AppGate SDP Name Resolving in Azure

Step-by-Step Setup Guide

Last revised Aug 8, 2017

Table of Contents

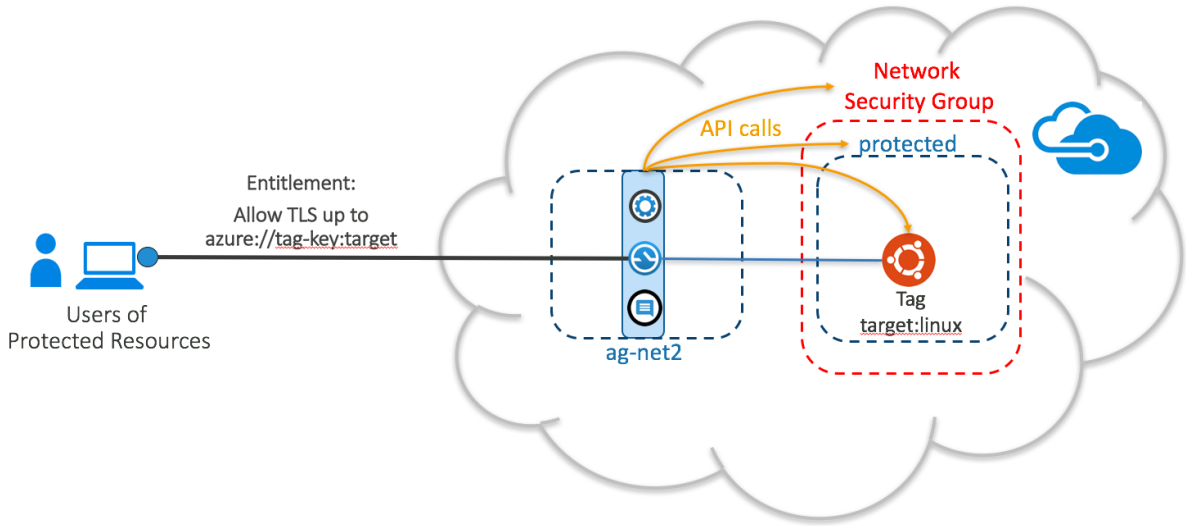
1 Overview

Enterprises continue to rapidly embrace Microsoft Azure, but securing access to these cloud-based workloads can be a challenge. AppGate is purpose-built for the Azure environment and draws on user context to dynamically create a secure, encrypted network segment of one that's tailored for each user session. It dramatically simplifies the Cloud resource user access challenge and eliminates IP-based over-entitled network access. AppGate provides a means for security teams to efficiently and effectively control user access to Azure resources.

AppGate is a distributed network access control system that creates a unique access filter for each user/device combination. This patented access system dynamically matches the context information from the user and device with the context information it polls in real-time from the Cloud provider. This ability to adapt in real-time to changing conditions in the Azure infrastructure means that every new instance that is added or removed will now automatically be tracked and added or removed from the access filter, without the need to change any policies.

Let's take a look at how we need to configure both Azure and AppGate to be able to use this capability to adapt in real-time to changing conditions in Azure.

All user traffic is tunneled from their device (via a virtual network adapter, similar to a VPN client), and passed through the AppGate gateway to the protected resources. The Entitlement in this case does not define an IP address or a host name but instead uses a *resource name.* The Gateway can then make a number of different API calls into the Azure environment to discover hosts or the contents of a Virtual Network or a Network Security Group.



This **Getting Started** guide will take you through the steps necessary to set up and configure both AppGate and Azure so that these *resource names* can be used and resolved in real time.

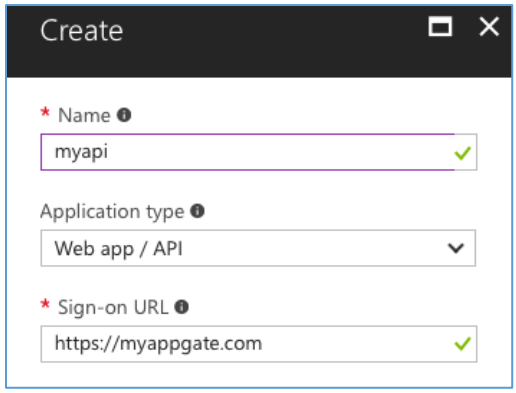
2 Allow API access into Azure

By default AppGate is not allowed to use any Azure APIs so we need to configure Azure to allow this. Additionally there are 4 pieces of information we need to be able to configure the AppGate side so we need to extract these from Azure.

Register AppGate as an API user and collect the related details

Begin by logging into your Azure Portal.

From the start page select **more services** from the left side menu. In the **Filter box** type *app* and select *app registrations*. Click on **New application registration** and a new blade will open.



Enter a name such as *myapi*

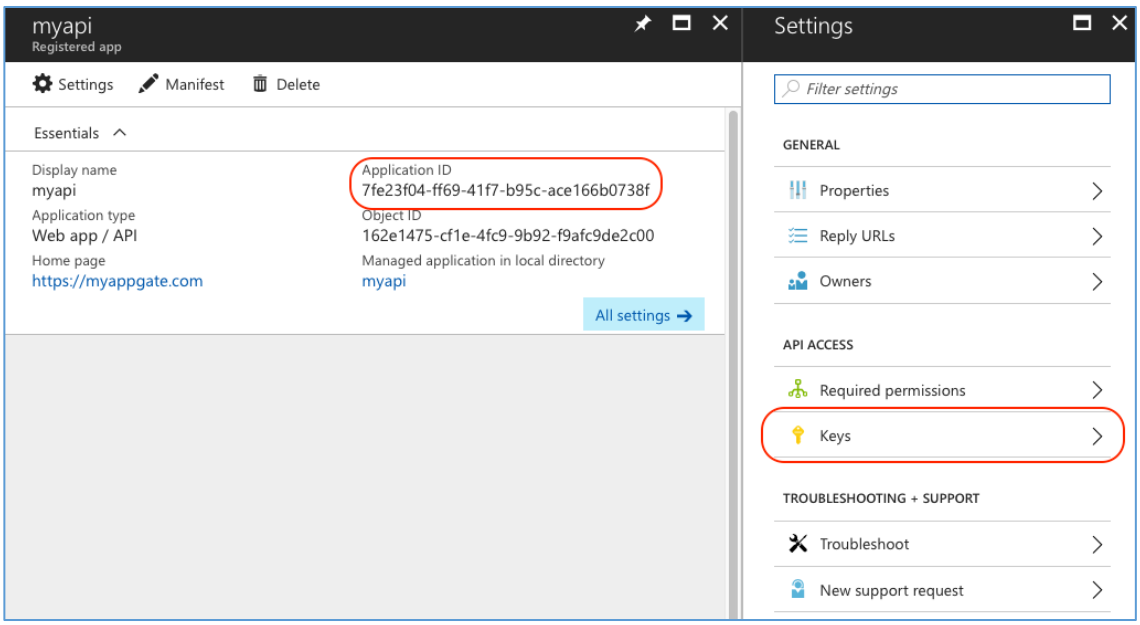
Leave the Application type as *Web app / API*

Sign-on URL is not used but is required. So enter something valid such as *https://myappgate.com*

Then click on **Create**.

The blade will then close.

Now find the new app registration from the list and open it. Two blades will open:



Make a note of the **Application ID** and then open **Keys.** In the Keys blade add a new key and make a note of the **Value**. This will appear when you hit Save but is gone thereafter so don't lose it!

For the last two pieces of information we need it is easiest to use the Azure CLI 2.0; so please install this on the PC you are using. It is also possible to find it in the Azure portal as mentioned in the table in section 3.

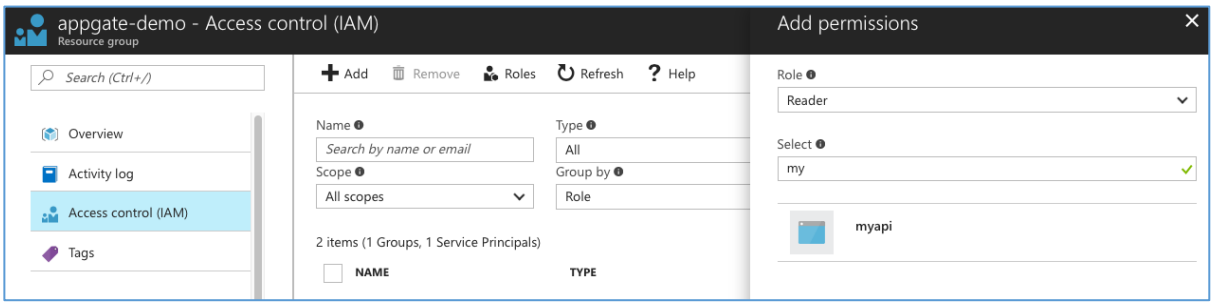
Give the new API rights to the Resource Group

From the Azure portal start page select *Resource groups* from the left side menu.

Select the Resource group where the protected resources behind the AppGate Gateway reside. In the first blade click on **Access control (IAM).**

In the next blade click on **+Add**.

In the Add permissions blade: **Role** – choose *Reader* role; **Select** – search for the new app we created earlier.



Select the new app and then click **Save**.

3 Adding name resolver information to Azure resources

In the Azure UI we need to add some information for our resolver to use. This may not be required if the information is already available in the current configuration.

The full set of *resource name* formats can be found here:
https://help.cryptzone.com/adminguide/hosts-resolving.html

These normally involve resolving resource NAMES, tag NAME, tag VAUE or tag NAME:VALUE. If we want to use tags to resolve hosts then we need to update our target hosts/networks/etc with a tag NAME:VALUE we can look for.

Open the Resource and click on Tags. In the Tags blade add *Name* and *Value*. The AppGate API call will now try to resolve these when a user connects.

4 Finding Subscription & Tenant IDs

Open the terminal and type *azure login*. This will redirect you to login in your browser. Once logged in issue the command: *azure account show*.

You should see something like:

```
info:    Executing command account show
data:    Name                  : Microsoft Azure Enterprise
data:    ID                    : 01234567-792e-4119-825b-a1b2c3d4e5f6
data:    State                 : Enabled
data:    Tenant ID             : fedcba98-d025-4c03-aaa5-a1b2b3b4b5b6
data:    Is Default            : true
data:    Environment           : AzureCloud
data:    Has Certificate       : No
data:    Has Access Token      : Yes
data:    User name             : username@company.com
```

Make a note of the **ID** and the **Tenant ID**.

Now it is time to move on to configure the AppGate side of things.

5 AppGate - Name Resolver

Name resolvers are configured on a per Site basis – so all you need to configure is in one place:

Configure the AppGate Site

Go to **Sites>Name Resolution>Azure resolvers**:

Click **+** for a new resolver.

Make the Name reflect the Resource Group in Azure. Leave the update interval at its default (this is how frequently the AppGate server calls into Azure to check for server changes). Then enter the 4 pieces of information we collected earlier:

Subscription ID:	Use *ID*. Also available in Azure portal: Resource Group > Overview
Tenant ID:	Use *Tenant ID*. Also available in Azure portal by hovering over your user name (top right)
Client ID:	Use *Application ID*.
Secret:	Use *Key Value*.

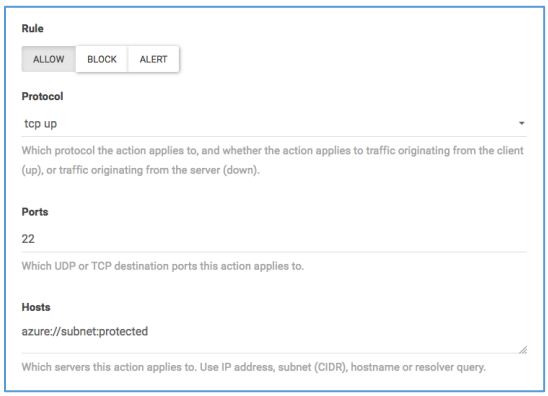
6 AppGate - Entitlement

Configuring the entitlement to use a *resource name* in place of an IP address or a host name is the final step.

Edit your entitlement and in the **Actions**>**Hosts** use the *resource name* you require.

azure://subnet:protected

In this case the API will return all the IPs of host located in the subnet *protected*.



7 Resources and Community

Cryptzone has an online AppGate SDP for Azure community here: https://cryptzone.vbulletin.net/

We encourage you to register and join the conversation! Here you will find information from other AppGate SDP for Azure users and the experiences they've had getting up and running as well as using it on a day-to-basis.

In addition to the AppGate SDP for Azure online community, you'll find additional resources on the Cryptzone website here: https://www.cryptzone.com/downloadcenter/appgate-sdp-for-azure

And the AppGate SDP product documentation is available here:

- Admin Guide: https://help.cryptzone.com/adminguide
- Client User Guide: https://help.cryptzone.com/userguide

Thank you, and we hope you find AppGate SDP to be a valuable solution to your security challenges.