Microsoft Teams Deployment Guide

AudioCodes SmartTAP™ 360° for Microsoft Teams

SmartTAP 360° for Microsoft Teams

SmartTAP 360° Enterprise Recording Solution

Version 5.2

Notice

Information contained in this document is believed to be accurate and reliable at the time of printing. However, due to ongoing product improvements and revisions, AudioCodes cannot guarantee accuracy of printed material after the Date Published nor can it accept responsibility for errors or omissions. Updates to this document can be downloaded from https://www.audiocodes.com/library/technical-documents.

This document is subject to change without notice.

Date Published: October-20-2020

WEEE EU Directive

Pursuant to the WEEE EU Directive, electronic and electrical waste must not be disposed of with unsorted waste. Please contact your local recycling authority for disposal of this product.

Customer Support

Customer technical support and services are provided by AudioCodes or by an authorized AudioCodes Service Partner. For more information on how to buy technical support for AudioCodes products and for contact information, please visit our website at https://www.audiocodes.com/services-support/maintenance-and-support.

Documentation Feedback

AudioCodes continually strives to produce high quality documentation. If you have any comments (suggestions or errors) regarding this document, please fill out the Documentation Feedback form on our website at https://online.audiocodes.com/documentation-feedback.

Stay in the Loop with AudioCodes

Related Documentation

Document Name
SmartTAP 360° for Microsoft Teams Release Notes
SmartTAP 360° for Microsoft Teams Administrator Guide
SmartTAP 360° for Microsoft Teams Installation Guide

Document Revision Record

LTRT	Description
27325	Initial document release.
27326	Sections Updated: SmartTAP 360° for Microsoft Teams Specifications; Step 1 Create Service Fabric Cluster; Step 2-1 Configure Service Channel; Step 3-1 Prepare Local Machine for Deployment on Service Fabric; Step 3-2 Deploy BOT Package
27327	Sections Updated: Overview; SmartTAP 360° for Microsoft Teams Specifications; Step 2 Create Service Fabric Cluster; Grant API Permissions to BOT Service Sections Added: Purpose; Deploy SmartTAP 360° On-premises Hybrid model; Create Service Bus Sections Removed: Set Azure Active Directory Read Permissions; Create Application Instance; Configure Microsoft Blob Storage

Table of Contents

1 Purpose

This document describes the default deployment of the simplest form of SmartTAP 360° for Teams in an Azure subscription. The deployment instructions assumes the creation of Azure resources described in SmartTAP 360° for Microsoft Teams Specifications on page 5. It uses a public IP address for the communication between the BOT hosted on the VMSS to Microsoft Graph API.

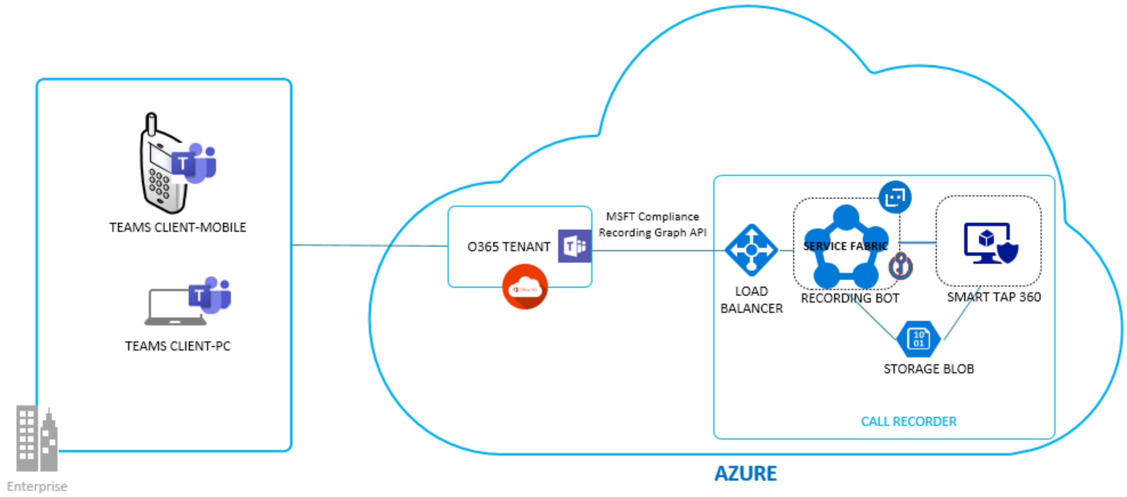
2 Overview

SmartTAP 360° for Teams is sold as part of AudioCodes Live offering with Managed Services Pump model. SmartTAP for Teams Compliance Recording is part of AudioCodes Live offering with the OPEX model including per user per month managed service payments. This offering can either be hosted fully in either AudioCodes Azure subscription or in the customer's Azure subscription. When the solution is hosted in the customer's subscription, SmartTAP Server with media SMB storage can alternatively be deployed On-Premises as a Hybrid model. This topology may also reflect organization policies to deploy all AudioCodes products on Microsoft Azure. This solution includes the following components:

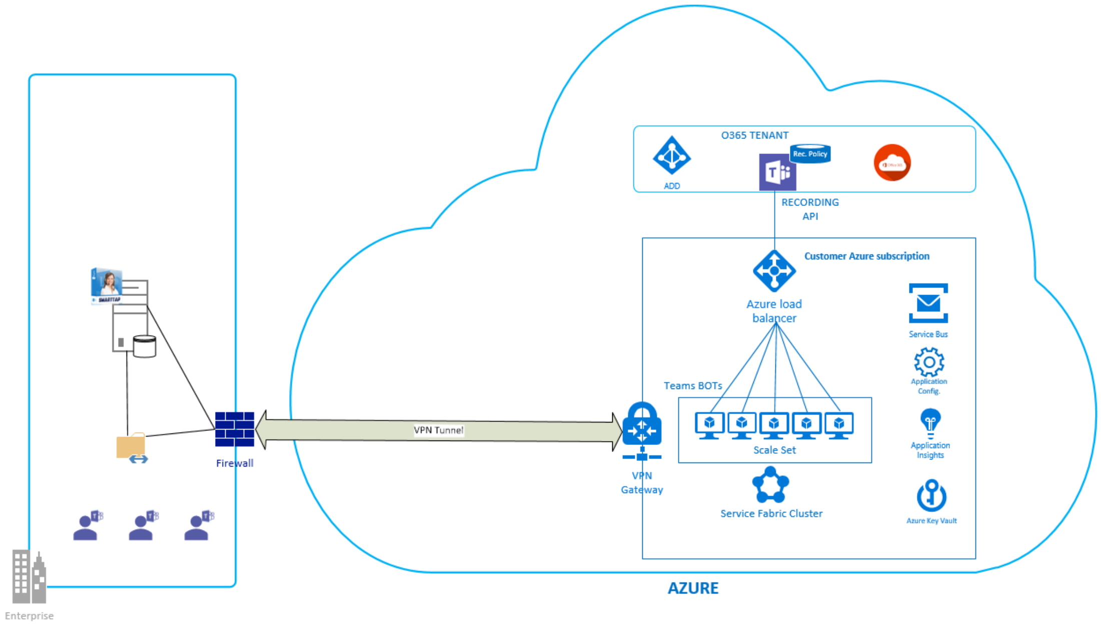
- **Microsoft Teams Compliance Recording BOT:** A component consisting of one or several VMs working that are working together in a Azure Service Fabric Cluster which manages and balances between the VMs . The BOT connects to the customer's Teams subscription and enables recording of Teams communications by receiving the call data and media and uploading it to the SmartTAP 360° recording server.
- **Audiocodes SmartTAP 360° Recording server:** Consists of one or more servers (VMs) recording calls' metadata and media. In its simplest form, one server is required, hosting all SmartTAP 360° components. Storage consists of OS disk and Logs/DB data disks.
- **Microsoft Blob Storage:** Stores recorded media holding the recorded calls media (voice/video) are configured on Microsoft Azure Blob.

The figures below illustrates the different deployment topology offerings.

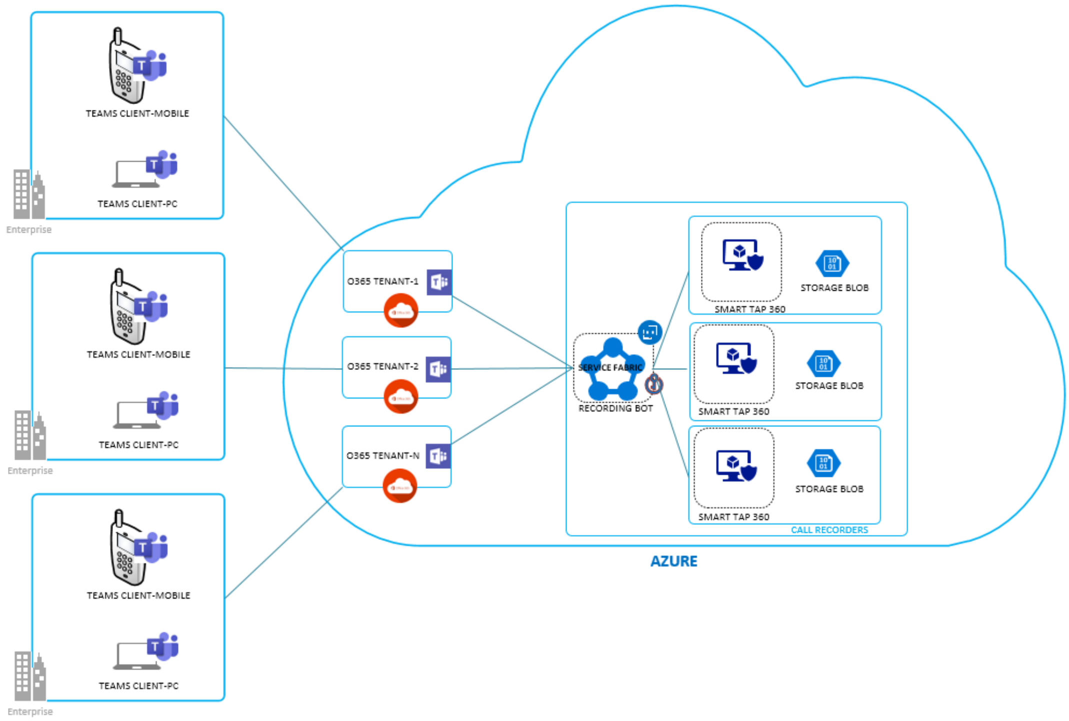
- **Customer Subscription Model:**
  - All entities are deployed on Microsoft Azure:



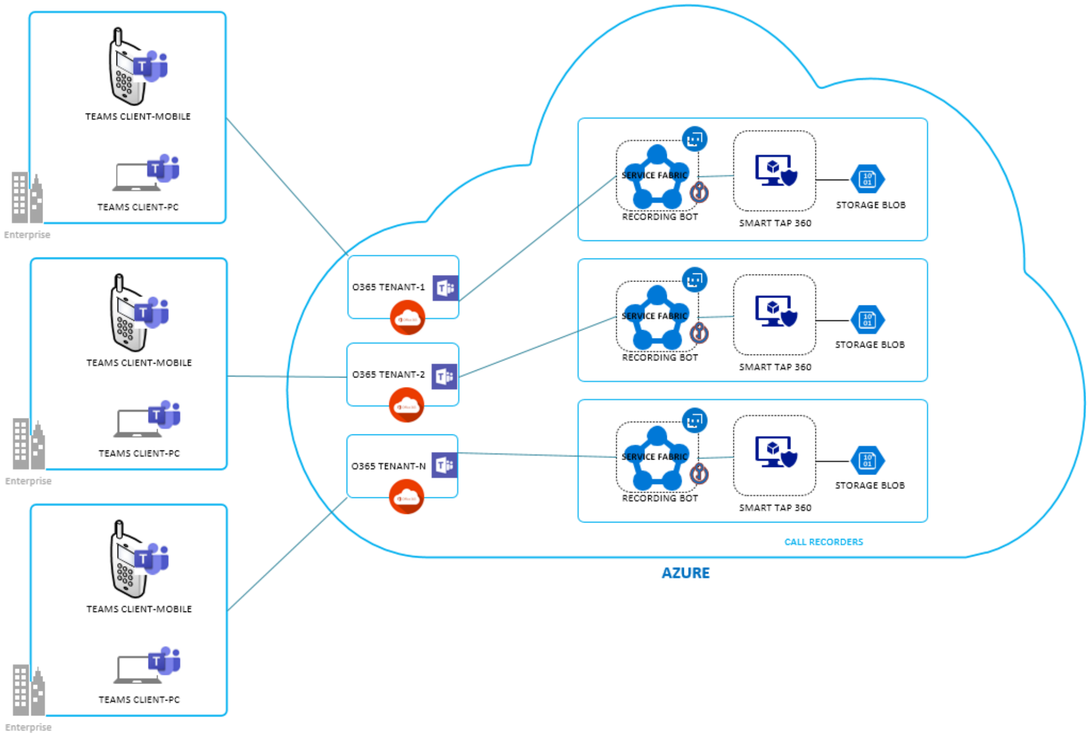
  - **Hybrid Model:** Microsoft Teams BOT is deployed in the customer's Azure subscription and SmartTAP server and media SMB storage are deployed On-premises



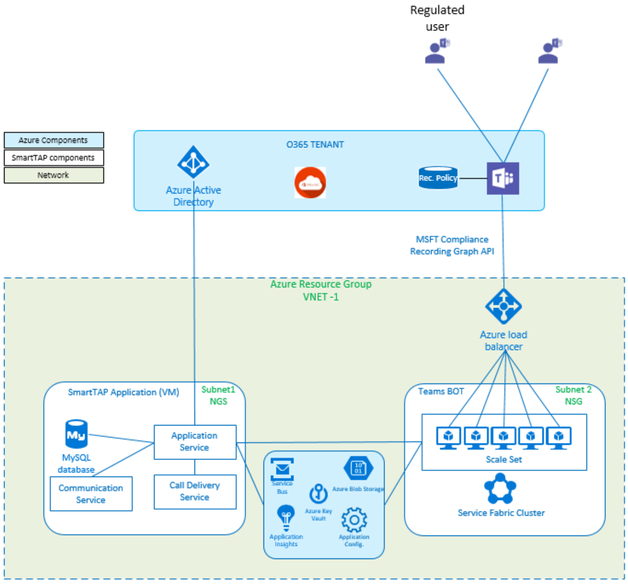
- Deployment in the AudioCodes subscription **for up to 500 users:**



- Deployment in the AudioCodes subscription for **more than 500 users:**



SmartTAP Components and scenarios:



Scenarios:

Recording flow
1. Call started
2. MSFT checks recording policy
3. MSFT auto-invite recording BOT
4. BOT joins the call
5. MSFT presents the recording notification to participants (audio to PSTN parties):
⚠ **Recording started.** Policy applied to one or more participants requires this call to be recorded.
6. BOT records the call media locally
7. Call ended
8. BOT transfers the media to the Blob Storage/SMB on premise

Playback flow
1. User select a call recording and clicks play in UI
2. Application Service downloads media files from Blob storage
3. Application Service plays the recording to the user

3 SmartTAP 360° for Microsoft Teams Specifications

This section describes the recommended specifications for the Microsoft Teams BOT Cluster and the SmartTAP 360° Recording solution.

- **SmartTAP 360° Server:**
  - Operating System: Microsoft Windows Server 2016 or Microsoft Windows Server 2019
  - SmartTAP 360° server with the specifications below can handle up to 3000 targeted users and 500 audio call recordings:
    - Virtual Machine: Tier=Standard, Instance=DS2 v2 (2 vCPUs, 7 GB RAM, 14 GB Temporary storage)
  - SmartTAP 360° server with the specifications below can handle up to a 3000 targeted users and a combination of 500 audio/Desktop Application Sharing (DAS) call recordings.
    - Virtual Machine: Tier=Standard, Instance=F8s v2 (8 vCPUs, 16 GB RAM, 64 GB Temporary storage)
  - An additional managed disk is required for database storage. The estimated size of the required disk can be calculated using the SmartTAP storage calculator. The additional managed disk is not required for POC if the SmartTAP Server's OS disk has sufficient space to hold the database. The disk should be a premium SSD managed disk.

> ⚠ When the SmartTAP 360° server is deployed On-premises in the "Hybrid" model, refer to "Server Configurations" in the *SmartTAP 360° Installation Guide.*

- **Microsoft Teams BOT Cluster:**
  - Service Fabric Cluster with Silver Durability with a minimum of 5 nodes (for testing or POCs, Bronze Durability with 1 or 3 nodes can be used). For more information, refer to Microsoft Service Fabric Cluster.
  - Single BOT node with the specifications below can handle up to 40 concurrent DAS calls or up to 50 concurrent audio calls. For example, the recording of 150 DAS and 150 audio calls requires 7 nodes:
    - Virtual Machine: Tier=Standard, Instance=DS2 V2 (2 vCPUs, 7 GB RAM, 100 GB Temporary storage)
    - Windows Server 2019 Datacenter - with Containers
  - Additional mandatory Azure resources:
    - Load Balancer for BOT Service Fabric Cluster
    - Public IP address for the Load Balancer
    - Virtual Machine ScaleSet – VMs for BOT Service Fabric Cluster
    - Key Vault to store BOT Service Fabric Cluster certificates

  - Microsoft Azure Blob Storage
- Optional Azure resources:
  - Application Insights to store BOT logs
  - App Configuration to store BOT configuration

- **SmartTAP 360° for Microsoft Teams availability:** SmartTAP 360° for Microsoft Teams availability is based on Azure Virtual Machines (VM) Service Level Agreement (SLA):
  - SmartTAP 360° Server on Azure VM - SLA is 99.9% for one instance and 99.99% can be achieved by deploying the two servers in different Availability Zones (optionally available at extra cost). Refer to Azure VM SLA.
  - SmartTAP 360° Teams BOT on Azure VM - SLA 99.9%. Refer to Azure VM SLA.
  - SmartTAP 360° Media on Azure BLOB – SLA is 99.9% for Hot tier, and 99% for Cool Tier. Refer to Azure Blob Storage SLA.
  - The durability of Azure BLOB using Locally Redundant Storage (LRS) is 11 nines. Refer to Azure Blob Storage Durability.
- **SmartTAP 360° for Microsoft Teams Backup/Restore:** Azure Virtual Machines (VM) backup/restore procedures are highly recommended.

- For integrations with third-party applications, a custom specification is required.
- DAS call recordings are limited to up to two concurrent DAS recording playbacks or downloads.

4 Prerequisites

The following describes the prerequisite actions to perform for generating certificates on Microsoft Azure:

- Generate certificate before configuring the installation FQDN for SmartTAP 360° Server
- Generate certificate before configuring the installation FQDN for Teams BOTs
- Create a certificate(s) for the services above and have it signed (wildcards are supported)
- Create an Azure key vault and upload the certificate to be used to the vault. This certificate is used for the following purposes:
  - For service fabric cluster
  - For BOT package deployment
  - For SmartTAP 360° server HTTPS connection

> ⚠ For information on generating Azure key vaults, refer to: https://docs.microsoft.com/en-us/azure/key-vault/

- Copy the certificate thumbprint Secret Identifier to a text file as it is later required for configuration.

5 Deployment Procedures Overview

The deployment includes the following procedures:

6 Step 1 Deploying SmartTAP 360° Server

SmartTAP 360° Server can be deployed using one of the following storage topologies:

- Option 1 Deploy SmartTAP 360° Server on Azure with Blob Storage below
- Option 2 Deploy SmartTAP 360° Server On-premises (Hybrid Model) on page 16

Option 1 Deploy SmartTAP 360° Server on Azure with Blob Storage

This procedure describes how to deploy SmartTAP 360° Server on Azure and how to configure Blob storage. Do the following:

1. Create SmartTAP 360° Virtual Machine below
2. Configure Microsoft Blob Storage on page 15

Once you have completed the above, refer to the *SmartTAP Administrators Guide* to perform the following:

- Map Azure Active Directory Users (see 'Azure Active Directory User Mapping')
- Setup Azure Active Directory (see 'Azure Active Directory User Authentication')

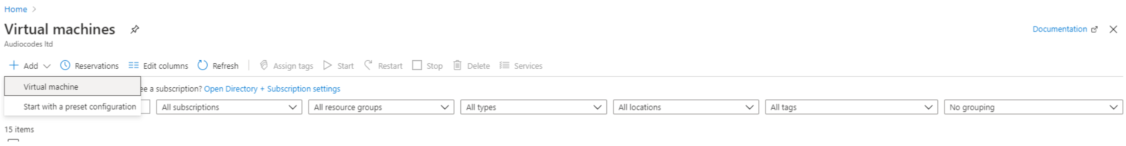
Create SmartTAP 360° Virtual Machine

This section describes how to create the new VM from the Azure Portal in the customer or AUDC subscription and install SmartTAP suite on the newly deployed VM.

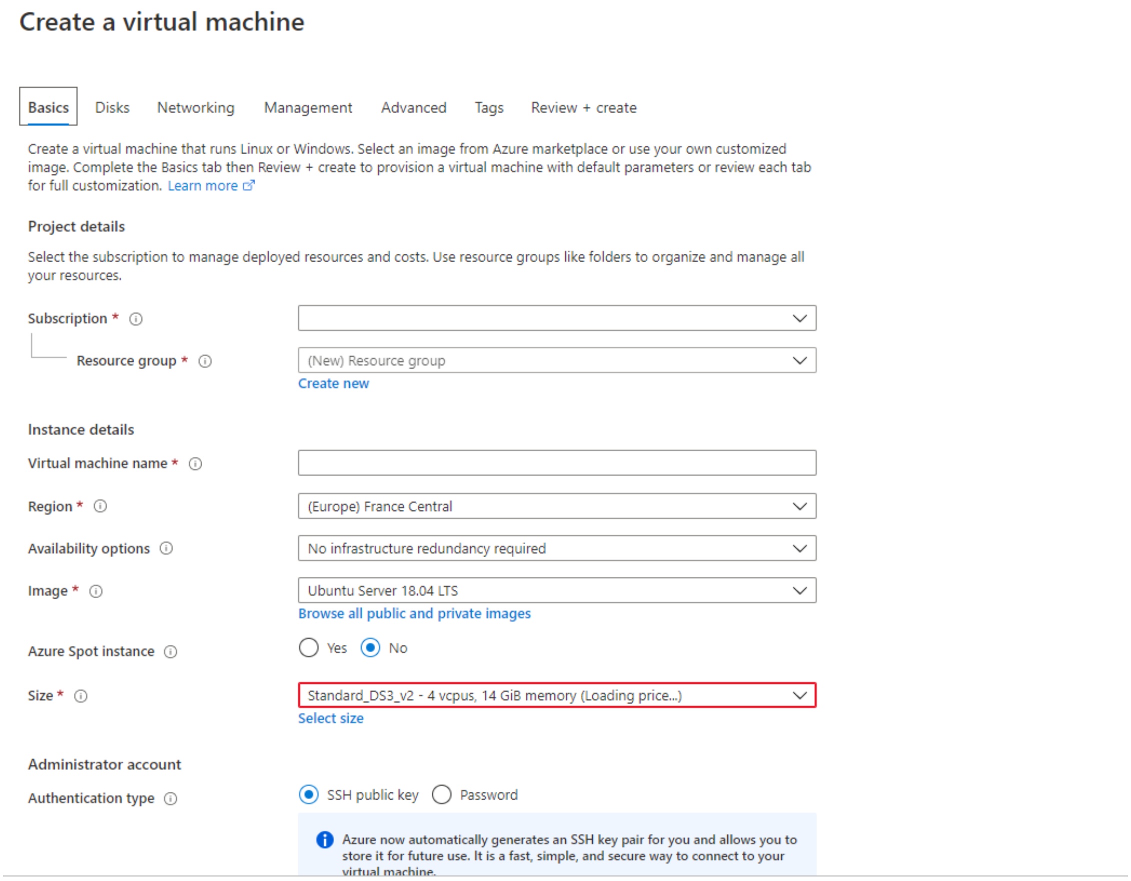
➢ **Do the following:**

1. Log on to the Azure portal and go to your subscription directory .

Figure 6-1: Create Virtual Machine



2. Click **Virtual machine** to create a virtual machine.



Create a virtual machine

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. Learn more

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

Resource group * (New) Resource group

Create new

Instance details

Virtual machine name *

Region * (Europe) France Central

Availability options No infrastructure redundancy required

Image * Ubuntu Server 18.04 LTS

Browse all public and private images

Azure Spot instance ○ Yes ● No

Size * Standard_DS3_v2 - 4 vcpus, 14 GiB memory (Loading price...)

Select size

Administrator account

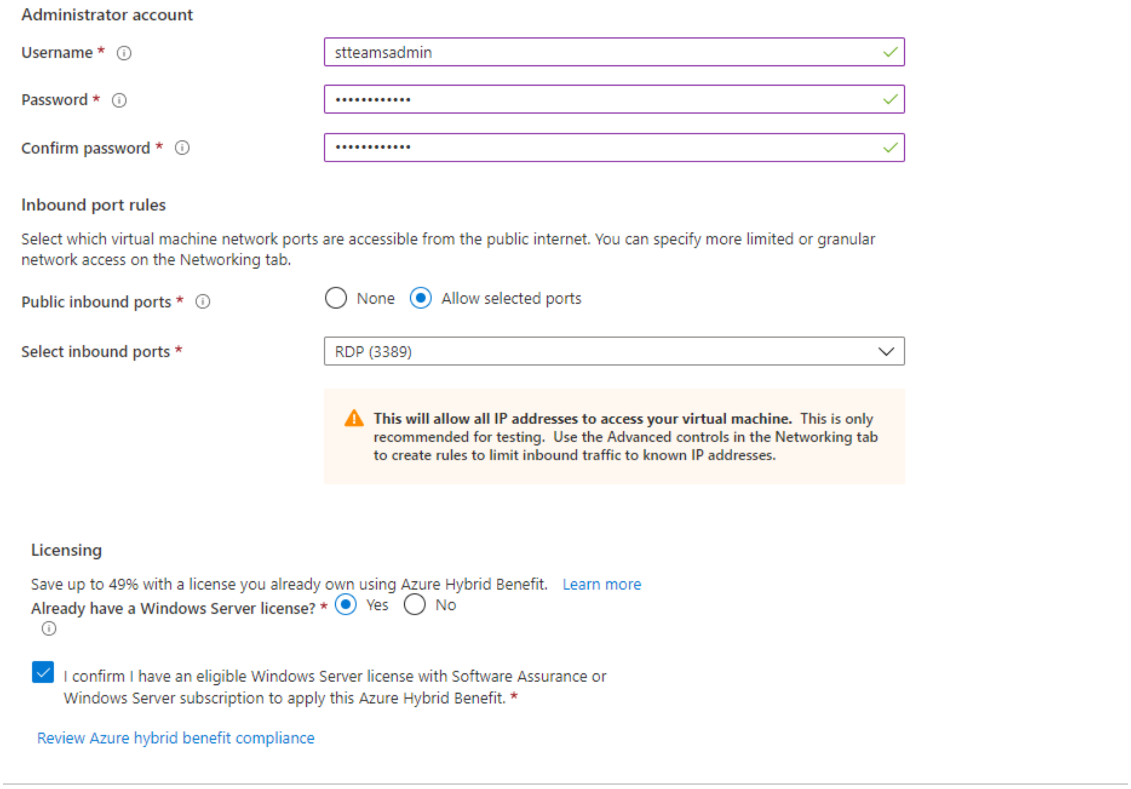
Authentication type ● SSH public key ○ Password

Azure now automatically generates an SSH key pair for you and allows you to store it for future use. It is a fast, simple, and secure way to connect to your virtual machine.

3. Fill in the relevant customer information : subscription , Resource group, region, virtual machine name, user and password.

4. Select the relevant Virtual Machine specifications according to SmartTAP 360° for Microsoft Teams Specifications on page 5 and then click **Next**.

Figure 6-2: Administrator Account



Administrator account

Username * ⓘ stteamsadmin ✓

Password * ⓘ •••••••••••• ✓

Confirm password * ⓘ •••••••••••• ✓

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports * ⓘ ○ None ● Allow selected ports

Select inbound ports * RDP (3389)

⚠ **This will allow all IP addresses to access your virtual machine.** This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

Licensing

Save up to 49% with a license you already own using Azure Hybrid Benefit. Learn more

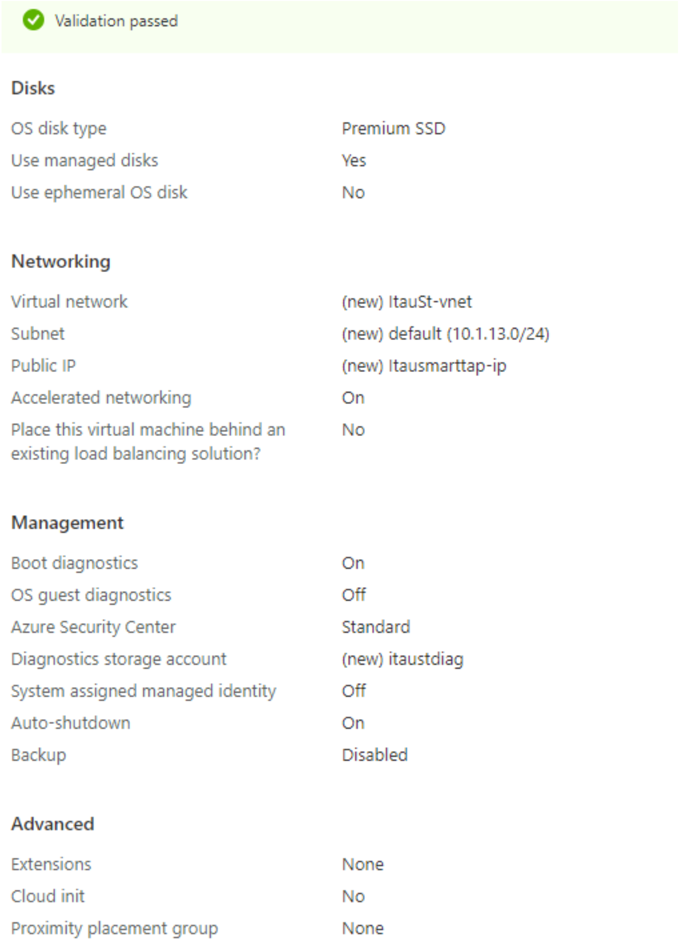
Already have a Windows Server license? * ⓘ ● Yes ○ No

☑ I confirm I have an eligible Windows Server license with Software Assurance or Windows Server subscription to apply this Azure Hybrid Benefit. *

Review Azure hybrid benefit compliance

5. Review the details and then click **Review and Create**.

Figure 6-3: Review and Create



✓ Validation passed

Disks

OS disk type	Premium SSD
Use managed disks	Yes
Use ephemeral OS disk	No

Networking

Virtual network	(new) ItauSt-vnet
Subnet	(new) default (10.1.13.0/24)
Public IP	(new) Itausmarttap-ip
Accelerated networking	On
Place this virtual machine behind an existing load balancing solution?	No

Management

Boot diagnostics	On
OS guest diagnostics	Off
Azure Security Center	Standard
Diagnostics storage account	(new) itaustdiag
System assigned managed identity	Off
Auto-shutdown	On
Backup	Disabled

Advanced

Extensions	None
Cloud init	No
Proximity placement group	None

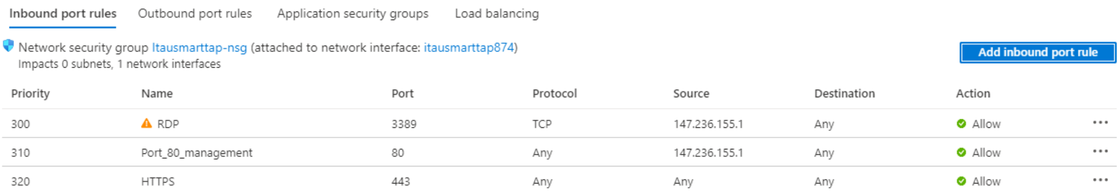
6. Install SmartTap server from Installation Suite All-In-One mode on the VM that you just created. Refer to the *SmartTAP 360° Installation Guide* for details.

7. Run firewall rules script to enable the relevant ports for traffic (part of Installation Kit). This script is located in the Installation Suite at the following location:

   ..\tools\Users\stteamsadmin\Downloads\SmartTAP_<SmartTApVersion>\SmartTAP\Tools\EnableFWRules

8. Configure Azure Network Security Group Inbound rules for port 80, 443 and block RDP port to only allow access to listed IPs.

Figure 6-4: Inbound Firewall Rules



Network security group Itausmarttap-nsg (attached to network interface: itausmarttap874)
Impacts 0 subnets, 1 network interfaces

Priority	Name	Port	Protocol	Source	Destination	Action
300	⚠ RDP	3389	TCP	147.236.155.1	Any	✓ Allow
310	Port_80_management	80	Any	147.236.155.1	Any	✓ Allow
320	HTTPS	443	Any	Any	Any	✓ Allow

Table 6-1: Firewall Rules

Protocol	Ports	Connection	Port Flow	Description
TCP	80/443	BOT VMs ⇔ SmartTAP server	Bi-directional	Used for Management/Signaling between BOT and SmartTAP (On SmartTAP Azure NSG).
TCP	80/9441	BOT VMs ⇔ BOT VMs	Bi-directional	Used by Load Balancer Address Pool (part of SFC deployment script).
TCP	19080/19081/19000	BOT VMs ⇔ BOT VMs	Bi-directional	Used for Load Balancer HTTP Fabric Gateway Probe (part of SFC deployment script).
TCP	443	BOT VMs ⇒Teams	Send-only	Used for signaling from BOT VMs to Teams.
TCP	9444-9544	BOT VMs ⇐ Teams	Receive-only	Used for signaling from Teams to BOT VMs (part of SFC deployment script).
TCP	8445-8545	BOT VMs ⇐ Teams	Receive-only	Used for media TCP traffic from Teams to BOT VMs (part of SFC deployment script)
HTTP/S	8861	OVOC Main Agent⇐ OVOC client agents	Receive-only	Used for managing status events sent from OVOC client agents (SmartTAP VMs) that run SmartTAP components (e.g. BOT, RDD) to OVOC Main Agent.
HTTP	8862	Web Admin ⇔ OVOC Main Agent	Bi-directional	Used for Rest API communication between SmartTAP Web Admin interface and OVOC Main Agent for

Protocol	Ports	Connection	Port Flow	Description
				alarms and status updates
HTTP/S	8863	OVOC Main Agent ⇒ SmartTap client agents	Send-only	Used for managing requests from SmartTAP AS (Main OVOC Agent) to SmartTap Virtual Machines client agents (they also run on SmartTAP AS).
UDP	3478-3481	Teams ⇔ BOT VMs	Bi-directional	Used for media relay (part of SFC deployment script).
TCP	3389-33xxxx	BOT VMs ⇐ Teams	Receive-only	(Optional) Used for RDP traffic sent from Teams to BOT VMs (part of SFC deployment script).
UDP	161	SmartTAP ⇔ OVOC	Bi-directional	Used for SNMP traffic for managing traps/alarms between AS and OVOC
UDP	162	SmartTAP ⇒ OVOC	Send-only	Used for SNMP traffic for sending Keep-alive messages from AS to OVOC
UDP	1161	SmartTAP ⇒ OVOC	Send-only	Used for SNMP traffic for sending Keep-alive messages from AS to OVOC (this port is predominantly used when AS is installed behind a NAT)

9. It is also recommended to assign compliance recording policies to all targeted users. Instead of assigning each user separately, you can alternatively assign the recording policy to a Security Group and then add all targeted users to this group. Refer to the following links:

- https://docs.microsoft.com/en-us/microsoftteams/assign-policies#assign-a-policy-to-a-group
- https://docs.microsoft.com/en-us/powershell/module/teams/new-csgrouppolicyassignment?view=teams-ps#description

Configure Microsoft Blob Storage

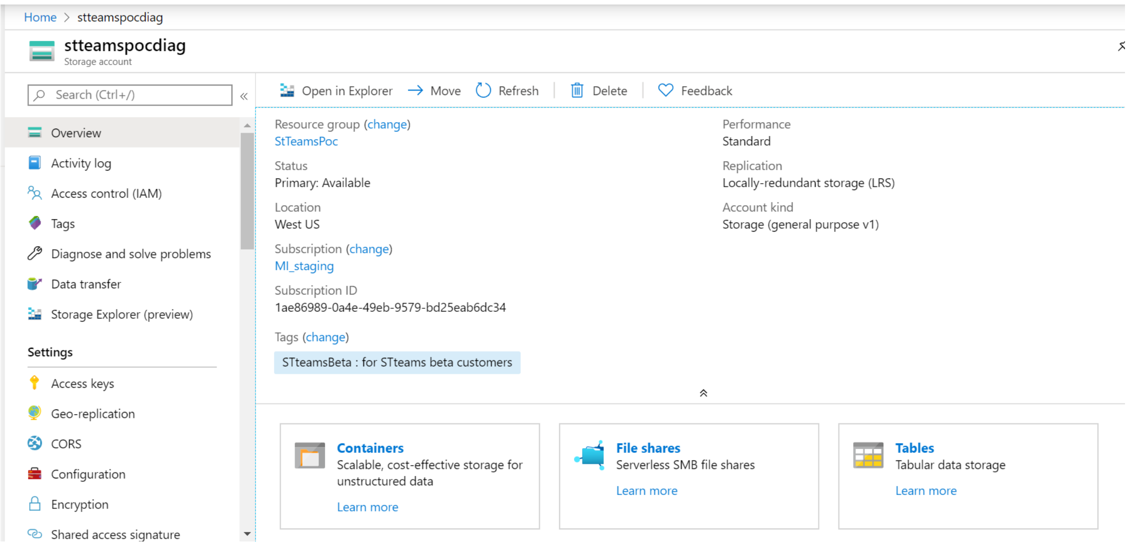
This section describes how to configure Microsoft Blob Storage as the external storage platform for storing recorded media. You should create or user an existing storage account. The created storage should be of type general purpose **v2** and **cool access tier**.

> ⚠ When the Microsoft Teams deployment is hosted in the customer subscription, SmartTAP Server and Media Server Message Block (SMB) storage can alternatively be deployed On-Premises (described in Configuring Media). You cannot configure both On-Premises and Blob Storage simultaneously.

➢ To configure Microsoft Blob:

1. Log on to the Azure portal and open the Storage account settings page.
2. Create or use existing storage account.

Figure 6-5: Microsoft Blob Storage Account



Resource group (change)
StTeamsPoc

Status
Primary: Available

Location
West US

Subscription (change)
MI_staging

Subscription ID
1ae86989-0a4e-49eb-9579-bd25eab6dc34

Tags (change)
STteamsBeta : for STteams beta customers

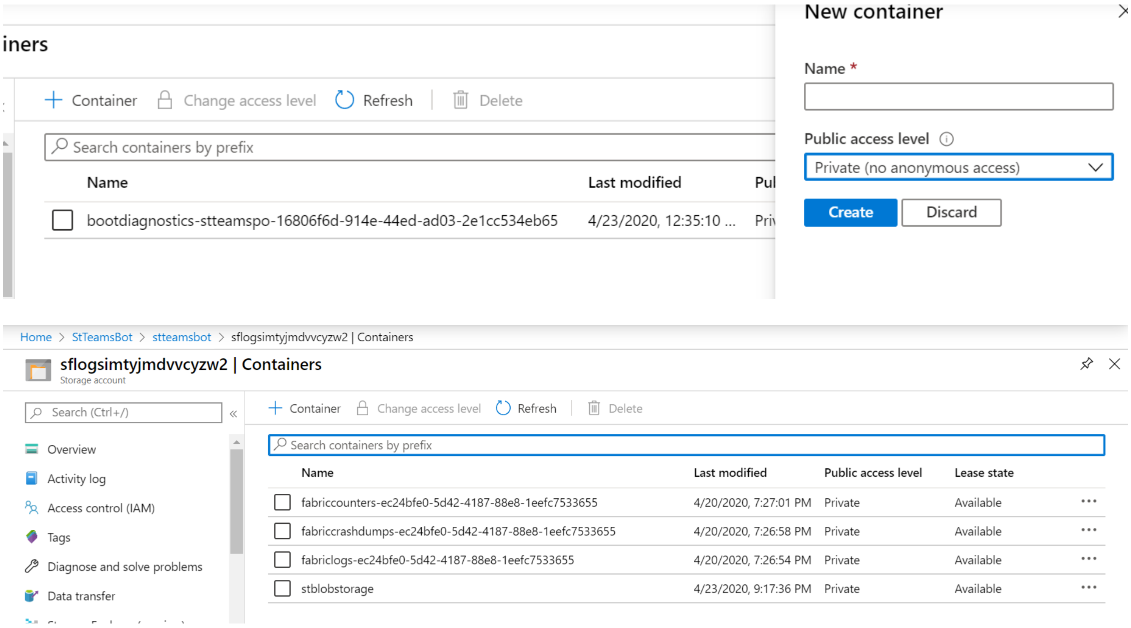
Performance
Standard

Replication
Locally-redundant storage (LRS)

Account kind
Storage (general purpose v1)

3. Save the storage name for SmartTAP 360° settings.
4. Create a new container for BLOB media storage and save the name.

Figure 6-6: Create New Blob Container



Name	Last modified	Pul
□ bootdiagnostics-stteamspo-16806f6d-914e-44ed-ad03-2e1cc534eb65	4/23/2020, 12:35:10 ...	Pri

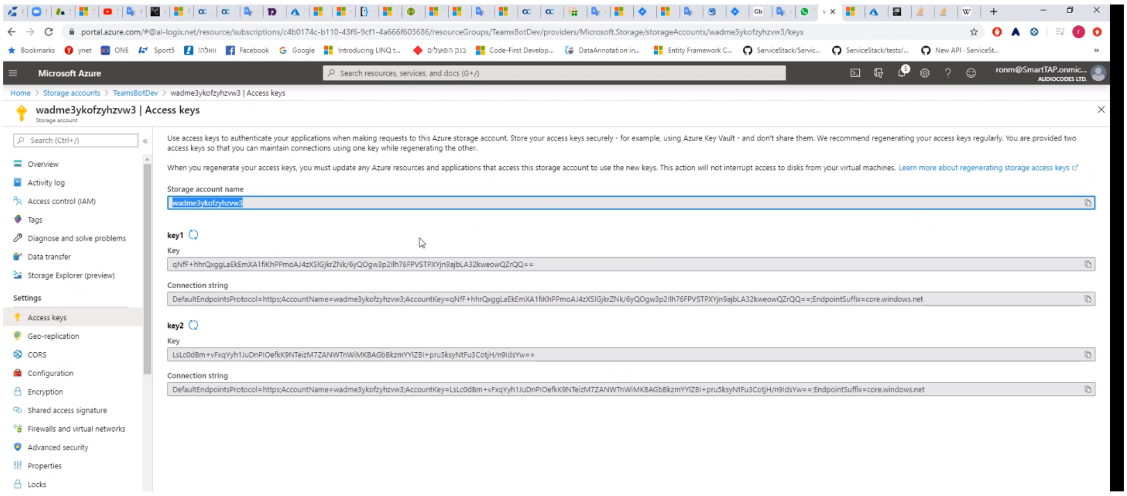
Name *

Public access level: Private (no anonymous access)

Name	Last modified	Public access level	Lease state
□ fabriccounters-ec24bfe0-5d42-4187-88e8-1eefc7533655	4/20/2020, 7:27:01 PM	Private	Available
□ fabriccrashdumps-ec24bfe0-5d42-4187-88e8-1eefc7533655	4/20/2020, 7:26:58 PM	Private	Available
□ fabriclogs-ec24bfe0-5d42-4187-88e8-1eefc7533655	4/20/2020, 7:26:54 PM	Private	Available
□ stblobstorage	4/23/2020, 9:17:36 PM	Private	Available

5. Save the storage name and credentials.

Figure 6-7: Storage Name and Credentials



Use access keys to authenticate your applications when making requests to this Azure storage account. Store your access keys securely - for example, using Azure Key Vault - and don't share them. We recommend regenerating your access keys regularly. You are provided two access keys so that you can maintain connections using one key while regenerating the other.

When you regenerate your access keys, you must update any Azure resources and applications that access this storage account to use the new keys. This action will not interrupt access to disks from your virtual machines. Learn more about regenerating storage access keys

Storage account name: wadme3ykofzyhzvw3

key1

Key: qNfF+hhrQxggLaEkEmXA1fiKhPPmoAJ4zXSlGjkrZNk/6yQOgw3p2Ilh76FPVSTPXYjn9ajbLA32kweowQZrQQ==

Connection string: DefaultEndpointsProtocol=https;AccountName=wadme3ykofzyhzvw3;AccountKey=qNfF+hhrQxggLaEkEmXA1fiKhPPmoAJ4zXSlGjkrZNk/6yQOgw3p2Ilh76FPVSTPXYjn9ajbLA32kweowQZrQQ==;EndpointSuffix=core.windows.net

key2

Key: LsLc0dBm+vFxqYyh1JuDnPIOefkK9NTeizM7ZANWTnWiMKBAGbBkzmYYlZBI+pru5ksyNtFu3CotjH/n9IdsYw==

Connection string: DefaultEndpointsProtocol=https;AccountName=wadme3ykofzyhzvw3;AccountKey=LsLc0dBm+vFxqYyh1JuDnPIOefkK9NTeizM7ZANWTnWiMKBAGbBkzmYYlZBI+pru5ksyNtFu3CotjH/n9IdsYw==;EndpointSuffix=core.windows.net

6. Define Blob Storage account credentials (refer to the *SmartTAP 360° Administrator Guide*).

7. Add Recording Location in SmartTAP Web interface (refer to the *SmartTAP 360° Administrator Guide*).

Option 2 Deploy SmartTAP 360° Server On-premises (Hybrid Model)

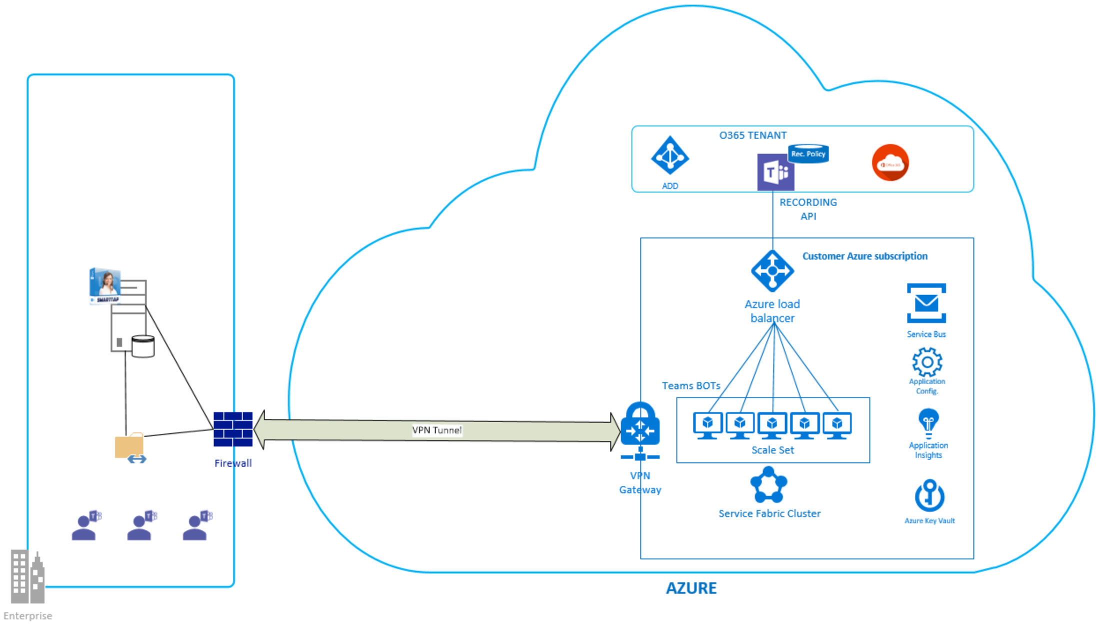
SmartTAP 360° for Microsoft Teams with AudioCodes Live can be deployed in a Hybrid model where SmartTAP Teams BOT Service Fabric Cluster is deployed in the customer Azure subscription and the SmartTAP Server On-premises, utilizing the Server Message Block (SMB) storage. This model enables customers to store data On-premises in order to comply with regulations and policies.

Microsoft Teams is a cloud-based service, where the backend infrastructure is hosted on Microsoft Azure. For the purposes of recording Microsoft Teams sessions, the Service Fabric Cluster and the Service BOT channel for SmartTAP recording must reside on Microsoft Azure as well. However, sometimes due to regulatory or policy reasons, it is preferable to store management and media files On-premises.

The figure below illustrates this topology. The connection between the local and the Azure deployment is secured over an IPsec VPN connection. Media is captured on the Teams deployment and sent to SmartTAP On-premises where it can be saved in a user configured SMB Schema Fileshare directory.

Figure 6-8: VPN Setup for SmartTAP Teams Hybrid

Figure 6-9:



Before proceeding with the setup, observe the prerequisites: Prerequisites for Hybrid Deployment on the next page

This section includes the following procedures:

1. Set VPN Firewall Rules on the next page
2. Create a Virtual Network on page 19
3. Create the VPN Gateway on page 22
4. Create the Local Network Gateway on page 26
5. Configure your VPN Device on page 28
6. Create the VPN Connection on page 29
7. Verify the VPN Connection on page 30
8. Connect to a Virtual Machine on page 30

9. Change a Gateway SKU (resize a Gateway) on page 31

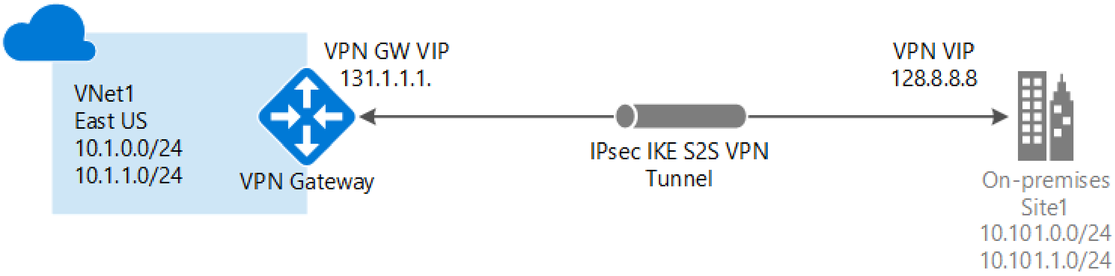
Prerequisites for Hybrid Deployment

- Make sure you have a compatible VPN device and a support engineer who is qualified to configure it.
- You have an externally facing public IPv4 address for your VPN device.
- When you create this configuration, you must specify the IP address range prefixes that Azure will route to your on-premises location. None of the subnets of your on-premises network can overlap with the virtual network subnets to which you wish to connect.
- **A Shared Key:** This key is used by both ends (On-premises, VPN device and Azure Virtual Network Gateway) for their initial handshake and configuration. Must be entered on both devices.

Creating a Site-to-Site Connection

This section describes how to use the Azure portal to create a Site-to-Site VPN gateway connection from your on-premises network to the Azure VNet. A Site-to-Site VPN gateway connection is used to connect your On-premises network to an Azure virtual network over an IPsec/IKE (IKEv1 or IKEv2) VPN tunnel. This type of connection requires a VPN device located on-premises that has an externally facing public IP address assigned to it.

Figure 6-10: Site-to-Site Connection



This section does not describe configuration of the on-premises VPN device, which may vary between vendors. For more information, see this list for Microsoft supported devices.

Set VPN Firewall Rules

The following are firewall rules to set on the on-premises Firewall/VPN device for the VPN tunnel.

Table 6-2: VPN Firewall Rules

Protocol	Ports	Connection	Port Flow	Description
TCP	80,443	Azure VNet ⇔ On-	Bi-	HTTP/S

Protocol	Ports	Connection	Port Flow	Description
		prem site	directional	between sites
TCP	443	On-prem ⇒ SmartTAP Þ Any	Send-only	HTTPS to Azure
TCP+UDP	53	Azure VNet ⇔ On-prem site	Bi-directional	DNS between sites
TCP	445	Azure VNet ⇒ On-prem site	Send-only	CIFS (SMB) Access
	Echo Request	Azure VNet ⇔ On-prem site	Bi-directional	Echo Request (Ping)
TCP	3389	On-prem site ⇒ Azure VNet	Send-only	Remote-Desktop

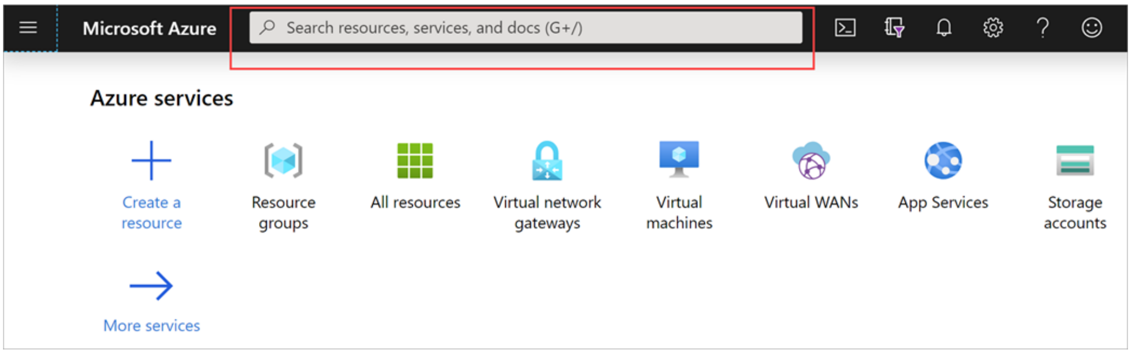
Create a Virtual Network

This section describes how to create a virtual network.

➢ To create a Virtual network:

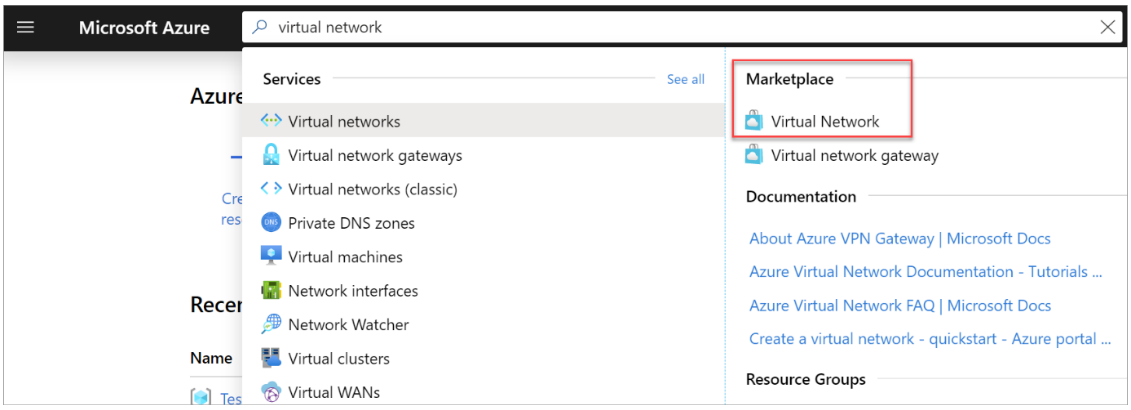
1. Sign into the Azure Portal.

Figure 6-11: Search Resources



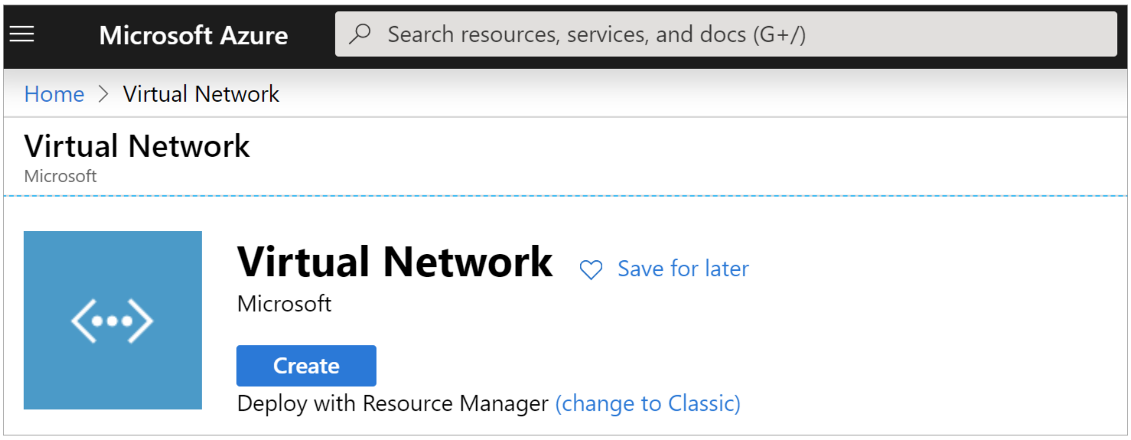
2. Select Virtual Network from the Marketplace search results.

Figure 6-12: Select Virtual Network



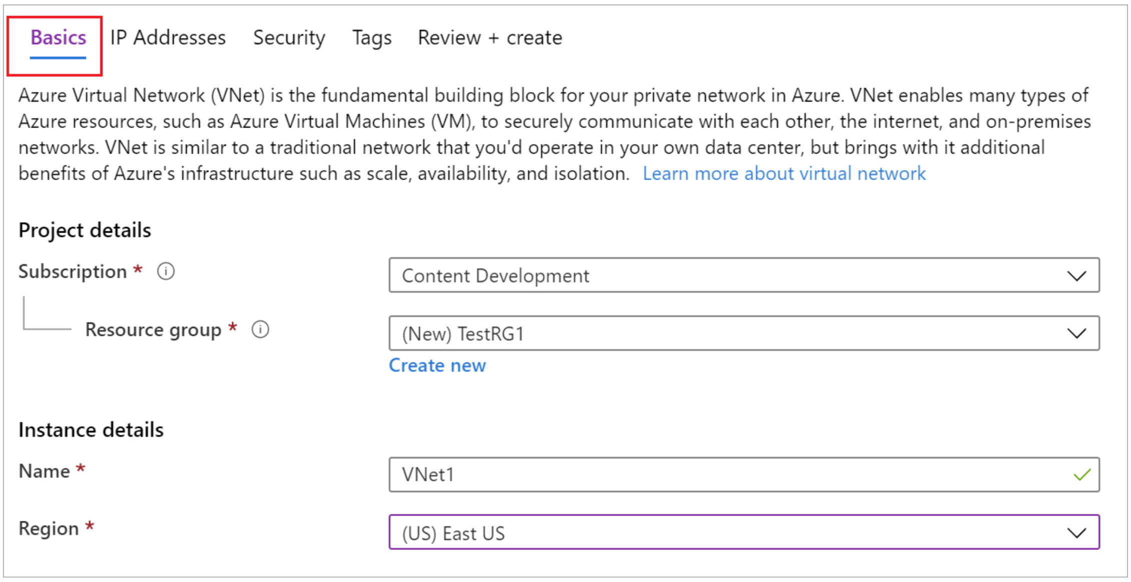
4. **3.** On the Virtual Network page, select **Create**; the Create virtual network page opens.

Figure 6-13: Create Virtual Network



5. **4.** Select the **Basics** tab and configure Project details and Instance details VNet settings.

Figure 6-14: Basics



Azure Virtual Network (VNet) is the fundamental building block for your private network in Azure. VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks. VNet is similar to a traditional network that you'd operate in your own data center, but brings with it additional benefits of Azure's infrastructure such as scale, availability, and isolation. Learn more about virtual network

Project details

Subscription * ⓘ Content Development

Resource group * ⓘ (New) TestRG1

Create new

Instance details

Name * VNet1 ✓

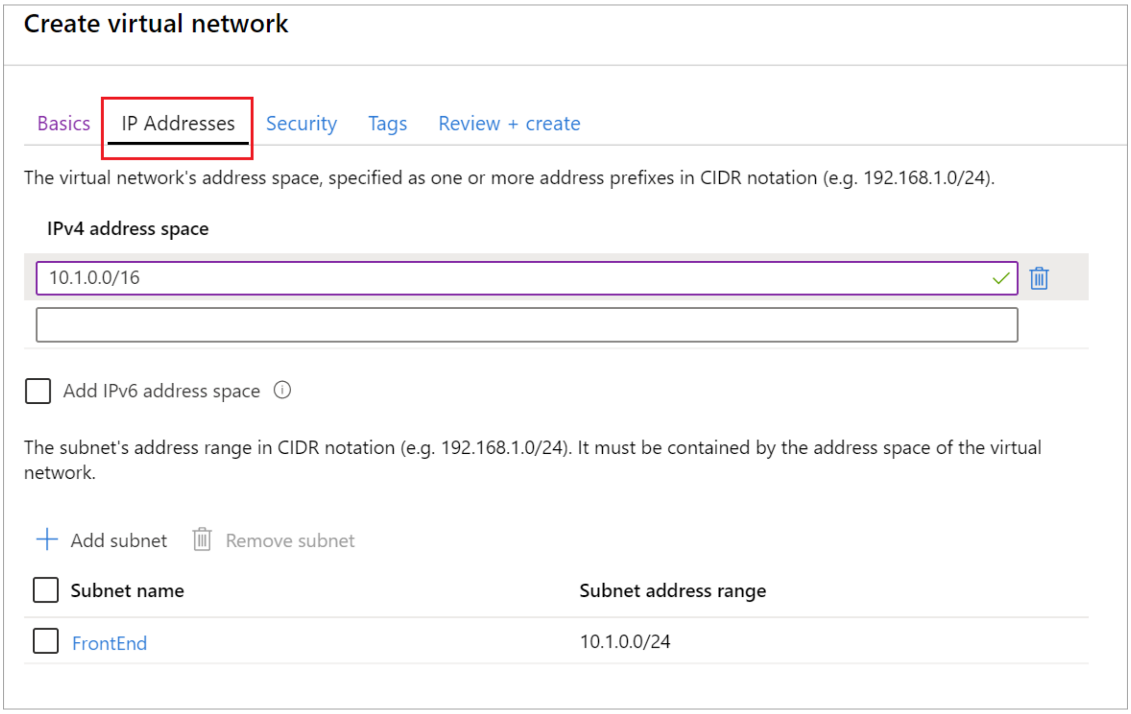
Region * (US) East US

7. When you fill in the fields in this screen, a green check mark appears when the characters you enter in the field are validated. Some values are autofilled, which you can replace with your own values:

- ● **Subscription:** Verify that the subscription listed is the correct one. You can change subscriptions by using the drop-down.
- ● **Resource group:** Select an existing resource group, or click Create new to create a new one.
- ● **Name:** Enter the name for your virtual network.
- ● **Region:** Select the location for your VNet. The location determines where the resources that you deploy to this VNet will live.

8. 5. On the **IP Addresses** tab, configure the values. The values shown in the examples below are for demonstration purposes. Modify these values according to your network configuration.

Figure 6-15: IP Networking



Create virtual network

The virtual network's address space, specified as one or more address prefixes in CIDR notation (e.g. 192.168.1.0/24).

IPv4 address space

10.1.0.0/16

□ Add IPv6 address space ⓘ

The subnet's address range in CIDR notation (e.g. 192.168.1.0/24). It must be contained by the address space of the virtual network.

□ Subnet name	Subnet address range
□ FrontEnd	10.1.0.0/24

- ● **IPv4 address space:** By default, an address space is automatically created. You can click the address space to adjust it to reflect your own values. You can also add additional address spaces.
- ● **Subnet:** If you use the default address space, a default subnet is created automatically. If you change the address space, you need to add a subnet. Select + Add subnet to open the Add subnet window. Configure the following settings and then select Add to add the values:
- ● **Subnet name:** In this example, we named the subnet "FrontEnd".
- ● **Subnet address range:** The address range for this subnet.

6. Select the **Security** tab, leave the default values:
   - DDos protection: Basic
   - Firewall: Disabled
7. Select **Review + create** to validate the virtual network settings.
8. After the settings have been validated, select **Create**.

Create the VPN Gateway

This step describes how to create the virtual network gateway for your VNet. The virtual network gateway uses a specific subnet called the gateway subnet. The gateway subnet is part of the virtual network IP address range that you specify when configuring your virtual network. It contains the IP addresses that the virtual network gateway resources and services use.

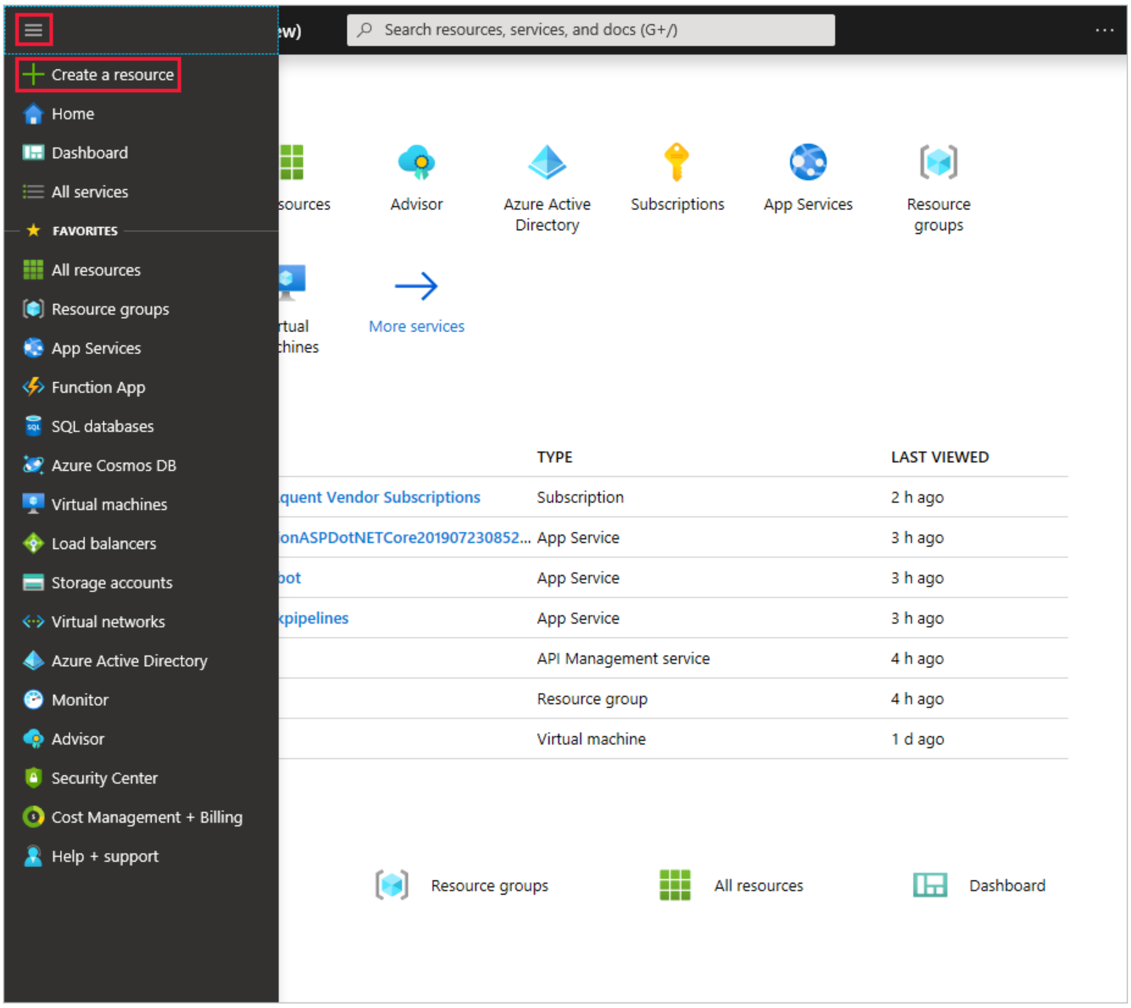
When you create the gateway subnet, you specify the number of IP addresses that the subnet contains. The number of required IP addresses depends on the VPN gateway configuration that you wish to create. Some configurations require more IP addresses than others. We recommend that you create a gateway subnet that uses a /27 or /28.

If you see an error that specifies that the address space overlaps with a subnet, or that the subnet is not contained within the address space for your virtual network, check your VNet address range. You may not have sufficient IP addresses available in the address range you created for your virtual network. For example, if your default subnet encompasses the entire address range, there are no IP addresses left to create additional subnets. In this case, you can either adjust your subnets within the existing address space to free up IP addresses, or specify an additional address range and create the gateway subnet in this range.

➢ To create the VPN Gateway:

1. From the Azure portal menu, select **Create a resource**.

Figure 6-16: Create a Resource



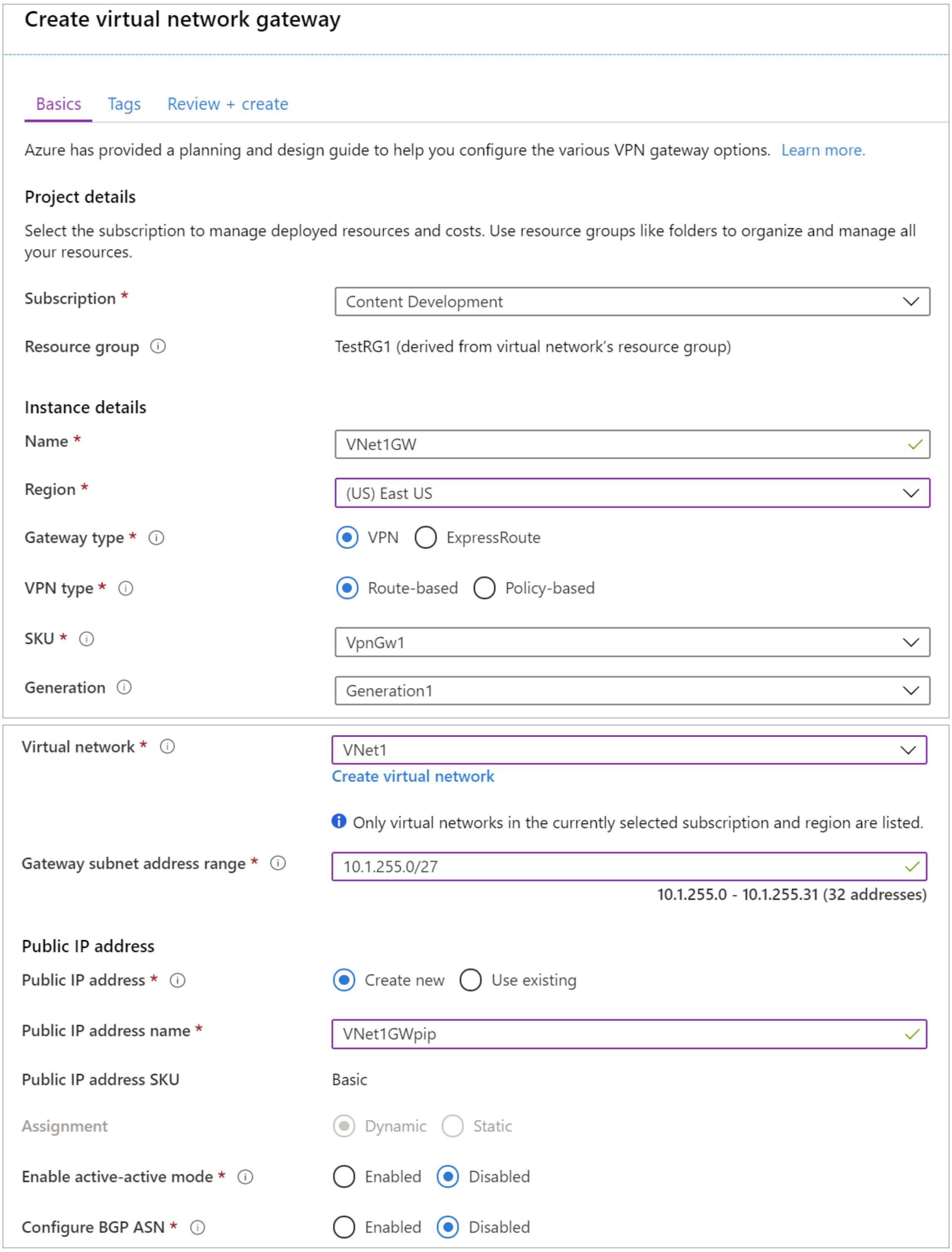
	TYPE	LAST VIEWED
quent Vendor Subscriptions	Subscription	2 h ago
ionASPDotNETCore201907230852...	App Service	3 h ago
bot	App Service	3 h ago
kpipelines	App Service	3 h ago
	API Management service	4 h ago
	Resource group	4 h ago
	Virtual machine	1 d ago

2. 2. In the Search the Marketplace field, type 'Virtual Network Gateway'.

3. Locate Virtual network gateway in the search results and select the entry.

4. On the Virtual network gateway page, select **Create**; the Create virtual network gateway page opens.

Figure 6-17: Virtual Network Gateway- Project Details



Create virtual network gateway

Azure has provided a planning and design guide to help you configure the various VPN gateway options. Learn more.

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Content Development
Resource group ⓘ	TestRG1 (derived from virtual network's resource group)

Instance details

Name *	VNet1GW ✓
Region *	(US) East US
Gateway type * ⓘ	● VPN ○ ExpressRoute
VPN type * ⓘ	● Route-based ○ Policy-based
SKU * ⓘ	VpnGw1
Generation ⓘ	Generation1
Virtual network * ⓘ	VNet1 Create virtual network ⓘ Only virtual networks in the currently selected subscription and region are listed.
Gateway subnet address range * ⓘ	10.1.255.0/27 ✓ 10.1.255.0 - 10.1.255.31 (32 addresses)

Public IP address

Public IP address * ⓘ	● Create new ○ Use existing
Public IP address name *	VNet1GWpip ✓
Public IP address SKU	Basic
Assignment	● Dynamic ○ Static
Enable active-active mode * ⓘ	○ Enabled ● Disabled
Configure BGP ASN * ⓘ	○ Enabled ● Disabled

3. **5.** Select the **Basics** tab and enter the values for your virtual network gateway:

- **Subscription:** Select the subscription you want to use from the dropdown.
- **Resource Group:** This setting is autofilled when you select your virtual network on this page.
- **Instance details:**

  - ◆ **Name:** Name your gateway. Naming your gateway not the same as naming a gateway subnet. It is the name of the gateway object you are creating.
  - ◆ **Region:** Select the region in which you want to create this resource. The region for the gateway must be the same as the virtual network.
  - ◆ **Gateway type:** Select VPN. VPN gateways use the virtual network gateway type VPN.
  - ◆ **VPN type:** Select the VPN type that is specified for your configuration. Most configurations require a Route-based VPN type.
  - ◆ **SKU:** Select the gateway SKU from the dropdown. The SKUs listed in the dropdown depend on the VPN type you select. For more information about gateway SKUs, see Gateway SKUs.

- **Generation:** For information about VPN Gateway Generation, see Gateway SKUs.
- **Virtual network:** From the dropdown, select the virtual network to which you want to add this gateway.
- **Gateway subnet address range:** This field only appears if your VNet doesn't have a gateway subnet. If possible, make the range /27 or larger (/26,/25 etc.). We don't recommend creating a range any smaller than /28. If you already have a gateway subnet, you can view Gateway Subnet details by navigating to your virtual network. Click Subnets to view the range. If you want to change the range, you can delete and recreate the GatewaySubnet.
- **Public IP address:** This setting specifies the public IP address object that gets associated to the VPN gateway. The public IP address is dynamically assigned to this object when the VPN gateway is created. The only time the Public IP address changes is when the gateway is deleted and re-created. It doesn't change across resizing, resetting, or other internal maintenance/upgrades of your VPN gateway.
  - ◆ **Public IP Address:** Leave Create new selected.
  - ◆ **Public IP address name:** In the text box, type a name for your public IP address instance.
  - ◆ **Assignment:** VPN gateway supports only Dynamic.
  - ◆ **Active-Active mode:** Only select Enable active-active mode if you are creating an active-active gateway configuration. Otherwise, leave this setting unselected.
  - ◆ Leave **Configure BGP ASN** deselected, unless your configuration specifically requires this setting.

4. 6. Select **Review + create** to run validation. Once validation passes, select **Create** to deploy the VPN gateway. A gateway can take up to 45 minutes to fully create and deploy. You can see the deployment status on the Overview page for your gateway. After the gateway is created, you can view the IP address that has been assigned to it by viewing the virtual network in the portal. The gateway appears as a connected device.

When working with gateway subnets, avoid associating a network security group (NSG) to the gateway subnet. Associating a network security group may cause your Virtual Network gateway (VPN, Express Route gateway) to stop functioning as expected.

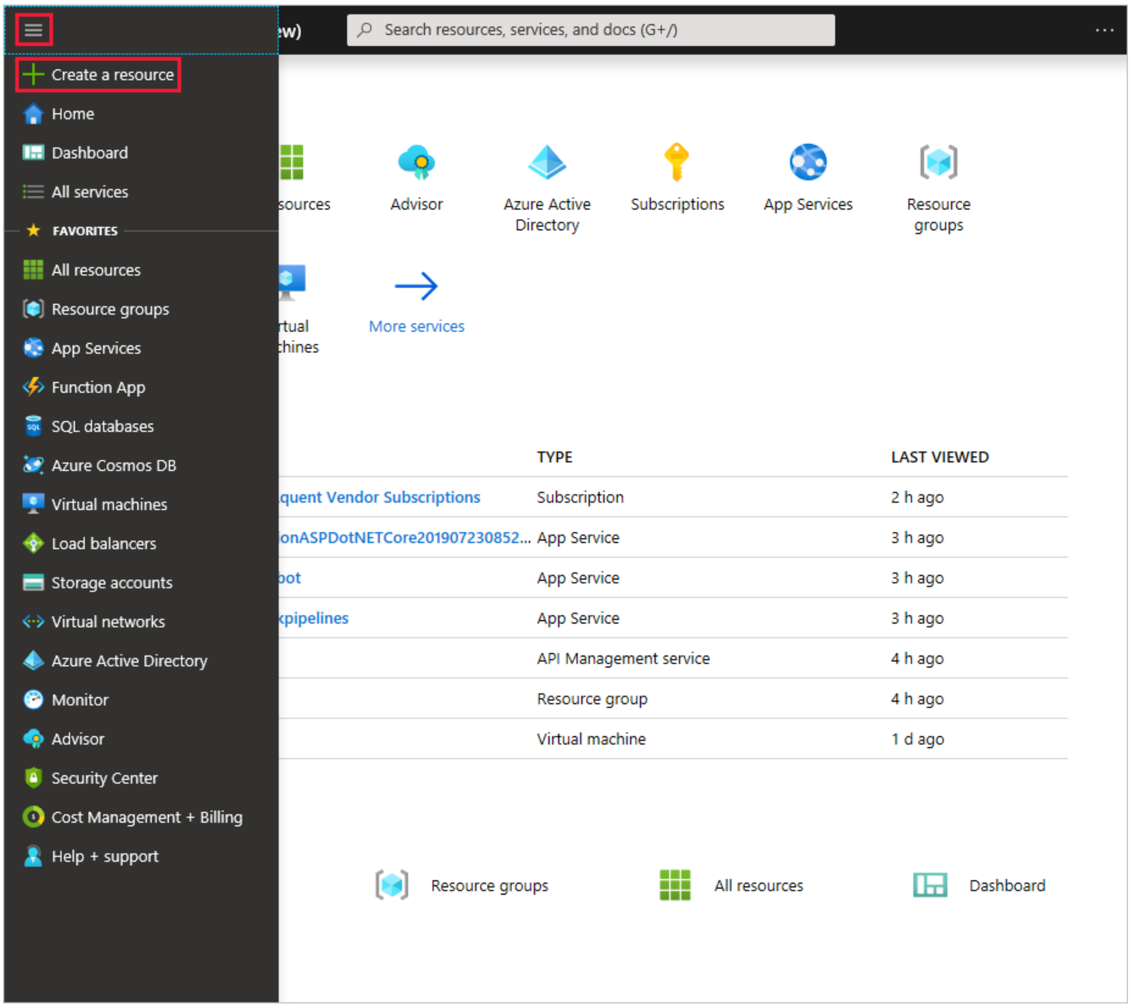
Create the Local Network Gateway

The local network gateway is a specific object that represents your on-premises location (the site) for routing purposes. You define a name for the site for which Azure can refer to it, then specify the IP address of the On-premises VPN device to which you will create a connection. You also specify the IP address prefixes that are to be routed through the VPN gateway to the VPN device. The address prefixes you specify are the prefixes located on your on-premises network. If your On-premises network changes or you need to change the public IP address for the VPN device, you can easily later update these values.

➢ To create the local network gateway:

1. From the Azure portal menu, select Create a resource.

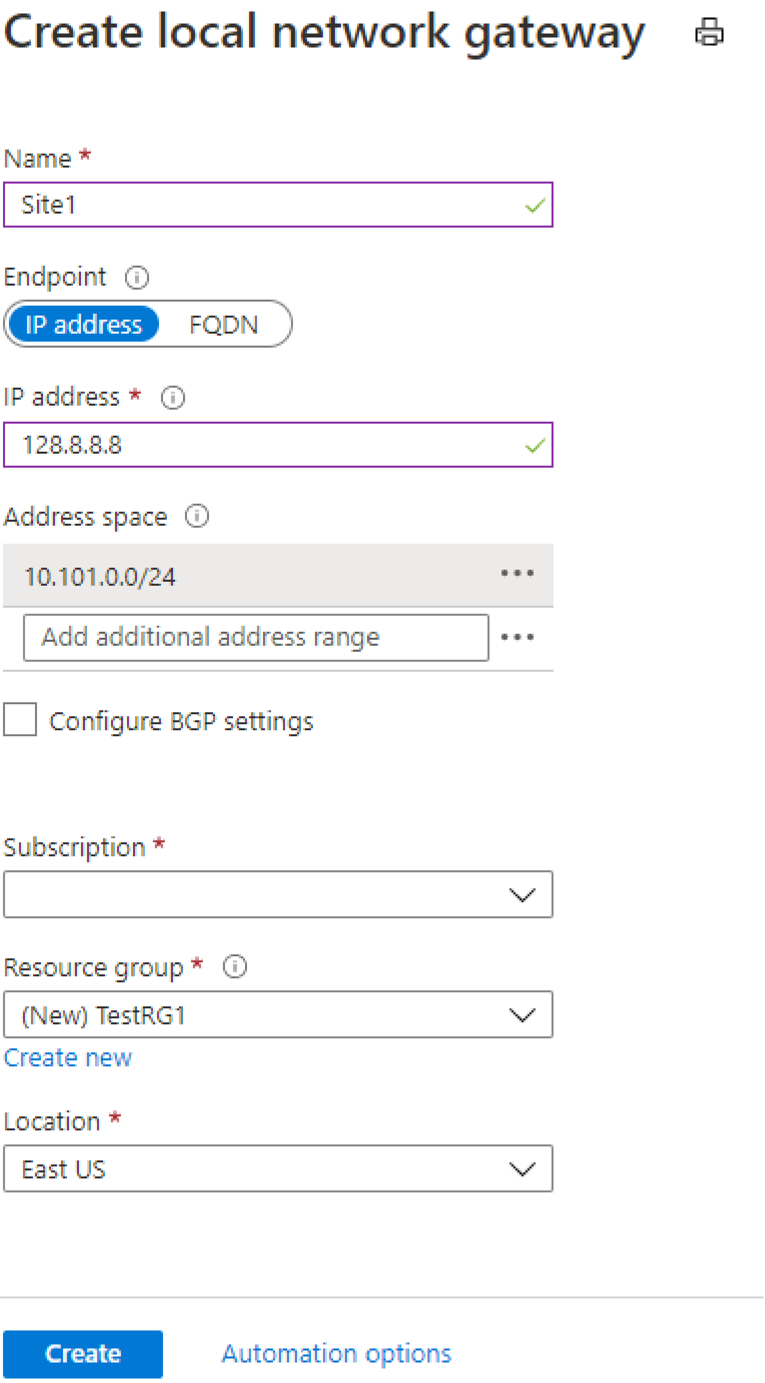
Figure 6-18: Create a Resource



	TYPE	LAST VIEWED
quent Vendor Subscriptions	Subscription	2 h ago
onASPDotNETCore201907230852...	App Service	3 h ago
oot	App Service	3 h ago
xpipelines	App Service	3 h ago
	API Management service	4 h ago
	Resource group	4 h ago
	Virtual machine	1 d ago

2. **2.** In the Search the marketplace field, type Local network gateway, then press Enter to activate the search; a list of results is returned. Click Local network gateway, then click **Create** to open the Create local network gateway page:

Figure 6-19: Create Local Network Gateway



Create local network gateway

Name * Site1 ✓

Endpoint ⓘ
● IP address ○ FQDN

IP address * ⓘ 128.8.8.8 ✓

Address space ⓘ
10.101.0.0/24 •••
Add additional address range •••

□ Configure BGP settings

Subscription *

Resource group * ⓘ (New) TestRG1
Create new

Location * East US

3. **3.** On the Create local network gateway page, specify the values for your local network gateway:

- **Name:** Specify a name for your local network gateway object.
- **Endpoint:** Select the endpoint type for the on-premises VPN device - IP address or FQDN (Fully Qualified Domain Name).
- **IP address:** If you have a static public IP address allocated from your Internet service provider for your VPN device, select the IP address option and fill in the IP address as

shown in the example. This is the public IP address of the VPN device for which you wish the Azure VPN gateway to connect. If you don't have the IP address at this point in time, you can temporarily use the values shown in the example; however, you will need to later replace your placeholder IP address with the public IP address of your VPN device, otherwise Azure will not be able to connect.

- **FQDN:** If you have a dynamic public IP address that could change after certain period of time, usually determined by your Internet service provider, you can use a constant DNS name with a Dynamic DNS service to point to your current public IP address of your VPN device. Your Azure VPN gateway will resolve the FQDN to determine the public IP address to connect to. A screenshot below shows an example of using FQDN instead of IP address.
- **Address Space:** Refers to the address ranges for the network that this local network represents. You can add multiple address space ranges. Make sure that the ranges you specify here do not overlap with ranges of other networks that you want to connect to. Azure will route the address range that you specify to the on-premises VPN device IP address. Use your own values here if you want to connect to your on-premises site, not the values shown in the example.
- **Configure BGP settings:** Use only when configuring BGP. Otherwise, don't select this.
- **Subscription:** Verify that the correct subscription is showing.
- **Resource Group:** Select the resource group that you want to use. You can either create a new resource group, or select one that you have already created.
- **Location:** The location is the same as Region in other settings. Select the location that this object will be created in. You may want to select the same location that your VNet resides in, but you are not required to do so.

4. **4.** When you have finished specifying the values, select **Create** at the bottom of the page to create the local network gateway.

- Azure VPN supports only one IPv4 address for each FQDN. If the domain name resolves to multiple IP addresses, Azure VPN Gateway will use the first IP address returned by the DNS servers. To eliminate the uncertainty, we recommend that your FQDN always resolve to a single IPv4 address. IPv6 is not supported.
- Azure VPN Gateway maintains a DNS cache refreshed every 5 minutes. The gateway tries to resolve the FQDNs for disconnected tunnels only. Resetting the gateway will also trigger FQDN resolution.

Configure your VPN Device

Site-to-Site connections to an On-premises network requires a VPN device. This procedure provides the basic setup instructions for configuring your VPN device regarding shared properties. When configuring your VPN device, you need the following:

- **A shared key**. This is the same shared key that you specify when creating your Site-to-Site VPN connection. We recommend that you generate a complex key for use.
- The **Public IP address** of your virtual network gateway. You can view the public IP address by using the Azure portal. To find the Public IP address of your VPN gateway using the Azure portal, navigate to Virtual network gateways, then click the name of your gateway.

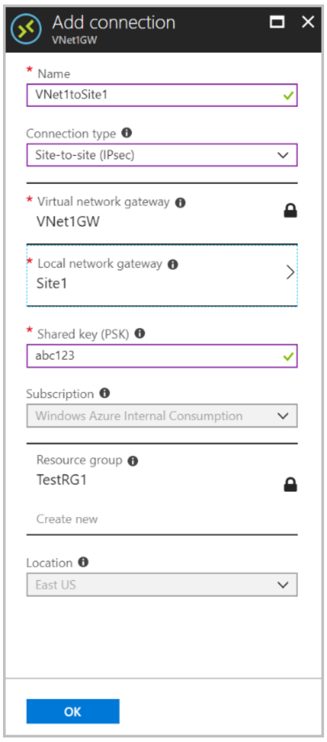
Create the VPN Connection

This procedure describes how to create the Site-to-Site VPN connection between your virtual network gateway and your on-premises VPN device.

➢ To create the VPN Connection:

1. Open the page for your virtual network gateway. You can navigate to the gateway (**Name of your VNet** > **Overview** > **Connected devices** > **Name of your gateway**).
2. On the page for the gateway, click **Connections**. At the top of the Connections page, click +Add to open the Add connection page.

Figure 6-20: Add Connection



Add connection
VNet1GW

* Name
VNet1toSite1

Connection type
Site-to-site (IPsec)

* Virtual network gateway
VNet1GW

* Local network gateway
Site1

* Shared key (PSK)
abc123

Subscription
Windows Azure Internal Consumption

Resource group
TestRG1
Create new

Location
East US

OK

Enter the following details:

- **Name:** Name your connection.
- **Connection type:** Select Site-to-site(IPSec).
- **Virtual network gateway:** The value is fixed because you are connecting from this gateway.
- **Local network gateway:** Click Choose a local network gateway and select the local network gateway that you wish to use.
- **Shared Key:** the value here must match the value that you are using for your local on-premises VPN device. The example in the figure above uses 'abc123', however you can define a more complex string. It is important to note that the value you specify for this key must be the same value that you specify when configuring your VPN device.

The remaining values for Subscription, Resource Group are determined by the specific Location.

3. 3. Click **OK** to create your connection. The message "Creating Connection" appears on the screen. You can view the connection in the Connections page of the virtual network gateway. The Status will change from "Unknown" to "Connecting" and then to "Succeeded".

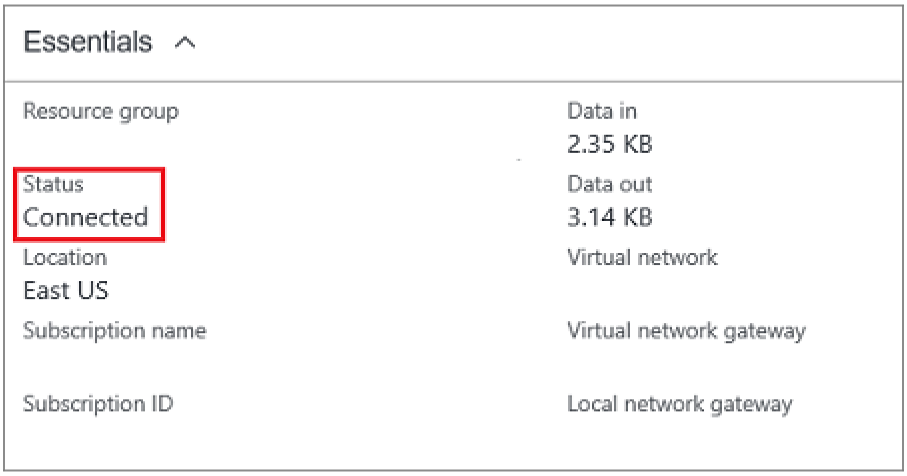
Verify the VPN Connection

In the Azure portal, you can view the connection status of a Resource Manager VPN Gateway by navigating to the connection. The following describes one method to navigate to your connection and verify.

➢ To verify connection:

1. In the Azure portal menu, select **All resources** or search for and select All resources from any page.
2. Select to your virtual network gateway.
3. On the blade for your virtual network gateway, click **Connections**. You can view the status of each connection.

Figure 6-21: Verify Connection



Essentials ^

Resource group

Status
Connected

Location
East US

Subscription name

Subscription ID

Data in
2.35 KB

Data out
3.14 KB

Virtual network

Virtual network gateway

Local network gateway

4. 4. Click the name of the connection whose "Essentials" you wish to verify. In "Essentials", you can view more information about your connection. When you have established a successful connection, the statuses 'Succeeded' and 'Connected' are displayed.

Connect to a Virtual Machine

You can connect to a VM that is deployed to your VNet by creating a Remote Desktop Connection to your VM. The best way to initially verify that you can connect to your VM is to connect using its private IP address, instead of its computer name. Using this method, you can test to see if you can connect and not whether name resolution is configured correctly.

Reset a VPN Gateway

Resetting an Azure VPN gateway is useful if you lose cross-premises VPN connectivity on one or more Site-to-Site VPN tunnels. In this situation, your On-premises VPN devices are all working correctly, however are not able to establish IPsec tunnels with the Azure VPN gateways. For details, see Reset a VPN gateway.

Change a Gateway SKU (resize a Gateway)

For the steps to change a gateway SKU, see Gateway SKUs.

7 Step 2 Create Service Fabric Cluster

This procedure describes how to deploy the Service Fabric Cluster using the Azure Resource Manager template which uses Jason files and power shell scripts for creating the the Service Fabric Cluster instead of using the Azure portal.

➢ To create a service fabric cluster:

1. Extract the SFC Deployment script package from the following location to your local machine:
   ..\Release\Publish\HueBot_Deployment_Package\SFCDeploymentscript

   This directory includes the following files:

   - AzureDeploy.json
   - AzureDeploy.Parameters.json
   - Deploy.ps1

2. Using a text editor, open the file AzureDeploy.Parameters.json and set the following parameters:

```
"parameters": {
"clusterLocation": {
"value": "westus"
},
"clusterName": {
"value": "teamsbotclustetest"},
"adminUserName": {
"value": "huebot"
},
"adminPassword": {
"value": "Password!1"},
```

```
"nt0InstanceCount": {
"value": 3
},
```

```
"vmNodeType0Size": {
"value": "Standard_D2_V2" # change to  Standard_DS2_V2
```

```
"vmImageSku":
{value": "2016-Datacenter-with-Containers"
}
```

The above parameter is set according to the cluster durability settings for the number of instances in the Service Fabric Cluster. If Durability and reliability need to be changed according to the required number of instances,using a text editor, open the file AzureDeploy.json and set the following parameters:

```
"publisher": "Microsoft.Azure.ServiceFabric",
"settings": {
"clusterEndpoint": "[reference(parameters('clusterName')).clusterEndpoint]",
"nodeTypeRef": "[variables('vmNodeType0Name')]",
"dataPath": "D:\\SvcFab",
"durabilityLevel": "Bronze",  "clientConnectionEndpointPort": "[variables
('nt0fabricTcpGatewayPort')]",
"durabilityLevel": "Bronze",
"provisioningState": "Default",
"reliabilityLevel": "Silver",
```

3. Using a text editor, open file deploy.ps1 and set the following parameters:

```
$subscriptionName="%replace_with_azure_subscription_name%"
```

```
$resourceGroupName="<resourceGroupName>"
```

```
$keyvaultName="%replace_with_azure_keyvault_name%"
```

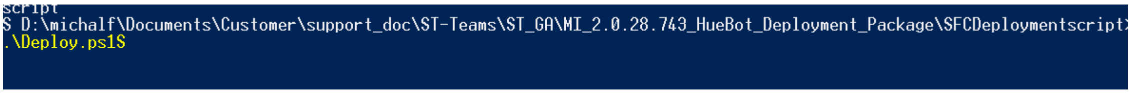
```
$parameterFilePath="%replace_with_path_to_repos_folder%\service-
shared_platform_samples\LocalMediaSamples\HueBot\HueBot\ARM_
Deployment\AzureDeploy.Parameters.json"
```

```
$templateFilePath="%replace_with_path_to_repos_folder%\service-shared_
platform_samples\LocalMediaSamples\HueBot\HueBot\ARM_
Deployment\AzureDeploy.json"
```

```
$secretID="%replace_with_secret_id_of_certificate_from_keyvault%"
```

4. Open PowerShell window as 'admin', run the Deploy.ps1 script from the folder location to which you extracted this file.

Figure 7-1: Run Deploy Script



```
script
$ D:\michalf\Documents\Customer\support_doc\ST-Teams\ST_GA\MI_2.0.28.743_HueBot_Deployment_Package\SFCDeploymentscript>
.\Deploy.ps1$
```

5. Install the following Prerequisites programs on each deployed Service Fabric node in the Service Fabric cluster.

> ⚠ All program files are located in the Prerequisites_installation package folder.

- Azure SDK for service fabric
- Microsoft Speech Platform Runtime v11 (x64).
- Microsoft Speech Recognition en-US.
- Microsoft TTS en-US
- Microsoft Visual C++ Redistributable 2019
- Net 4.8 framework
- RTS install (part of prerequisites script)
- OVOC client

> ⚠ For Multi-tenancy deployments, where each tenant has a dedicated Service Fabric node, this step should be performed for all deployed Service Fabric nodes that are deployed in the Service Fabric cluster.

6. Restart all nodes.

8 Step 3 Create Service BOT Channel

This procedure describes how to create a service BOT channel (see below) on Microsoft Azure. This step also includes the following procedures:

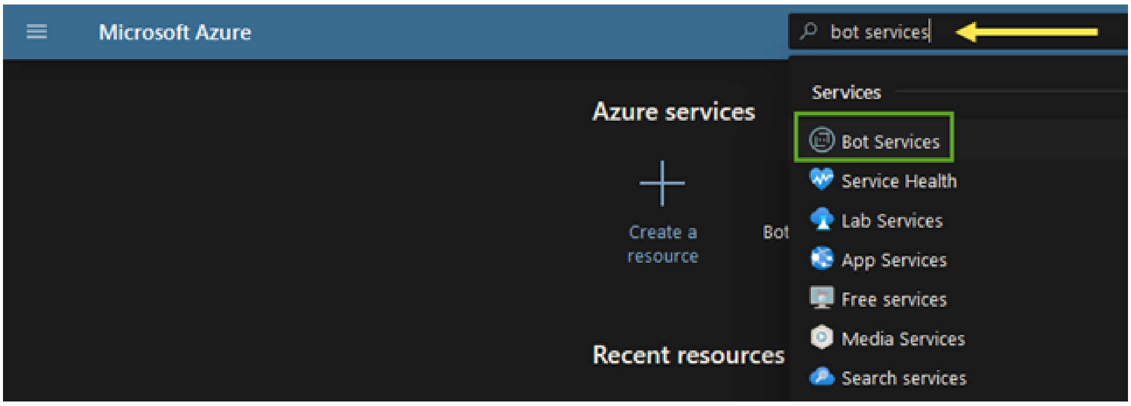
1. Configure Service Channel on page 40
2. Grant API Permissions to BOT Service on page 44

> ⚠ Before deploying your SmartTAP 360° BOT in production, you must provide AudioCodes SmartTAP 360° Teams BOT application ID and respective deployment Teams Tenant ID to AudioCodes support. This is necessary to enable traffic throttling exceptions, otherwise the call recording maybe throttled in the event of higher loads or longer calls.

➢ To create a service BOT channel:

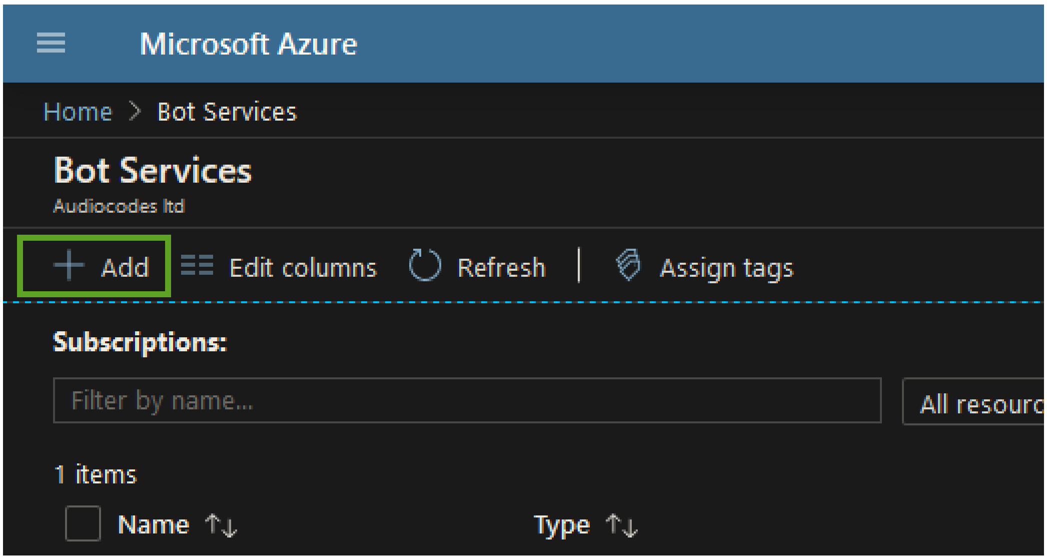
1. In the Azure portal, open the BOT Services screen (**Services** > **Bot Services**).

Figure 8-1: Azure Services



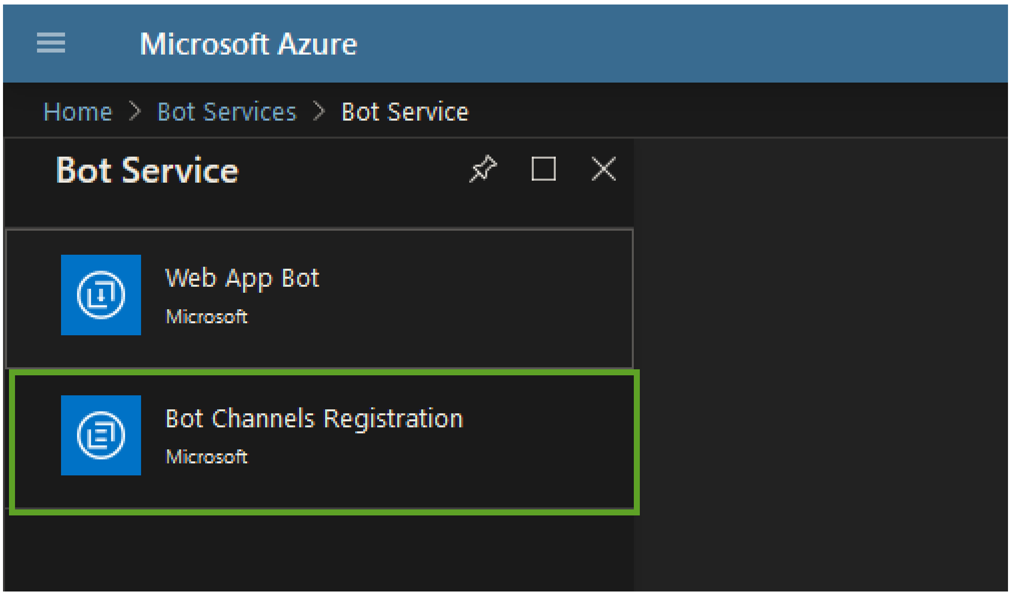
2. Click **Add** to add a new Bot service.

Figure 8-2: Add BOT Service



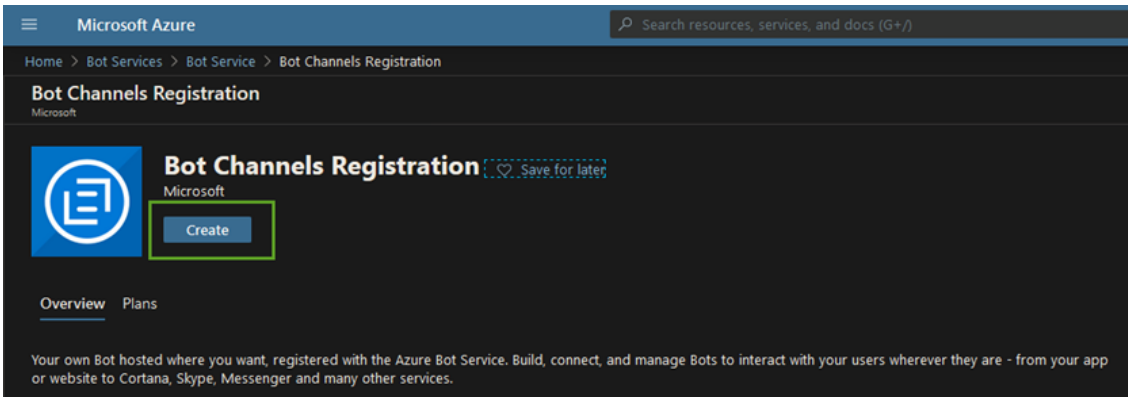
3. Click **Bot Channels Registration**.

Figure 8-3: BOT Channels Registration



4. Click **Create** to create the service.

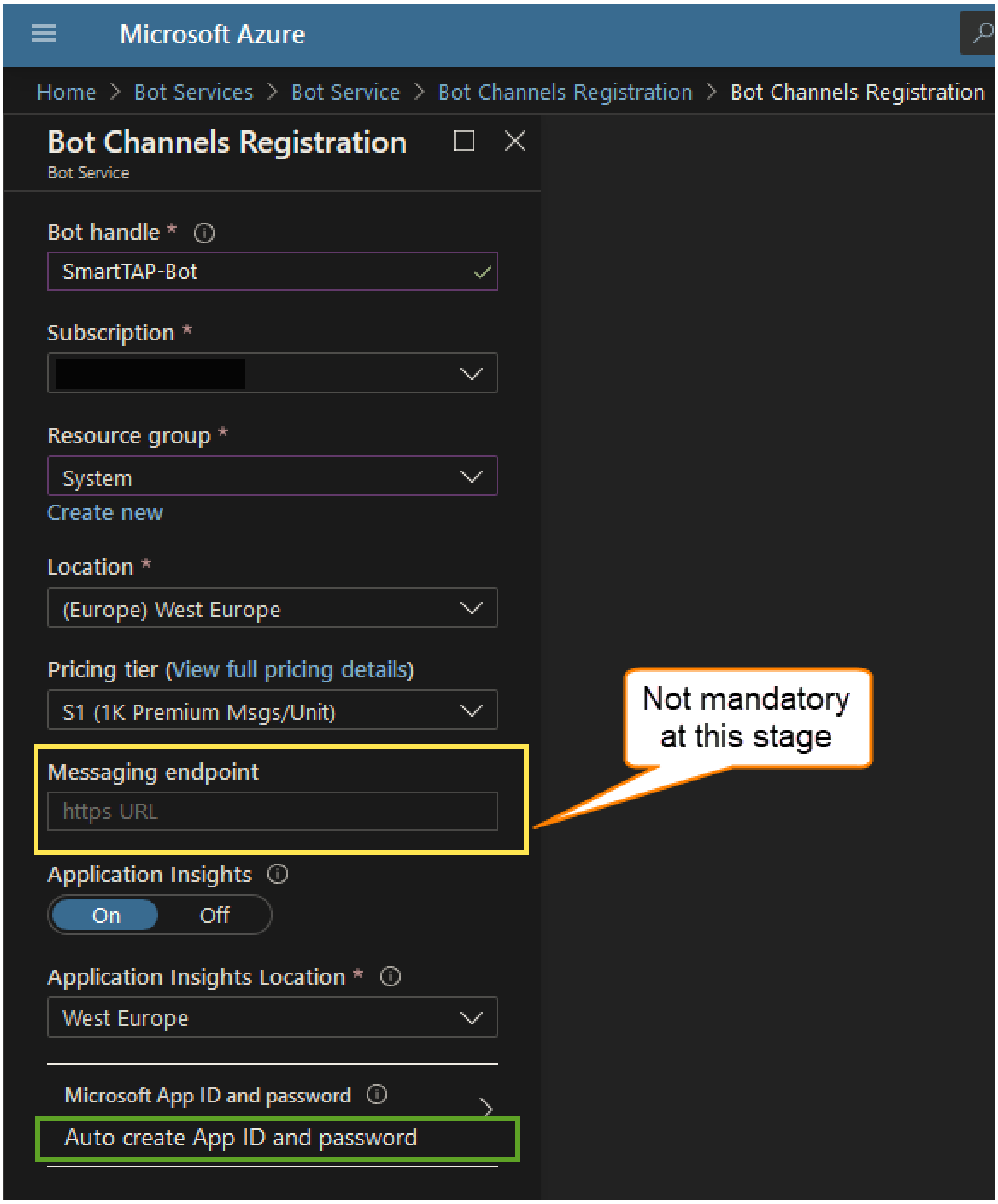
Figure 8-4: Create the Service


Microsoft Azure
Search resources, services, and docs (G+/)
Home > Bot Services > Bot Service > Bot Channels Registration
Bot Channels Registration
Microsoft
Bot Channels Registration
Save for later
Microsoft
Create
Overview
Plans

Your own Bot hosted where you want, registered with the Azure Bot Service. Build, connect, and manage Bots to interact with your users wherever they are - from your app or website to Cortana, Skype, Messenger and many other services.

5. Set the relevant parameters shown in the Bot Channels Registration screen below.

Figure 8-5: Parameter Configuration



Microsoft Azure

Home > Bot Services > Bot Service > Bot Channels Registration > Bot Channels Registration

Bot Channels Registration
Bot Service

Bot handle *: SmartTAP-Bot

Subscription *: [redacted]

Resource group *: System

Create new

Location *: (Europe) West Europe

Pricing tier (View full pricing details): S1 (1K Premium Msgs/Unit)

Messaging endpoint: https URL

Application Insights: On Off

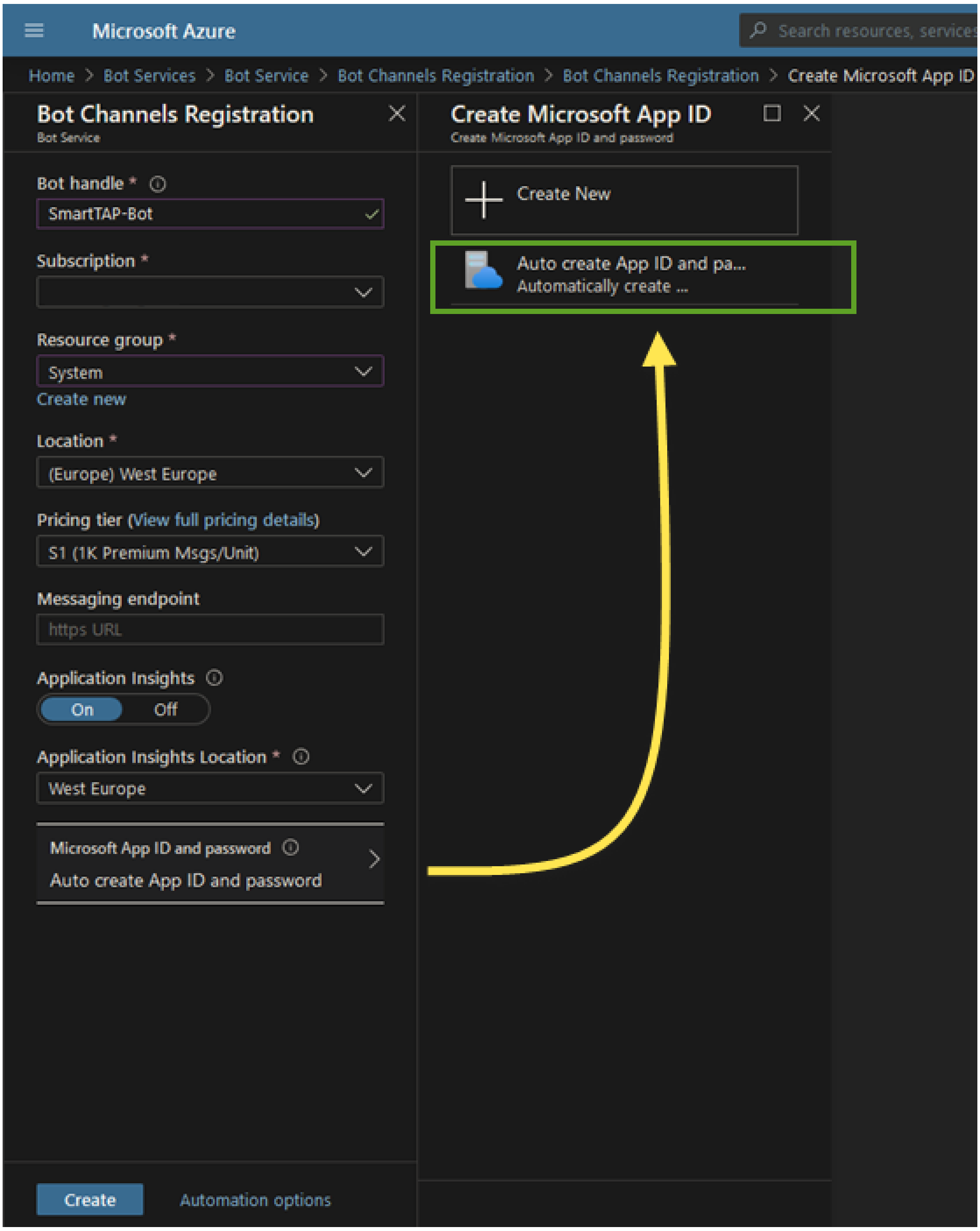
Application Insights Location *: West Europe

Microsoft App ID and password: Auto create App ID and password

Not mandatory at this stage

6. Select **Auto create App ID and password** to create the Microsoft App ID (copy to notepad as this value is configured in later in Step 6 Enable Users with Compliance Recordings on page 60).

Figure 8-6: Auto Create App ID



Microsoft Azure
Search resources, services
Home > Bot Services > Bot Service > Bot Channels Registration > Bot Channels Registration > Create Microsoft App ID
Bot Channels Registration
Bot Service
Create Microsoft App ID
Create Microsoft App ID and password
Create New
Auto create App ID and pa...
Automatically create ...
Create
Automation options

Bot handle * SmartTAP-Bot

Subscription *

Resource group * System

Create new

Location * (Europe) West Europe

Pricing tier (View full pricing details) S1 (1K Premium Msgs/Unit)

Messaging endpoint https URL

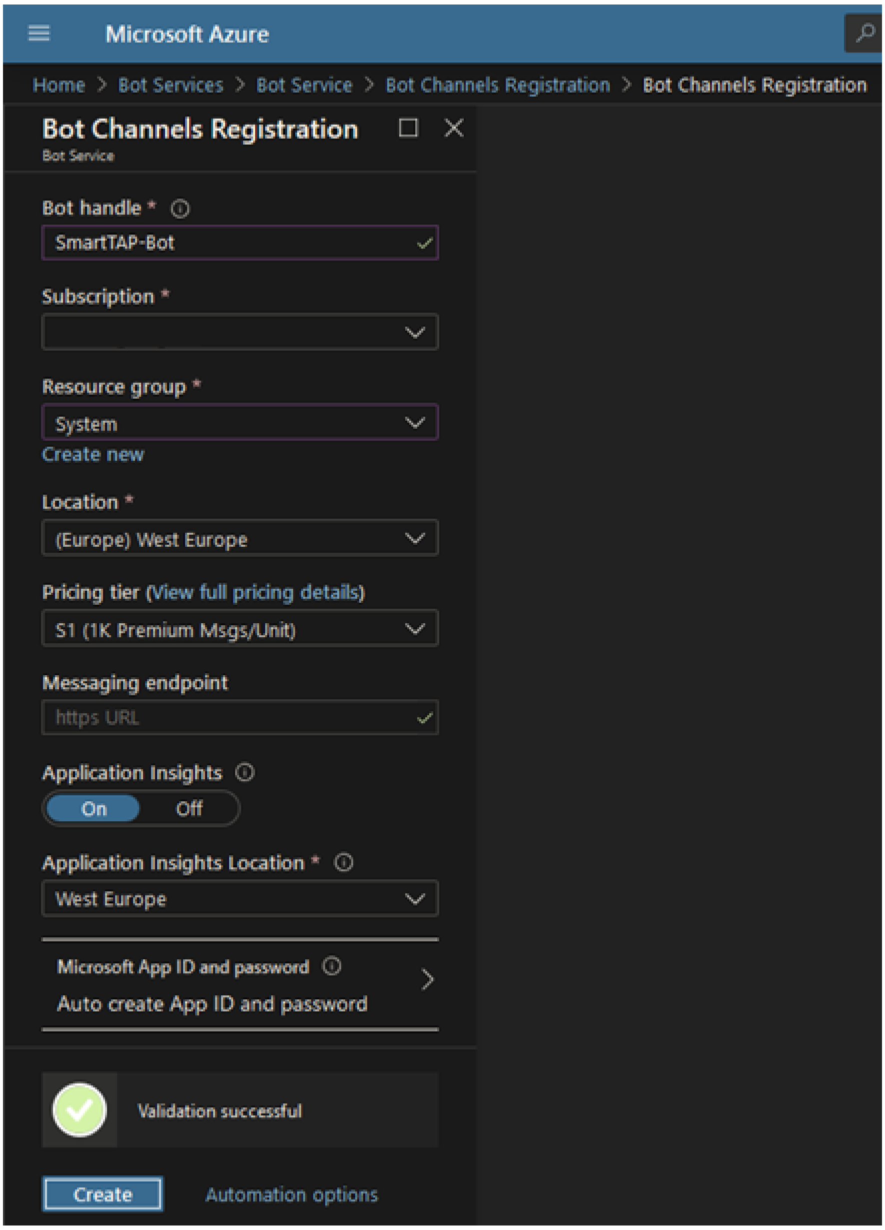
Application Insights On Off

Application Insights Location * West Europe

Microsoft App ID and password: Auto create App ID and password

7. Click **Create.**

Figure 8-7: Bot Channels Registration Details



Microsoft Azure
Home > Bot Services > Bot Service > Bot Channels Registration > Bot Channels Registration

Bot Channels Registration

Bot Service

Bot handle * : SmartTAP-Bot

Subscription * :

Resource group * : System

Create new

Location * : (Europe) West Europe

Pricing tier (View full pricing details) : S1 (1K Premium Msgs/Unit)

Messaging endpoint : https URL

Application Insights : On Off

Application Insights Location * : West Europe

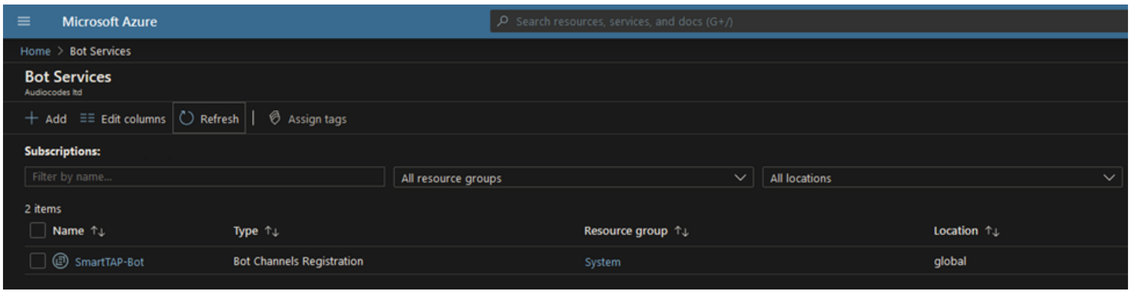
Microsoft App ID and password : Auto create App ID and password

Validation successful

8. Once Validation is successful, click **Create** to create the service.

The resource is created and you are prompted to display the resource; confirm and the new resource is displayed:

Figure 8-8: New Resource



Bot Services

Audiocodes ltd

Subscriptions:

2 items

Name	Type	Resource group	Location
SmartTAP-Bot	Bot Channels Registration	System	global

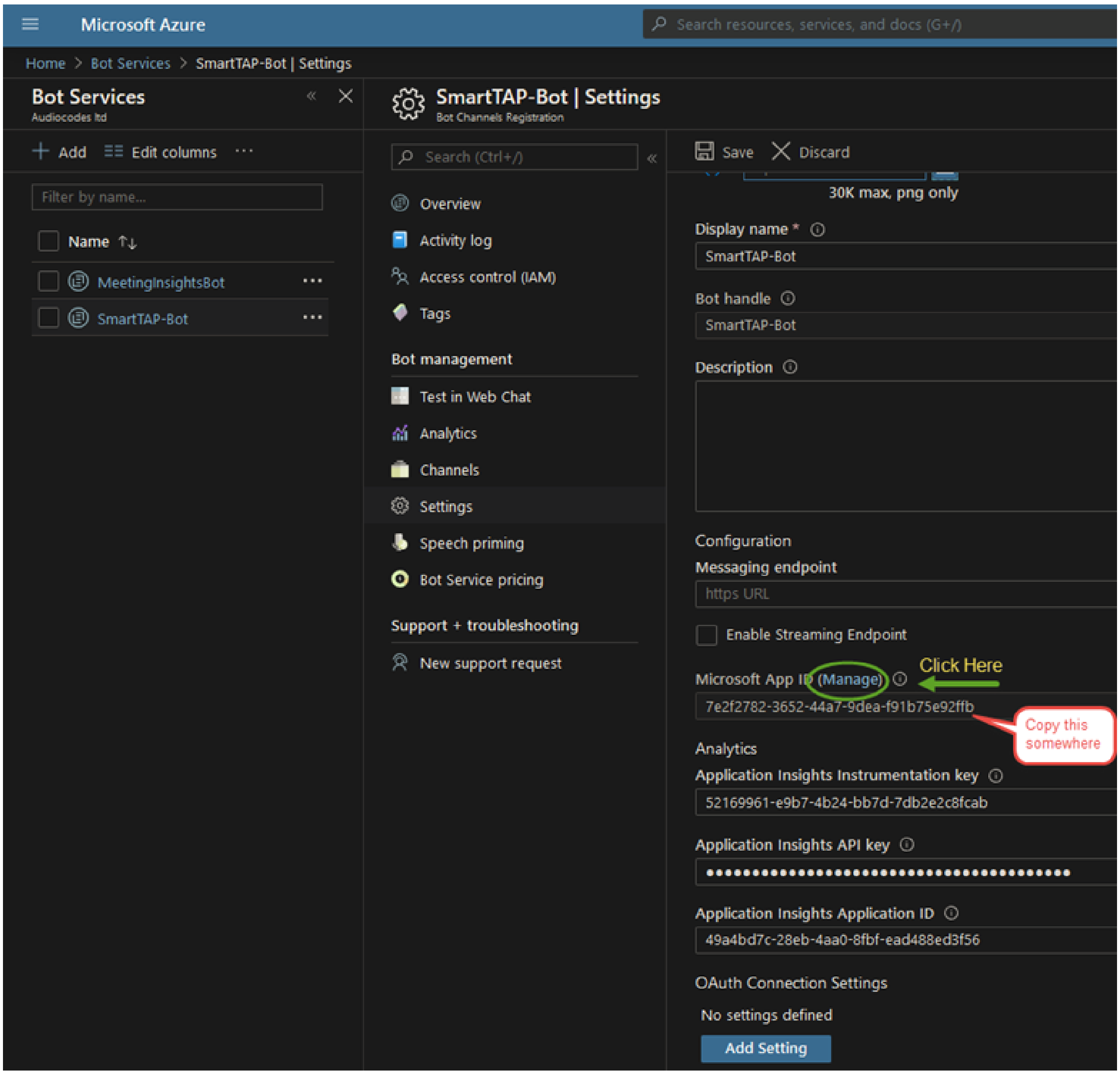
Configure Service Channel

This procedure describes how to configure the service channel.

➢ To configure the service channel:

1. Click **Edit**.

Figure 8-9: Settings



Display name * SmartTAP-Bot

Bot handle SmartTAP-Bot

Description

Configuration

Messaging endpoint https URL

□ Enable Streaming Endpoint

Microsoft App ID (Manage) 7e2f2782-3652-44a7-9dea-f91b75e92ffb

Click Here

Copy this somewhere

Analytics

Application Insights Instrumentation key 52169961-e9b7-4b24-bb7d-7db2e2c8fcab

Application Insights API key ●●

Application Insights Application ID 49a4bd7c-28eb-4aa0-8fbf-ead488ed3f56

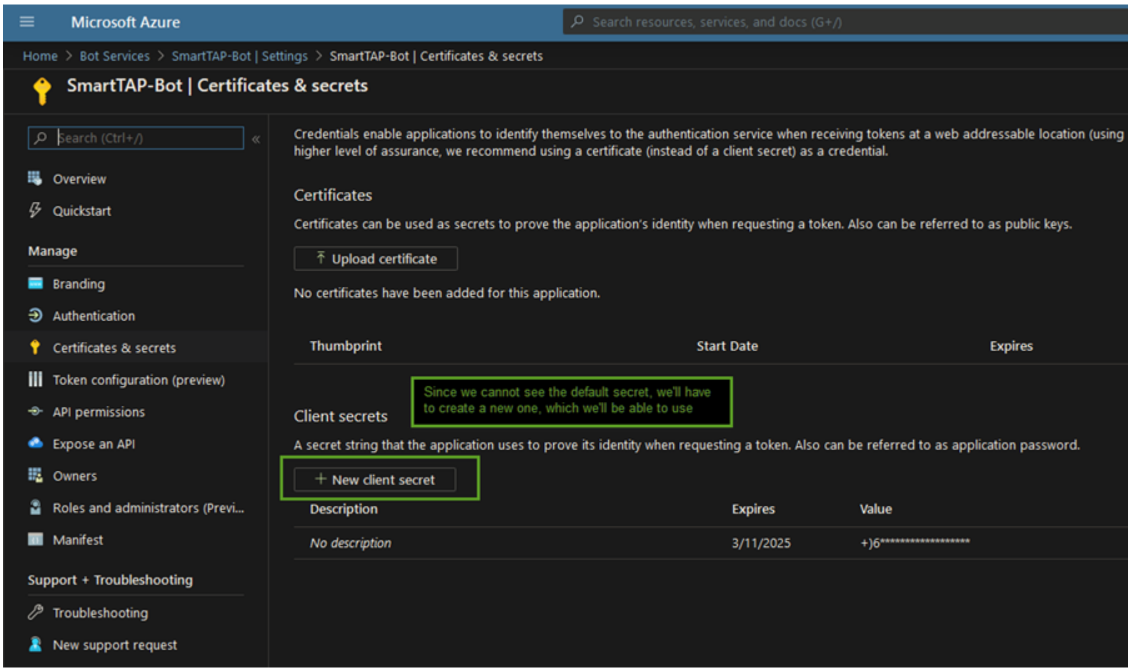
OAuth Connection Settings

No settings defined

Add Setting

2. Click **Manage** to configure the Microsoft App ID.

Figure 8-10: Certificates and Secrets


Microsoft Azure
Search resources, services, and docs (G+/)
Home > Bot Services > SmartTAP-Bot | Settings > SmartTAP-Bot | Certificates & secrets
Search (Ctrl+/)
Overview
Quickstart
Manage
Branding
Authentication
Certificates & secrets
Token configuration (preview)
API permissions
Expose an API
Owners
Roles and administrators (Previ...
Manifest
Support + Troubleshooting
Troubleshooting
New support request
Upload certificate
New client secret

SmartTAP-Bot | Certificates & secrets

Credentials enable applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Certificates

Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be referred to as public keys.

No certificates have been added for this application.

Thumbprint	Start Date	Expires

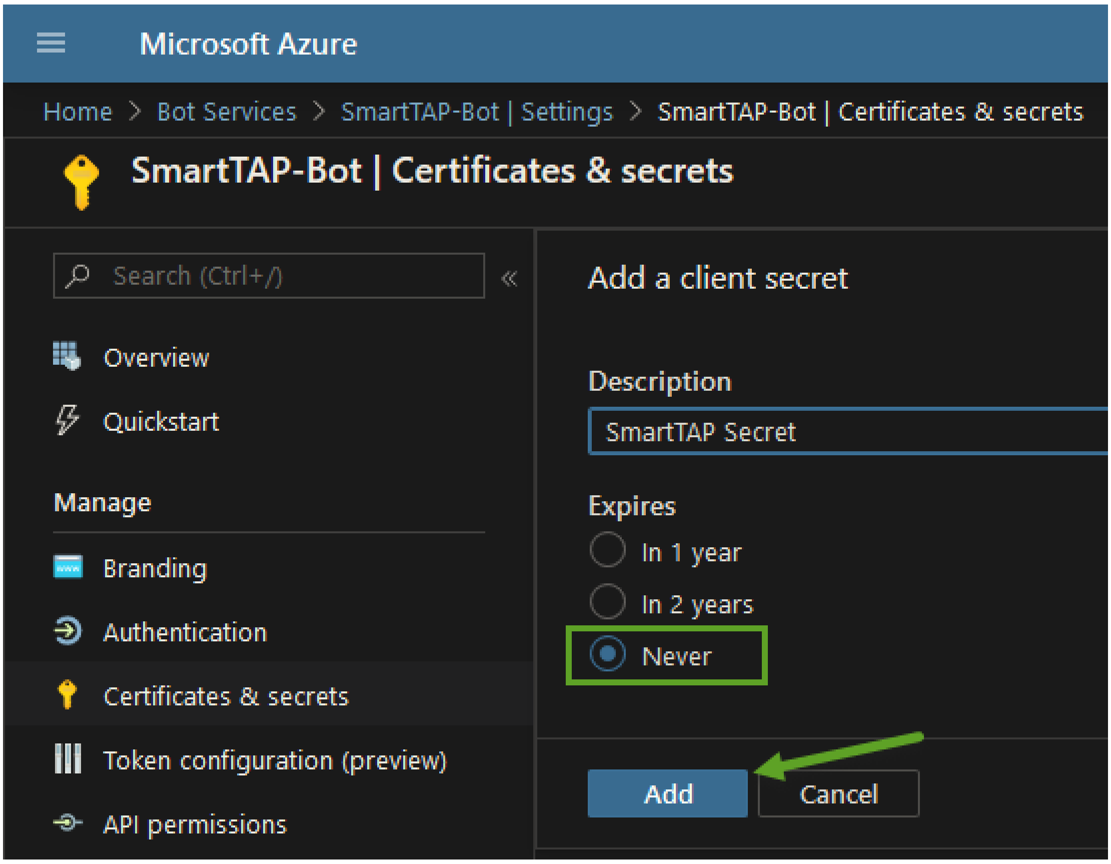
Since we cannot see the default secret, we'll have to create a new one, which we'll be able to use

Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

Description	Expires	Value
No description	3/11/2025	+)6*****************

3. Click **New client secret** to create a new APP secret.



SmartTAP-Bot | Certificates & secrets

Add a client secret

Description: SmartTAP Secret

Expires

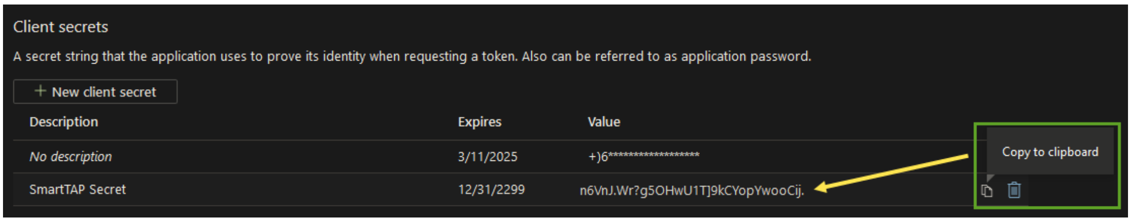
○ In 1 year

○ In 2 years

● Never

4. In the Expires pane, select **Never** and then click **Add**.

Figure 8-11: Client Secrets



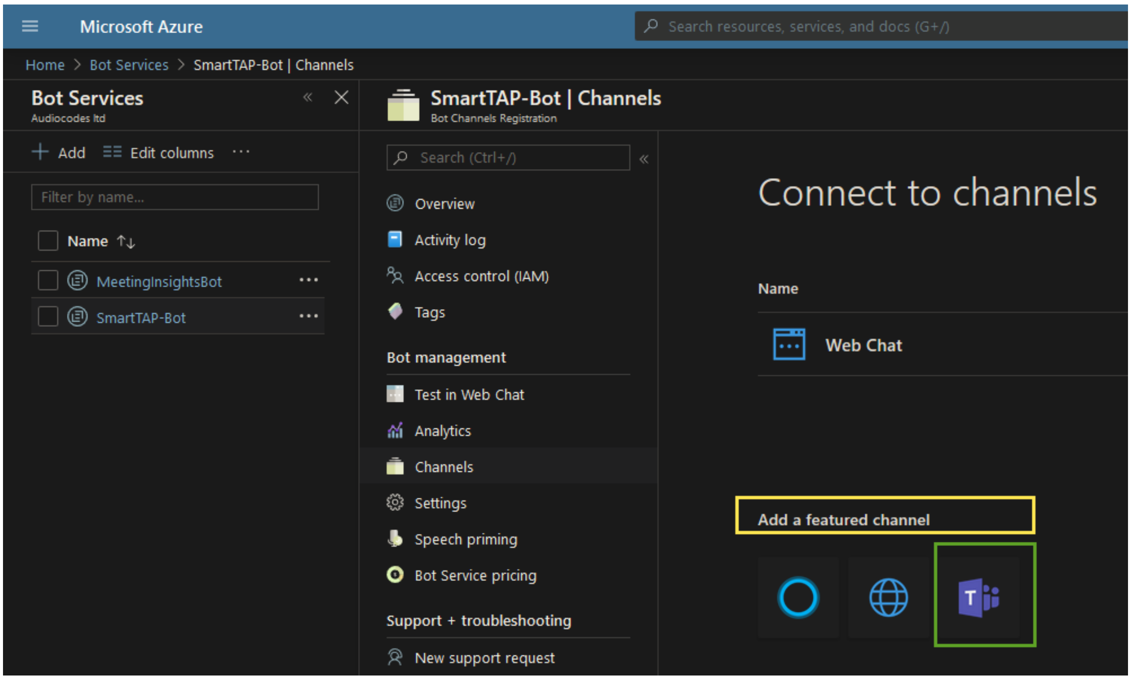
Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

Description	Expires	Value
No description	3/11/2025	+)6******************
SmartTAP Secret	12/31/2299	n6VnJ.Wr?g5OHwU1TJ9kCYopYwooCij.

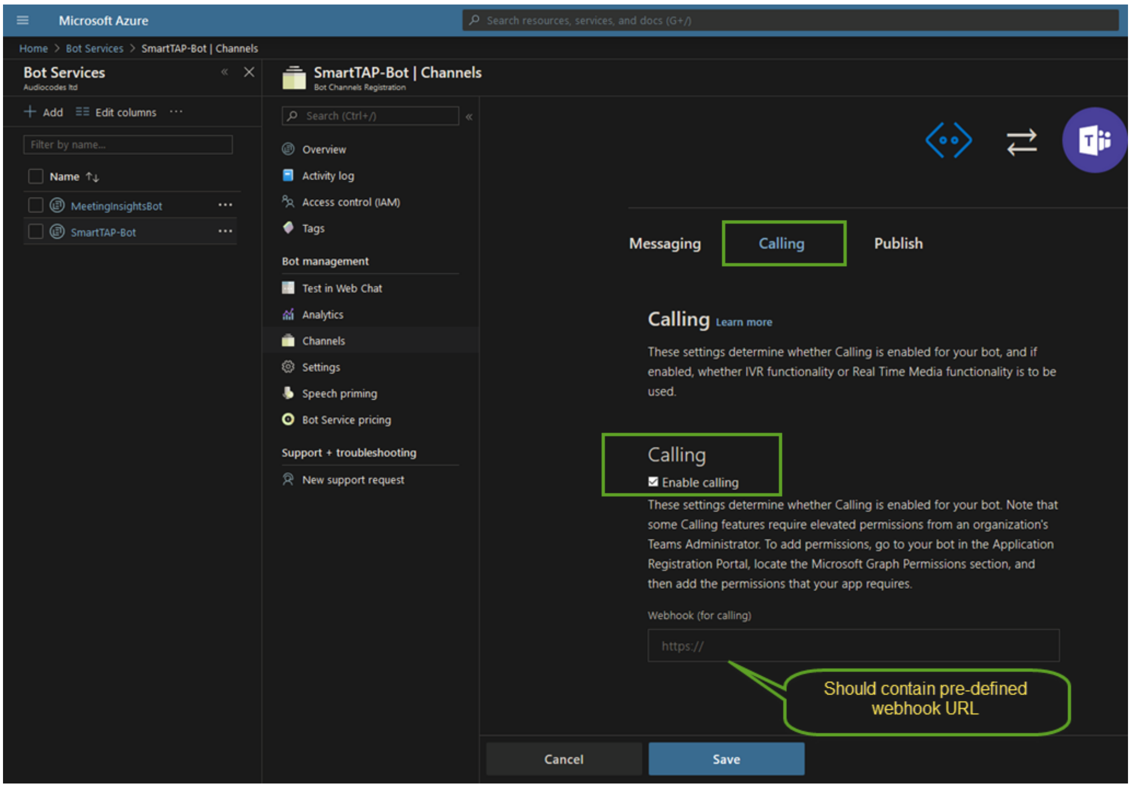
5. Copy the SmartTAP 360° Secret to the clipboard or notepad as it must be configured in a later procedure.
6. Open the Channels screen (**Home** > **BoT Services** > **SmartTAP 360°-BoT** > **Channels**).
7. Select **Add a featured Channel** > **Teams** icon.

Figure 8-12: Teams Feature



8. Click the **Calling** tab.

Figure 8-13: Calling Option



Calling Learn more

These settings determine whether Calling is enabled for your bot, and if enabled, whether IVR functionality or Real Time Media functionality is to be used.

Calling

☑ Enable calling

These settings determine whether Calling is enabled for your bot. Note that some Calling features require elevated permissions from an organization's Teams Administrator. To add permissions, go to your bot in the Application Registration Portal, locate the Microsoft Graph Permissions section, and then add the permissions that your app requires.

Webhook (for calling)

https://

Should contain pre-defined webhook URL

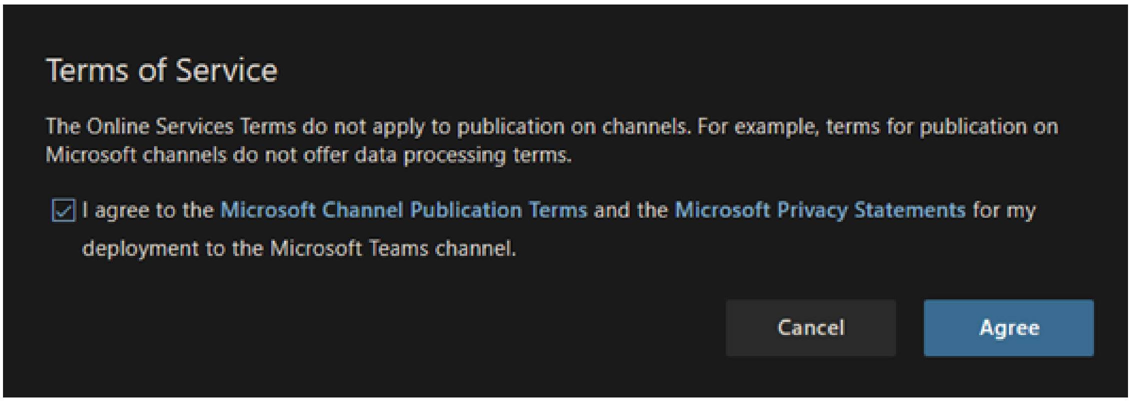
9. Select the 'Enable Calling' check box.

10. Paste the pre-defined webhook URL as follows:

    https://<Service Fabric Cluster FQDN>:9441/api/calls

    Where the URL is the service fabric DNS name given to the Service Fabric Cluster admin (see Prerequisites on page 7).

Figure 8-14: Terms of Service



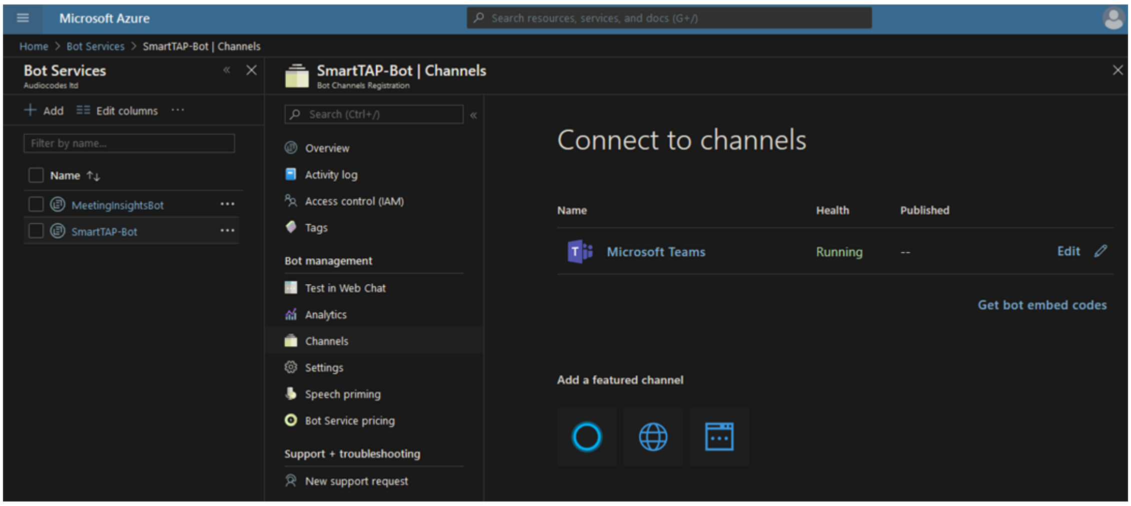
Terms of Service

The Online Services Terms do not apply to publication on channels. For example, terms for publication on Microsoft channels do not offer data processing terms.

☑ I agree to the Microsoft Channel Publication Terms and the Microsoft Privacy Statements for my deployment to the Microsoft Teams channel.

11. Click **Agree** to agree to the terms of service.

Figure 8-15: Connect to Channels



Connect to channels

Name	Health	Published	
Microsoft Teams	Running	--	Edit

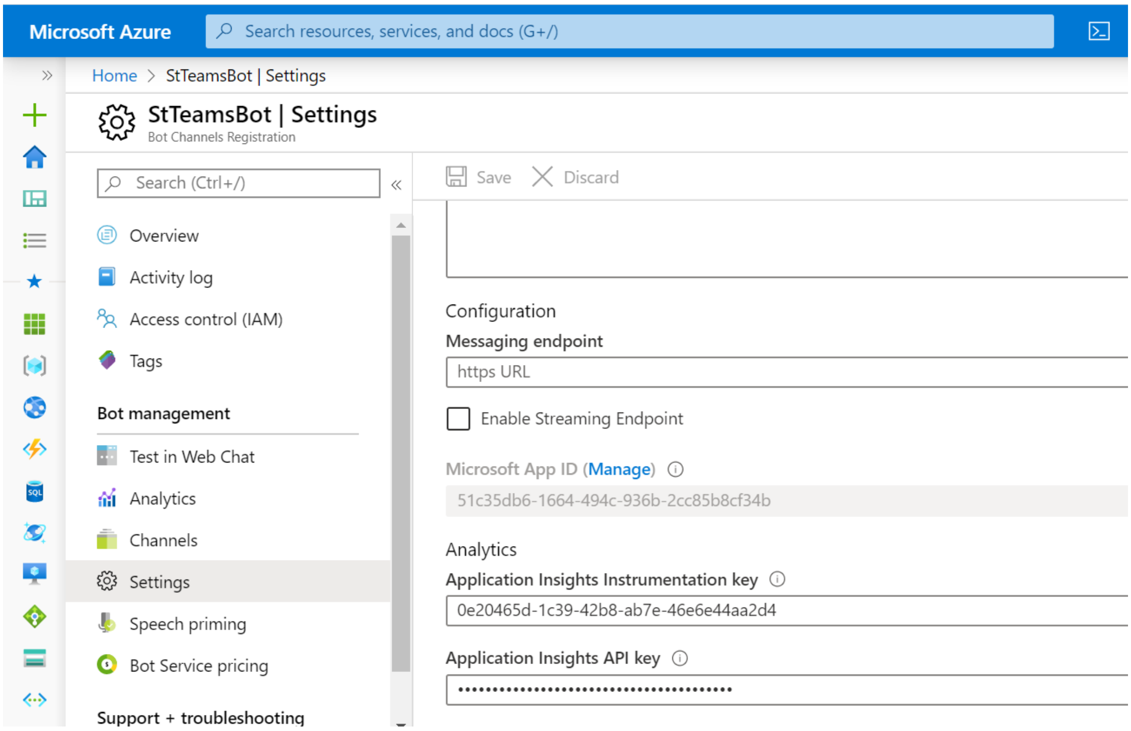
Grant API Permissions to BOT Service

This procedure describes how to grant API permissions to the BOT service.

➢ To grant API permissions to the BOT service:

1. In the Azure portal, open the Settings page (**Home** > **StTeamsBOT** > **Settings**).

Figure 8-16: BOT Settings



Configuration

Messaging endpoint: https URL

☐ Enable Streaming Endpoint

Microsoft App ID (Manage): 51c35db6-1664-494c-936b-2cc85b8cf34b

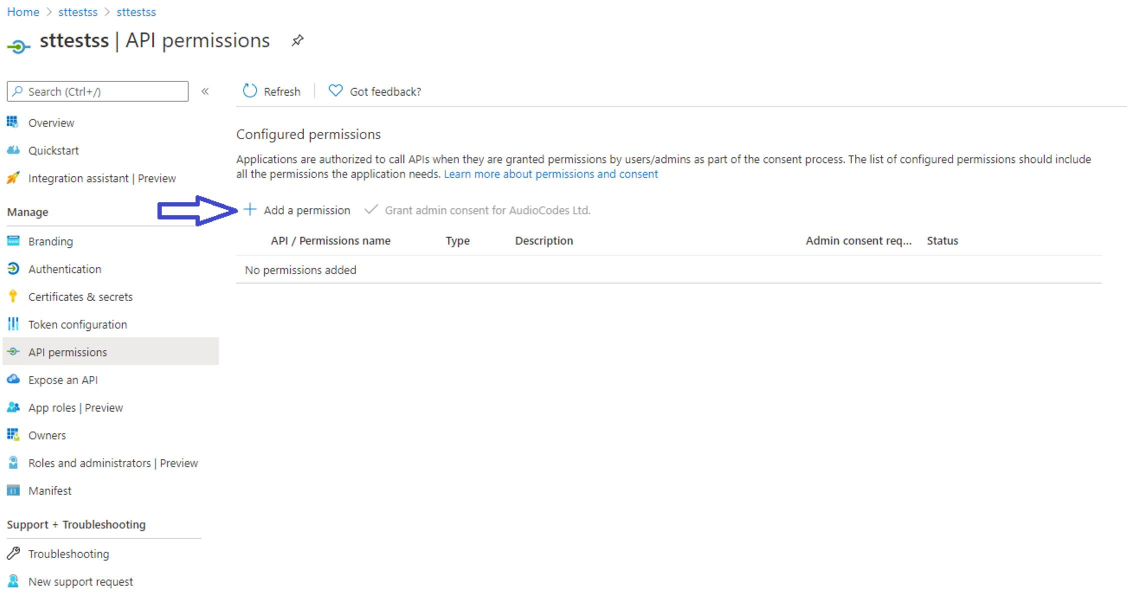
Analytics

Application Insights Instrumentation key: 0e20465d-1c39-42b8-ab7e-46e6e44aa2d4

Application Insights API key: ••••••••••••••••••••••••••••••••••••••

2. Open the Request API Permissions screen (**Manage** > **API** permissions > **Add a Permission**).

Figure 8-17: Add Permissions



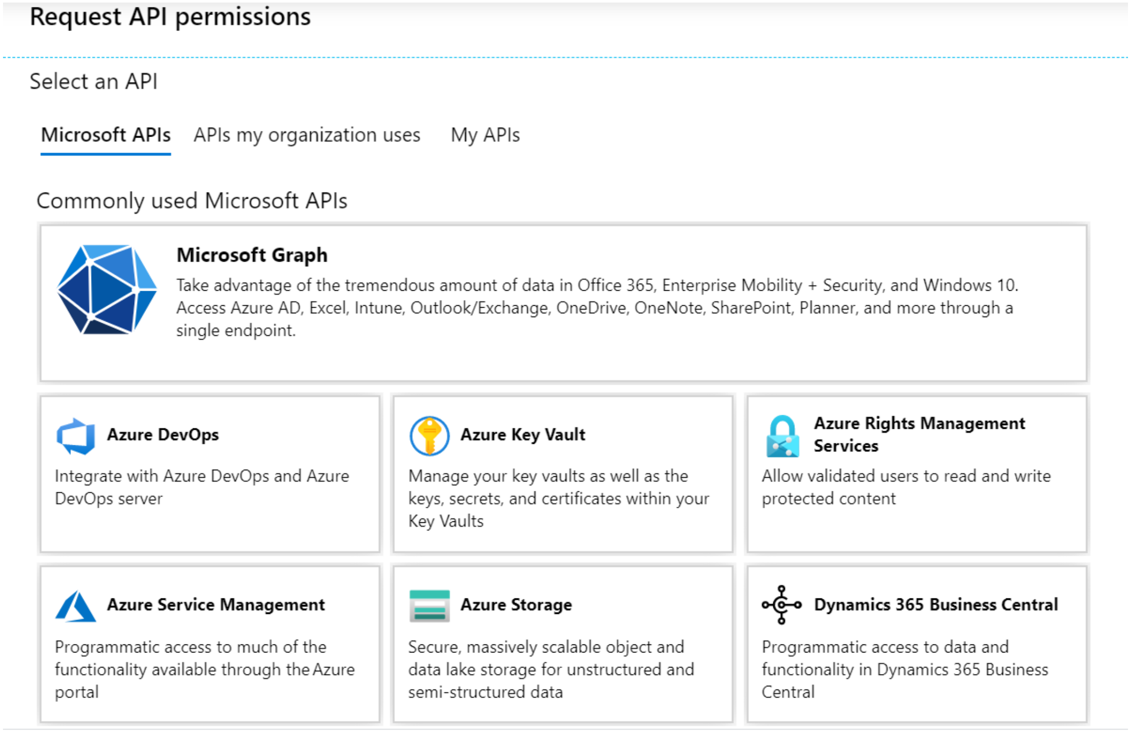
Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. Learn more about permissions and consent

API / Permissions name	Type	Description	Admin consent req...	Status
No permissions added				

The following screen is displayed:

Figure 8-18: Request API Permissions



Microsoft Graph
Take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, and Windows 10. Access Azure AD, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner, and more through a single endpoint.

Azure DevOps
Integrate with Azure DevOps and Azure DevOps server

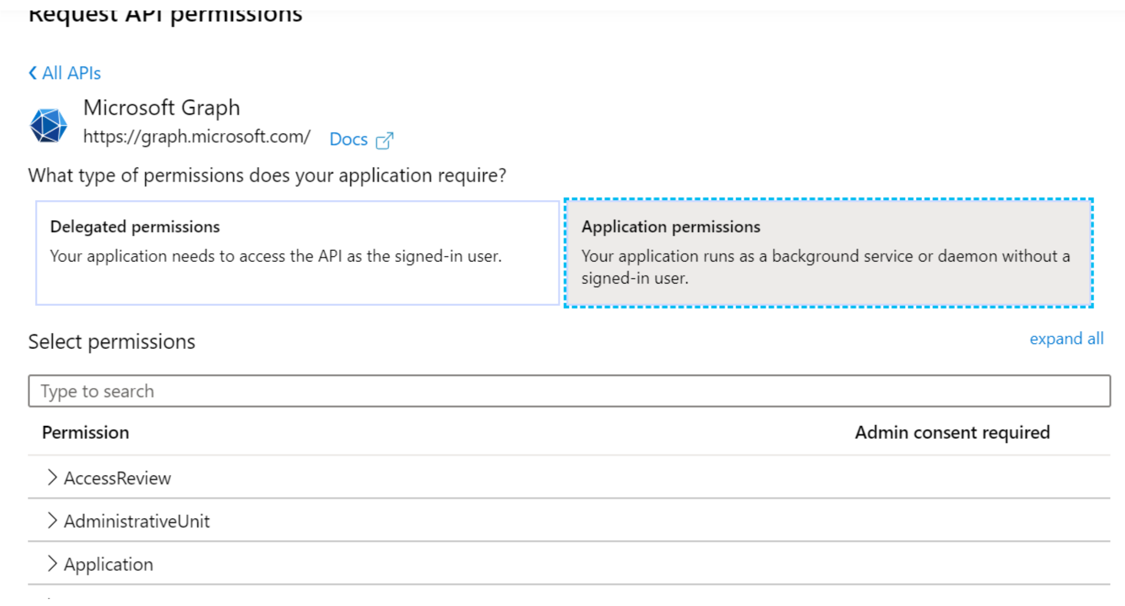
Azure Key Vault
Manage your key vaults as well as the keys, secrets, and certificates within your Key Vaults

Azure Rights Management Services
Allow validated users to read and write protected content

Azure Service Management
Programmatic access to much of the functionality available through the Azure portal

Azure Storage
Secure, massively scalable object and data lake storage for unstructured and semi-structured data

Dynamics 365 Business Central
Programmatic access to data and functionality in Dynamics 365 Business Central



Request API permissions

< All APIs

Microsoft Graph
https://graph.microsoft.com/ Docs

What type of permissions does your application require?

Delegated permissions
Your application needs to access the API as the signed-in user.

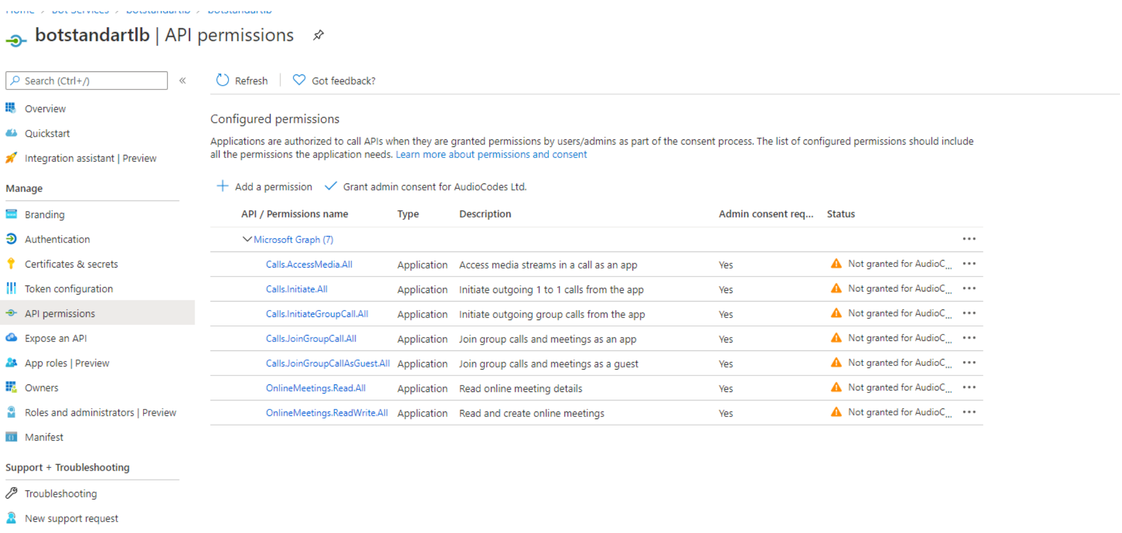
Application permissions
Your application runs as a background service or daemon without a signed-in user.

Select permissions

Permission	Admin consent required
> AccessReview	
> AdministrativeUnit	
> Application	

3. Add the listed permissions below and grant admin consent.

Figure 8-19: Listed API Permissions



botstandartlb | API permissions

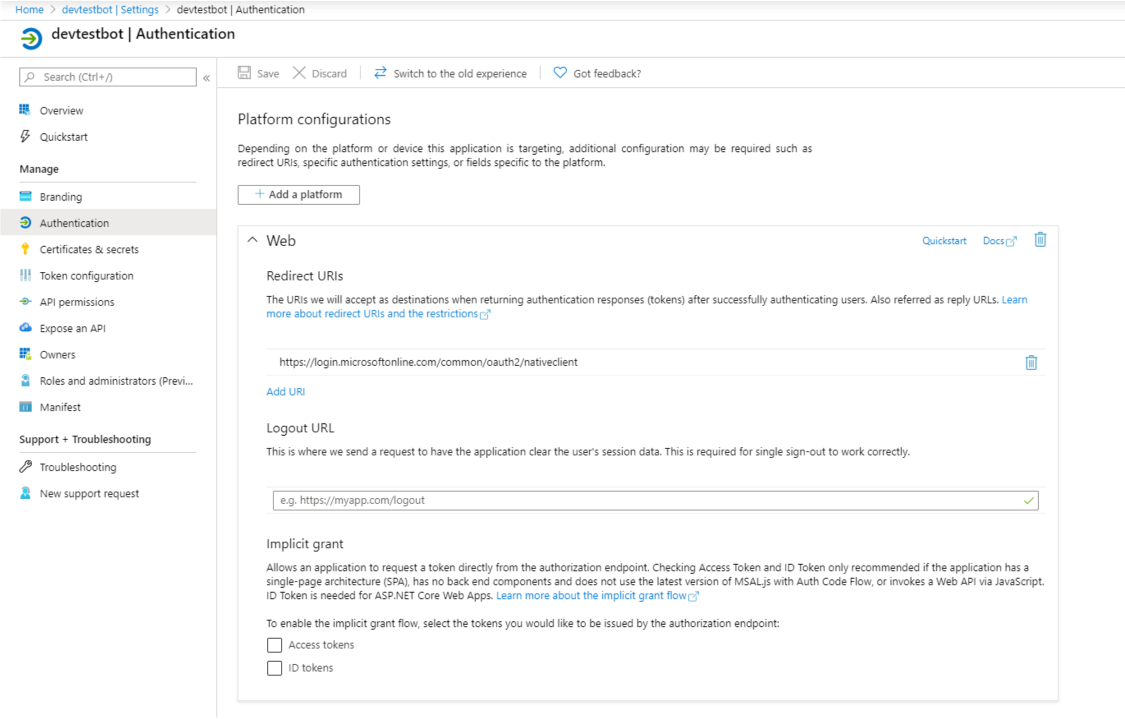
Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. Learn more about permissions and consent

API / Permissions name	Type	Description	Admin consent req...	Status
∨ Microsoft Graph (7)				
Calls.AccessMedia.All	Application	Access media streams in a call as an app	Yes	⚠ Not granted for AudioC...
Calls.Initiate.All	Application	Initiate outgoing 1 to 1 calls from the app	Yes	⚠ Not granted for AudioC...
Calls.InitiateGroupCall.All	Application	Initiate outgoing group calls from the app	Yes	⚠ Not granted for AudioC...
Calls.JoinGroupCall.All	Application	Join group calls and meetings as an app	Yes	⚠ Not granted for AudioC...
Calls.JoinGroupCallAsGuest.All	Application	Join group calls and meetings as a guest	Yes	⚠ Not granted for AudioC...
OnlineMeetings.Read.All	Application	Read online meeting details	Yes	⚠ Not granted for AudioC...
OnlineMeetings.ReadWrite.All	Application	Read and create online meetings	Yes	⚠ Not granted for AudioC...

4. Open the Authentication screen (**Home** > **<Botname> Settings** > <Botname> **Authentication**).

Figure 8-20: Authentication



Home > devtestbot | Settings > devtestbot | Authentication

devtestbot | Authentication

Platform configurations

Depending on the platform or device this application is targeting, additional configuration may be required such as redirect URIs, specific authentication settings, or fields specific to the platform.

Web

Redirect URIs

The URIs we will accept as destinations when returning authentication responses (tokens) after successfully authenticating users. Also referred as reply URLs. Learn more about redirect URIs and the restrictions

https://login.microsoftonline.com/common/oauth2/nativeclient

Logout URL

This is where we send a request to have the application clear the user's session data. This is required for single sign-out to work correctly.

e.g. https://myapp.com/logout

Implicit grant

Allows an application to request a token directly from the authorization endpoint. Checking Access Token and ID Token only recommended if the application has a single-page architecture (SPA), has no back end components and does not use the latest version of MSAL.js with Auth Code Flow, or invokes a Web API via JavaScript. ID Token is needed for ASP.NET Core Web Apps. Learn more about the implicit grant flow

To enable the implicit grant flow, select the tokens you would like to be issued by the authorization endpoint:

☐ Access tokens

☐ ID tokens

5. Copy the following link and paste it in the redirect URL: https://login.microsoftonline.com/common/oauth2/nativeclient

   Where 'nativeclient' is the SmartTAP 360° Bot app ID from BOT service that was created in Configure Service Channel on page 40. This is required to authenticate your Azure subscription.

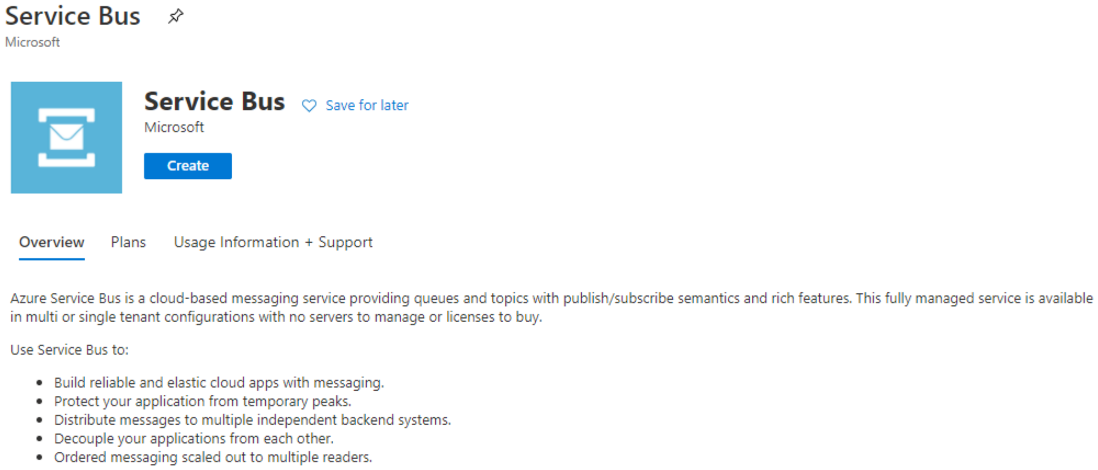
9 Step 4 Create Service Bus

This section describes how to create a service bus.

➢ To create a service bus:

1. Under the same resource group, add a new resource (**service bus** > **Create**).

Figure 9-1: Create Service Bus



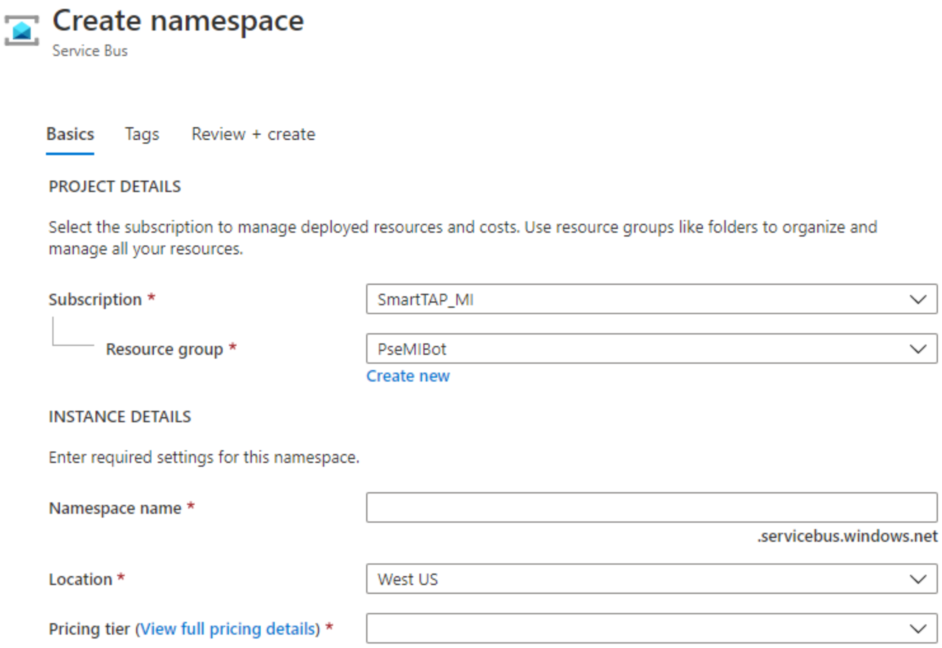
Azure Service Bus is a cloud-based messaging service providing queues and topics with publish/subscribe semantics and rich features. This fully managed service is available in multi or single tenant configurations with no servers to manage or licenses to buy.

Use Service Bus to:

- Build reliable and elastic cloud apps with messaging.
- Protect your application from temporary peaks.
- Distribute messages to multiple independent backend systems.
- Decouple your applications from each other.
- Ordered messaging scaled out to multiple readers.

The following screen is displayed:

Figure 9-2: Create NameSpace



Create namespace

Service Bus

PROJECT DETAILS

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * SmartTAP_MI

Resource group * PseMIBot

INSTANCE DETAILS

Enter required settings for this namespace.

Namespace name *

.servicebus.windows.net

Location * West US

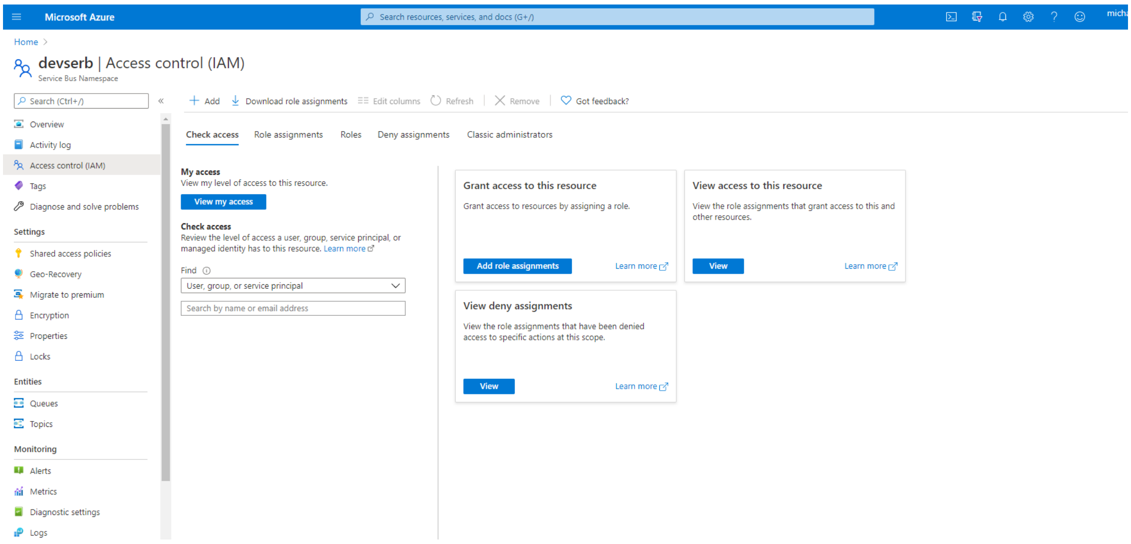
Pricing tier (View full pricing details) *

2. From the Pricing tier drop-down list, choose ‘STANDARD’

3. Create the service bus.

4. Go to **Access Control** > **Add role assignment**.

Figure 9-3: Add Role Assignment



My access

View my level of access to this resource.

Check access

Review the level of access a user, group, service principal, or managed identity has to this resource. Learn more

Find: User, group, or service principal

Search by name or email address

Grant access to this resource

Grant access to resources by assigning a role.

View access to this resource

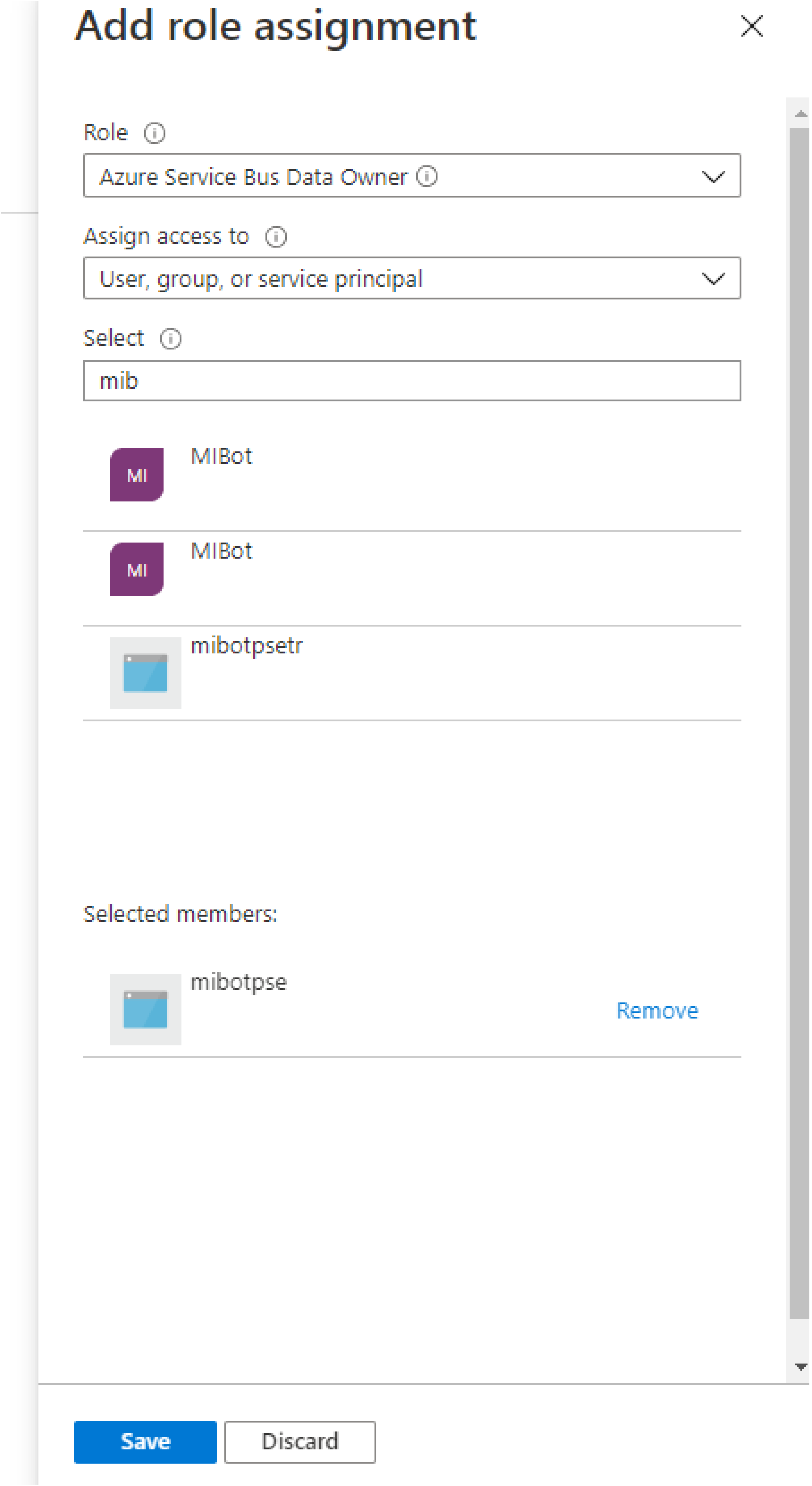
View the role assignments that grant access to this and other resources.

View deny assignments

View the role assignments that have been denied access to specific actions at this scope.

5. Select the following roles ‘Azure service bus data owner’, assign access to, choose ‘Azure AD user, group, or service principle’, select: here look for relevant bot app registration, and then click **Save**.

Figure 9-4: Add Role Assignment



Add role assignment

Role: Azure Service Bus Data Owner

Assign access to: User, group, or service principal

Select: mib

MIBot

MIBot

mibotpsetr

Selected members:

mibotpse

10 Step 5 Deploy BOT Package on Service Fabric Cluster

This procedure describes how to deploy BOT Package on the Service Fabric Cluster on the local machine or from inside one of the cluster nodes including the following procedures:

1. Prepare Local Machine for Deployment on Service Fabric below
2. Deploy BOT Package on page 59

Prepare Local Machine for Deployment on Service Fabric

This procedure describes how to prepare the local virtual machine for deployment on the Service Fabric.

➢ To prepare machine for deployment on service fabric:

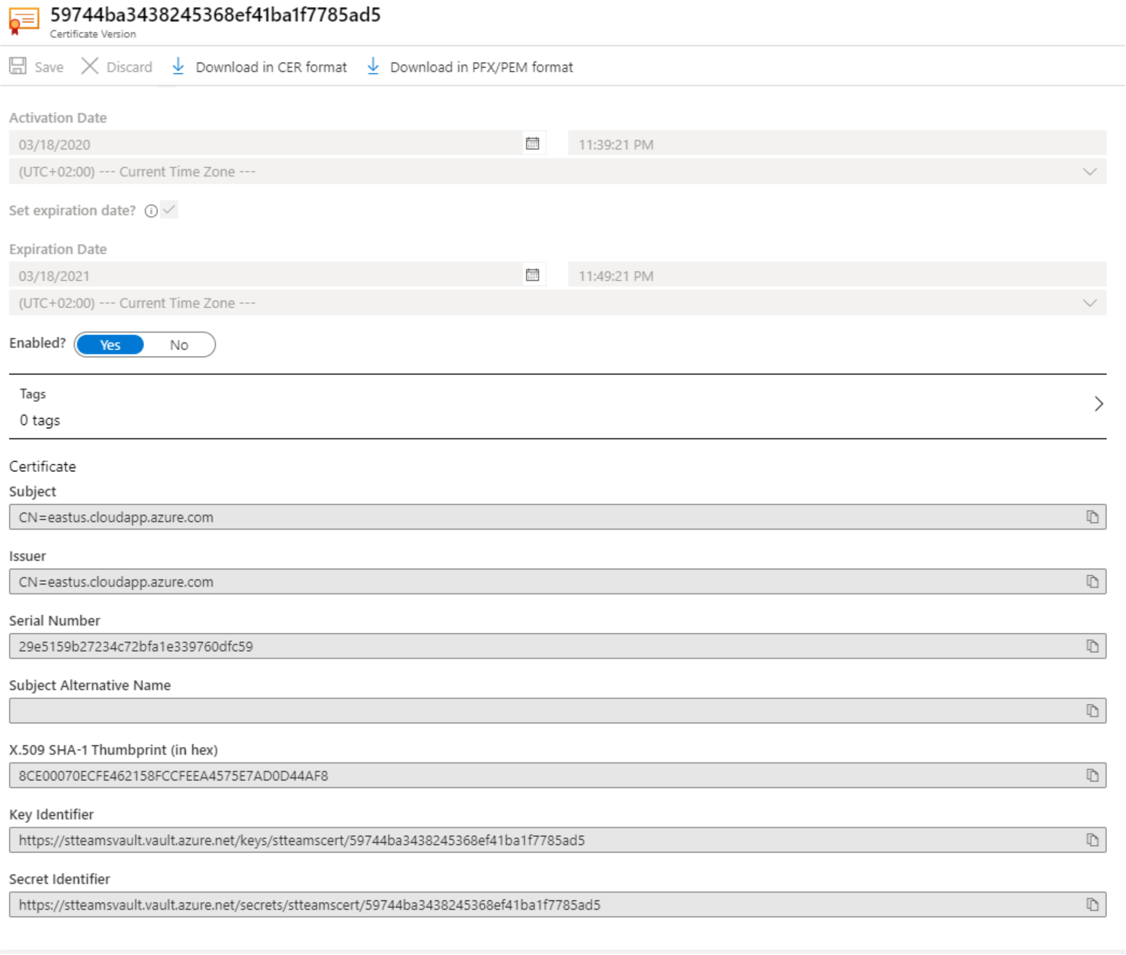
1. Extract the SFC Deployment script package from the following location to your local machine:
   …\Release\Publish\STTeamsBOT_Deployment_Package\BotDeploymentScript:

   This directory includes the following files:

   - connectArgs.psd1
   - deployBOT.ps1
2. Enable PowerShell script execution:

   ```
   PS .:\> “Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Force -Scope
   CurrentUser”
   ```
3. Install Azure SDK for Service Fabric.
4. Download generated certificate (see Prerequisites on page 7) in PFX/PEM format to the local machine.

Figure 10-1: Download Certificate



59744ba3438245368ef41ba1f7785ad5
Certificate Version

Activation Date
03/18/2020 11:39:21 PM
(UTC+02:00) --- Current Time Zone ---

Set expiration date? ✓

Expiration Date
03/18/2021 11:49:21 PM
(UTC+02:00) --- Current Time Zone ---

Enabled? [✓ Yes] No

Tags
0 tags

Certificate

Subject
CN=eastus.cloudapp.azure.com

Issuer
CN=eastus.cloudapp.azure.com

Serial Number
29e5159b27234c72bfa1e339760dfc59

Subject Alternative Name

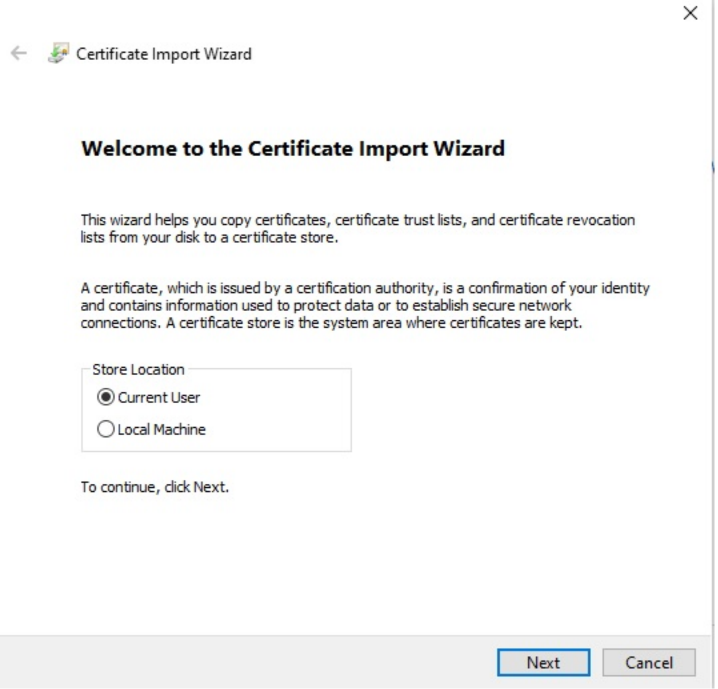
X.509 SHA-1 Thumbprint (in hex)
8CE00070ECFE462158FCCFEEA4575E7AD0D44AF8

Key Identifier
https://stteamsvault.vault.azure.net/keys/stteamscert/59744ba3438245368ef41ba1f7785ad5

Secret Identifier
https://stteamsvault.vault.azure.net/secrets/stteamscert/59744ba3438245368ef41ba1f7785ad5

5. Install the certificate to personal store.

Figure 10-2: Certificate Import Wizard



Certificate Import Wizard

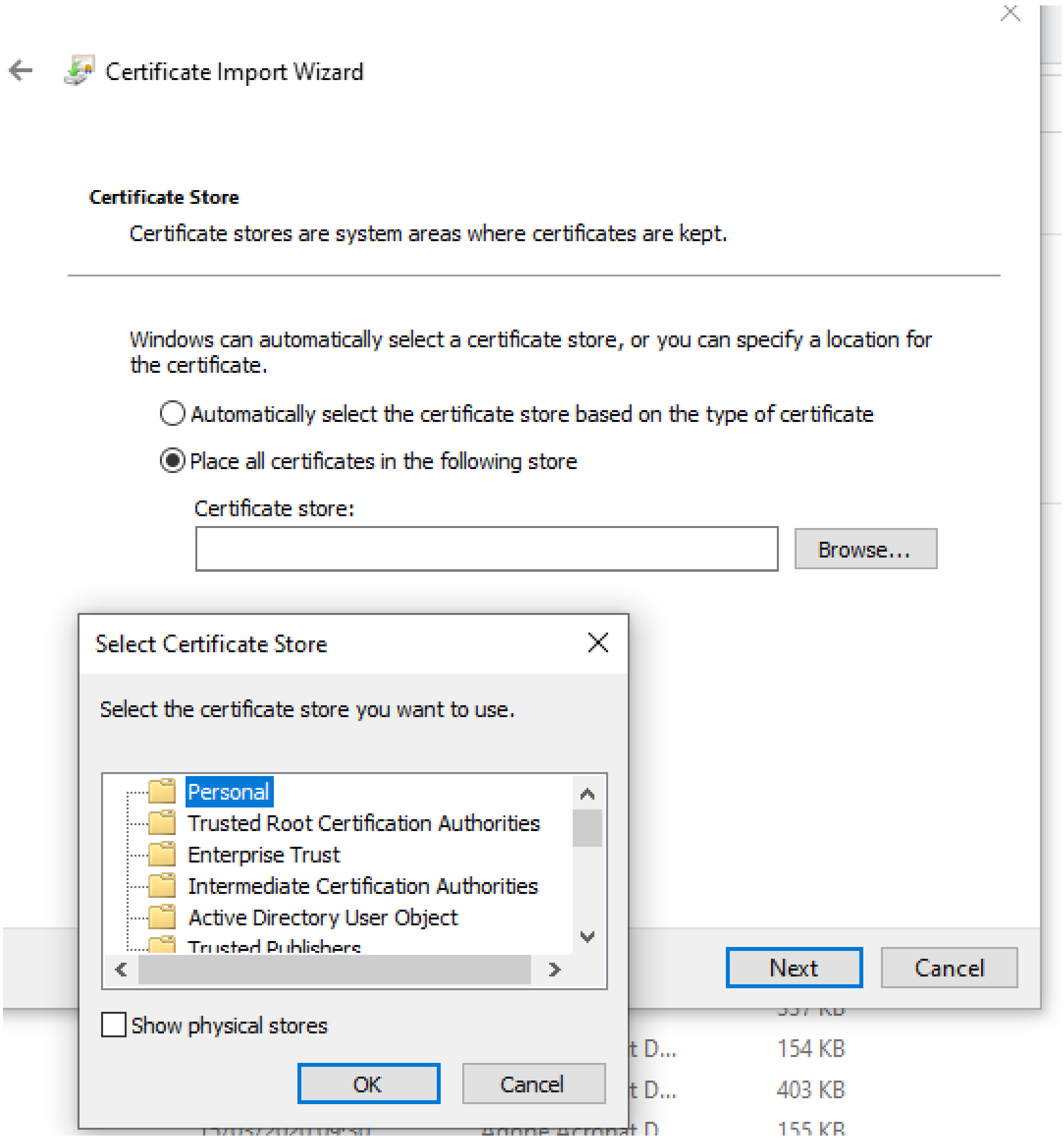
Welcome to the Certificate Import Wizard

This wizard helps you copy certificates, certificate trust lists, and certificate revocation lists from your disk to a certificate store.

A certificate, which is issued by a certification authority, is a confirmation of your identity and contains information used to protect data or to establish secure network connections. A certificate store is the system area where certificates are kept.

Store Location
● Current User
○ Local Machine

To continue, click Next.



Certificate Import Wizard

Certificate Store

Certificate stores are system areas where certificates are kept.

Windows can automatically select a certificate store, or you can specify a location for the certificate.

○ Automatically select the certificate store based on the type of certificate

● Place all certificates in the following store

Certificate store:

Select Certificate Store

Select the certificate store you want to use.

- Personal
- Trusted Root Certification Authorities
- Enterprise Trust
- Intermediate Certification Authorities
- Active Directory User Object
- Trusted Publishers

□ Show physical stores

6. Using a text editor, update the connectArgs.psd1 file (from the BOT Deployment script package) as highlighted below:

```
ConnectionEndpoint = '<AzureFQDN:port>'
```

```
ServerCommonName = "<Server Common Name>"
```

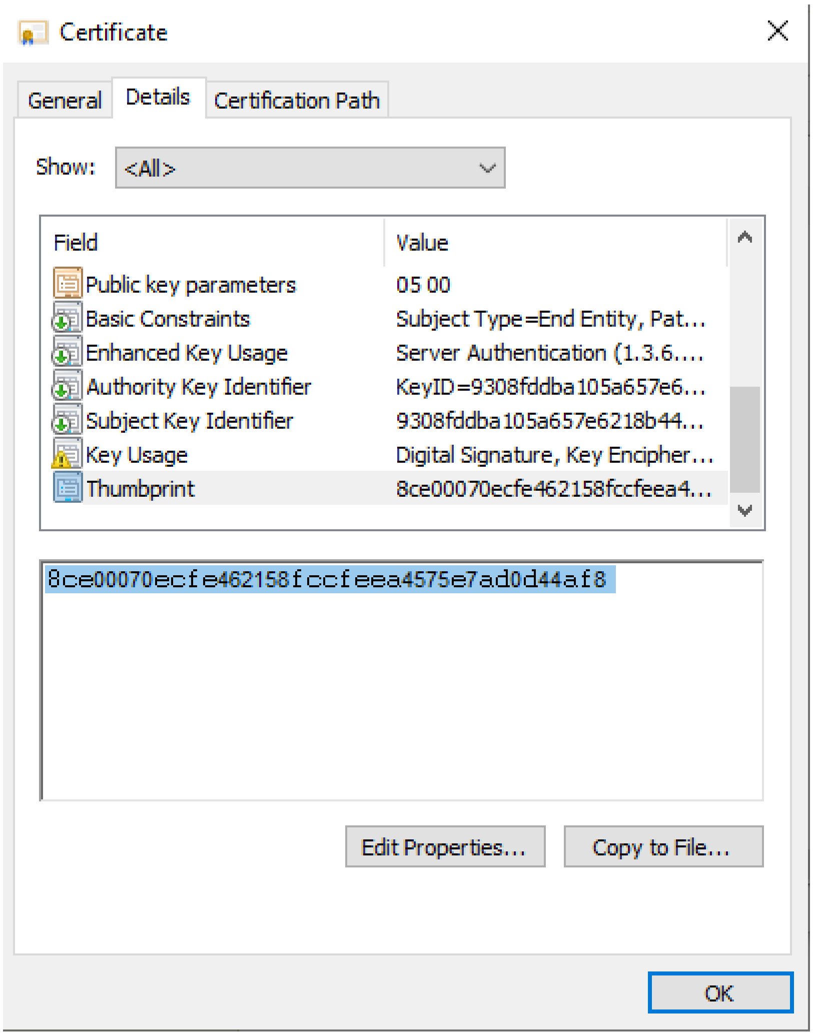
```
FindValue = "<ClientCertificateThumbprintValue>"
```

Where:

- <AzureFQDN:port> is the FQDN of the Microsoft Azure Cloud platform which can be extracted from the client certificate or from the azure portal in Service Fabric Cluster view.

- <Server Common Name> is the common name of the Windows Server that is extracted from the Subject field in the Client certificate example shown in the figure below.
- <ClientCertificateThumbprintValue>" is the thumbprint value that is extracted from the client certificate example below.

Figure 10-3: Client Certificate

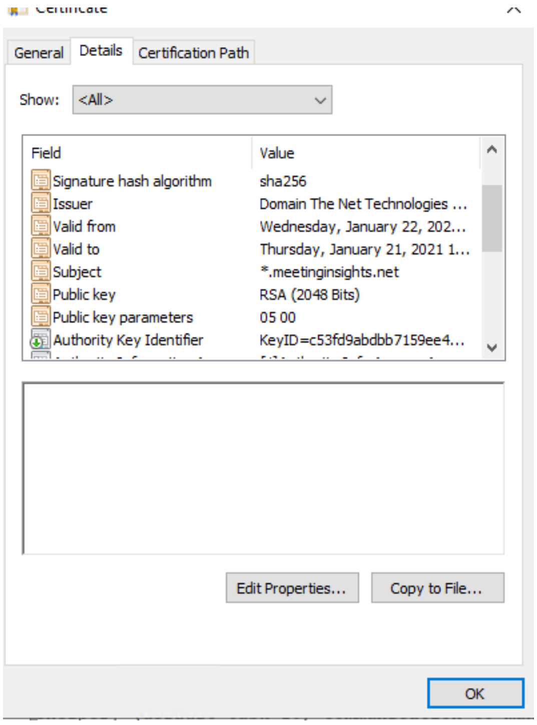


Show: <All>

Field	Value
Public key parameters	05 00
Basic Constraints	Subject Type=End Entity, Pat...
Enhanced Key Usage	Server Authentication (1.3.6....
Authority Key Identifier	KeyID=9308fddba105a657e6...
Subject Key Identifier	9308fddba105a657e6218b44...
Key Usage	Digital Signature, Key Encipher...
Thumbprint	8ce00070ecfe462158fccfeea4...

8ce00070ecfe462158fccfeea4575e7ad0d44af8

Figure 10-4: Server Common Name



Show: <All>

Field	Value
Signature hash algorithm	sha256
Issuer	Domain The Net Technologies ...
Valid from	Wednesday, January 22, 202...
Valid to	Thursday, January 21, 2021 1...
Subject	*.meetinginsights.net
Public key	RSA (2048 Bits)
Public key parameters	05 00
Authority Key Identifier	KeyID=c53fd9abdbb7159ee4...

7. Using a text editor, update ApplicationManifest.xml: ApplicationManifest.xml (..\HueBot\) as highlighted:

```
<Parameter Name="STTeamsBOT_PartitionCount" DefaultValue="3" /># insert the amount of instances in the service fabric cluster
```

8. Using a text editor, update file appsettingsST.json (\STTeamsBOT\STTeamsBOTPkg\Code\AppSettingsTemplates) as highlighted below:

```
"AppId": "53210052-c601-4d74-bfdc-cc3863e9b375", # Taken from Bot service (see image below)
```

"AppSecret": "<Appsecret>", *# App secret copied during Bot channel creation above.*

"BotBaseUrl": "<BotBaseUrl>", *#Created according to BOT DNS with signaling port. For example, https://stteamsbotsfpoc.meetinginsights.net:9444/api/calls*

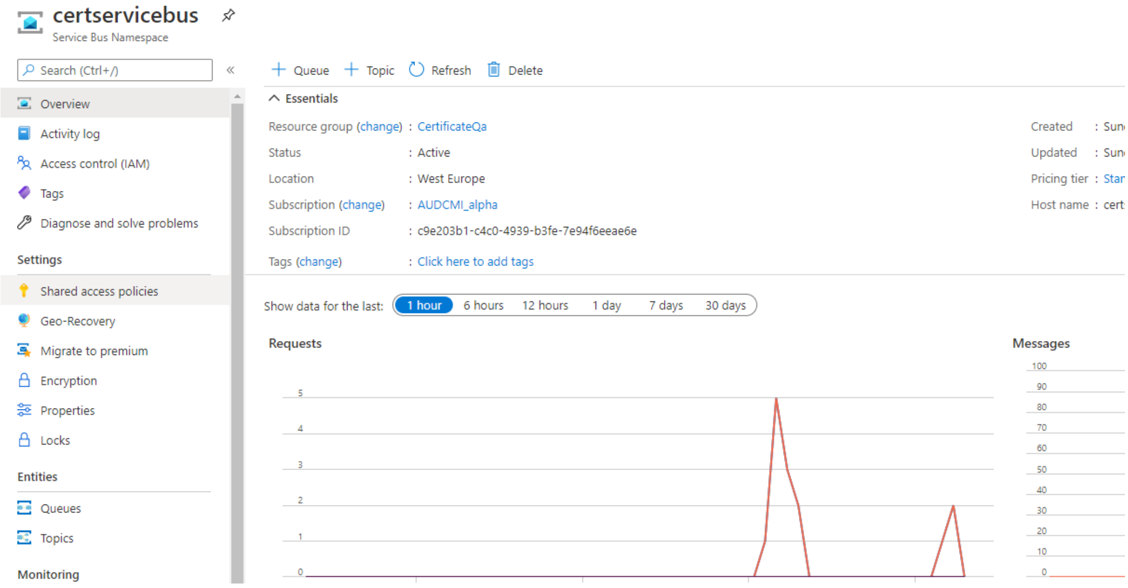
"BotMediaProcessorUrl": "<BotMediaProcessorUrl>", *# For example, port tcp://teamsbotclustersftest.meetinginsights.net:8445*

"Certificate": "<ClientCertificateThumbprint>", *# insert the certificate Thumbprint here*

```
"Deployment": {

"IsLocal": false,

"ServiceFqdn": "<ServiceFqdn>", #DNS record pointing to service cluster.

"EnableBinaryWriter": false,

"App": {

"AppMode": "ST",

 "TenantId": "<TenantId>",

"ApplicationInsights": {

"InstrumentationKey": "<InstrumentationKey>" #set from BOT services, see
example image below.

"BackEndBaseUrl": "http://<SmartTAP_IP or FQDN>/",# set to SmartTAP IP
address or FQDN

"ServiceBusData": {
        "TopicName": "<TopicName>", # is a topic name
        "SBConnStrOrEndpoint": "<SBConnStrOrEndpoint>", -# service bus
connection string. This string can be taken from the Azure portal under Service Bus
> Shared access policies.
        "ServiceBusSubscriptionId": "<ServiceBusSubscriptionId>", # service bus
subscription id. This string can be taken from the Azure portal under Service Bus,
main window.
        "ResourceGroupName": "<ResourceGroupName>", # the name of the
resource group from which the service bus was created.
        "NameSpaceName": "<NameSpaceName>" # Service Bus Namespace that
represents the name of the created service bus resource which can be taken from
the Azure portal.
      }
```

The figures below display examples for Service Bus configuration.

Figure 10-5: Service Bus



Resource group (change) : CertificateQa

Status : Active

Location : West Europe

Subscription (change) : AUDCMI_alpha

Subscription ID : c9e203b1-c4c0-4939-b3fe-7e94f6eeae6e

Tags (change) : Click here to add tags

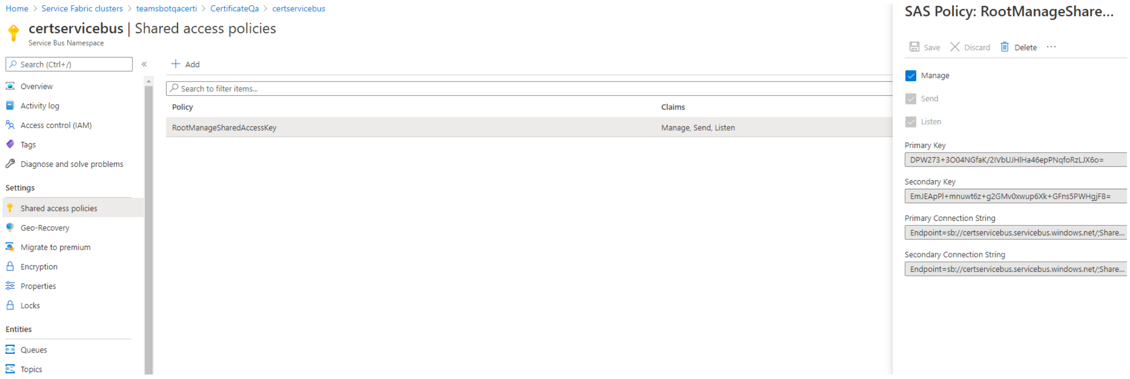
Created : Sun

Updated : Sun

Pricing tier : Star

Host name : cert

Figure 10-6: Shared Access Policies for Service Bus



Policy	Claims
RootManageSharedAccessKey	Manage, Send, Listen

☑ Manage

☑ Send

☑ Listen

Primary Key

DPW273+3O04NGfaK/2IVbUJHlHa46epPNqfoRzLJX6o=

Secondary Key

EmJEApPl+mnuwt6z+g2GMv0xwup6Xk+GFns5PWHgjF8=

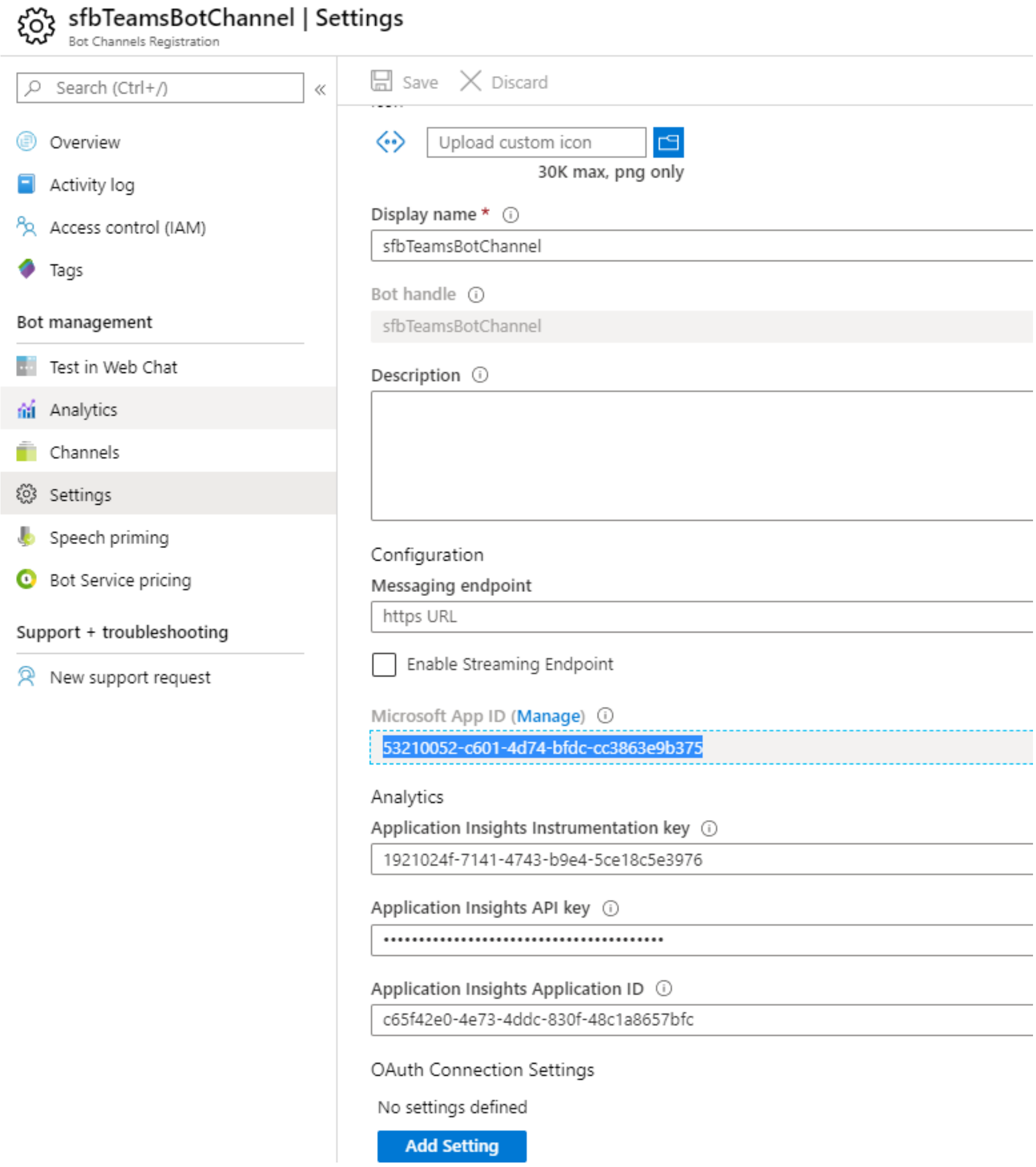
Primary Connection String

Endpoint=sb://certservicebus.servicebus.windows.net/;Share...

Secondary Connection String

Endpoint=sb://certservicebus.servicebus.windows.net/;Share...

Figure 10-7: Configure Microsoft App ID



30K max, png only

Display name * : sfbTeamsBotChannel

Bot handle: sfbTeamsBotChannel

Description:

Configuration

Messaging endpoint: https URL

□ Enable Streaming Endpoint

Microsoft App ID (Manage): 53210052-c601-4d74-bfdc-cc3863e9b375

Analytics

Application Insights Instrumentation key: 1921024f-7141-4743-b9e4-5ce18c5e3976

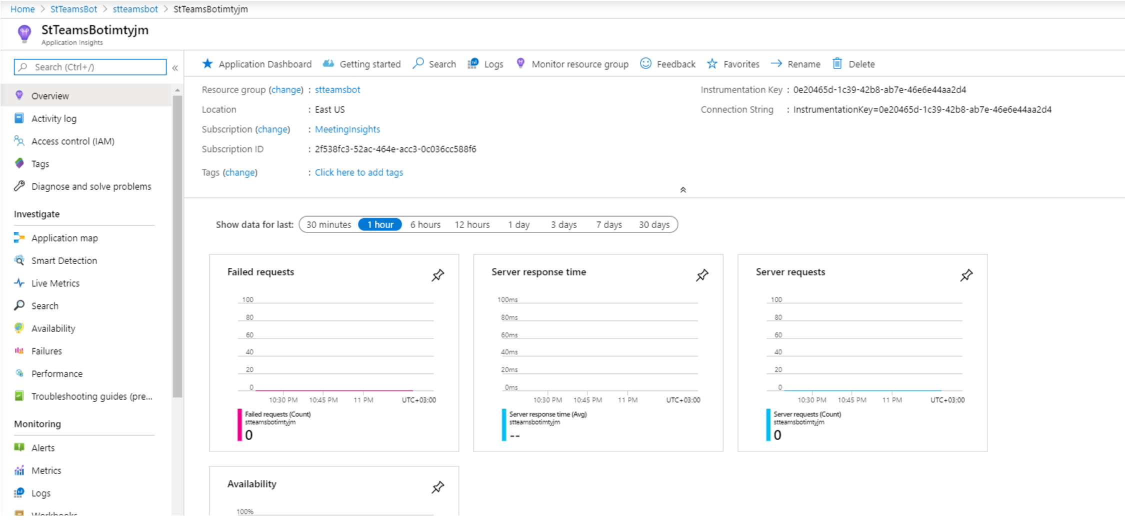
Application Insights API key: ••

Application Insights Application ID: c65f42e0-4e73-4ddc-830f-48c1a8657bfc

OAuth Connection Settings

No settings defined

Figure 10-8: Configure Instrumentation Key



Resource group (change) : stteamsbot

Location : East US

Subscription (change) : MeetingInsights

Subscription ID : 2f538fc3-52ac-464e-acc3-0c036cc588f6

Tags (change) : Click here to add tags

Instrumentation Key : 0e20465d-1c39-42b8-ab7e-46e6e44aa2d4

Connection String : InstrumentationKey=0e20465d-1c39-42b8-ab7e-46e6e44aa2d4

Show data for last: 30 minutes | **1 hour** | 6 hours | 12 hours | 1 day | 3 days | 7 days | 30 days

Failed requests (Count) stteamsbotimtyjm 0

Server response time (Avg) stteamsbotimtyjm --

Server requests (Count) stteamsbotimtyjm 0

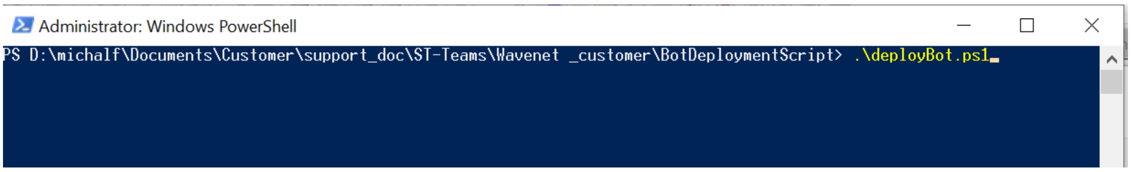
Deploy BOT Package

This procedure describes how to deploy the BOT Package.

➢ To deploy SFC:

1. Run the script deployBOT.ps1 from the folder location to which you extracted this file from the BOT Deployment script package.

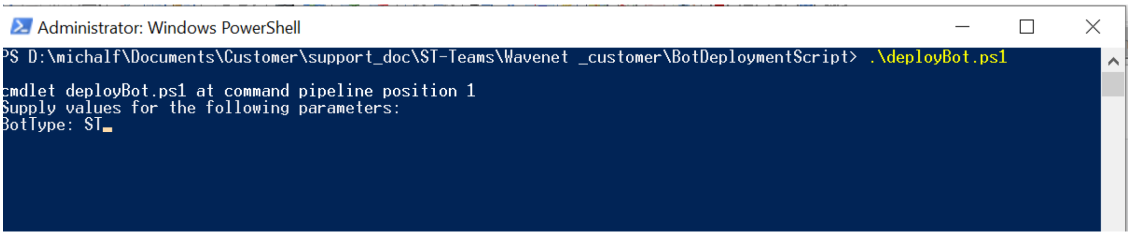
Figure 10-9: Run Script



```
PS D:\michalf\Documents\Customer\support_doc\ST-Teams\Wavenet _customer\BotDeploymentScript> .\deployBot.ps1_
```

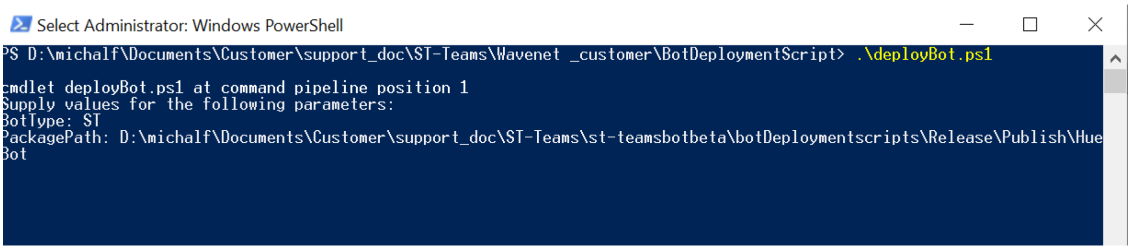
2. Enter BOT type : **ST** and press enter.

Figure 10-10: BOT Type



```
PS D:\michalf\Documents\Customer\support_doc\ST-Teams\Wavenet _customer\BotDeploymentScript> .\deployBot.ps1

cmdlet deployBot.ps1 at command pipeline position 1
Supply values for the following parameters:
BotType: ST_
```

Figure 10-11: SmartTAPTeamsBOT



```
PS D:\michalf\Documents\Customer\support_doc\ST-Teams\Wavenet _customer\BotDeploymentScript> .\deployBot.ps1

cmdlet deployBot.ps1 at command pipeline position 1
Supply values for the following parameters:
BotType: ST
PackagePath: D:\michalf\Documents\Customer\support_doc\ST-Teams\st-teamsbotbeta\botDeploymentscripts\Release\Publish\Hue
Bot
```

Once the deployment is complete, related information is displayed in the PowerShell window and on the Microsoft Azure portal.

11 Step 6 Enable Users with Compliance Recordings

This procedure describes how to enable users with Compliance Recordings using PowerShell scripts on the local machine that need to run with permissions on the required Teams environment. This step includes the following procedures:

- Prerequisite - Join Calls in Teams Tenant below
- Create Compliance Recording Policy on the next page

Prerequisite - Join Calls in Teams Tenant

This procedure describes how to provide SmartTAP 360° with permissions to join calls in your Teams' tenant. The procedure should be performed by your Office 365 Administrator.

➢ **To join calls in your Teams tenant:**

1. Paste the following URL in your browser with parameters shown below: https://login.microsoftonline.com/common/adminconsent?

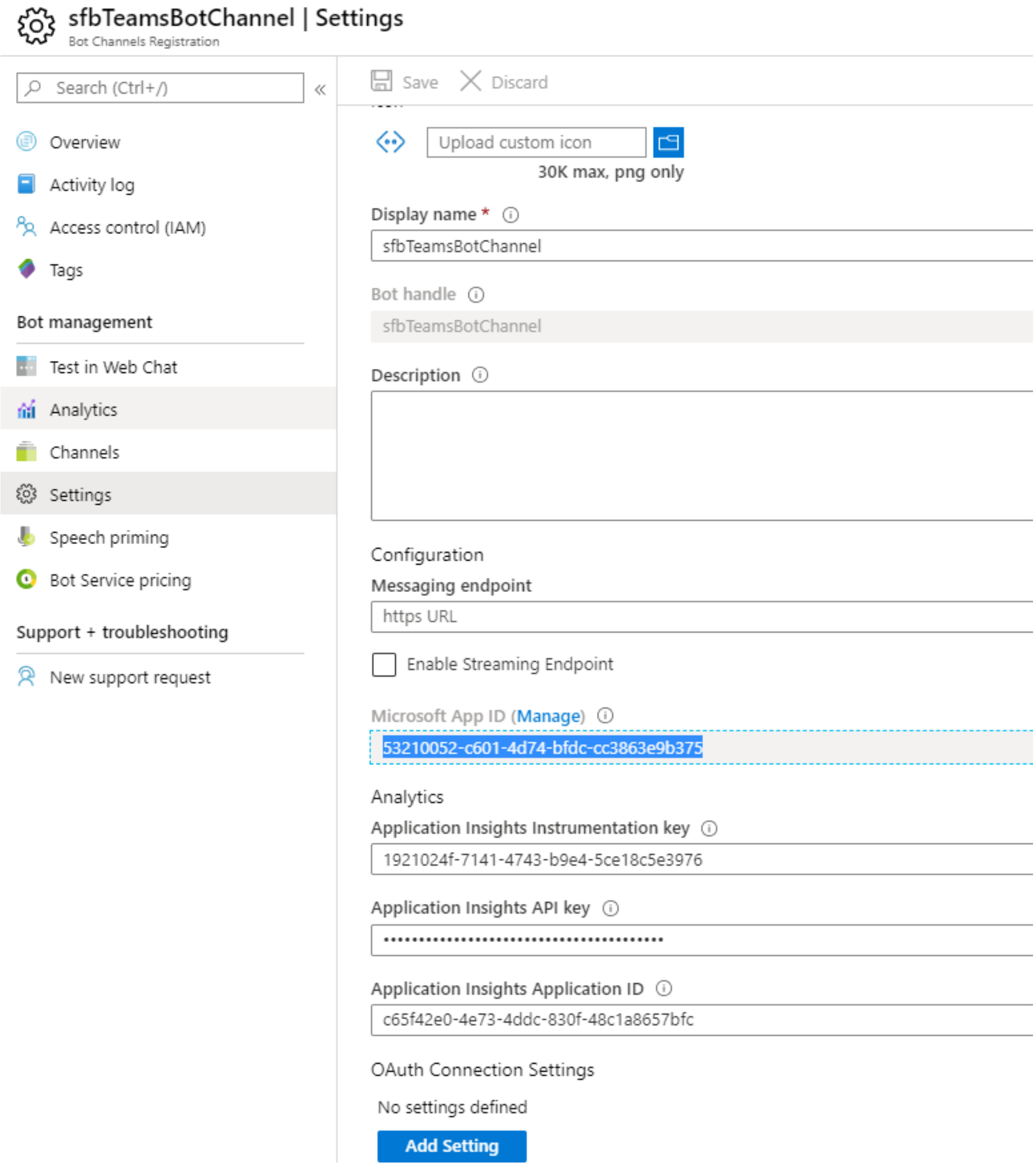
   - client_id=XXXX

   Where XXXX is the SmartTAP 360° Bot app ID from BOT service that was created in Configure Service Channel on page 40 which can be extracted from Manage > BoT Service. This is required to authenticate your Azure subscription.

   - &state=12345
   - &redirect_uri=https://login.microsoftonline.com/common/oauth2/nativeclient
     - 'nativeclient' is the SmartTAP 360° Bot app ID from BOT service that was created and which can be extracted from the Manage > BoT Service page. This is required to authenticate your Azure subscription.
   - &scope=
   - https://graph.microsoft.com/Calls.AccessMedia.All
   - https://graph.microsoft.com/Calls.Initiate.All
   - https://graph.microsoft.com/Calls.InitiateGroupCall.All
   - https://graph.microsoft.com/Calls.JoinGroupCall.All
   - https://graph.microsoft.com/Calls.JoinGroupCallAsGuest.All
   - https://graph.microsoft.com/OnlineMeetings.Read.All
   - https://graph.microsoft.com/OnlineMeetings.ReadWrite.All

   The Authentication Settings are displayed and the connection is authenticated.

Figure 11-1: BOT Channel Settings



sfbTeamsBotChannel | Settings

Bot Channels Registration

Display name * : sfbTeamsBotChannel

Bot handle: sfbTeamsBotChannel

Description:

Configuration

Messaging endpoint: https URL

□ Enable Streaming Endpoint

Microsoft App ID (Manage): 53210052-c601-4d74-bfdc-cc3863e9b375

Analytics

Application Insights Instrumentation key: 1921024f-7141-4743-b9e4-5ce18c5e3976

Application Insights API key: ••

Application Insights Application ID: c65f42e0-4e73-4ddc-830f-48c1a8657bfc

OAuth Connection Settings

No settings defined

Create Compliance Recording Policy

This procedure describes how to create a Compliance Recording Policy:

1. Create Application Instance below
2. Create New Compliance Recording Policy on page 63
3. Set Compliance Recording Policy on page 64
4. Grant Policy to a Recorded User on page 65

Create Application Instance

This procedure describes how to create an Application Instance on the local machine. This action can be performed by 'Admin' user.

➢ To create an Application instance:

1. Download Skype for Business module to be able to record Teams users with SmartTAP 360°. The Microsoft Teams Administrator must create a Compliance Recording Policy for SmartTAP 360° and assign it to the recorded users. Refer to the following link:

   https://docs.microsoft.com/en-us/skypeforbusiness/set-up-your-computer-for-windows-powershell/download-and-install-the-skype-for-business-online-connector

2. Create a new session with the relevant Teams tenant:

   ```
   PS .:\> Import-Module SkypeOnlineConnector
   ```

   ```
   PS .:\> $sfbSession = New-CsOnlineSession
   ```

   ```
   PS .:\> Import-PSSession $sfbSession
   ```

   Refer to: https://docs.microsoft.com/en-us/office365/enterprise/powershell/manage-skype-for-business-online-with-office-365-powershell

3. Enter the following commands:

   ```
   PS .:\> New-CsOnlineApplicationInstance -UserPrincipalName <User
   Principal Name> -DisplayName <displayName> -ApplicationId
   <SmartTAPBOTID>
   ```

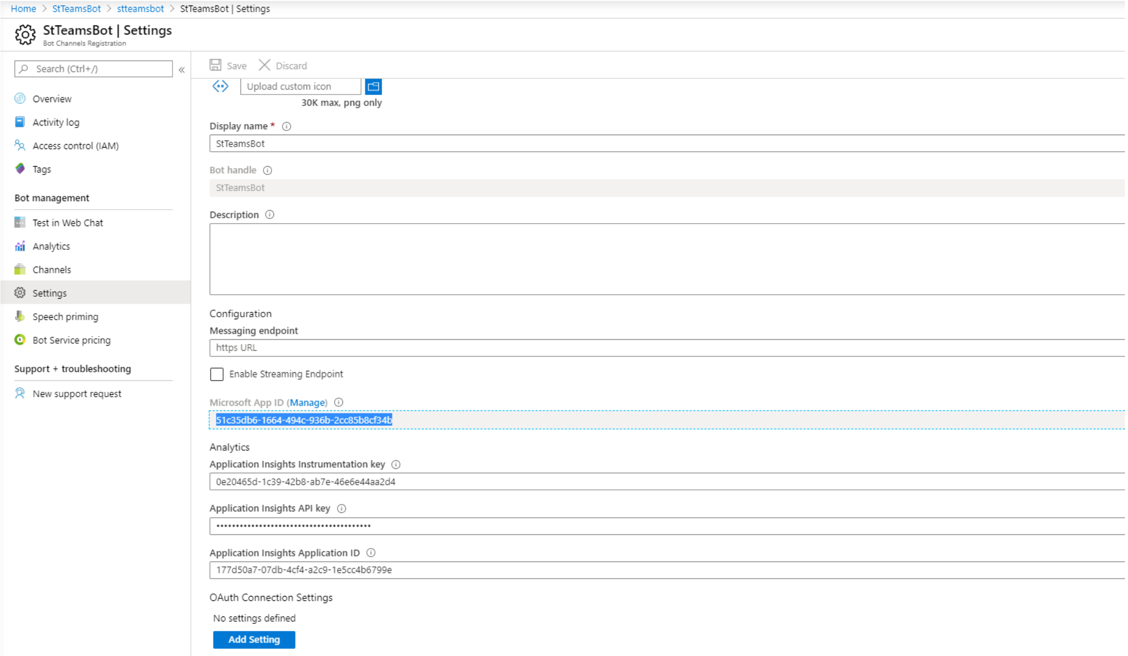
   Where:

   - <UserPrincipalName>: AD BOT entity - Organizational user with onmicrosoft.com domain that is assigned to the BOT.
   - <SmartTAPBOTID> -Application ID that was created during the creation of the BOT Service channel (see Configure Service Channel on page 40). This value can extracted from the Settings screen (see example figure below).
   - <displayName>: Free text Description field

   Output similar to the following is displayed:

   ```
   RunspaceId      : 15eea8f7-970e-4061-893e-3573cb5e973b
   ObjectId        : fd13dab0-dd31-4b58-86d6-122fa07e250f
   TenantId        : ad41d6c3-67f0-47cc-9de3-e07fd185c1c7
   UserPrincipalName : STTeamsbotstandartlb2@smarttap.onmicrosoft.com
   ApplicationId   : ff6fc00a-fc73-4062-b99f-55ff0e09b779
   DisplayName     : STTeamsbotstandartlb2
   PhoneNumber     :
   ```

Figure 11-2: Create Application Instance



Display name * StTeamsBot

Bot handle StTeamsBot

Description

Configuration

Messaging endpoint https URL

☐ Enable Streaming Endpoint

Microsoft App ID (Manage) 51c35db6-1664-494c-936b-2cc85b8cf34b

Analytics

Application Insights Instrumentation key 0e20465d-1c39-42b8-ab7e-46e6e44aa2d4

Application Insights API key ••

Application Insights Application ID 177d50a7-07db-4cf4-a2c9-1e5cc4b6799e

OAuth Connection Settings

No settings defined

4. Enter the following command:

```
PS .:\> Sync-CsOnlineApplicationInstance -ObjectId <ObjectID>
```

Where <ObjectID> is the ObjectID that was generated from the above command. Note this value for procedure in Set Compliance Recording Policy on the next page.

Create New Compliance Recording Policy

This procedure describes how to create a new Compliance Recording Policy.

➢ To create a new compliance recording policy:

1. Enter the following commands:

```
PS ..\> New-CsTeamsComplianceRecordingPolicy -Tenant <TenantID> -
Enabled $true -Description <free text> <ComplianceRecordingBot_
PolicyName>
```

- <TenantID>: Azure tenant ID of customer's Microsoft Azure subscription (Microsoft App ID)
- <ComplianceRecordingBot_PolicyName>: User-defined name of the Compliance Recording Policy

2. After 30-60 seconds, the policy should be displayed. Enter the following command to verify that your policy was added correctly:

```
PS ..\> Get-CsTeamsComplianceRecordingPolicy
<ComplianceRecordingBot_PolicyName>
```

⚠ For more information, refer to: Create New Compliance Recording Policy

Set Compliance Recording Policy

This procedure describes how to set the Compliance Recording policy.

➢ **To set the Compliance Recording Policy:**

1. Enter the following commands:

```
PS .:\> Set-CsTeamsComplianceRecordingPolicy -Tenant <TenantID> -
Identity <ComplianceRecordingBot_PolicyName> -Tenant <TenantID> -
Parent ComplianceRecordingBot -Id <ObjectID>  -<policy-based recording
application behavior> $true/false
```

- **<TenantID>:** Azure tenant ID of customer's Microsoft Azure subscription (Microsoft App ID)
- **<ComplianceRecordingBot_PolicyName>:** User-defined name of the Compliance Recording Policy that was defined in Create New Compliance Recording Policy on the previous page
- **<ObjectID>:** Object ID that was generated in Create Application Instance on page 61
- **<policy-based recording application behavior>:** $true/false

  Where <policy-based recording application behavior> is one of the following:

  - **RequiredBeforeCallEstablishment (default: false):** Indicates whether the policy-based recording application must be in the call before the call is allowed to establish. If this is set to True, the call will be cancelled if the policy-based recording application fails to join the call. If this is set to False, call establishment will proceed normally if the policy-based recording application fails to join the call.
  - **RequiredBeforeMeetingJoin (default: false):** Indicates whether the policy-based recording application must be in the meeting before the user is allowed to join the meeting. If this is set to True, the user will not be allowed to join the meeting if the policy-based recording application fails to join the meeting. The meeting will still continue for users who are in the meeting. If this is set to False, the user will be allowed to join the meeting even if the policy-based recording application fails to join the meeting.
  - **RequiredDuringCall (default: false):** Indicates whether the policy-based recording application must be in the call while the call is active. If this is set to True, the call will be cancelled if the policy-based recording application leaves the

call or is dropped from the call. If this is set to False, call establishment will proceed normally if the policy-based recording application leaves the call or is dropped from the call.

- **RequiredDuringMeeting (default: false):** Indicates whether the policy-based recording application must be in the meeting while the user is in the meeting. If this is set to True, the user will be ejected from the meeting if the policy-based recording application leaves the meeting or is dropped from the meeting. The meeting will still continue for users who are in the meeting. If this is set to False, the user will not be ejected from the meeting if the policy-based recording application leaves the meeting or is dropped from the meeting.
- **Priority:** Determines the order in which the policy-based recording applications are displayed in the output of the Get-CsTeamsComplianceRecordingPolicy cmdlet.
- **ConcurrentInvitationCount:** Determines the number of invites to send out to the application instance of the policy-based recording application.

2. After 30-60 seconds, the policy should be displayed. Enter the following command to verify that your policy was updated correctly:

```
PS .:\> Get-CsTeamsComplianceRecordingPolicy
<ComplianceRecordingBot_PolicyName>
```

⚠ For more information, refer to Set Compliance Recording Application

Grant Policy to a Recorded User

This procedure describes how to grant policies to a recorded user.

➢ To grant policies to a recorded user:

- Enter the following commands:

```
PS .:\> Grant-CsTeamsComplianceRecordingPolicy -Identity <Identity> -
PolicyName ComplianceRecordingBot -Tenant <TenantID>
```

Where:

- Identity: UPN of recording-targeted user
- <TenantID>: Azure tenant ID of customer's Microsoft Azure subscription (Microsoft App ID)

⚠ For more information, refer to https://docs.microsoft.com/en-us/powershell/module/skype/grant-csteamscompliancerecordingpolicy?view=skype-ps

This page is intentionally left blank.

International Headquarters

1 Hayarden Street,

Airport City

Lod 7019900, Israel

Tel: +972-3-976-4000

Fax: +972-3-976-4040

AudioCodes Inc.

200 Cottontail Lane

Suite A101E

Somerset NJ 08873

Tel: +1-732-469-0880

Fax: +1-732-469-2298

Contact us: https://www.audiocodes.com/corporate/offices-worldwide

Website: https://www.audiocodes.com/

Documentation Feedback: https://online.audiocodes.com/documentation-feedback

Document #: LTRT-27327