

Azure VM Monitoring

This azure function will monitor some pre-defined and user defined tags for assets deployed in one or more Resource Groups (RGs) in your azure environment
These tags are then pushed to a firewall of your choice.
This feature is similar to the VM-Monitoring feature built into PAN-OS.

<https://www.paloaltonetworks.com/documentation/70/pan-os/pan-os/policy/enable-vm-monitoring-to-track-changes-on-the-virtual-network>

Overview

The solution is distributed as a python script that runs as an Azure Function.
<https://azure.microsoft.com/en-us/services/functions/>

The script has been purposely written to have no external dependencies since python support in Azure Functions is still experimental!

Use portal to create an Azure Active Directory application and service principal that can access resources

In order to make API calls into the Azure environment one needs to first and foremost create a service principal and then register your application with the Azure AD environment. This will provide you with various keys/IDs that will be used to generate an Azure Bearer Token that will be used in the header during the REST calls.

Please follow the instructions here to create an Azure Active Directory application and service principal:

<https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-create-service-principal-portal>

Ideally you can start from the “Create an Azure Active Directory application” portion (if your account has the ability to register applications). If not, you may need to talk to the account admin to allow the ability to register applications.

To be able to assign a role to a Service Principal, you need to have the “User Access Administrator” assigned to your account. A potential workaround could be to have an administrator assign “Contributor” role to the Service Principal on the user’s behalf.

Once you have registered your application (you can name it anything you like) you will need to note down a few things (these are also listed in the link above):

Application ID

Secret key

Note: Make sure you note this down because the key will be hidden once you navigate away

Tenant ID

Lastly, Subscription ID: [<https://blogs.msdn.microsoft.com/mschray/2016/03/18/getting-your-azure-subscription-guid-new-portal/>]

In the link above when you get to the “Assign application to role” part it is possible that you are the admin of your account and can assign an IAM role to your app. If so, add your application and assign it a reader role.

Create an Azure Function using the Azure Portal

Log into the Azure portal (<https://portal.azure.com>) and create your Function App.

Microsoft Azure New

New

+ New

Dashboard

All resources

Resource groups

Virtual machines

Storage accounts

Network security groups

Route tables

Application gateways

Virtual networks

Availability sets

Cost Management + Billing

Container registries

Container services

Load balancers

Virtual machine scale sets

Azure Active Directory

App Services

Function Apps

Resource Explorer

Local network gateways

Virtual network gateways

More services >

Search the Marketplace

Azure Marketplace See all

Featured See all

Get started

Compute

Networking

Storage

Web + Mobile

Databases

Data + Analytics

AI + Cognitive Services

Internet of Things

Enterprise Integration

Security + Identity

Developer tools

Monitoring + Management

Add-ons

Containers

Blockchain

Windows Server 2016 Datacenter
Quickstart tutorial

Red Hat Enterprise Linux 7.2
Learn more

Ubuntu Server 16.04 LTS
Quickstart tutorial

SQL Server 2016 SP1 Enterprise on Windows Server 2016
Learn more

Virtual machine scale set
Learn more

Azure Container Service
Learn more

Azure Container Registry
Learn more

Web App for Containers
Learn more

Azure Container Instances (preview)
Learn more

Function App
Quickstart tutorial

The image is a screenshot of the Microsoft Azure portal's 'New' page. On the left is a dark sidebar with a list of services. The 'New' button at the top of this sidebar is highlighted with a red box. A red arrow points from this button to the 'Compute' category in the 'Azure Marketplace' section. The 'Compute' category is also highlighted with a red dashed box. Another red arrow points from 'Compute' down to the 'Function App' item in the 'Featured' section. The 'Function App' item is highlighted with a red box. The 'Function App' item includes a lightning bolt icon and a link to a 'Quickstart tutorial'.

On the next screen enter your App Name, Select your subscription that you want to use for this App, Create a new Resource Group (or use an existing one).

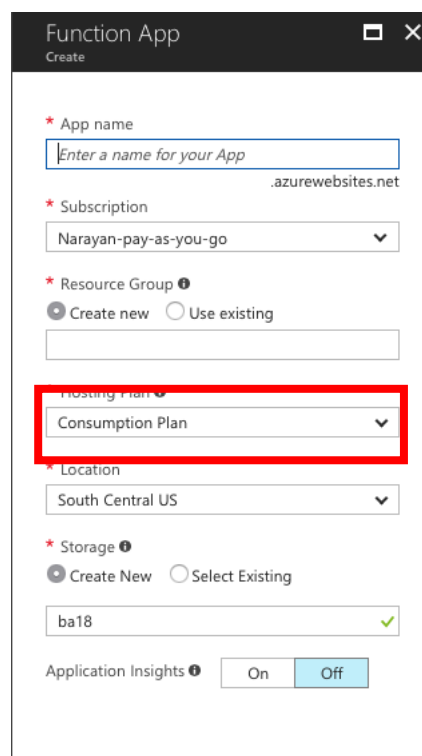
The next step is determined by the location of the firewall that receives the ip-to-tag mapping.

Publically addressable firewalls

For firewalls that are publicly accessible you can choose the **Consumption plan**.

This plan allows you to pay-per-execution (and dynamically allocates resources based on your app's load)

More information here: <https://azure.microsoft.com/en-us/pricing/details/functions/>



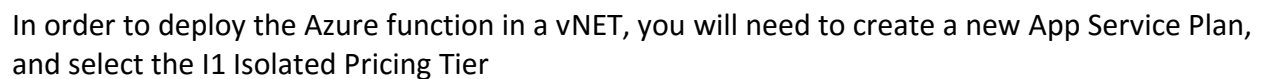
The screenshot shows the 'Function App Create' form in the Azure portal. The form includes the following fields and options:

- App name:** A text input field with the placeholder 'Enter a name for your App' and a '.azurewebsites.net' domain suffix.
- Subscription:** A dropdown menu showing 'Narayan-pay-as-you-go'.
- Resource Group:** Radio buttons for 'Create new' (selected) and 'Use existing', followed by an empty text input field.
- Hosting Plan:** A dropdown menu showing 'Consumption Plan', which is highlighted with a red rectangular box.
- Location:** A dropdown menu showing 'South Central US'.
- Storage:** Radio buttons for 'Create New' (selected) and 'Select Existing', followed by a text input field containing 'ba18' with a green checkmark.
- Application Insights:** A toggle switch with 'On' and 'Off' buttons, currently set to 'Off'.

Enter a name for your function app. You can choose to launch it in an existing Resource Group or create a new one. Same goes for your storage account. Pick a location, and hit Create

Firewalls deployed in Azure (in the same account/subscription)

For firewalls that do not have publically addressable management interfaces you can choose the Hosting Plan to be an App Service Plan. This will allow you to “associate” the App Service Plan with a vNET. This vNET should be the vNET where all the firewall(s) reside.



1	2	4	8	16	32	64	128	256	512	1024	2048	4096	8192	16384	32768	65536	131072	262144	524288	1048576	2097152	4194304	8388608	16777216	33554432	67108864	134217728	268435456	536870912	1073741824	2147483648	4294967296	8589934592	17179869184	34359738368	68719476736	137438953472	274877906944	549755813888	1099511627776	2199023255552	4398046511104	8796093022208	17592186044416	35184372088832	70368744177664	140737488355328	281474976710656	562949953421312	1125899906842624	2251799813685248	4503599627370496	9007199254740992	18014398509481984	36028797018963968	72057594037927936	144115188075855872	288230376151711744	576460752303423488	1152921504606846976	2305843009213693952	4611686018427387904	9223372036854775808	18446744073709551616	36893488147419103232	73786976294838206464	147573952589676412928	295147905179352825856	590295810358705651712	1180591620717411303424	2361183241434822606848	4722366482869645213696	9444732965739290427392	18889465931478580854784	37778931862957161709568	75557863725914323419136	151115727451828646838272	302231454903657293676544	604462909807314587353088	1208925819614629174706176	2417851639229258349412352	4835703278458516698824704	9671406556917033397649408	19342813113834066795298816	38685626227668133590597632	77371252455336267181195264	154742504910672534362390528	309485009821345068724781056	618970019642690137449562112	1237940039285380274899124224	2475880078570760549798248448	4951760157141521099596496896	9903520314283042199192993792	19807040628566084398385987584	39614081257132168796771975168	79228162514264337593543950336	158456325028528675187087900672	316912650057057350374175801344	633825300114114700748351602688	1267650600228229401496703205376	2535301200456458802993406410752	5070602400912917605986812821504	10141204801825835211973625643008	20282409603651670423947251286016	40564819207303340847894502572032	81129638414606681695789005144064	162259276829213363391578010288128	324518553658426726783156020576256	649037107316853453566312041152512	1298074214633706907132624082305024	2596148429267413814265248164610048	5192296858534827628530496329220096	10384593717069655257060992658440192	20769187434139310514121985316880384	41538374868278621028243970633760768	83076749736557242056487941267521536	166153499473114484112975882535043072	332306998946228968225951765070086144	664613997892457936451903530140172288	1329227995784915872903807060280344576	2658455991569831745807614120560689152	5316911983139663491615228241121378304	10633823966279326983230456482242756608	21267647932558653966460912964485513216	42535295865117307932921825928971026432	85070591730234615865843651857942052864	170141183460469231731687303715884105728	340282366920938463463374607431768211456	680564733841876926926749214863536422912	1361129467683753853853498429727072845824	272225893536750770770699685945414569152	544451787073501541541399371890829138304	1088903574147003083082798743781658276608	2177807148294006166165597487563316553216	4355614296588012332331194975126633106432	8711228593176024664662389950253266212864	174224571863520493293247799005065244256	348449143727040986586495598010130488512	696898287454081973172991196020260977024	1393796574908163946345982392040521954048	2787593149816327892691964784081043908096	5575186299632655785383929568162087816192	11150372599265311570767859136324173632384	2230074519853062314153571827264834726476
---	---	---	---	----	----	----	-----	-----	-----	------	------	------	------	-------	-------	-------	--------	--------	--------	---------	---------	---------	---------	----------	----------	----------	-----------	-----------	-----------	------------	------------	------------	------------	-------------	-------------	-------------	--------------	--------------	--------------	---------------	---------------	---------------	---------------	----------------	----------------	----------------	-----------------	-----------------	-----------------	------------------	------------------	------------------	------------------	-------------------	-------------------	-------------------	--------------------	--------------------	--------------------	---------------------	---------------------	---------------------	---------------------	----------------------	----------------------	----------------------	-----------------------	-----------------------	-----------------------	------------------------	------------------------	------------------------	------------------------	-------------------------	-------------------------	-------------------------	--------------------------	--------------------------	--------------------------	---------------------------	---------------------------	---------------------------	---------------------------	----------------------------	----------------------------	----------------------------	-----------------------------	-----------------------------	-----------------------------	------------------------------	------------------------------	------------------------------	------------------------------	-------------------------------	-------------------------------	-------------------------------	--------------------------------	--------------------------------	--------------------------------	---------------------------------	---------------------------------	---------------------------------	----------------------------------	----------------------------------	----------------------------------	----------------------------------	-----------------------------------	-----------------------------------	-----------------------------------	------------------------------------	------------------------------------	------------------------------------	-------------------------------------	-------------------------------------	-------------------------------------	-------------------------------------	--------------------------------------	--------------------------------------	--------------------------------------	---------------------------------------	---------------------------------------	---------------------------------------	--	--	--	--	---	---	---	--	---	---	--	--	--	--	---	---	---	--	--	--	---	--

You will need the tier that offers the option “Runs in your vNET (Network Isolated)”. As of writing this document, the cheapest tier that offers network isolation is the I1 tier.

The screenshot displays three panels from the Azure portal:

- Function App:** Shows the 'Create' page with fields for App name (vm-monitoring-app), Subscription (Narayan-pay-as-you-go), Resource Group (vm-monitoring-app), Hosting Plan (App Service Plan), App Service plan/Location (ServicePlan537e22ac-8345(South...)), Storage (vmmonitoringappbdc?), and Application Insights (On).
- App Service plan:** Shows the 'Select a plan for the web app' page with a 'Create New' button and a list of plans, including 'ServicePlan537e22ac-8345(51) (New)' in South Central US.
- New App Service Plan:** Shows the 'Create a plan for the web app' page with fields for App Service plan (app-service-plan-name), Location (South Central US), Pricing tier (I1 Isolated), App Service Environment Name (app-service-environment-name), Virtual Network (fwVNETa32a), Virtual Network Address Block (10.5.0.0/16), Subnet Name (random-subnet), and Subnet Block Size (16 Available Addresses (/28)).

Once you have selected your vNET, and filled in the required parameters, Create the function.

Note: Creating of the App service environment takes a very long time. Sometimes it can take about 2 hours.

Clone the GitHub Repository

The next step is to clone the repo that hold the Azure Function script.

Please follow instructions listed here:

<https://help.github.com/articles/cloning-a-repository/>

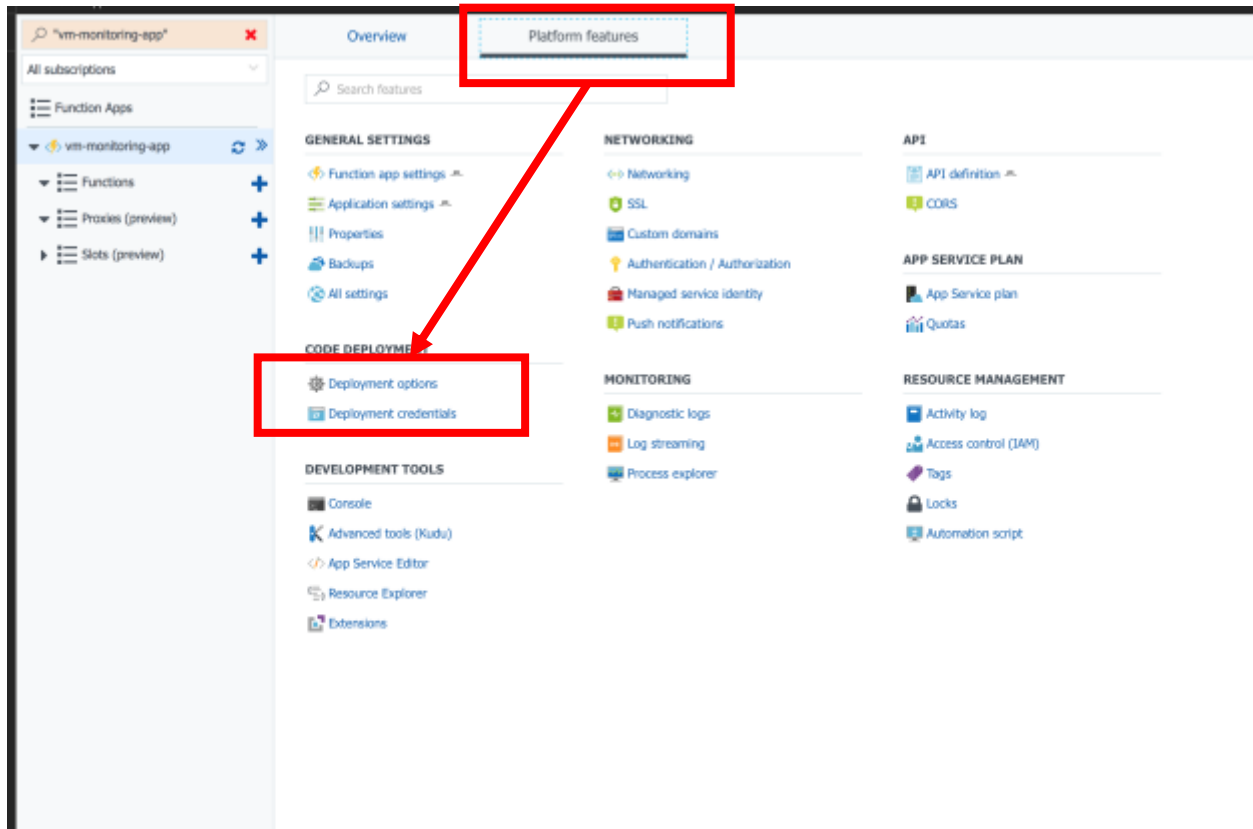
The URL for the repository to clone is:

<https://github.com/PaloAltoNetworks/azure-vm-monitoring.git>

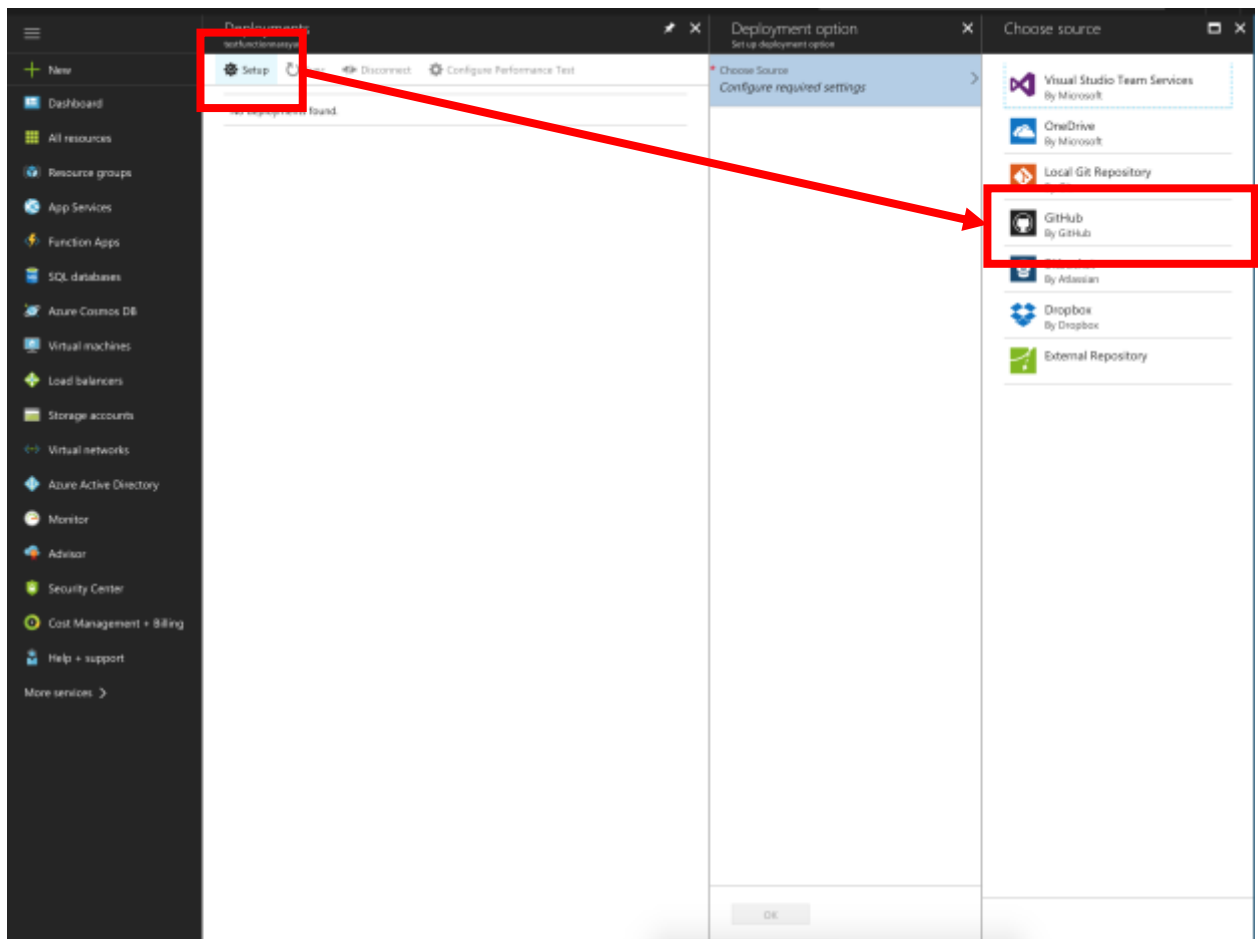
Setup Continuous Deployment for your Azure Function

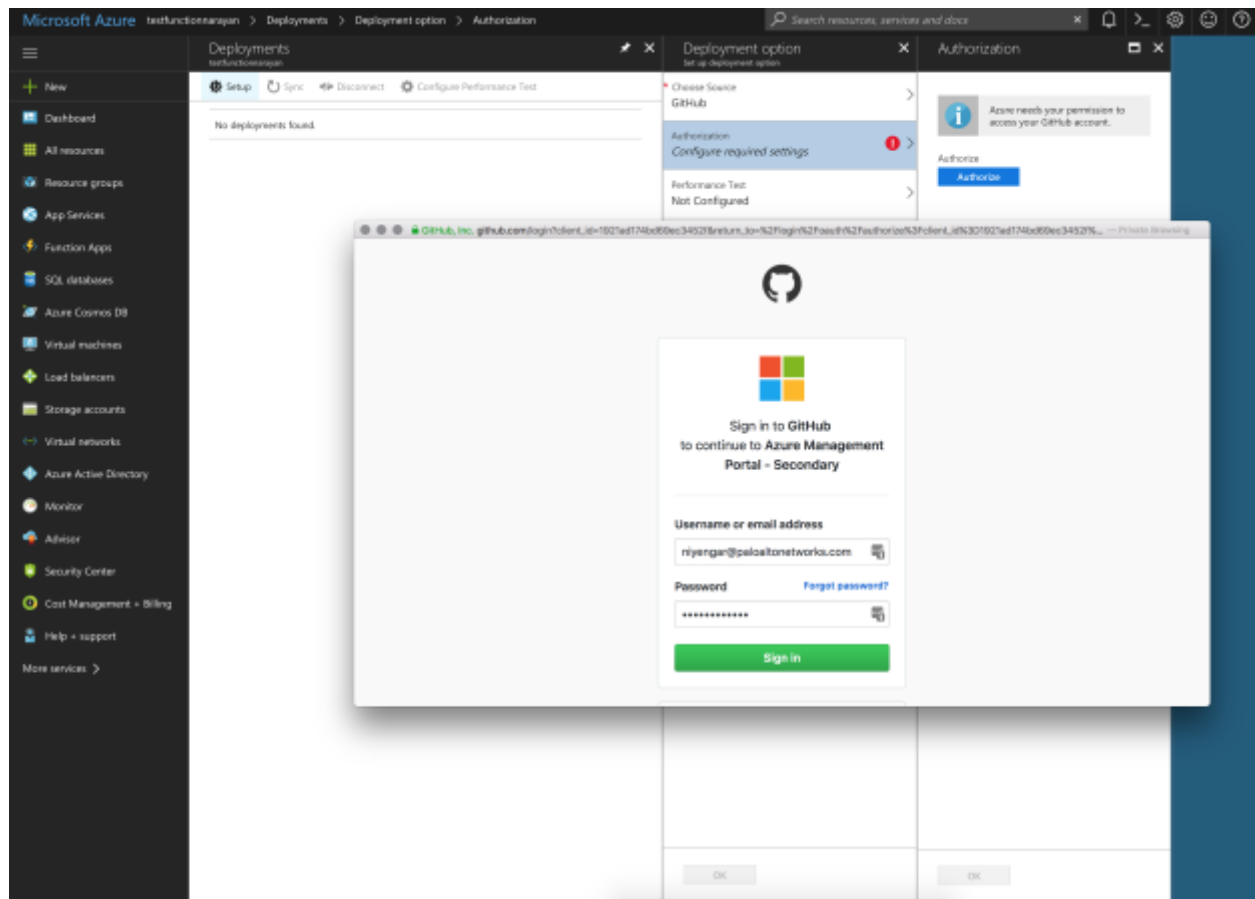
Once the Function App has been created, you can connect the GitHub repo to the function, to enable continuous deployment.

Navigate to the Function App blade on your Azure portal and click on the function that was just created. And click on the Platform Features tab, and click on Deployment Options



If you have never connected your GitHub account to Azure then you should be able to click on Setup and login and authorize Azure to access your GitHub profile





Once signed in and Authorized, you can select the GitHub repo where your code has been modified.

Select the appropriate repo and branch. You can leave the Performance Test option un-configured.

Setup

Sync

Disconnect

Configure Performance Test

No deployments found.

* Choose Source
GitHub


* Authorization
narayan-iyengar

* Choose your organization
PaloAltoNetworks


* Choose project
azure-vm-monitoring

* Choose branch
master

Performance Test
Not Configured



Click here if you are having trouble seeing your repositories.

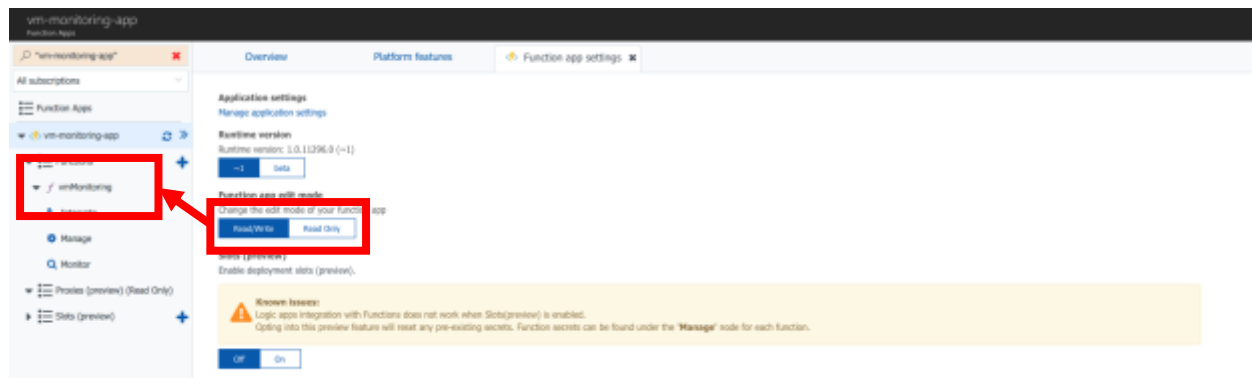
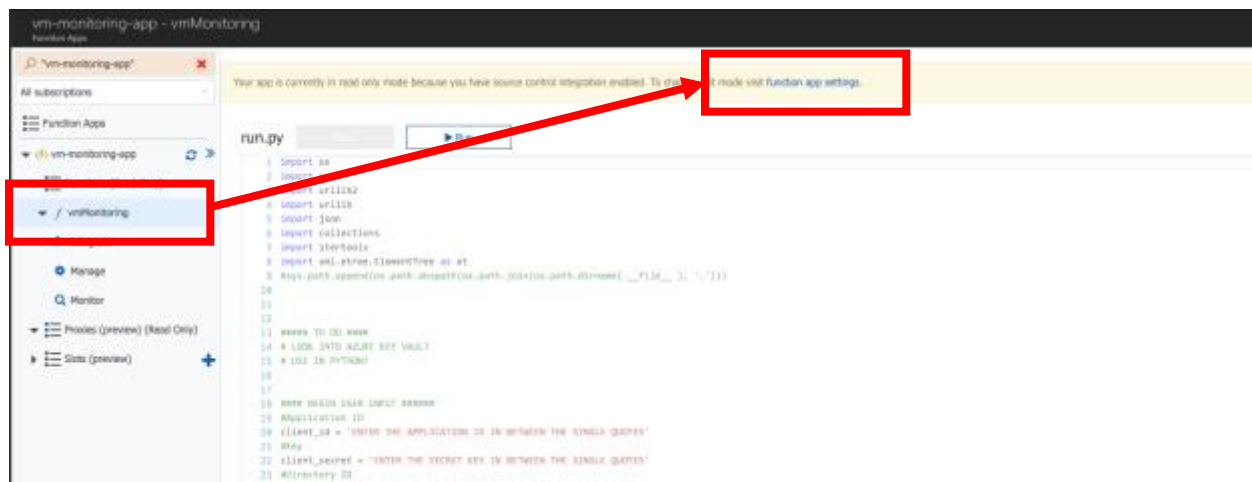


Once setup, the code is automatically pulled from GitHub and deployed.

Edit the script

Before the code will function correctly, the script needs to be edited.

In the Azure function portal edit the function app settings and make the function Read/Write:



Head back to the function and edit the file by following the instructions in the code.

You will need the following information:

Application ID

Access Key

Directory ID

Azure Subscription ID

Your app is currently in read\write mode because you've set the edit mode to read\write despite having source control enabled. Any changes you make may

run.py

Save

▶ Save and run

```
1 import os
2 import sys
3 import urllib2
4 import urllib
5 import json
6 import collections
7 import itertools
8 import xml.etree.ElementTree as et
9 #sys.path.append(os.path.abspath(os.path.join(os.path.dirname( __file__ ), '..')))
10
11
12
13 ##### TO DO #####
14 # LOOK INTO AZURE KEY VAULT
15 # USE IN PYTHON?
16
17
18 #### BEGIN USER INPUT #####
19 #Application ID
20 client_id = 'ENTER THE APPLICATION ID IN BETWEEN THE SINGLE QUOTES'
21 #Key
22 client_secret = 'ENTER THE SECRET KEY IN BETWEEN THE SINGLE QUOTES'
23 #Directory ID
24 tenant_id = 'ENTER THE DIRECTORY ID IN BETWEEN THE SINGLE QUOTES'
25 #Azure subscription ID
26 subscription_id = 'ENTER YOUR SUBSCRIPTION ID IN BETWEEN THE SINGLE QUOTES'
27
28 #Comma separated list of resource groups to be monitored.
29 #For example ResourceGroupList = ['rg1', 'rg2']
30 ResourceGroupList = ['Enter a comma separated list of resource groups to be monitored']
31
32 #Comma separated list of Firewall IPs or FQDNs of the management interface
33 #For example FirewallList = ['1.1.1.1', '2.2.2.2']
34 FirewallList= ['Comma separated list of firewall IPs or FQDNs']
35
36
37 #Comma separated list of API keys. Make sure the fw list and api key list match
38 #For example apikeyList = ['api key for fw with ip 1.1.1.1', 'api key for fw with ip 2.2.2.2']
39 apikeyList = ['Comma separated list of API keys for firewalls in FirewallList']
40
41 ##### END USER INPUT #####
42
```

The script will monitor all the resource groups listed in the ResourceGroupList and update all the firewalls in the FirewallList with the same ip-tag mapping.

Save the file. At this point you can choose to run the file and test for correct functionality.

run.py

Save

Run

```
1 import os
2 import sys
3 import urllib2
4 import urllib
5 import json
6 import collections
7 import itertools
8 import xml.etree.ElementTree as et
9 #sys.path.append(os.path.abspath(os.path.join(os.path.dirname( __file__ ), '..')))
10
11
12
13 ##### TO DO #####
14 # LOOK INTO AZURE KEY VAULT
15 # USE IN PYTHON?
16
17
18 ##### BEGIN USER INPUT #####
19 #Application ID
20 client_id = 'ENTER THE APPLICATION ID IN BETWEEN THE SINGLE QUOTES'
21 #Key
22 client_secret = 'ENTER THE SECRET KEY IN BETWEEN THE SINGLE QUOTES'
23 #Directory ID
24 tenant_id = 'ENTER THE DIRECTORY ID IN BETWEEN THE SINGLE QUOTES'
25 #Azure subscription ID
26 subscription_id = 'ENTER YOUR SUBSCRIPTION ID IN BETWEEN THE SINGLE QUOTES'
27
28 #Comma separated list of resource groups to be monitored.
29 #For example ResourceGroupList = ['rg1', 'rg2']
30 ResourceGroupList = ['Enter a comma separated list of resource groups to be monitored']
31
32 #Comma separated list of Firewall IPs or FQDNs of the management interface
33 #For example FirewallList = ['1.1.1.1', '2.2.2.2']
34 FirewallList = ['Comma separated list of firewall IPs or FQDNs']
35
36
37 #Comma separated list of API keys. Make sure the fw list and api key list match
38 #For example apiKeyList = ['api key for fw with ip 1.1.1.1', 'api key for fw with ip 2.2.2.2']
39 apiKeyList = ['Comma separated list of API keys for firewalls in FirewallList']
40
41 ##### END USER INPUT #####
42
```

Logs

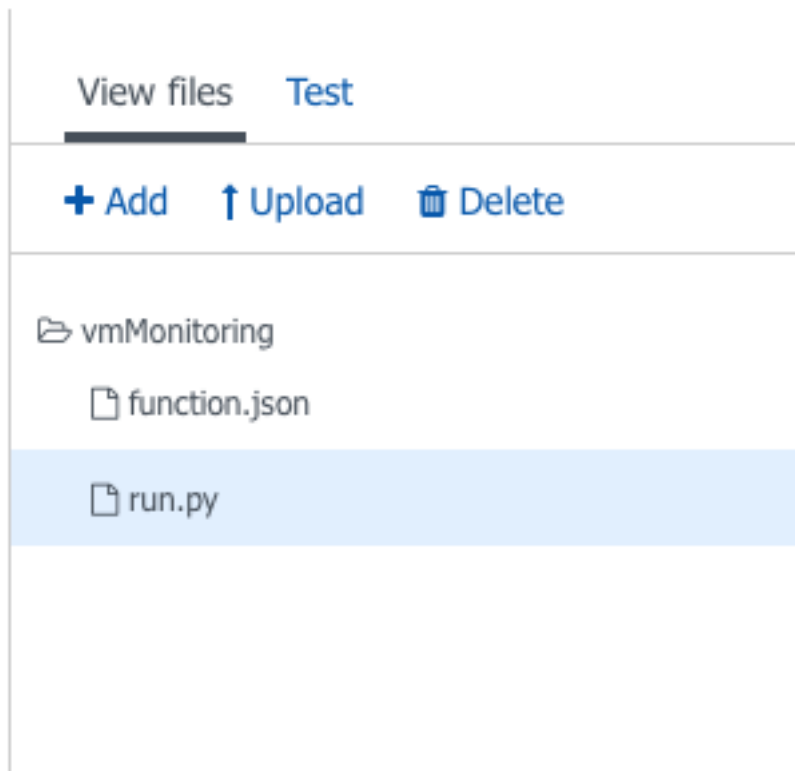
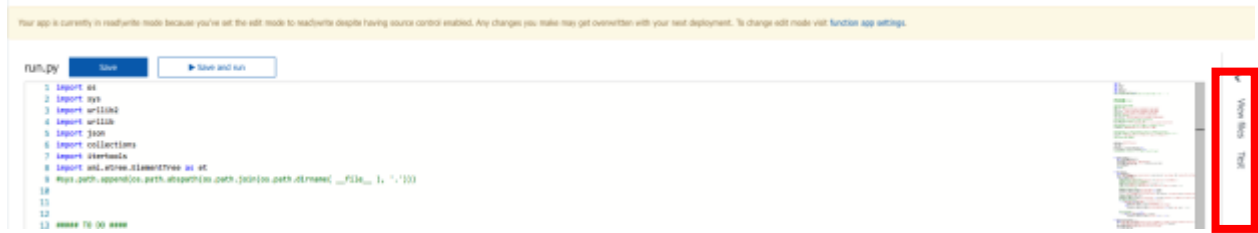
Pause Clear Copy logs Expand

```
2017-10-18T21:46:53 Welcome, you are now connected to log-streaming service.
2017-10-18T21:47:53 No new trace in the past 1 min(s).
2017-10-18T21:48:53 No new trace in the past 2 min(s).
2017-10-18T21:49:53 No new trace in the past 3 min(s).
2017-10-18T21:49:59.980 Function started (Id=eabb3778-5b83-44e8-97b7-296f79ae888f)
2017-10-18T21:50:00.012 Function started (Id=b39318b9-72e6-4095-8e4a-73081f5e0fae)
2017-10-18T21:50:03.783 Function completed (Success, Id=b39318b9-72e6-4095-8e4a-73081f5e0fae, Duration=3782ms)
2017-10-18T21:50:05.646 Function completed (Success, Id=eabb3778-5b83-44e8-97b7-296f79ae888f, Duration=5659ms)
```

Modify the timer [OPTIONAL]

By default the script runs every 5 minutes. If you want to change the timer, edit the function.json file.

In the Azure function portal click the View Files tab to bring up the file browser and click on function.json file



function.json

Save

▶ Run

```
1 {
2   "disabled": false,
3   "bindings": [
4     {
5       "name": "myTimer",
6       "type": "timerTrigger",
7       "direction": "in",
8       "schedule": "0 */5 * * * *"
9     }
10  ]
11 }
12
13
```

The value of schedule is a CRON expression. More here: <https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-timer>

For example: Change the 5 to 1, to run the function every 1 minute.

Save the changes to the function.json file

Your Azure function is now setup to monitor you resource groups and push ip-to-tag mappings to the firewalls of your choosing.

Note: The function will only push updates to the firewalls if it finds changes in the tags. It will register new ip addresses to tags and unregister ip addresses that are no longer present from the firewalls.

Now you can log into the firewall and check the ip to tag mappings.

Currently only a few tags are being monitored:

OS Type, VM Name, Private IP addresses, VM power state (stopped or running) and user defined tags.

