Cognizant CLOUD SERVICES

Blueprint – Azure Functions

Revision History

Date	Version	Author	Reviewer(s)	Comments
26-Feb-2018	1.0	Harini	Tressa	Initial Draft

Contents

1. Scope

This document provides the blueprint for the Functions service offered by Microsoft. This contains the below.

1. Overview
2. Service Usage
3. Provisioning Scripts
4. Support Objectives
5. Monitoring metrics
6. Monitoring Setup Scripts

2. Overview

Azure Functions is a server-less compute service to run code on-demand without having to explicitly provision or manage infrastructure. Azure Functions can be used to run a script or piece of code in response to a variety of events. Azure Functions is a solution for easily running small pieces of code, or "functions," in the cloud. Functions can help to write just the code needed for the problem at hand, without worrying about a whole application or the infrastructure to run it. Functions can make development even more productive, by using preferred development language of choice, such as C#, F#, Node.js, Java, or PHP. See the following link for more details: https://docs.microsoft.com/en-us/azure/azure-functions/

3. Service Usage

3.1 Service Connectivity

For intranet apps, restrict access using target source IP addresses in the firewall rules

3.2 Best Practices

- Function running on App Service plan to have "AlwaysOn" enabled for long running functions.
- Runtime 1.x is in GA, whereas 2.x is in Preview
- C#, JavaScript are recommended to be used in Runtime 1.x
- Python, Bash & PowerShell are experimental in Runtime 1.x and are not available in Runtime 2.x
- All integrations and connections are to be encrypted
- SSL Bindings and certificates to be used to streamline connectivity to specific networks or services
- IAM to be used to control access to users
- For consumption plan, it is recommended to have the storage account in the same region as the function app
- Integration with App Insights is recommended for monitoring and logging purposes.

3.3 Microsoft SLA

- 99.95% for App Service Plan.
- No SLA for consumption plan

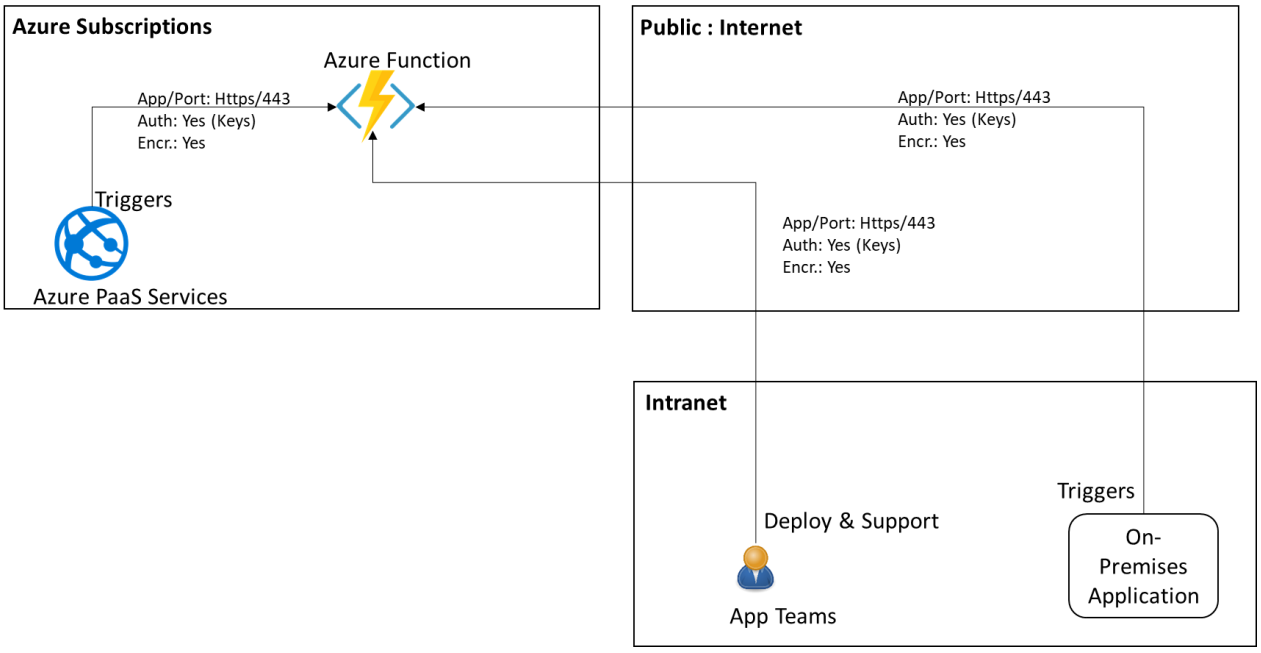
3.4 Additional Notes

A single function app will scale to a maximum of 200 instances. New instances will take at least 10 seconds to be allocated.

3.5 Recommended tiers for enterprise usage

- App Service Plan - Long running functions, Linux hosted
- Consumption plan - Maximum run time of 5 minutes. Can be increased up to 10 minutes

3.6 Service Usage Diagram



4. Provisioning Script

The below ARM template is to be used to provision an instance of the service.

This consists of the below parameters

Parameter Name	Description
FunctionappName	The name of the Application which will be using the Resource.

StorageName	The storage name which is used for logging.
Location	Provide location for WebApp Allowed Values: US South Central- Prod Workloads US South Central –Dev Workloads
AppInsightsResourceName	Name of the app insights instance for the function if in same resourcegroup
AppInsightsResourceId	Full resource ID of the app insights instance for the function
TagValues	Service Tags for the resource to categories
Pricing Tier	The pricing tier to be used for the FunctionApp.

Template Script

```
{
  "$schema": "http://schema.management.azure.com/schemas/2014-04-01-
preview/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",

  "parameters": {
    "functionAppName": {
      "type": "string"
    },
    "storageName": {
      "type": "string"
    },
    "storageResourceGroup": {
      "type": "string",
      "defaultValue": "",
      "metadata": {
        "description": "Leave empty if storage account is in the same
resource group as the function"
      }
    },
    "location": {
      "type": "String",
      "defaultValue": "resourceGroup",
      "allowedValues": [
        "resourceGroup",
        "North Central US",
        "South Central US"
      ],
      "metadata": {
        "description": "Enter the Azure locations in which the resources
exists, or to be created .."
```

```
      }
    },
    "tagValues": {
      "type": "object",
      "defaultValue": {
        "Tag1Name": "Tag1Value",
        "Tag2Name": "Tag2Value"
      }
    },
    "appInsightsResourceName": {
      "type": "string",
      "defaultValue": "",
      "metadata": {
        "description": "Name of the app insights instance for the function if
in same resourcegroup.  Specify this OR appInsightsResourceId."
      }
    },
    "appInsightsResourceId": {
      "type": "string",
      "defaultValue": "",
      "metadata": {
        "description": "Full resource ID of the app insights instance for the
function.  Specify this OR appInsightsName."
      }
    }
  },
  "variables": {
    "functionLocation": "[if(equals(parameters('location'), 'resourceGroup'),
resourceGroup().location, parameters('location'))]",
    "storageResourceGroupName":
"[if(equals(parameters('storageResourceGroup'), ''), resourceGroup().name,
parameters('storageResourceGroup'))]",
    "storageResourceId": "[resourceId(variables('storageResourceGroupName'),
'Microsoft.Storage/storageAccounts', parameters('storageName'))]",
    "appInsightsResourceId": "[if(equals(parameters('appInsightsResourceId'),
''),resourceId('microsoft.insights/components',
parameters('appInsightsResourceName')),
parameters('appInsightsResourceId'))]"
  },
  "resources": [
    {
      "apiVersion": "2016-03-01",
      "name": "[parameters('FunctionAppName')]",
      "type": "Microsoft.Web/sites",
      "tags": "[parameters('TagValues')]",
      "properties": {
        "name": "[parameters('FunctionAppName')]",
        "siteConfig": {
          "appSettings": [
            {
              "name": "AzureWebJobsDashboard",
              "value":
"[concat('DefaultEndpointsProtocol=https;AccountName=',parameters('storageNam
e'),';AccountKey=',listKeys(variables('storageResourceId'), '2015-05-01-
preview').key1)]"
            },
```

```
            {
              "name": "AzureWebJobsStorage",
              "value":
"[concat('DefaultEndpointsProtocol=https;AccountName=',parameters('storageNam
e'),';AccountKey=',listKeys(variables('storageResourceId'), '2015-05-01-
preview').key1)]"
            },
            {
              "name": "FUNCTIONS_EXTENSION_VERSION",
              "value": "~1"
            },
            {
              "name": "WEBSITE_CONTENTAZUREFILECONNECTIONSTRING",
              "value":
"[concat('DefaultEndpointsProtocol=https;AccountName=',parameters('storageNam
e'),';AccountKey=',listKeys(variables('storageResourceId'), '2015-05-01-
preview').key1)]"
            },
            {
              "name": "WEBSITE_NODE_DEFAULT_VERSION",
              "value": "6.5.0"
            },
            {
              "name": "APPINSIGHTS_INSTRUMENTATIONKEY",
              "value": "[reference(variables('appInsightsResourceId'),'2015-
05-01').InstrumentationKey]"
            }
          ]
        },
        "clientAffinityEnabled": false
      },
      "location": "[variables('functionLocation')]",
      "kind": "functionapp"
    }
  ]
}
```

5. Support Objectives

Below are the objectives to be fulfilled while providing support for instances of Azure Function.

1. Provisioning of Azure Functions
2. De-Provision of Azure Functions
3. De-Provision Azure App Service Plan
4. Request for Onboarding Azure Functions
    a. Setting up tags as defined by the enterprise
    b. Configuring monitoring metrics
5. Request for Service Configurations.
    a. Logging configuration with Storage accounts, App Insight Integration and Portal Dashboard.
    b. Add/Delete Function App's application settings

6. Monitoring Metrics

This section details the metrics which are to be monitored for instances of Azure Function App.

6.1 Recommended Metrics

The following metrics are recommended to be enabled by default.

Metrics	Category	Threshold	Unit	Frequency in Mins/Hrs	Reason & Required remedial action
Ping Test	Availability	=	Location 4xx/5xx	N/A	Prod: 5M Dev: 15M
Exception Rate	Performance	>	0.2 exceptions per second (12 per minute)	count	5M
Exception Count	Information	>	5 exceptions	count	5M

7. Monitoring Metrics Setup Script

Monitoring is setup using Azure App Insights.