the project into which the VPN will be deployed, or create a new project. More information on creating and managing projects can be found [here](#). To view the current network configuration for the project, select the Networking entry from the main services menu:



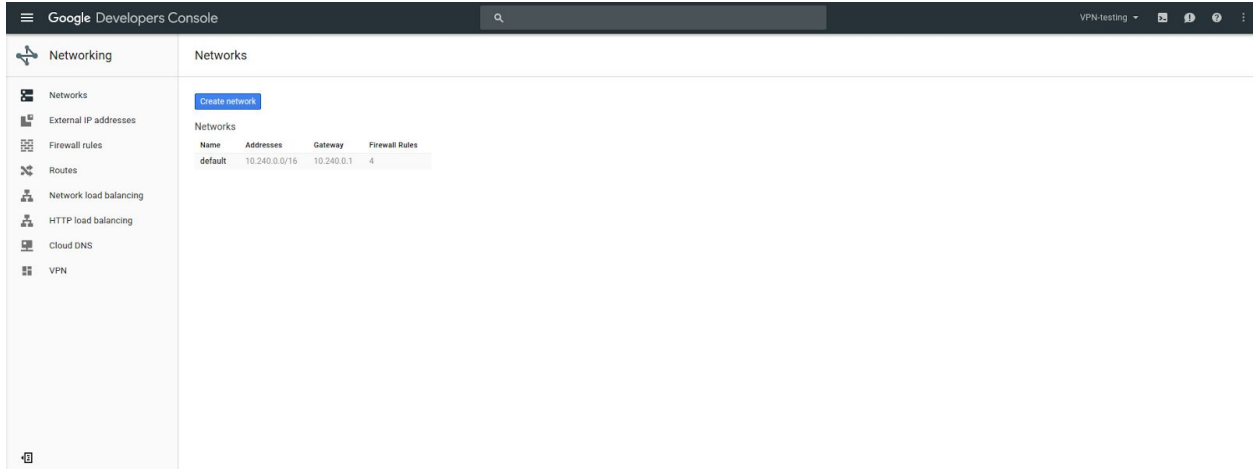
In GCP all projects start with a single network named *default* at time of creation. The default network is configured with a private IP space and a set of base firewall rules. The default network provides a sufficient starting point for creating a site-to-site IPsec VPN. More information on networking within the Google Cloud Platform can be found in the [Networking section](#) of the Google Compute Engine documentation.

To continue with IPsec site-to-site VPN configuration on the Azure side, two values are needed from GCP:

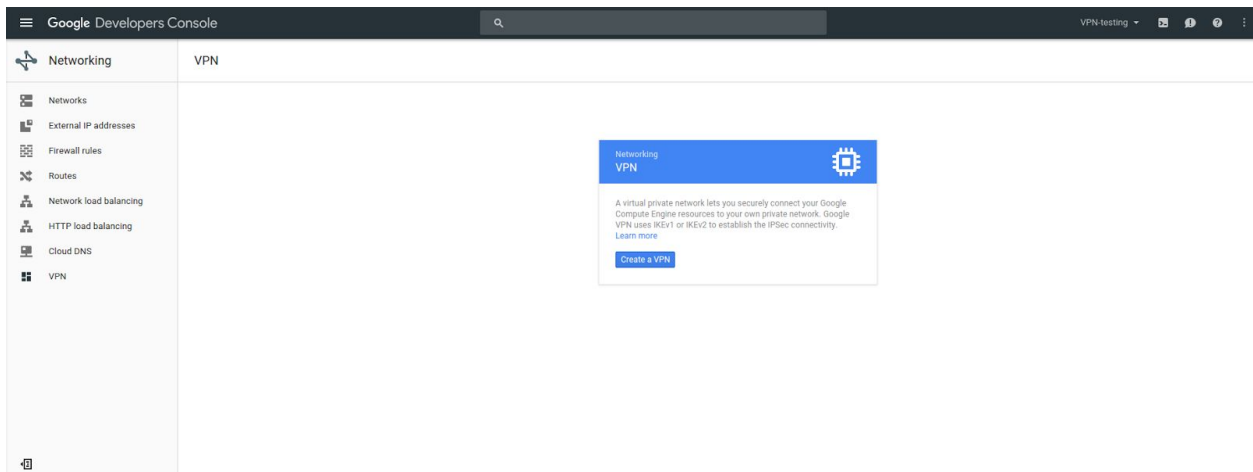
- **VPN Device IP Address:** the public IP address of the VPN gateway in Google Cloud
- **Address Space:** the private IP address space associated with the Google Cloud Platform Network

The address space is shown in the network overview and in our case is 10.240.0.0/16:





To get the VPN device IP address, we will need to create a Google Cloud VPN gateway. From the Networking menu, select VPN. Any existing VPN gateways will be listed in the main information panel. If no VPN gateways have been created, an option will be provided to create one:



Click "Create a VPN" to initiate the VPN creation workflow:

The VPN has several user configurable properties. At this stage we can set the following:

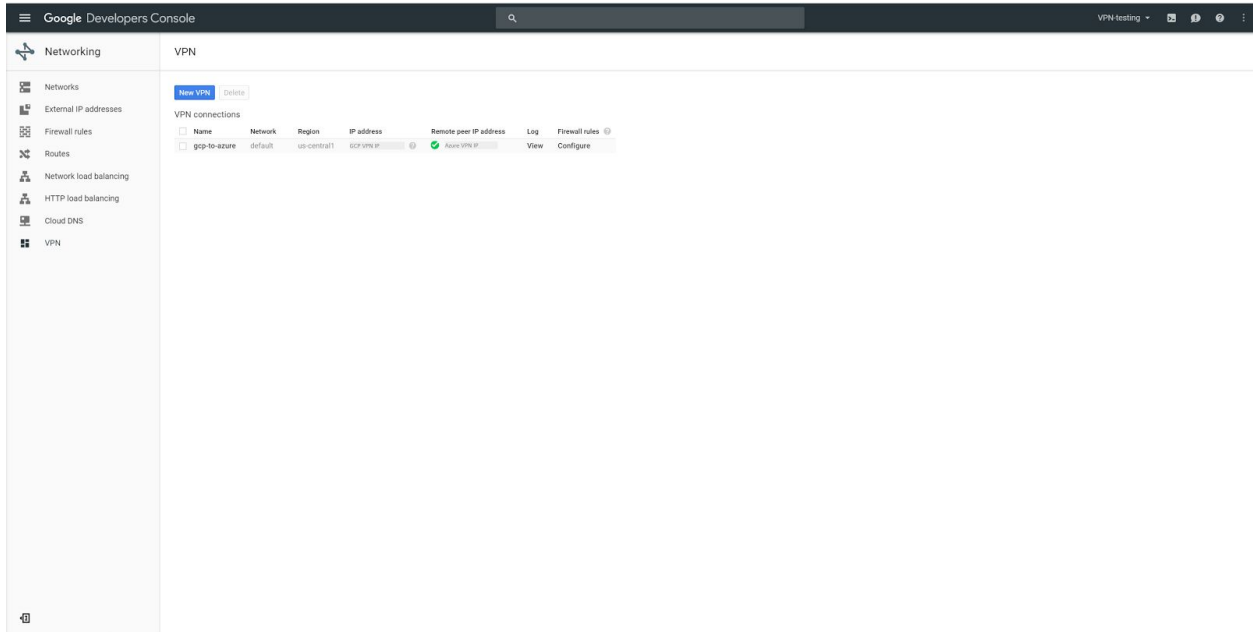
- **Name:** a representative name for the VPN connection (must be lowercase). In this case we have assigned “gcp-to-azure”
- **Description:** free form text for console administrators. In this case we described the source and destination environments of the IPsec connection.
- **Network:** the network to which the VPN gateway will be attached. In this example we have selected “default”
- **Region:** the region into which the VPN gateway will be deployed. We have selected “us-central-1”
- **IP address:** the static public IP address which will be assigned to the VPN gateway. A new static IP address can also be allocated at this stage. For this case we will go ahead and select “New Static IP address...” which will trigger the new IP address workflow:

## Finalizing the Google Cloud Platform Configuration

At this stage we can return to GCP to complete the configuration. Enter the Azure VPN Gateway IP and the pre-shared key collected in the previous step and click “create”. Note that Azure requires **IKEv2**:

The screenshot shows the Google Developers Console interface for configuring a new VPN connection. The left sidebar lists various networking services, with 'VPN' selected. The main content area is titled 'Create a new VPN connection' and contains several input fields and dropdown menus. The 'Name' field is filled with 'gcp-to-azure'. The 'Description' field has a placeholder text. The 'Network' dropdown is set to 'default', 'Region' is 'us-central1', and 'IP address' is 'gcp-to-azure (104.197.152.158)'. Under the 'Tunnels' section, the 'Remote peer IP address' field contains 'enter public IP' and has a red error message: 'Remote peer IP address is invalid'. The 'IKE version' dropdown is set to 'IKEv2'. The 'Shared secret' field contains 'enter shared secret here'. The 'Remote network IP ranges' field contains '10.241.0.0/16'. At the bottom, there are 'Create' and 'Cancel' buttons, and a link for 'Equivalent REST or command line'.

Once complete the VPN will attempt to connect. To check the VPN status monitor the developer console. If the VPN successfully connects a green check will mark the remote peer IP. Note that by default, new GCP Projects are deployed with default firewall rules in place allowing SSH, RDP and ICMP traffic from **any source**. If you have specific traffic requirements, a firewall rule will need to be created allowing the inbound traffic from the Azure source network on the required ports:



To verify connectivity on the Azure side, return to the Azure Virtual Network dashboard. The topology map will update to reflect connection status:

Microsoft Azure | CREDIT STATUS | Subscriptions

gcp-test

DASHBOARD | CONFIGURE | CERTIFICATES

virtual network

GCP-Test | GATEWAY | GCP

VPN

DATA IN: 0B | DATA OUT: 3.98KB | GATEWAY IP ADDRESS: AZURE VPN GATEWAY PUBLIC IP

NAME	ROLE	IP ADDRESS	SUBNET NAME
resources			

quick glance

- Download VPN Device Script
- STATUS: Created
- SUBSCRIPTION ID: ccb367f9-948a-4d8d-b9de-6f0913c5c208
- VIRTUAL NETWORK ID: 216d3735-ceca-49b2-8d2a-a32d7f7f8970
- LOCATION: South Central US
- GATEWAY TYPE: Dynamic Routing

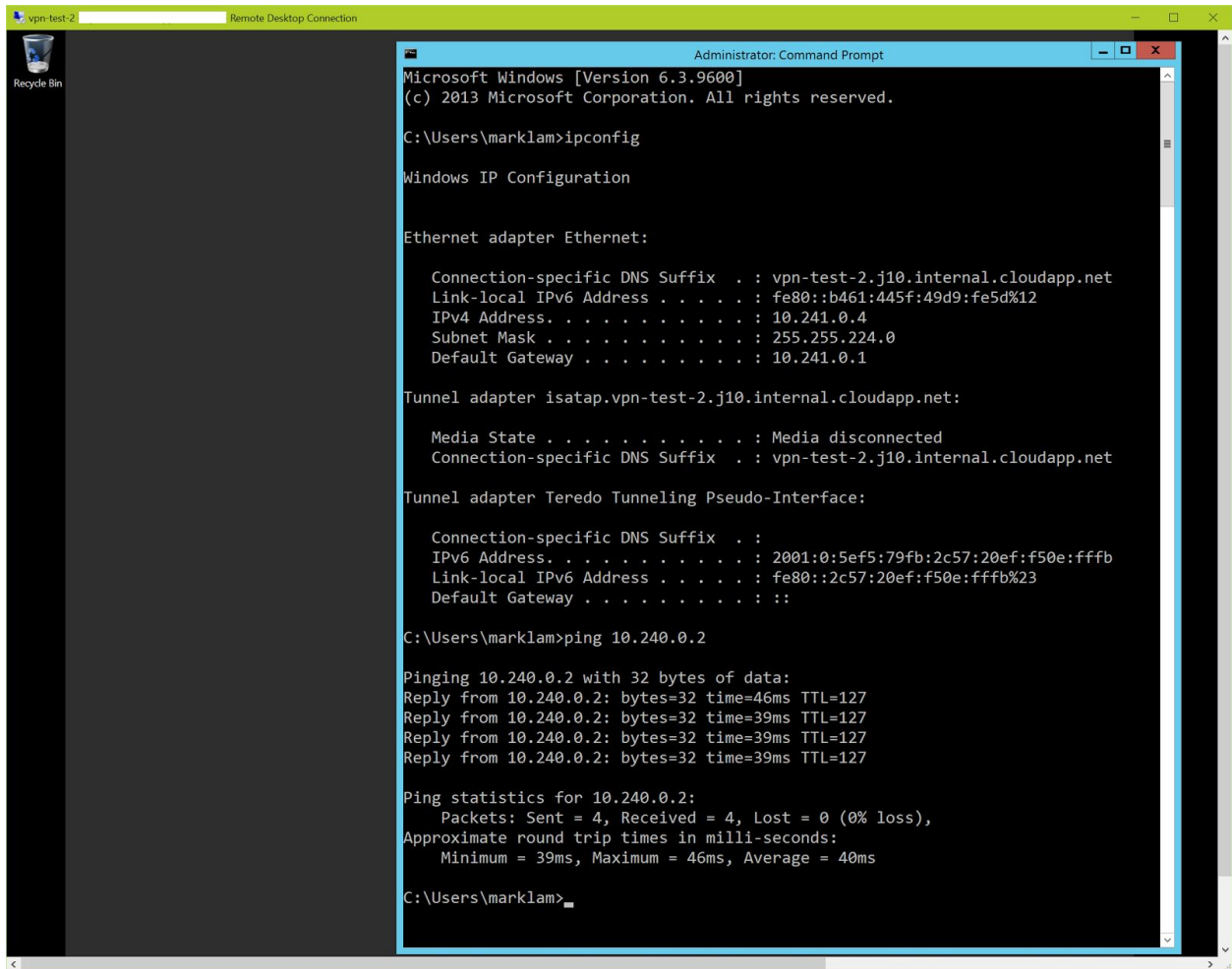
+ NEW | DELETE GATEWAY | DISCONNECT | EXPORT | MANAGE KEY | DELETE

## Testing the Site-to-Site VPN

With the site-to-site VPN online the tunnel is now ready for testing. To test, create virtual machines in both Azure and Google Compute Engine. Instructions for creating Azure virtual machines can be found [here](#). To learn how to create virtual machines in Google Compute Engine, visit the [Getting Started Guide](#).

Once virtual machines have been deployed on both platforms an ICMP echo test can ensure network connectivity. Note that on Azure the default firewall setting is **off** for ICMP and will have to be enabled for this test to work. A demonstration of a functional tunnel is below.

Azure virtual machine pinging the virtual machine in GCE:



```
Administrator: Command Prompt
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\marklam>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix  . : vpn-test-2.j10.internal.cloudapp.net
    Link-local IPv6 Address . . . . . : fe80::b461:445f:49d9:fe5d%12
    IPv4 Address. . . . . : 10.241.0.4
    Subnet Mask . . . . . : 255.255.224.0
    Default Gateway . . . . . : 10.241.0.1

Tunnel adapter isatap.vpn-test-2.j10.internal.cloudapp.net:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : vpn-test-2.j10.internal.cloudapp.net

Tunnel adapter Teredo Tunneling Pseudo-Interface:

    Connection-specific DNS Suffix  . :
    IPv6 Address. . . . . : 2001:0:5ef5:79fb:2c57:20ef:f50e:ffff
    Link-local IPv6 Address . . . . . : fe80::2c57:20ef:f50e:ffff%23
    Default Gateway . . . . . : ::

C:\Users\marklam>ping 10.240.0.2

Pinging 10.240.0.2 with 32 bytes of data:
Reply from 10.240.0.2: bytes=32 time=46ms TTL=127
Reply from 10.240.0.2: bytes=32 time=39ms TTL=127
Reply from 10.240.0.2: bytes=32 time=39ms TTL=127
Reply from 10.240.0.2: bytes=32 time=39ms TTL=127

Ping statistics for 10.240.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 39ms, Maximum = 46ms, Average = 40ms

C:\Users\marklam>
```

GCE virtual machine pinging the virtual machine in Azure:

