Microsoft Teams Deployment Guide

AudioCodes SmartTAP™ 360° for Microsoft Teams

SmartTAP 360° for Microsoft Teams

SmartTAP 360° Enterprise Recording Solution

Version 5.1

audiocodes

Notice

Date Published: August-30-2020

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	Updates to Section: 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

Table of Contents

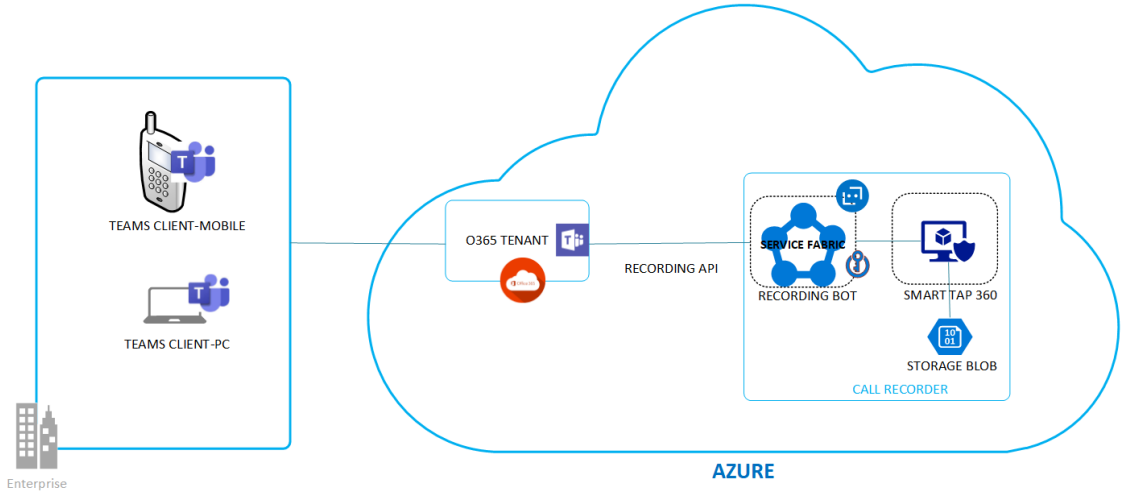
1 Overview

SmartTAP 360° Teams Compliance Recording may be hosted in-house entirely or on the customer's Microsoft Azure subscription. 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 figure below illustrates the basic deployment topology including a single Office 365 tenant and Service Fabric.

Figure 1-1: Basic Deployment



2 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/DAS call recordings.
    - Virtual Machine: Tier=Standard, Instance=F8s v2 (8 vCPUs, 16 GB RAM, 64 GB Temporary storage)
- **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 2016 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.

3 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.

4 Deployment Procedures Overview

The deployment includes the following procedures:

5 Step 1 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\STTeamsBOT_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
```

```
}
```

The above parameter is set according to the cluster durability settings for the number of instances in the Service Fabric Cluster.

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 5-1: Run Deploy Script

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

This step should be performed for all deployed Service Fabric nodes that are deployed in the Service Fabric cluster; for Multi-tenancy deployments where each tenant has a dedicated Service Fabric node.

6. Restart all nodes.

6 Step 2 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:

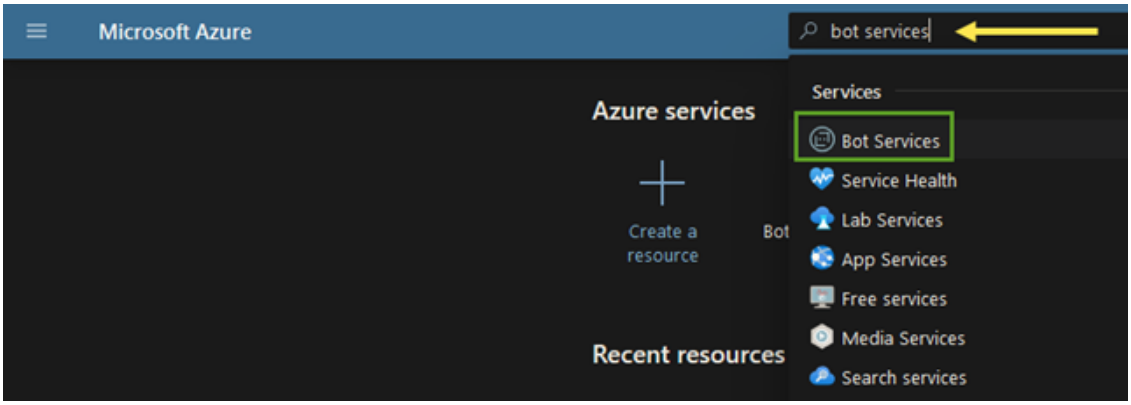
- Step 2-1 Configure Service Channel on page 14
- Step 2-2 Grant API Permissions to BOT Service on page 18

> 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:

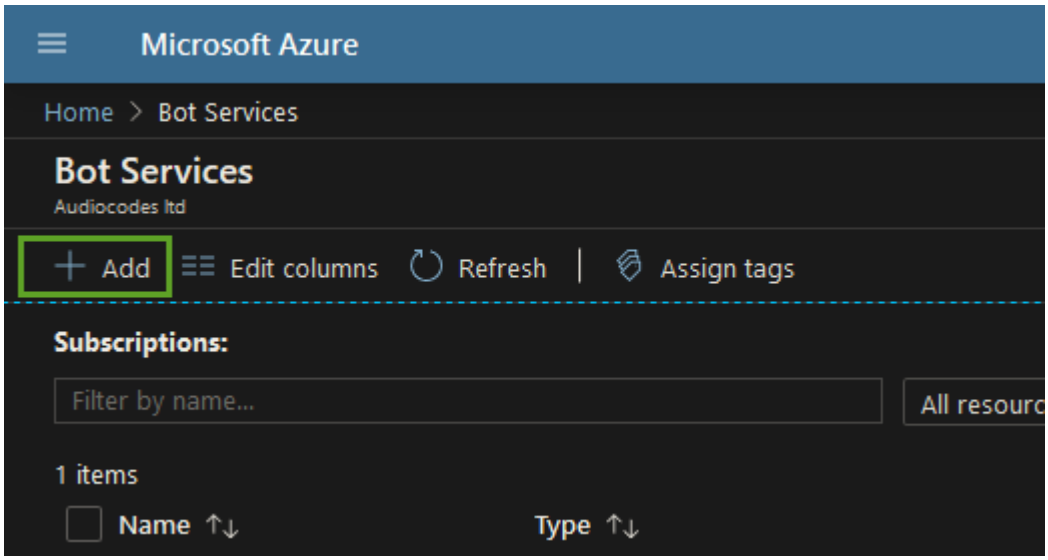
1. In the Azure portal, open the BOT Services screen (**Services** > **Bot Services**).

Figure 6-1: Azure Services



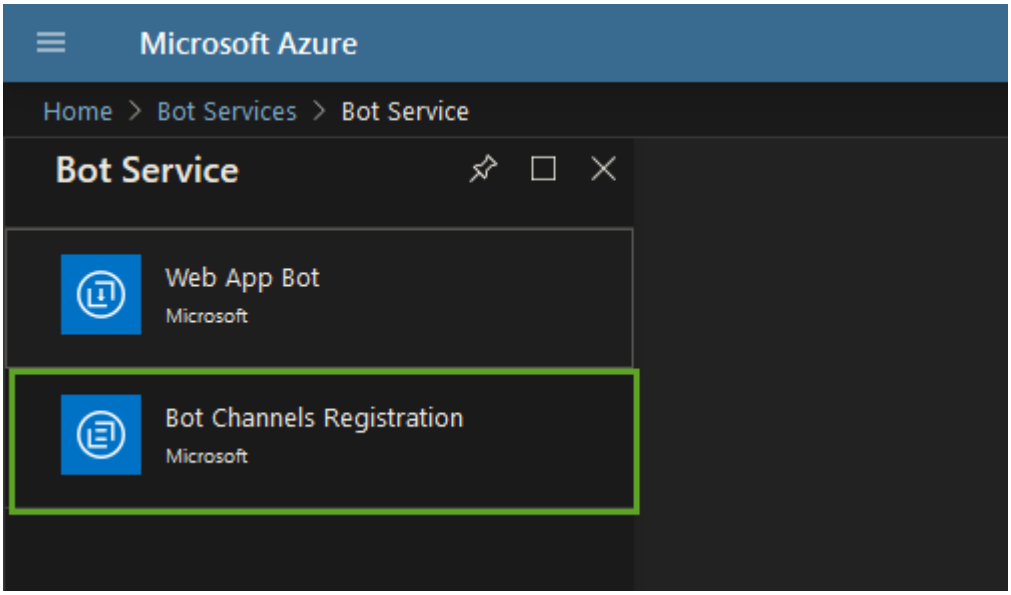
2. Click **Add** to add a new Bot service.

Figure 6-2: Add BOT Service



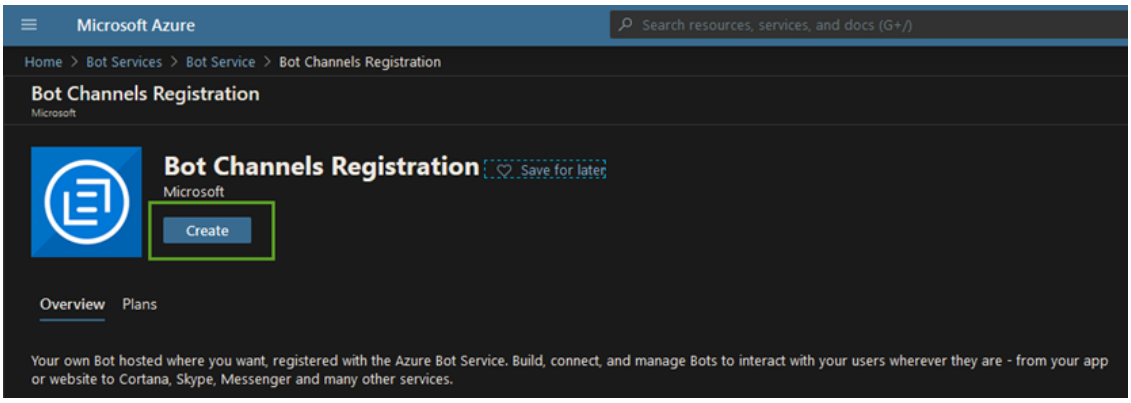
3. Click **Bot Channels Registration**.

Figure 6-3: BOT Channels Registration



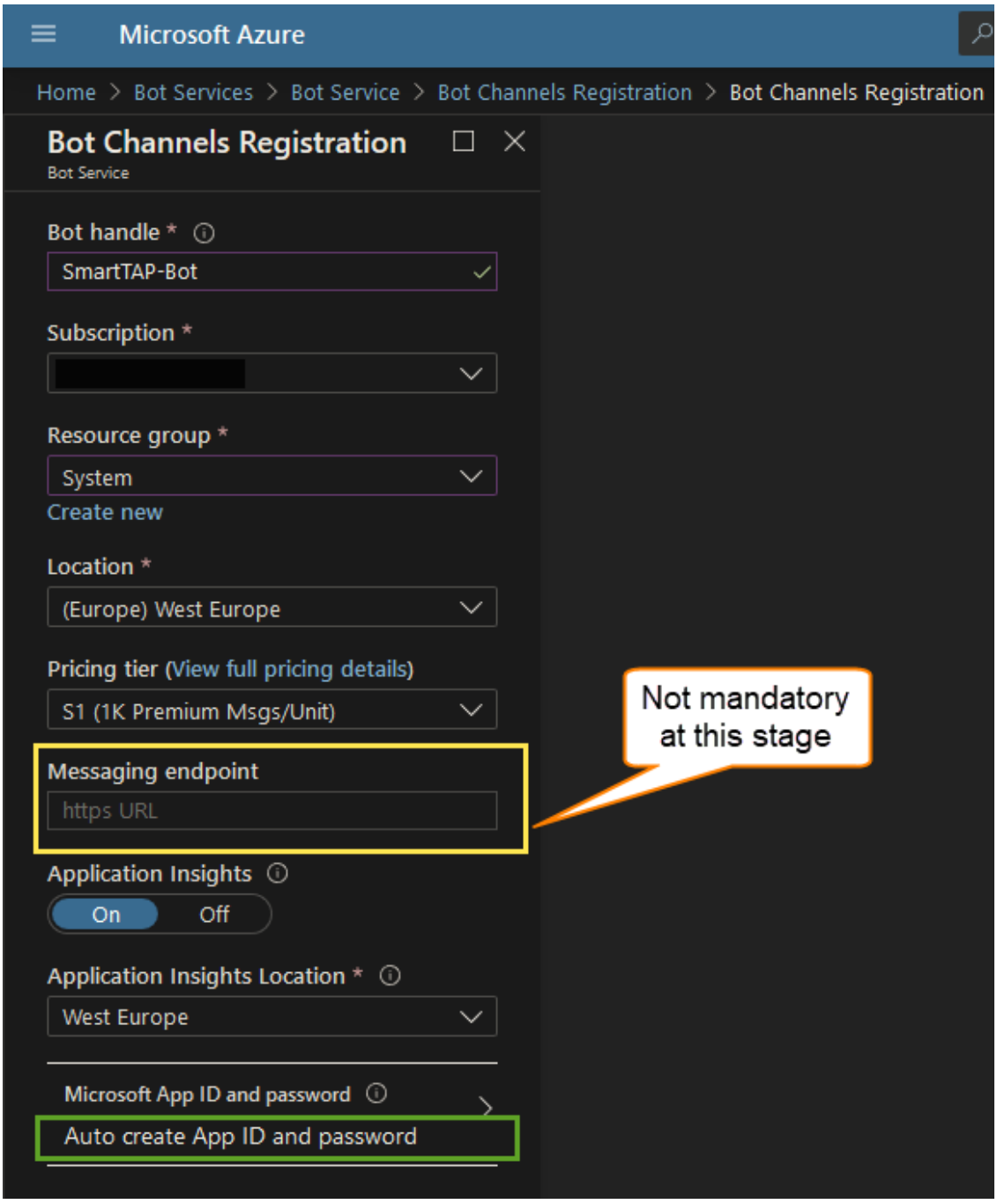
4. Click **Create** to create the service.

Figure 6-4: Create the Service



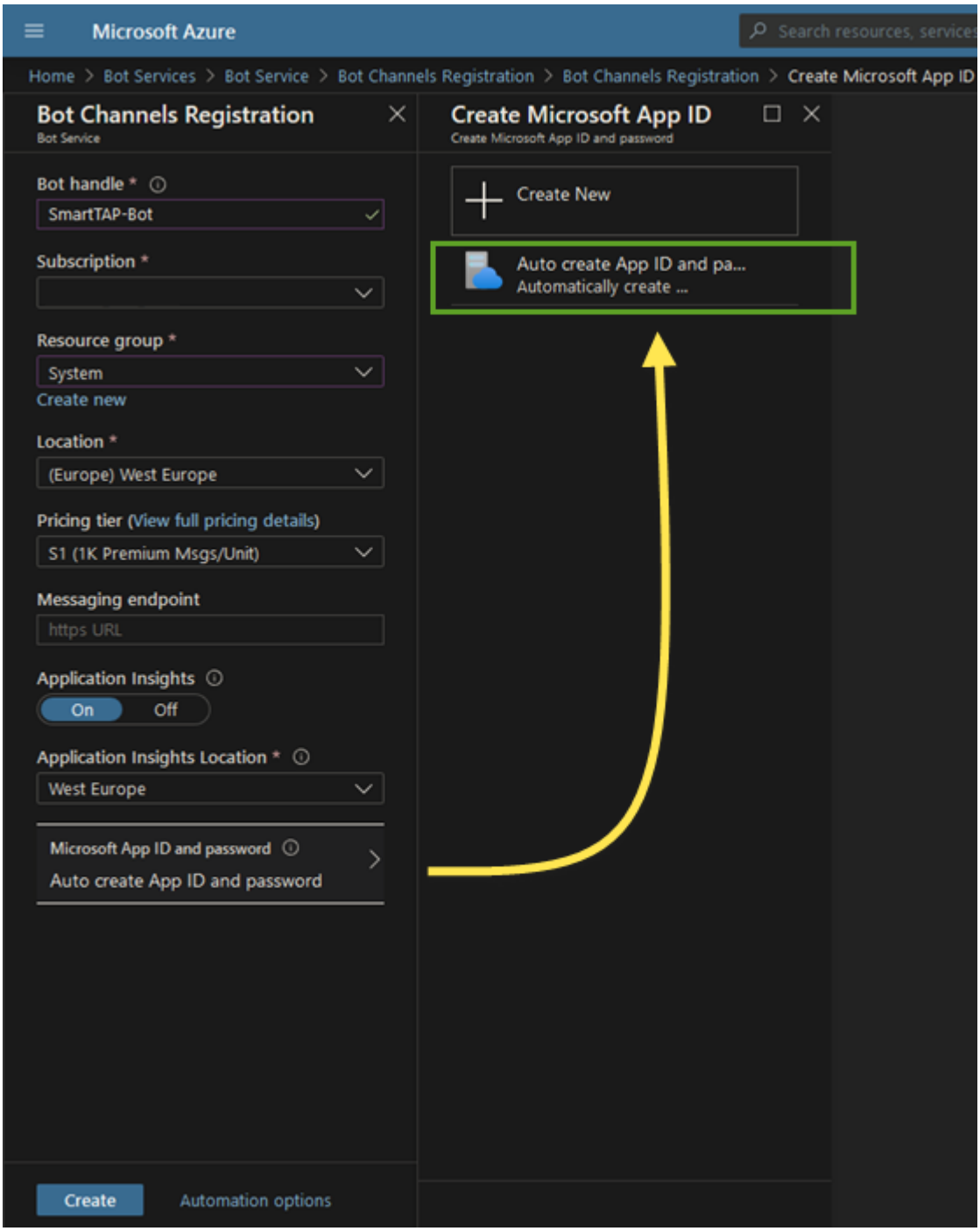
5. Set the relevant parameters shown in the Bot Channels Registration screen below.

Figure 6-5: Parameter Configuration



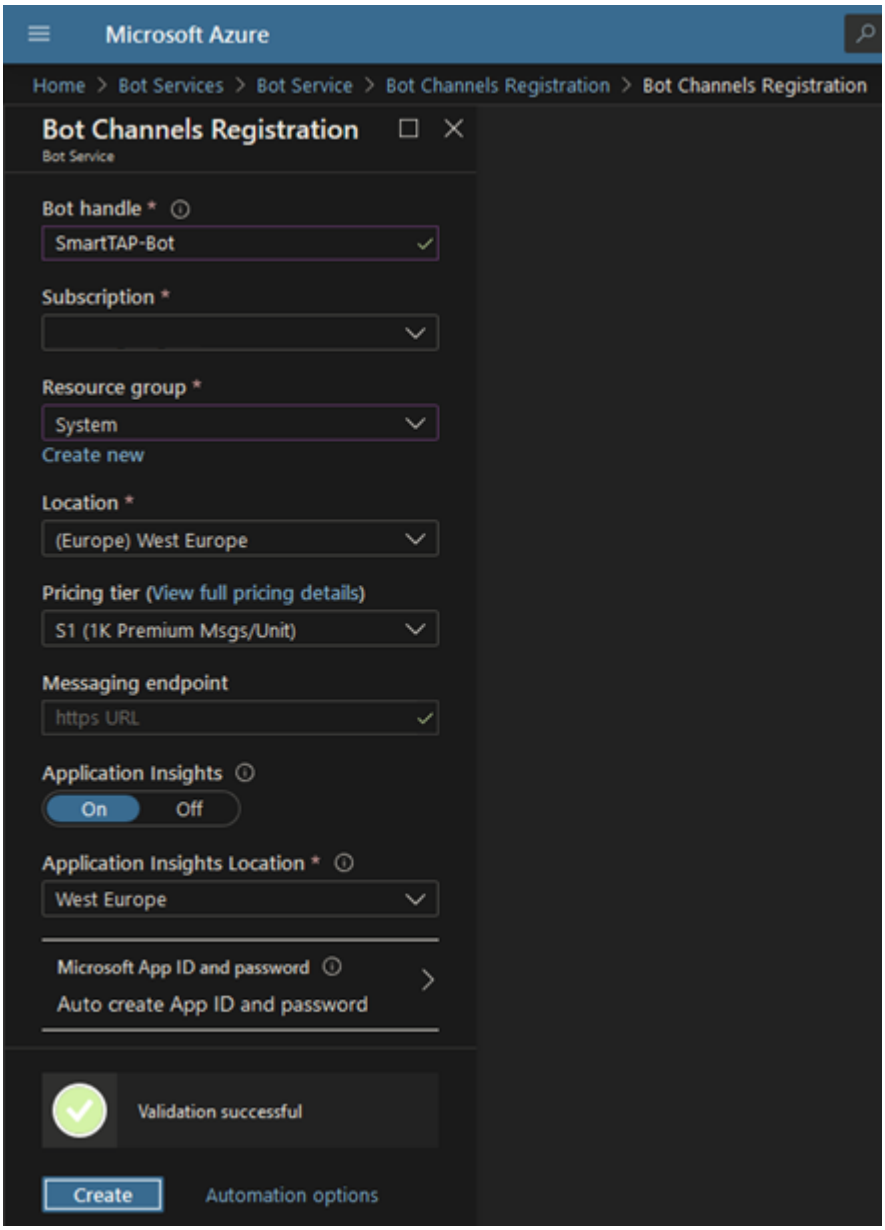
6. Select **Auto create App ID and password** to create the Microsoft App ID (copy to notepad as this value is configured in later in Step 4 Enable Users with Compliance Recordings on page 29).

Figure 6-6: Auto Create App ID



7. Click **Create.**

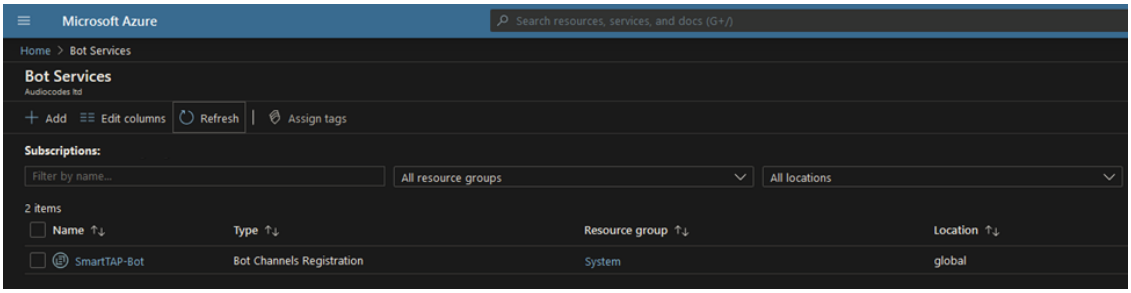
Figure 6-7: Bot Channels Registration Details



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 6-8: New Resource



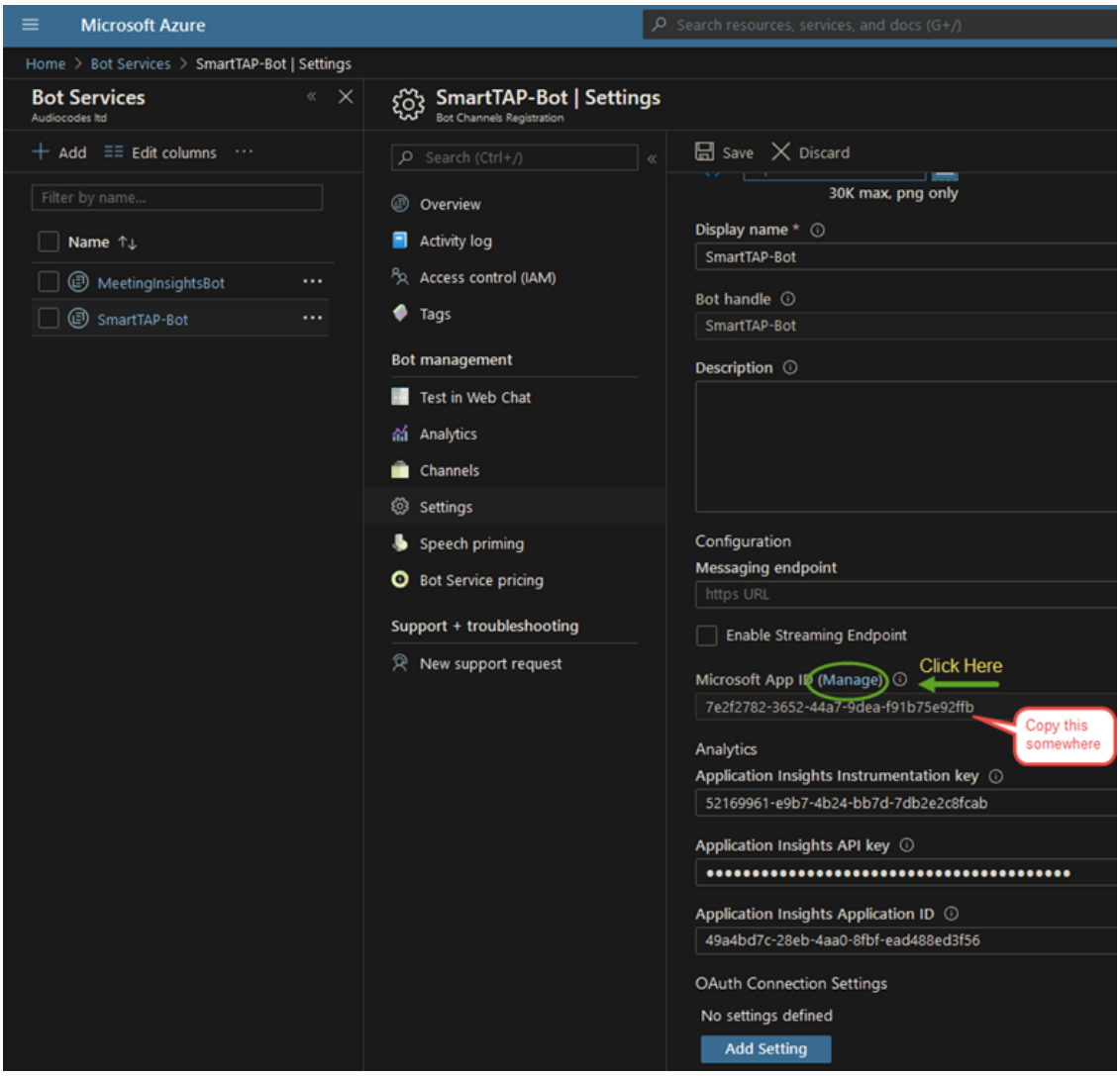
Step 2-1 Configure Service Channel

This procedure describes how to configure the service channel.

➢ To configure the service channel:

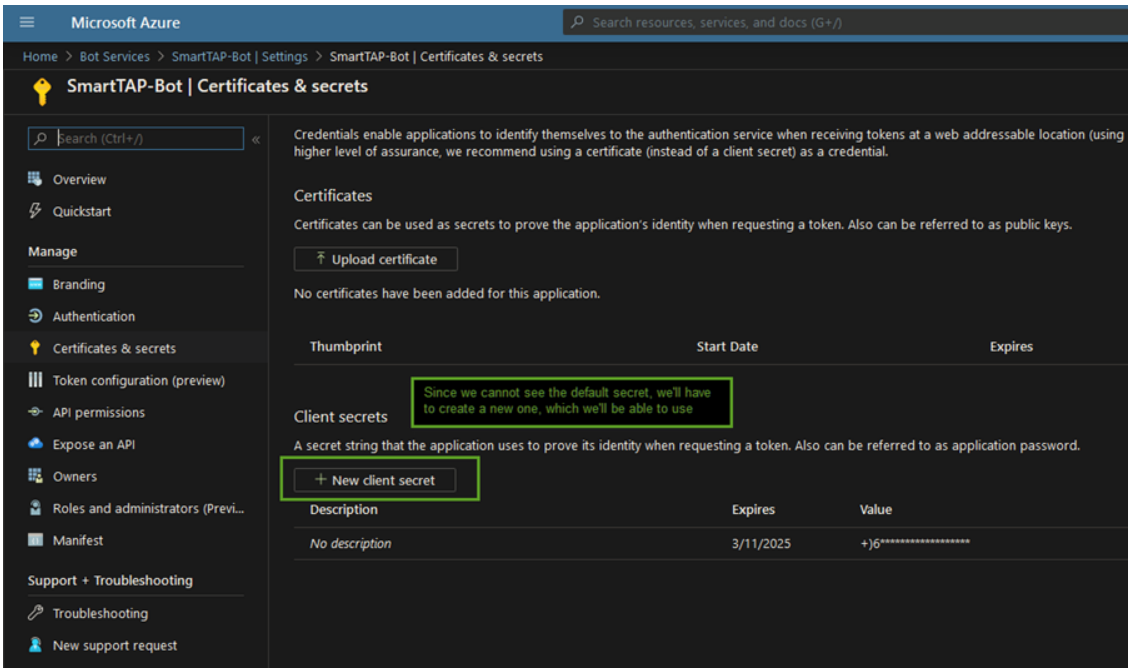
1. Click **Edit**.

Figure 6-9: Settings

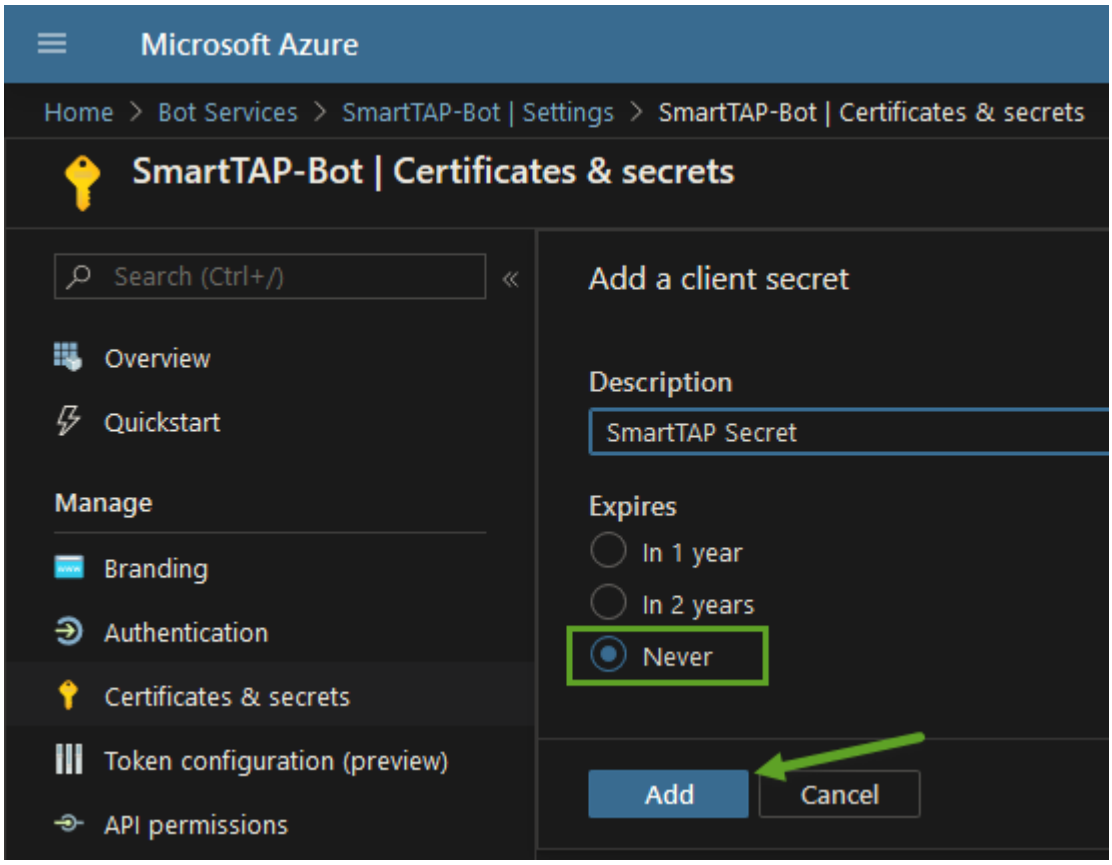


2. Click **Manage** to configure the Microsoft App ID.

Figure 6-10: Certificates and Secrets

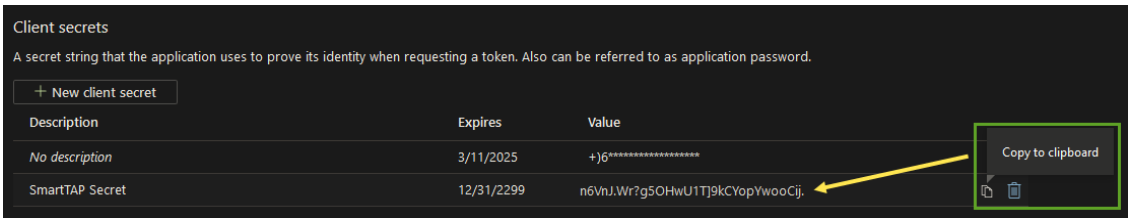


3. Click **New client secret** to create a new APP secret.



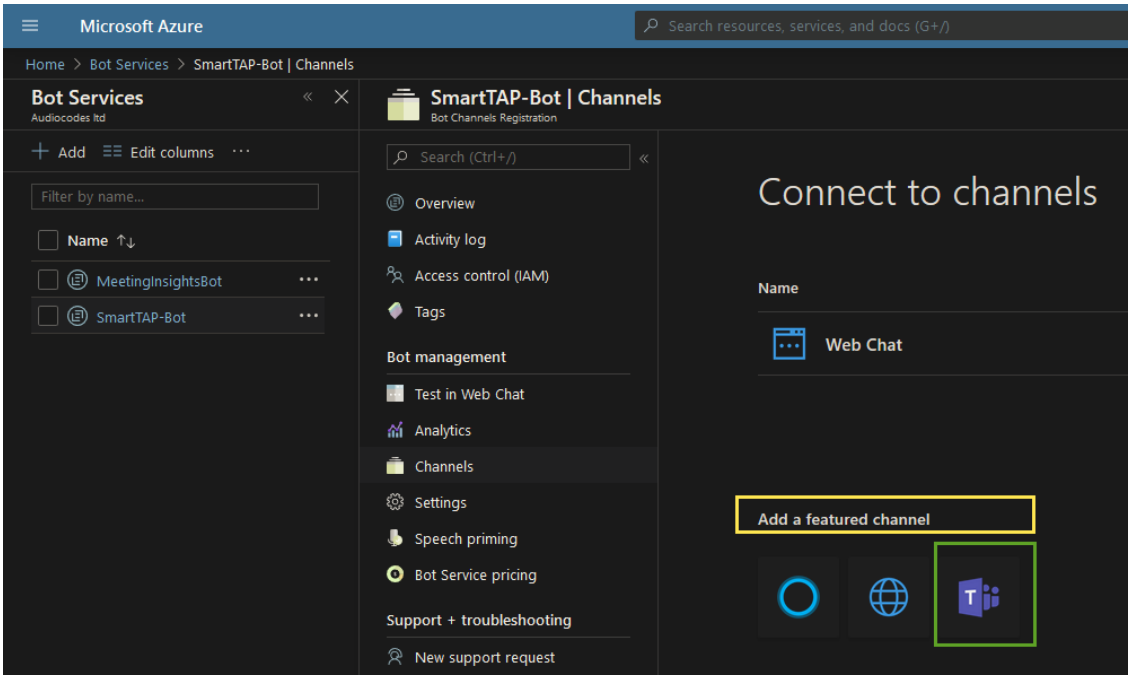
4. In the Expires pane, select **Never** and then click **Add**.

Figure 6-11: Client Secrets



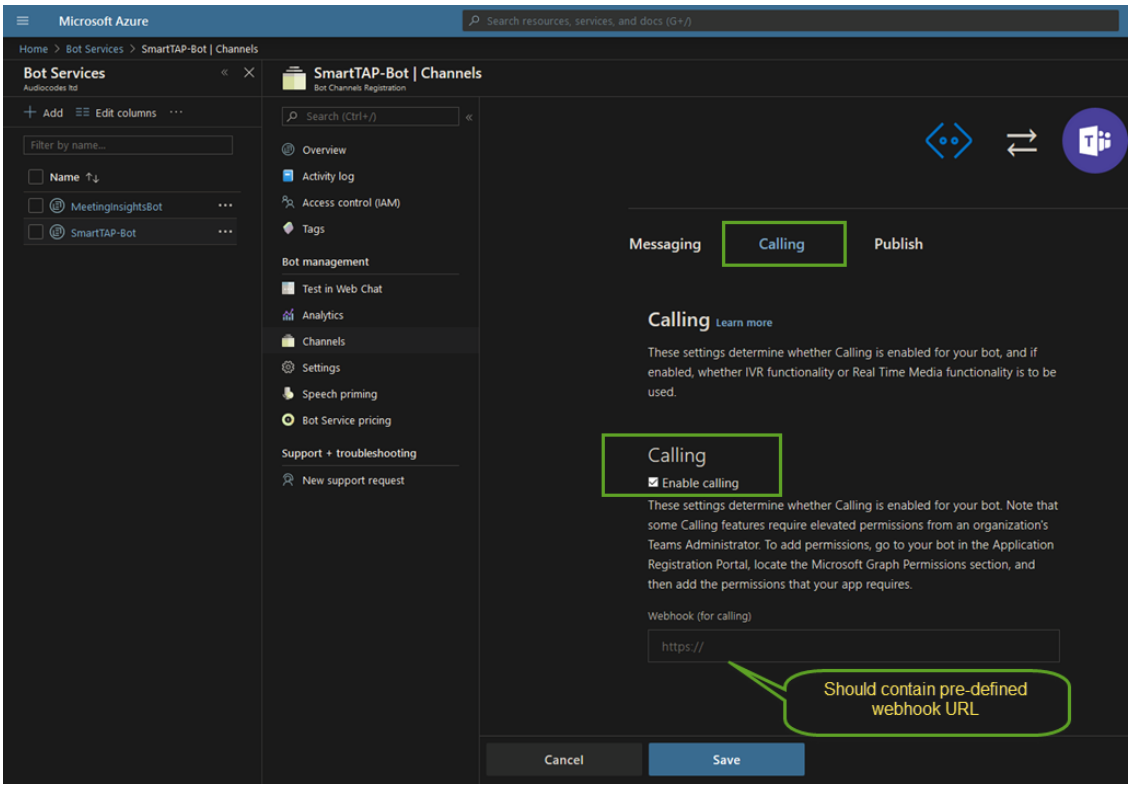
5. Copy the SmartTAP 360° Secret to the clipboard or notepad as it must be configured in a later procedure.
6. Open the Channels screen (**Home** > **BoT Services** > **SmartTAP 360°-BoT** > **Channels**).
7. Select **Add a featured Channel** > **Teams** icon.

Figure 6-12: Teams Feature



8. Click the **Calling** tab.

Figure 6-13: Calling Option

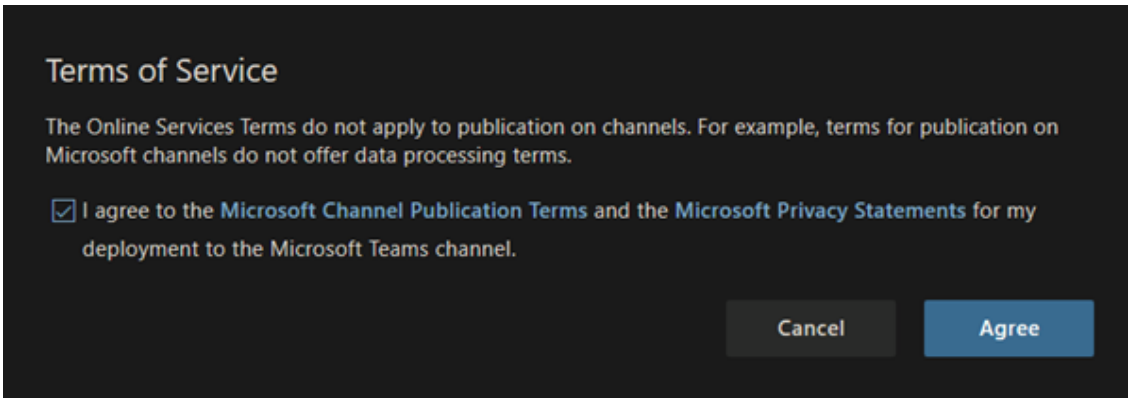


9. Select the 'Enable Calling' check box.
10. Paste the pre-defined webhook URL as follows:

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

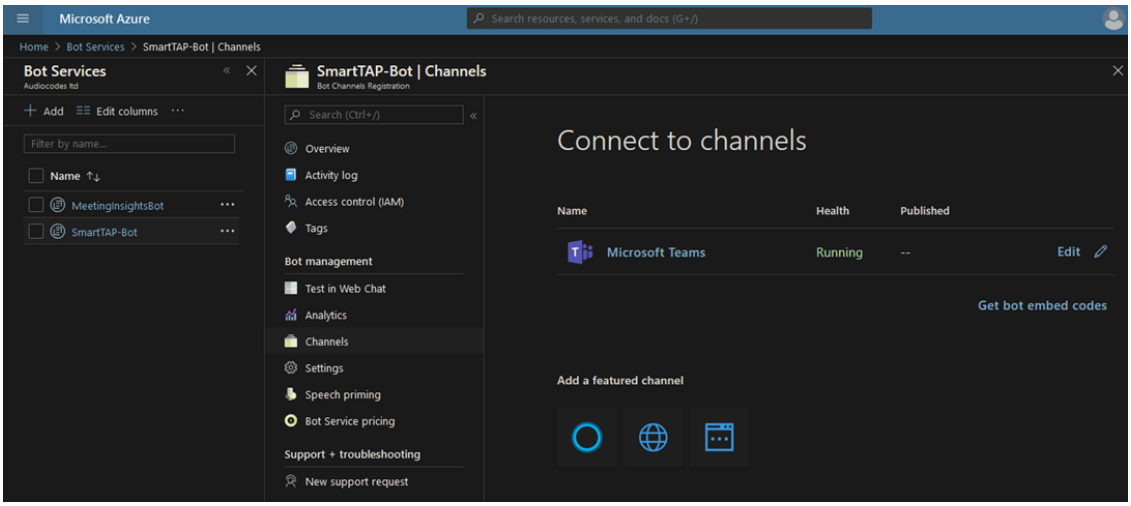
    Where the URL is the service fabric DNS name given to the Service Fabric Cluster admin (see Prerequisites on page 4).

Figure 6-14: Terms of Service



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

Figure 6-15: Connect to Channels



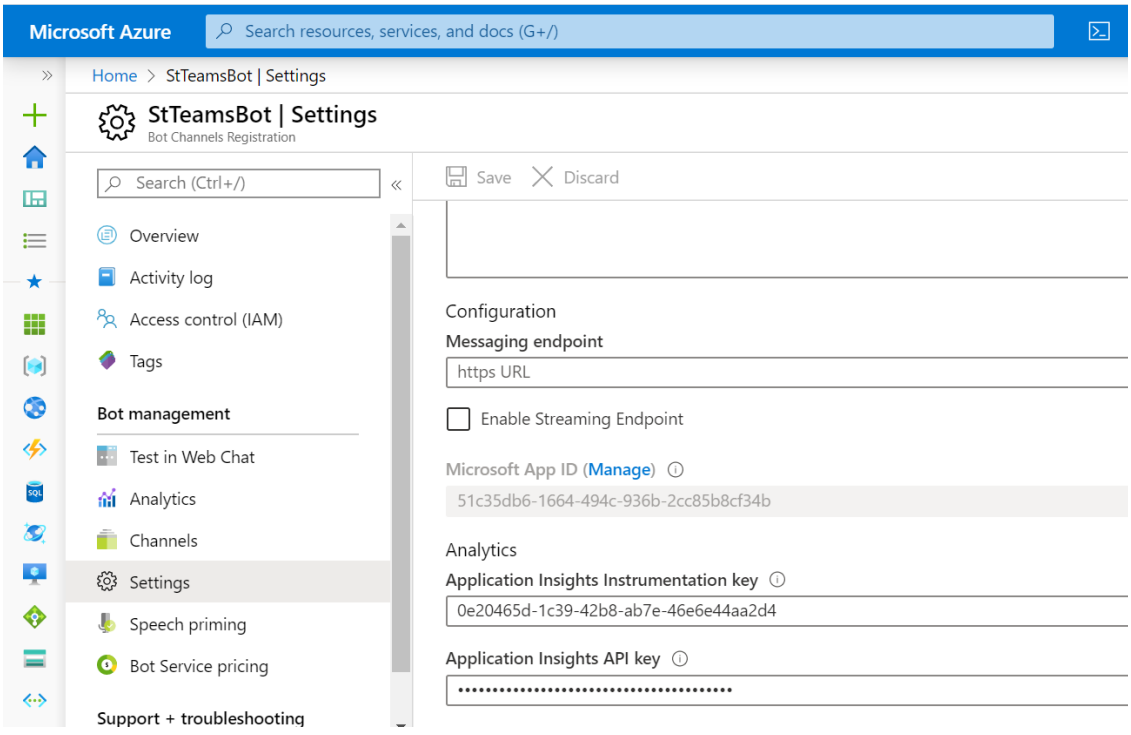
Step 2-2 Grant API Permissions to BOT Service

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

➢ To grant API permissions to the BOT service:

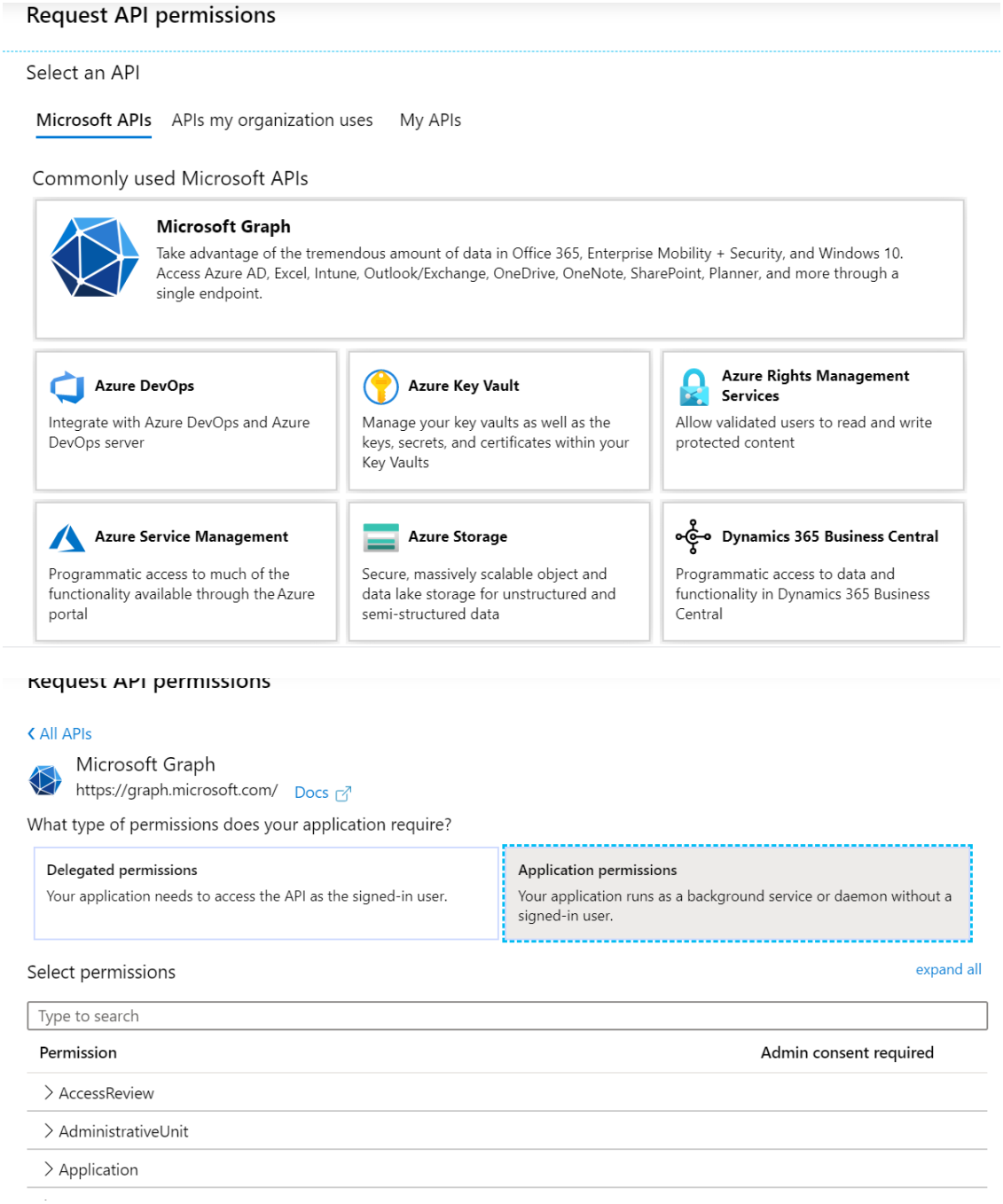
1. In the Azure portal, open the Settings page (**Home** > **StTeamsBOT** > **Settings**).

Figure 6-16: BOT Settings



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

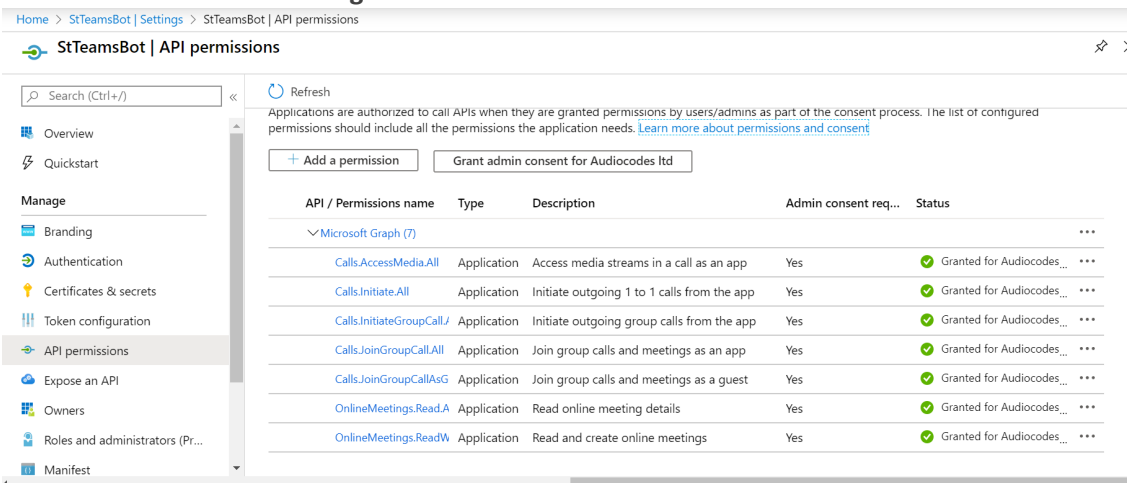
Figure 6-17: Add a Permission



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

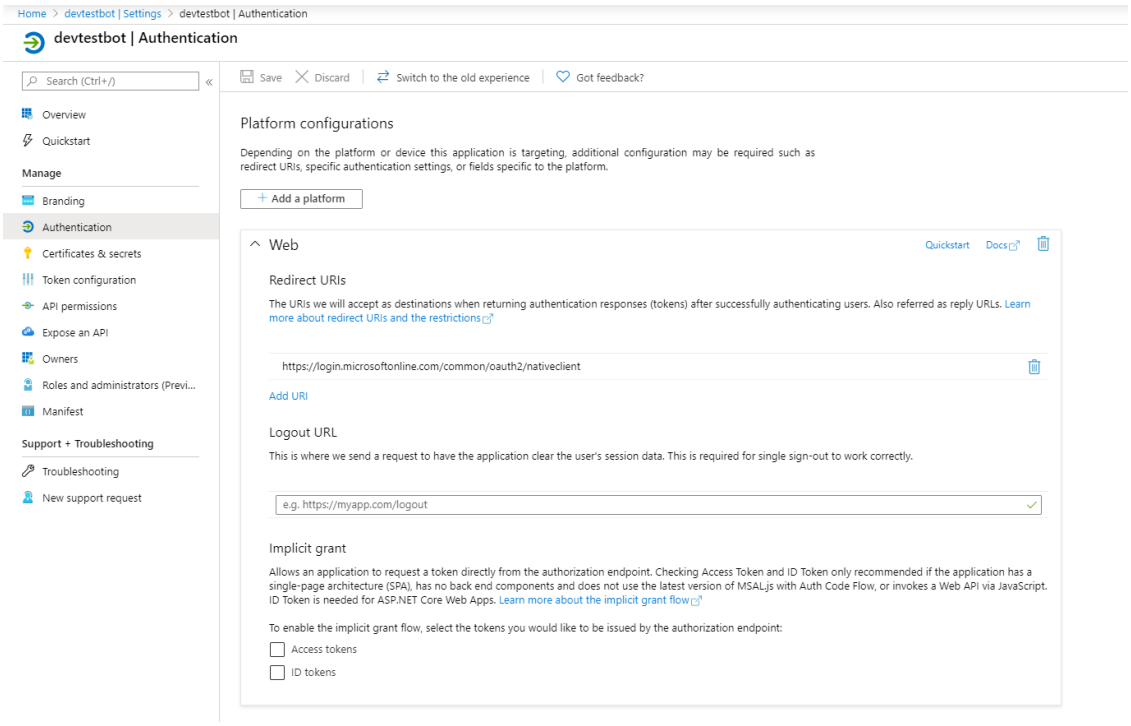
Figure 6-18: Listed Permission

Figure 6-19:



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

Figure 6-20: Authentication



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 Step 2-1 Configure Service Channel on page 14. This is required to authenticate your Azure subscription.

7 Step 3 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:

Step 3-1 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:**

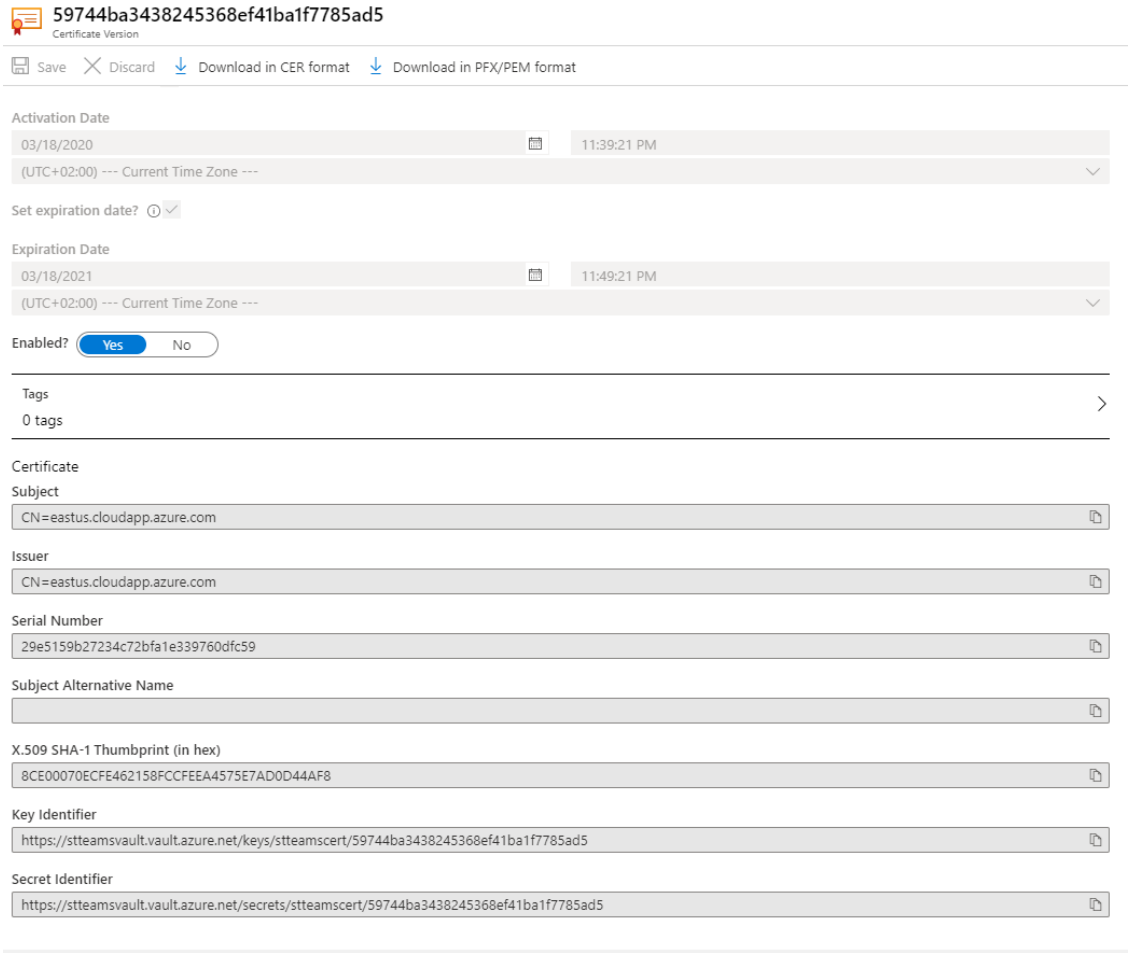
1. Extract the SFC Deployment script package from the following location to your local machine:
   …\Release\Publish\STTeamsBOT_Deployment_Package\BotDeploymentScript:

   This directory includes the following files:

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

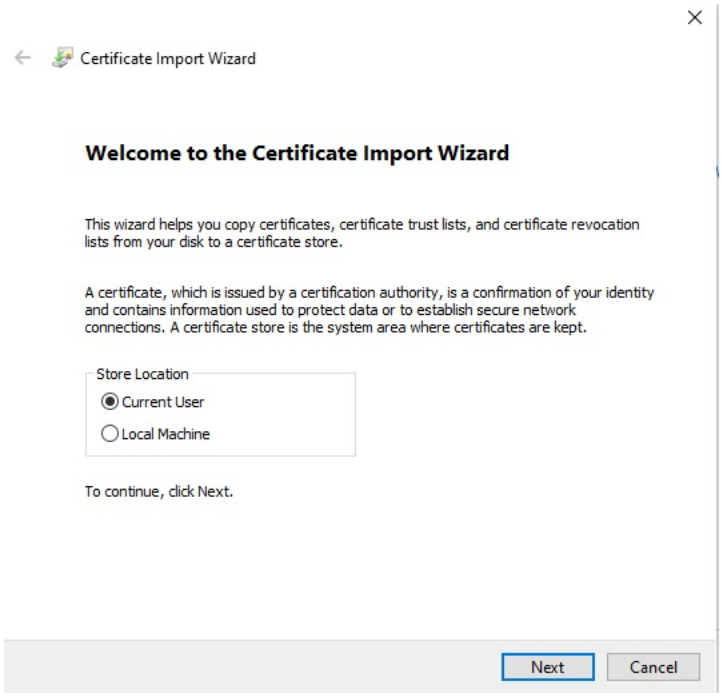
   ```
   PS .:\> “Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Force -Scope CurrentUser”
   ```
3. Install Azure SDK for Service Fabric.
4. Download generated certificate (see Prerequisites on page 4) in PFX/PEM format to the local machine.

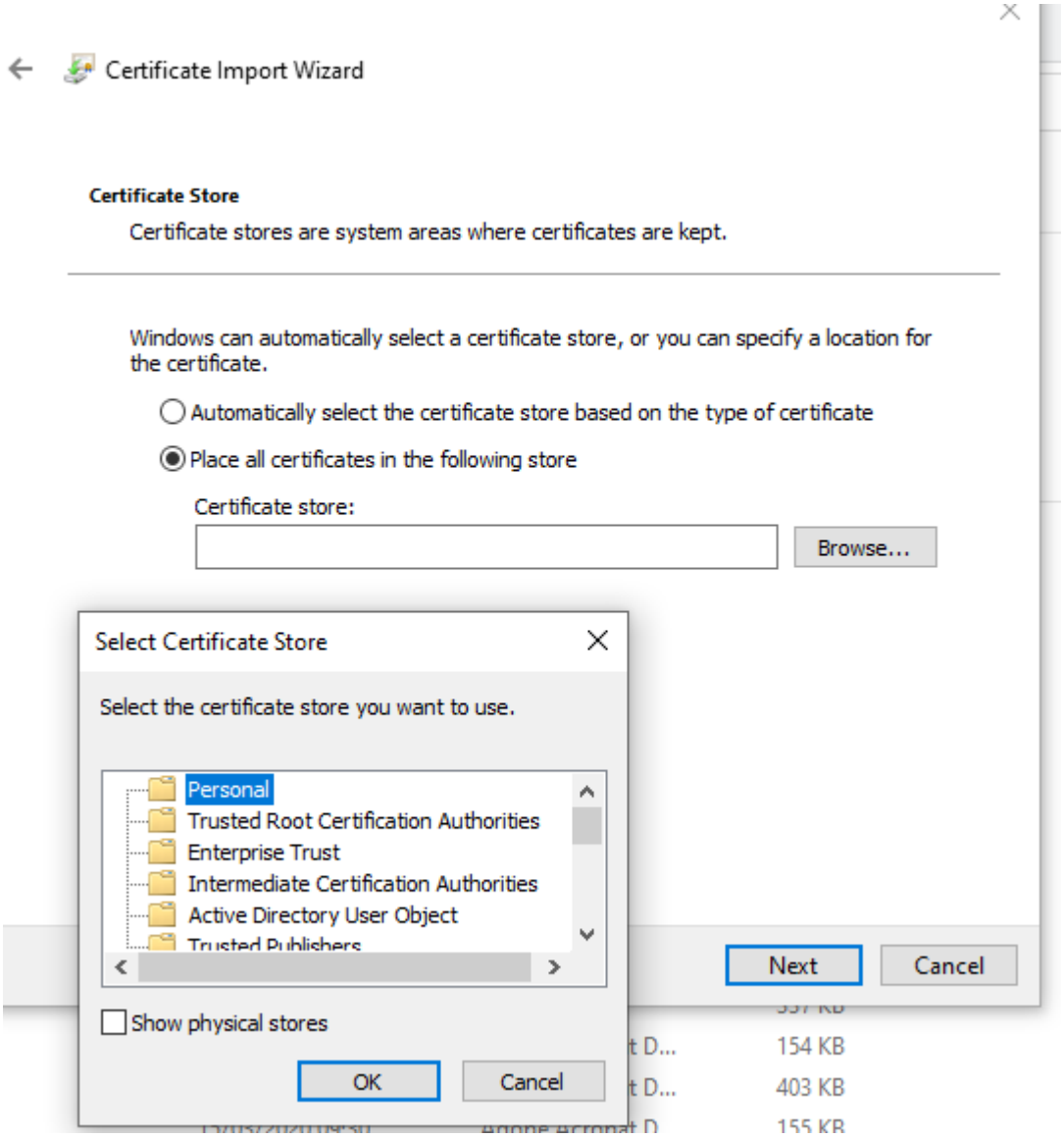
Figure 7-1: Download Certificate



5. Install the certificate to personal store.

Figure 7-2: Certificate Import Wizard





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 7-3: Client Certificate

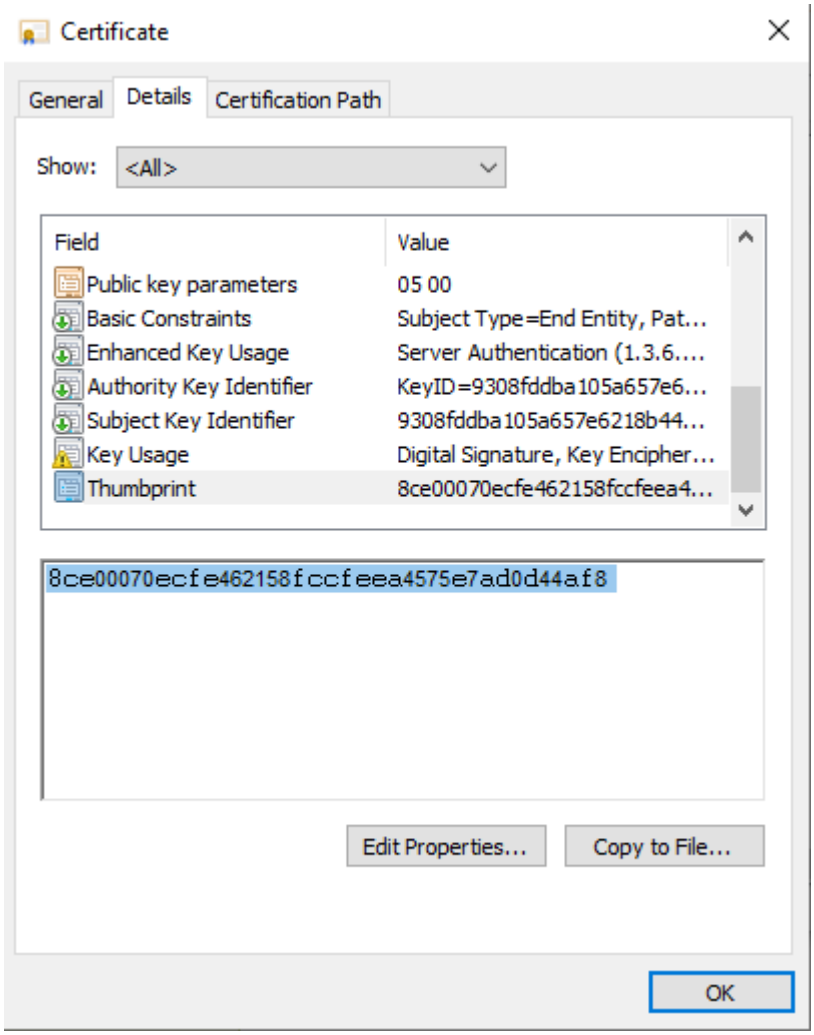
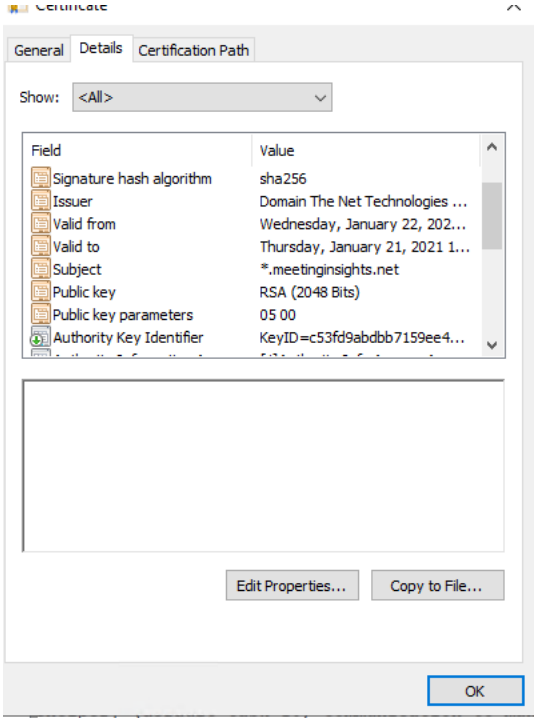


Figure 7-4: Server Common Name



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 (\STTeam-sBOT\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, https://stteamsbotsfpoc.meetinginsights.net:9444/api/calls*

"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>",
"Domain": "<Domain>",
"ApplicationInsights": {
"InstrumentationKey": "<InstrumentationKey>" #set from BOT services, see example image below.
"AppId": "<AppId>",
```

Figure 7-5: Configure Microsoft App ID

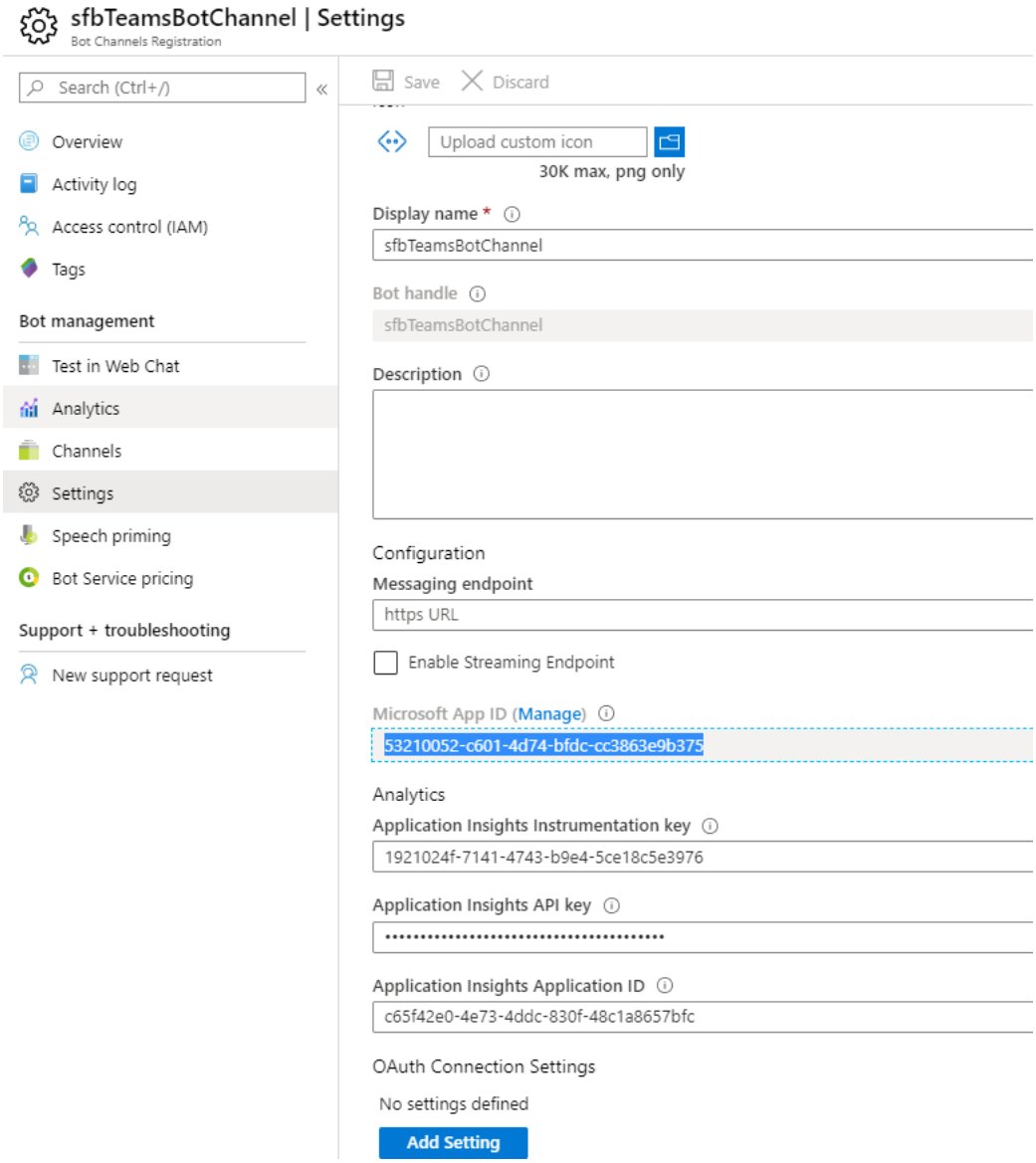
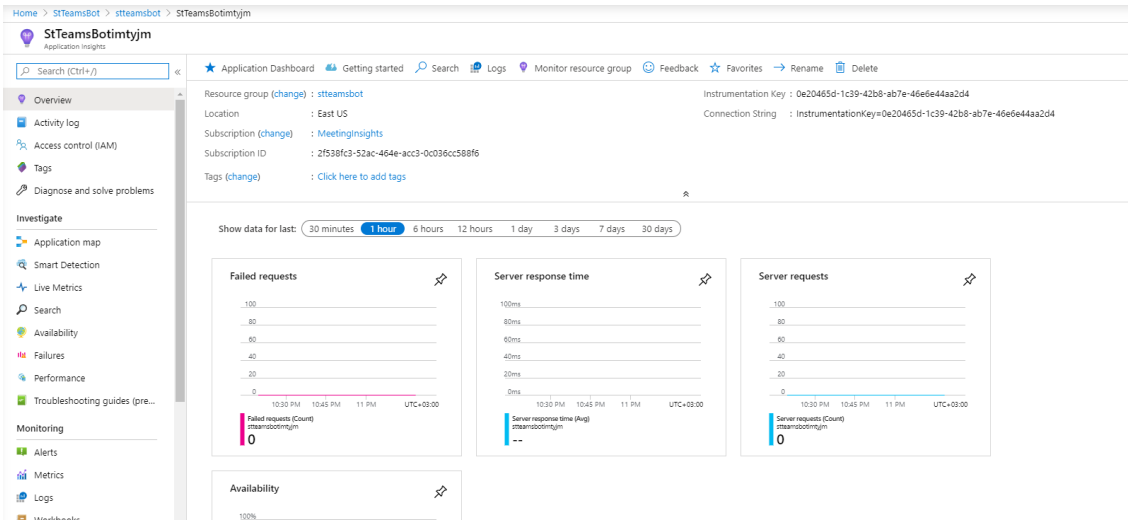


Figure 7-6: Configure Instrumentation Key



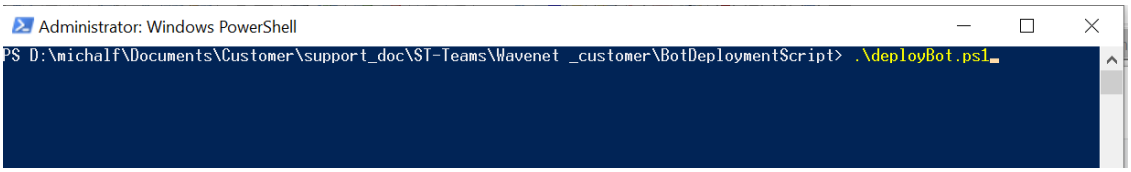
Step 3-2 Deploy BOT Package

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 7-7: Run Script



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

Figure 7-8: BOT Type

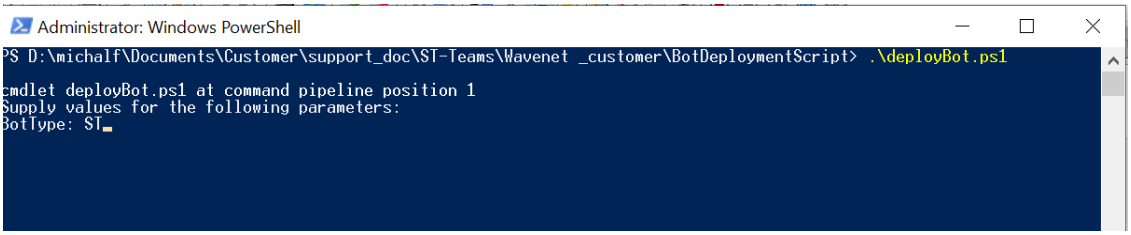
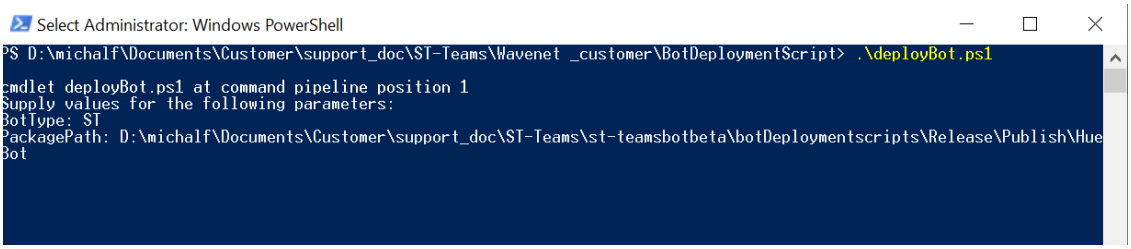


Figure 7-9: SmartTAPTeamsBOT



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

8 Step 4 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:

Step 4-1 Implement Prerequisites

The following prerequisites procedures must be implemented:

Step 4-1-1 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:**

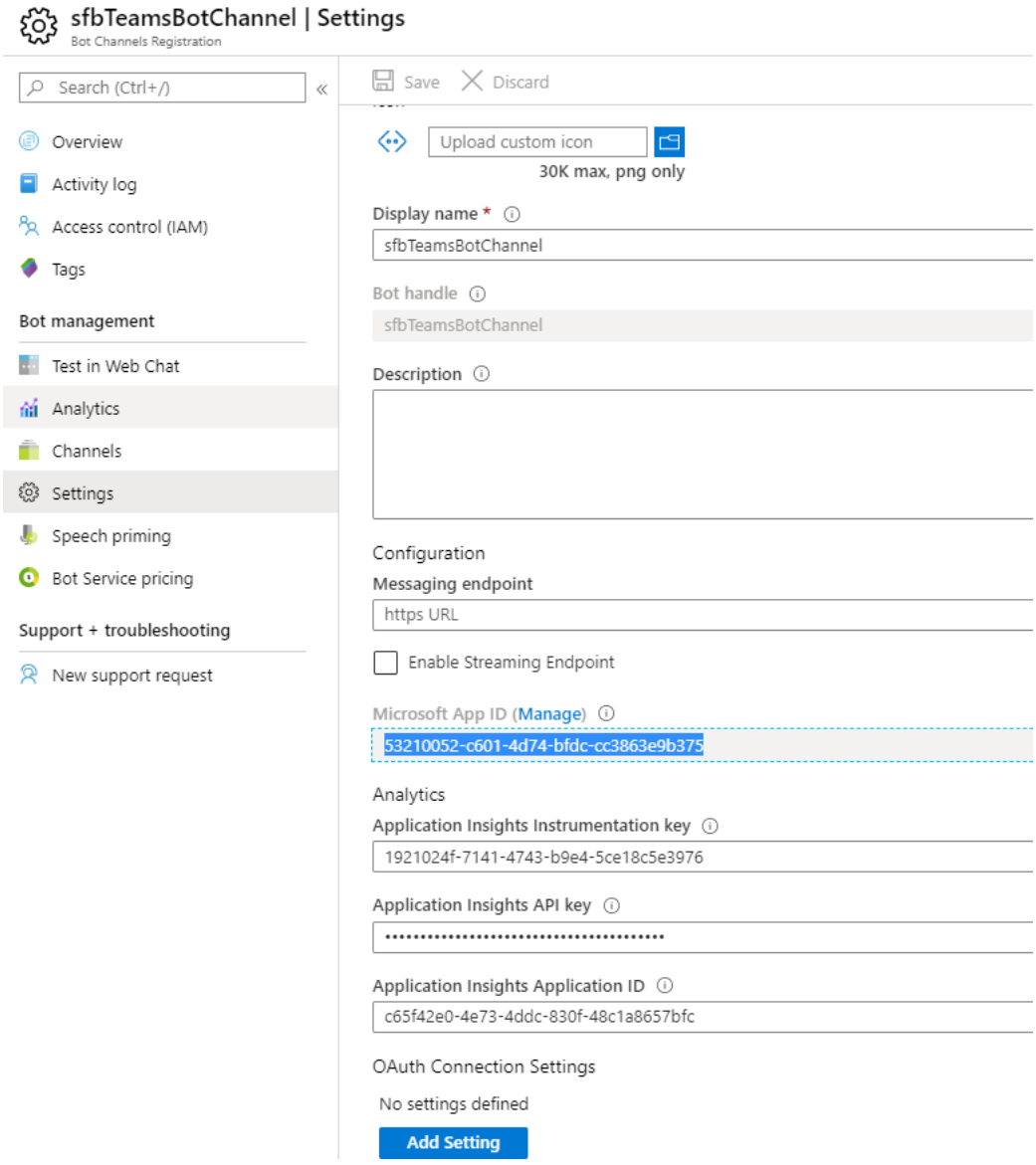
1. Paste the following URL in your browser with parameters shown below:
   https://login.microsoftonline.com/common/adminconsent?
   - client_id=XXXX

     Where XXXX is the SmartTAP 360° Bot app ID from BOT service that was created in Step 2-1 Configure Service Channel on page 14 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 8-1: BOT Channel Settings



Step 4-1-2 Set Azure Active Directory Read Permissions

This procedure describes how to set permissions to read Azure Active Directory. The procedure should be performed by the Office 365 administrator.

➢ **To set read permissions for Azure Active Directory:**

1. Paste the following URL with parameters shown below:

   https://login.microsoftonline.com/common/adminconsent?

   - client_id=YYYY

Where YYYY is the SmartTAP 360° Bot App ID from BOT service that was created in Step 2-1 Configure Service Channel on page 14 . This is required to authenticate your Azure subscription.

- &state=12345
- &redirect_uri=https://login.microsoftonline.com/common/oauth2/nativeclient
- &scope=

2. Paste the following URL with the parameters shown below:

   https://graph.microsoft.com/User.Read.All

   The Authentication Settings are displayed and the connection is authenticated.

3. Download SFB 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

4. Create 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

Step 4-2 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. Enter the following commands:

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

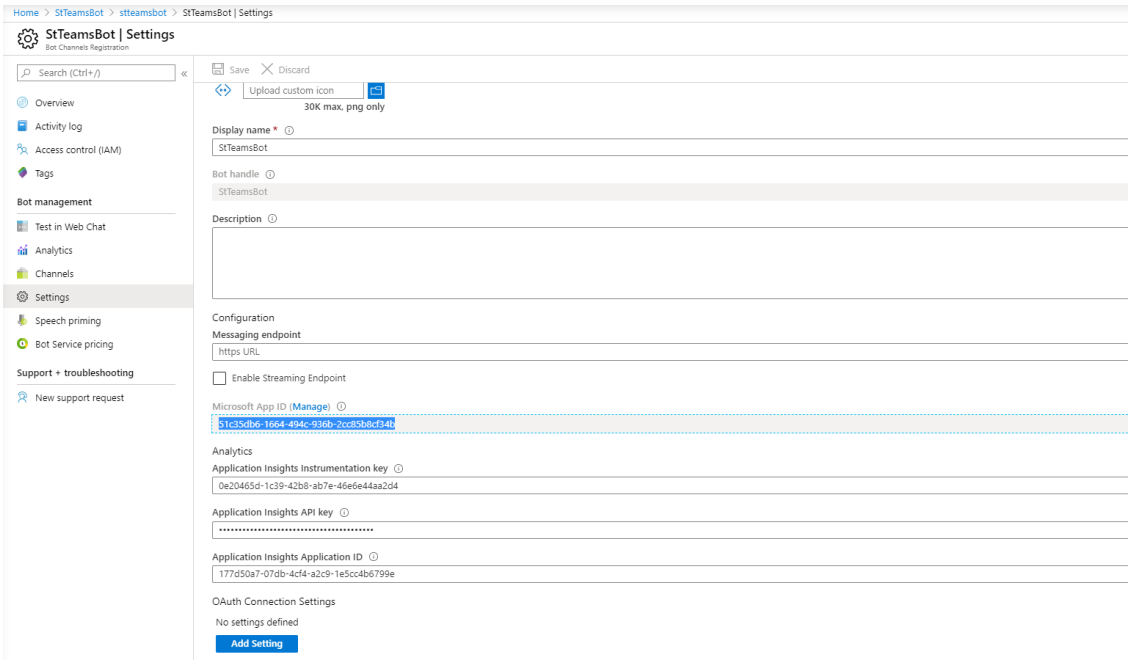
Where:

- <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 Step 2-1 Configure Service Channel on page 14). 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 8-2: Create Application Instance



2. 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 Step 4-3-2 Set Compliance Recording Policy on the next page.

Step 4-3 Create Compliance Recording Policy

This procedure describes how to create a Compliance Recording Policy.

Step 4-3-1 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

Step 4-3-2 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 Step 4-3-1 Create New Compliance Recording Policy on the previous page
- <ObjectID>: Object ID that was generated in Step 4-2 Create Application Instance on page 31
- -<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

Step 4-3-3 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

9 Step 5 Deploy SmartTAP 360° for Recording

This procedure describes how to deploy SmartTAP 360° including the following procedures:

- Step 5-1 Create SmartTAP 360° Virtual Machine below
- Step 5-2 Configure Microsoft Blob Storage on page 41

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

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

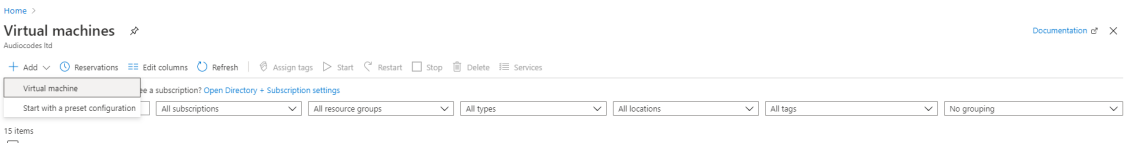
Step 5-1 Create SmartTAP 360° Virtual Machine

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.

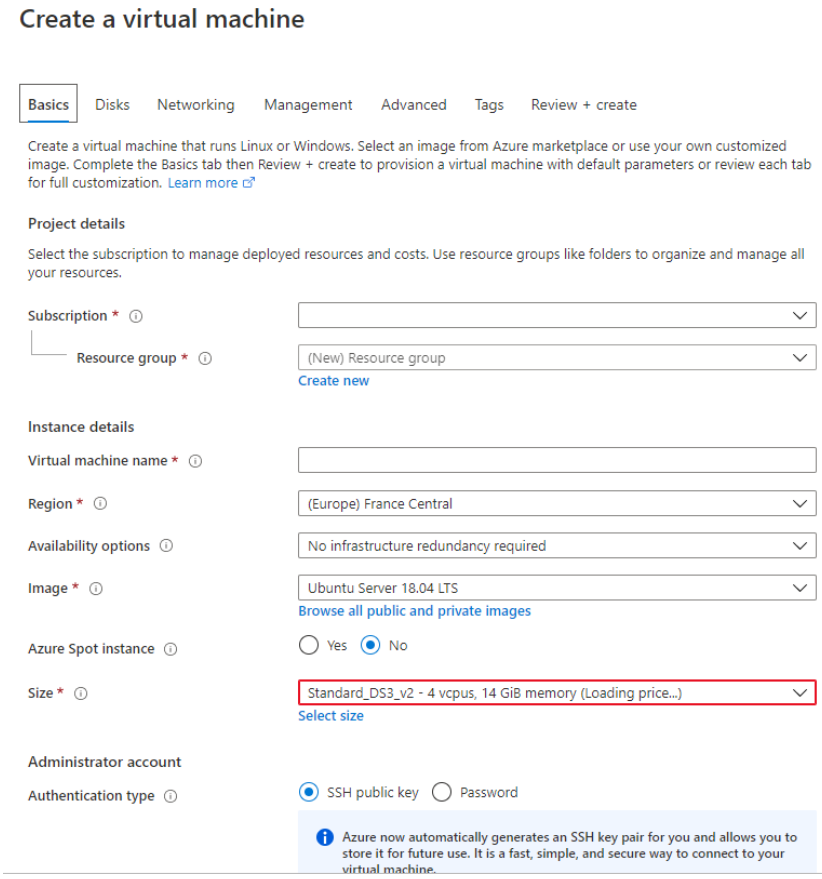
➢ Do the following:

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

Figure 9-1: Create Virtual Machine



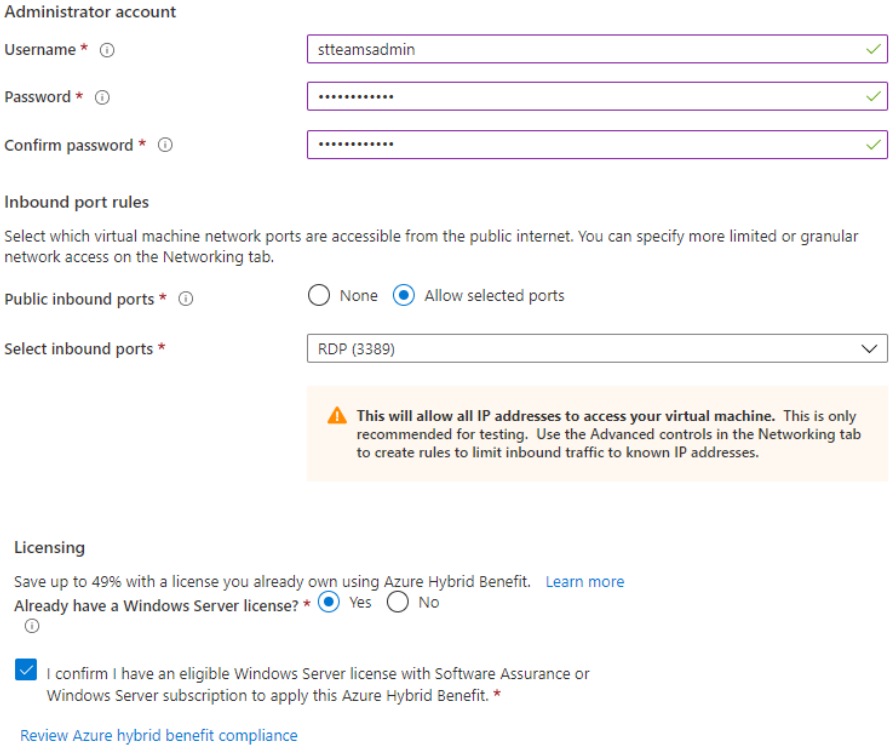
2. Click **Virtual machine** to create a virtual machine.



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

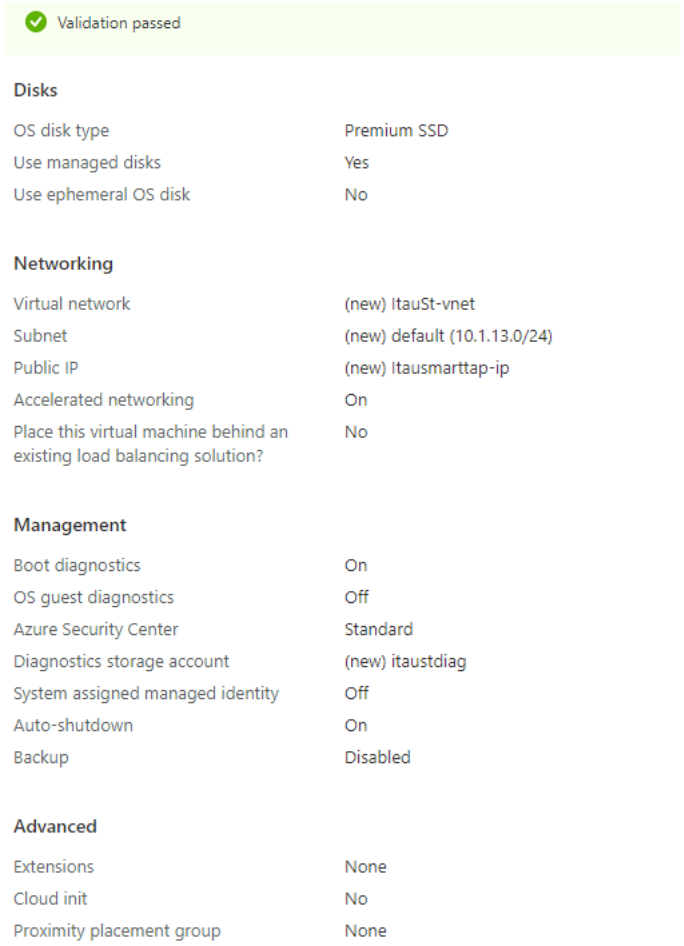
4. Select the relevant Virtual Machine specifications according to SmartTAP 360° for Microsoft Teams Specifications on page 2 and then click **Next**.

Figure 9-2: Administrator Account



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

Figure 9-3: Review and Create



6. Install SmartTap server from Installation Suite All-In-One mode on the VM that you just created. Refer to the SmartTAP 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 9-4: Inbound Firewall Rules

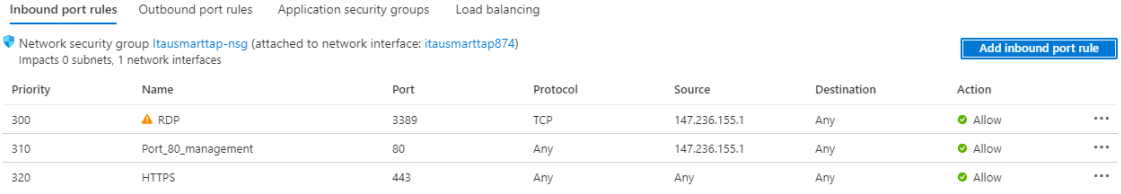


Table 9-1: Firewall Rules

Protocol	Ports	Connection	Description
TCP	80/443	Communication between BOT VMs and SmartTAP server in both directions.	Management/Signaling between BOT and SmartTAP (On SmartTAP Azure NSG).
TCP	80/9441	Communication between BOT VMs	Load Balancer Address Pool (part of SFC deployment script)
TCP	19080/ 19000	Communication between BOT VMs	Load Balancer HTTP Fabric Gateway Probe (part of SFC deployment script)
TCP	443	Communication from BOT VMs to Teams	Signaling connection
TCP	9444-9544	Communication between Teams to BOT VMs in both directions.	Signaling inbound NAT rules (part of SFC deployment script)
TCP	8445-8545	Communication between Teams to BOT VMs in both directions.	Media TCP inbound rules (part of SFC deployment script)
UDP	3478-3481	Communication between Teams to BOT VMs in both directions.	Media relay ports (part of SFC deployment script)
TCP	3389-3392	Recommended for BOT VMs access	RDP ports Inbound rules (part of SFC deployment script)

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

Step 5-2 Configure Microsoft Blob Storage

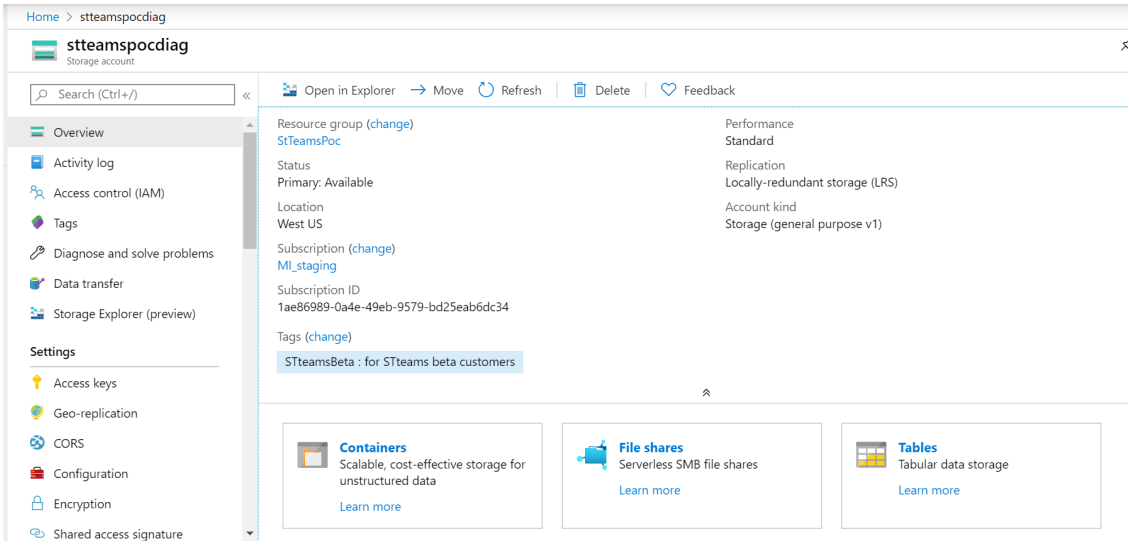
This section describes how to configure Microsoft Blob Storage as the external storage platform for storing recorded media.

> On-premises storage is currently not supported for SmartTAP 360° Microsoft Teams deployments.

➢ To configure Microsoft Blob:

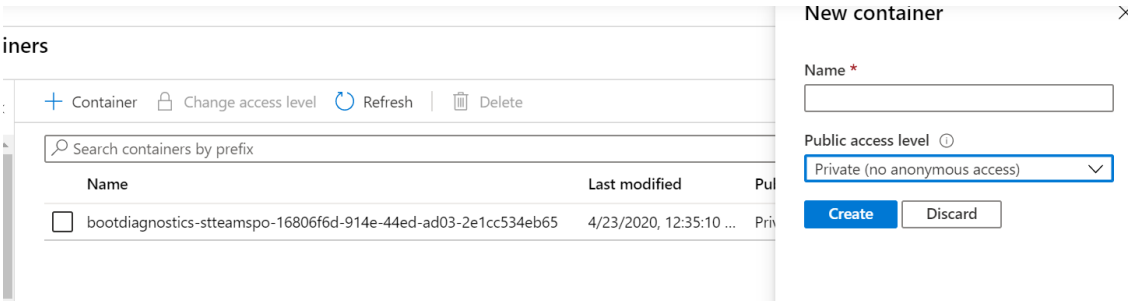
1. Log on to the Azure portal and open the Storage account settings page.
2. Create or use existing storage account.

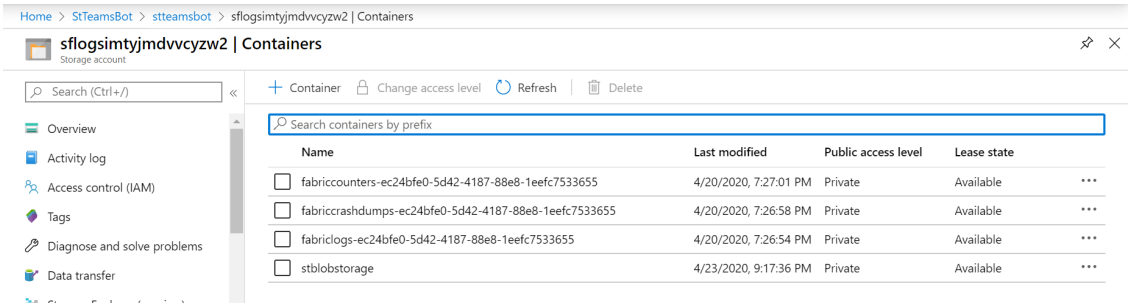
Figure 9-5: Microsoft Blob Storage Account



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

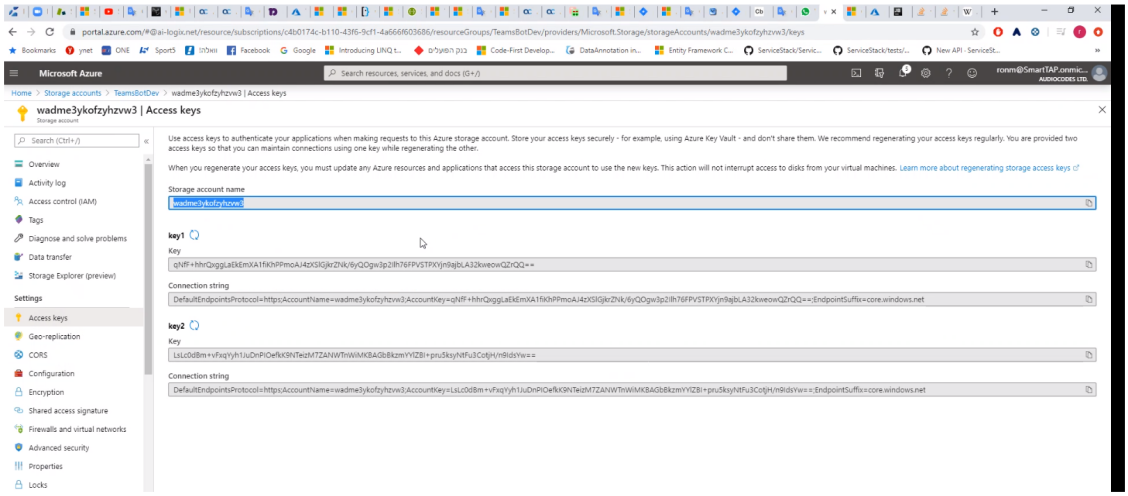
Figure 9-6: Create New Blob Container





5. Save the storage name and credentials.

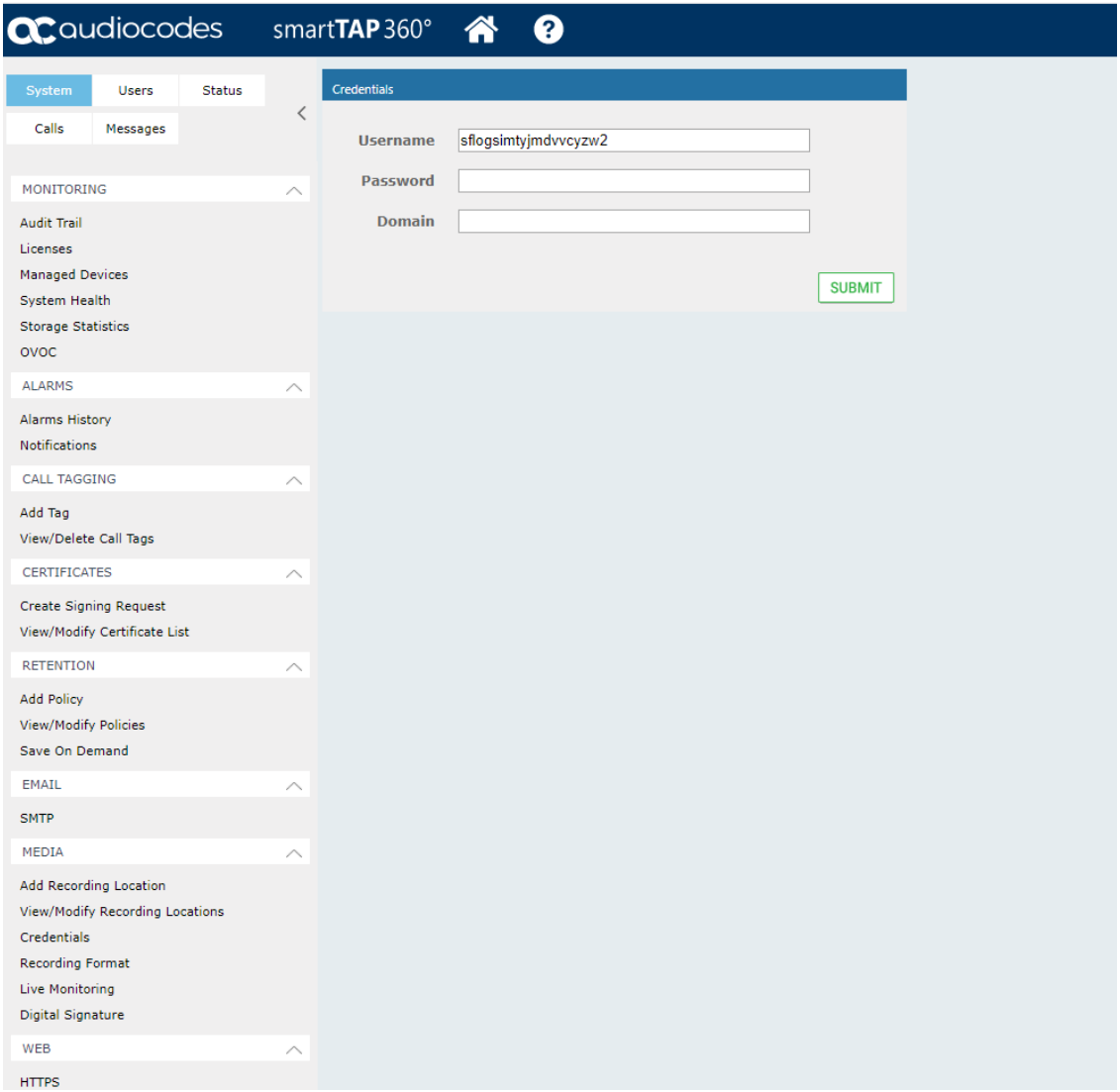
Figure 9-7: Storage Name and Credentials



6. Open the SmartTAP 360° Web interface and then open the Credentials screen (**System** menu > **Media** folder > **Credentials**):

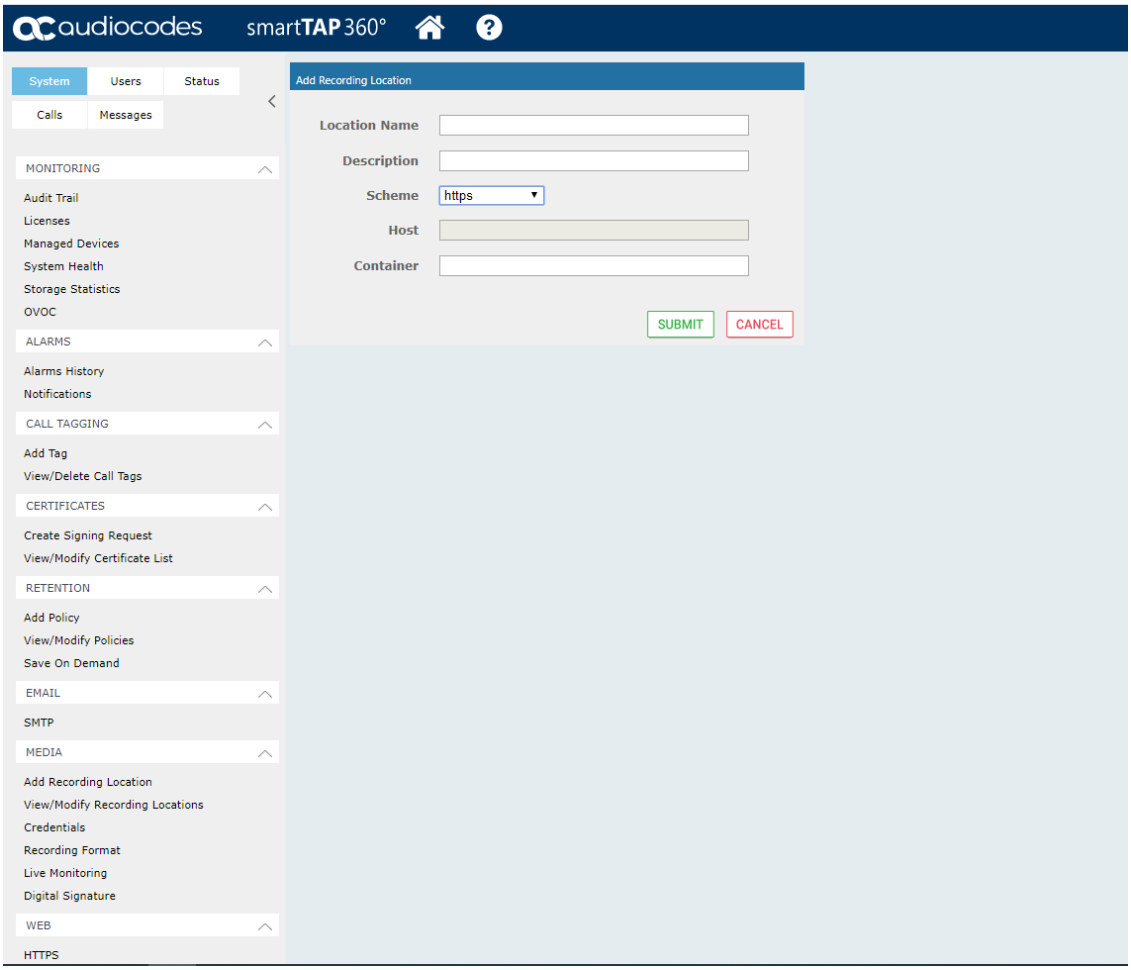
   - Set Username:" Storage account name" (where container was created).
   - Set Password: "access key" (from the storage account).

Figure 9-8: Credentials Screen



7. Open the Add Recording Location screen (**System** menu > **Media** folder > **Add Recording Location**).

Figure 9-9: Add Recording Location



8. From the Scheme drop-down list, select 'https' and insert the container name created for the Azure storage account.
9. Click SUBMIT.

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-27326