Microsoft Azure is a cloud computing service created by Microsoft for building, testing, deploying, and managing applications and services through a global network of Microsoft-managed data centers.

Things which we need to create VM:

1. Resource Group
2. Storage
3. VNetwork

1. VM is a Resource

2. Resource is a part of Resource Group

3. VM Has CPU, Memory, hard drive (file for my host so I need storage account)

4. To connect to outside world we need NIC which will be connected to a Switch (Physical)

For that we need virtual network.

Create a resource group and Vnetwork and put it in Resource group.

SUBNET AND ADDRESS SPACE:

Address space: 10.0.0.0/24

Subnet: 10.0.0.0/16

Next subnet will be: 10.0.1.0/24

Class A = 8

Class B = 16

Class C = 24

* All the VMs which are part of these subnet will connect to each other

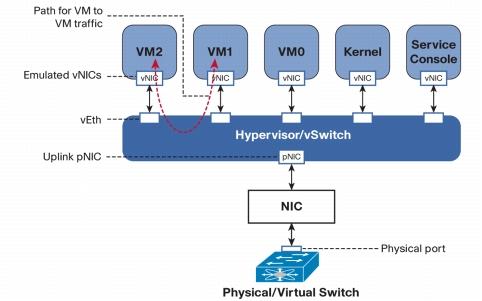
Now,

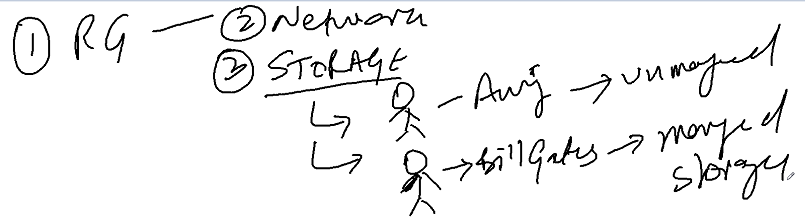
1. Resource
2. Network
3. Storage

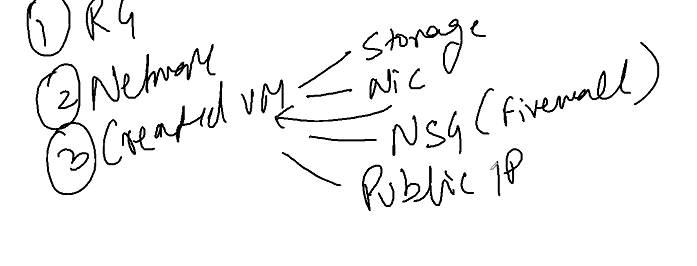
Either Microsoft will create storage or user will create

A network interface card (NIC) is a circuit board or card that is installed in a computer so that it can be connected to a network.

A network interface card provides the computer with a dedicated, full-time connection to a network. Personal computers and workstations on a local area network (LAN) typically contain a network interface card specifically designed for the LAN transmission technology.

[](javascript:;)





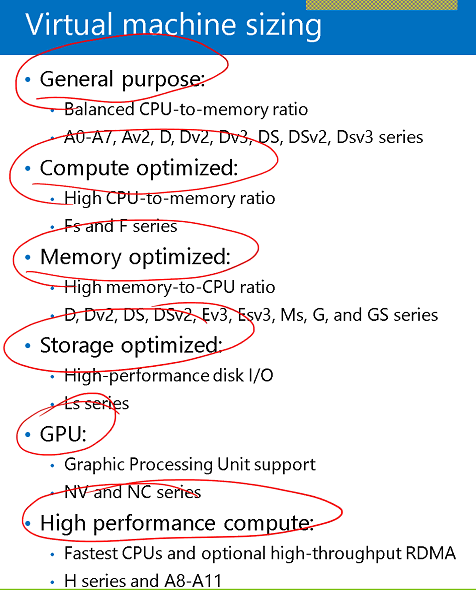
When VM is getting created... Storage, NIC , NSG and IP all get created same time.

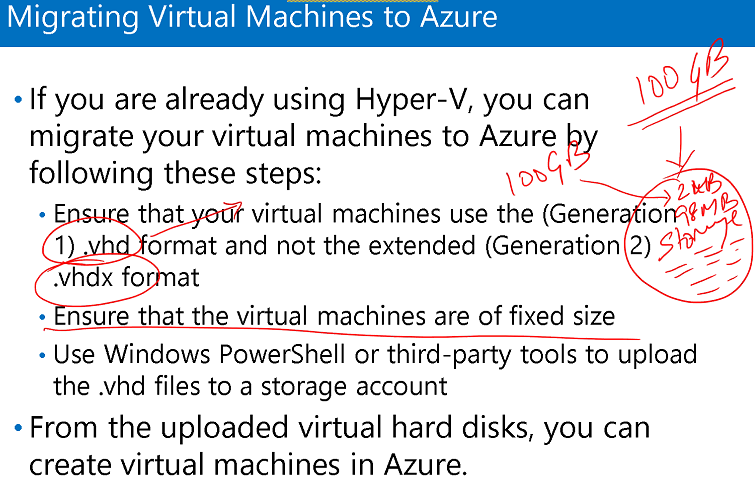
**Network Interface Card**, the **NIC** is also referred to as an **Ethernet card** and **network adapter**. It is an [expansion card](https://www.computerhope.com/jargon/e/expacard.htm) that enables a [computer](https://www.computerhope.com/jargon/c/computer.htm) to connect to a network; such as a home network etc.

## How does a computer with a network card connect to a network?

A network card can communicate with each other over the same [network](https://www.computerhope.com/jargon/n/network.htm) using a network [switch](https://www.computerhope.com/jargon/s/switch.htm) or if only two computers a direct connection. If computers on your network need to connect to a different network (e.g., the [Internet](https://www.computerhope.com/jargon/i/internet.htm)) they must be eventually connected to a [router](https://www.computerhope.com/jargon/r/router.htm) that allows networks to communicate with each other.

A network security group (NSG) contains a list of security rules that allow or deny network traffic to resources connected to Azure Virtual Networks (VNet). NSGs can be associated to subnets, individual VMs (classic), or individual network interfaces (NIC) attached to VMs (Resource Manager). When an NSG is associated to a subnet, the rules apply to all resources connected to the subnet. Traffic can further be restricted by also associating an NSG to a VM or NIC.

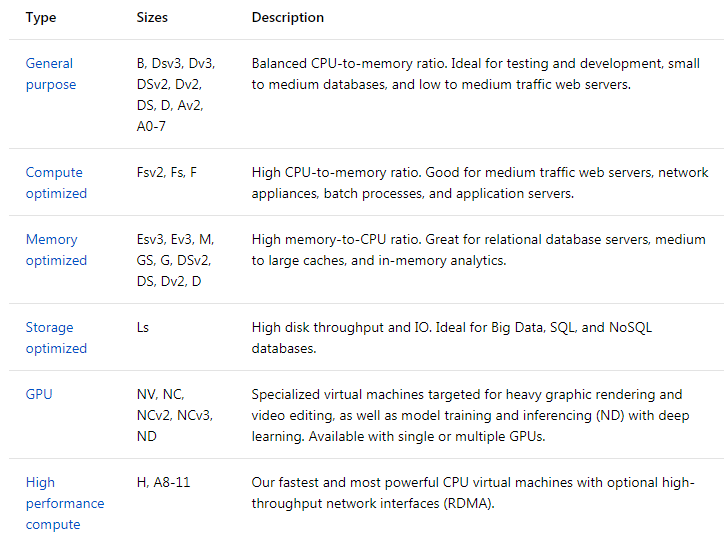




Labs:

1. Create VHD in local system
2. Detached this VHD
3. Create storage – blobs-container
4. Upload the .vhd

If this VHD has operating system that will work on VM.



What is the Marketplace?

The Marketplace is the premier destination for all your software needs - certified and optimized to run on Azure.

Find solutions you want, from open source container platforms to threat detection to block chain

Try software for free - take a Test Drive or deploy a free-trial

Provision end-to-end solutions quickly and reliably

What is load balancing?

A load balancer is a device that distributes network or application traffic across a cluster of servers. Load balancing improves responsiveness and increases availability of applications.

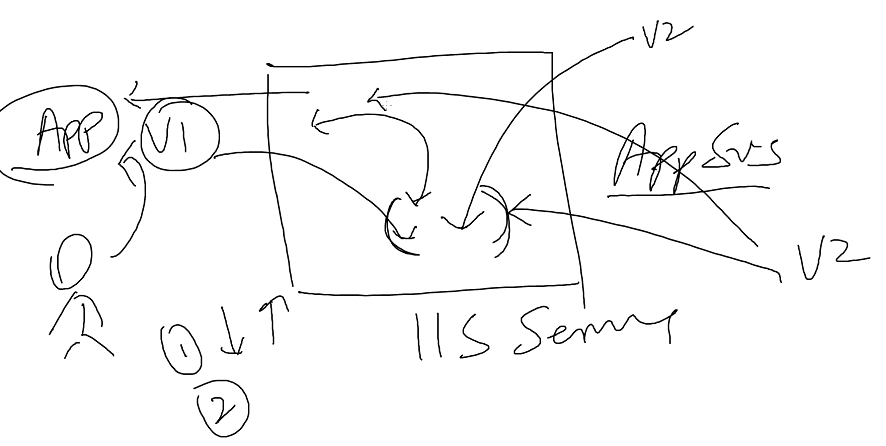
**Staging Publish**

Requirement: Update the Application:-

1. Bring the site down, update and bring it back up

2. Upload the content of v2 in different workspace which is a part of web service itself

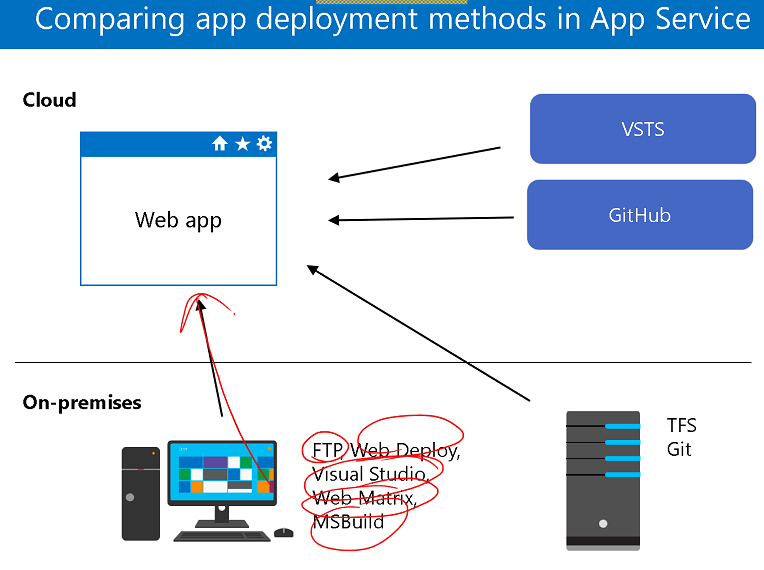
3. Then swap v1 and v2.



1. Create App service (Quickly and easily build web and mobile apps for any platform or device with Azure App Service.)
2. Using market place (can get different solutions), deploy web App
3. This works with DNS name(e.g. simplilearnweb)
4. Windows
5. Service plan
6. App Service Plan/ Location
7. Create App Service Plan(e.g. appsvsplan)
8. Select pricing

Website is ready, don’t have to build VM, configure IIS. My website is ready... just browse.

1. Put some template (download HTML5 Template)



**Using FTP:**

1. Copy the FTP hostname of App service and username and pate in FileZilla client.
2. We can use FTP, Web Deploy etc. to upload template to Web App

* Select Deployment Credentials (used with GIT and FTP) (download FillZilla, it uses port number: 21)
* Upload the files from local directories to remote Site( /site/wwwroot)

1. Create deployment slot i.e. staging area

* Upload the files from local directories to remote Site( /site/wwwroot)

1. Update production site, Now we can Swap from source to destination

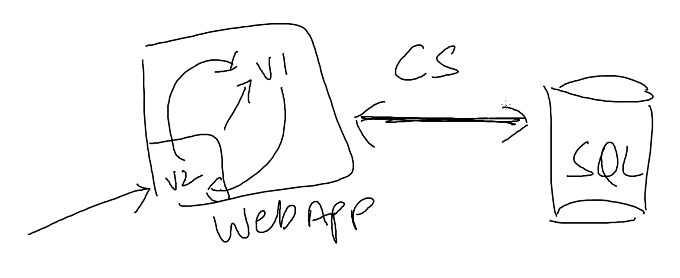
**Using Visual Studio:**

1. Right click project – Publish Web App – Select Microsoft azure - select subscription –View(resource Group)
2. Create
3. If we want to create new then click new
4. Create

It will upload in production

**Web App Configuration**

1. The web app deployment package and its configuration are intercepted and changed in the application during startup
2. Applications can be scaled by:
   * Creating IIS web sites using the Web Deploy package
   * Applying configuration options from the external store
3. Web Apps share functionality with IIS web sites
   * For Free and Shared instances, Web Sites are implemented similar to IIS web sites
   * For Standard instances, a reserved virtual machine is made available and each individual website is similar to an IIS web Site
   * Azure Websites can also be managed remotely using the IIS Manager



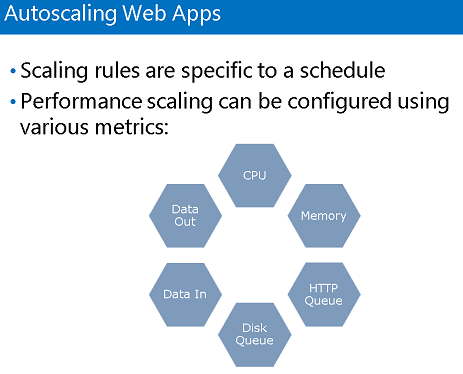
Requirement: we have to move V2 in production from V1, Web App is communicating with SQL in backend. So, when we swap V2 and V1 we need to change the Connection string and pointing to correct database. Initially V2 is pointing to test database and V1 pointing to deployment database.

**Custom domains**

Click Web App – Continuous delivery (preview) -

Custom domains (instead of using HOSTNAMES ASSIGNED TO SITE we can use abc.com as domain)

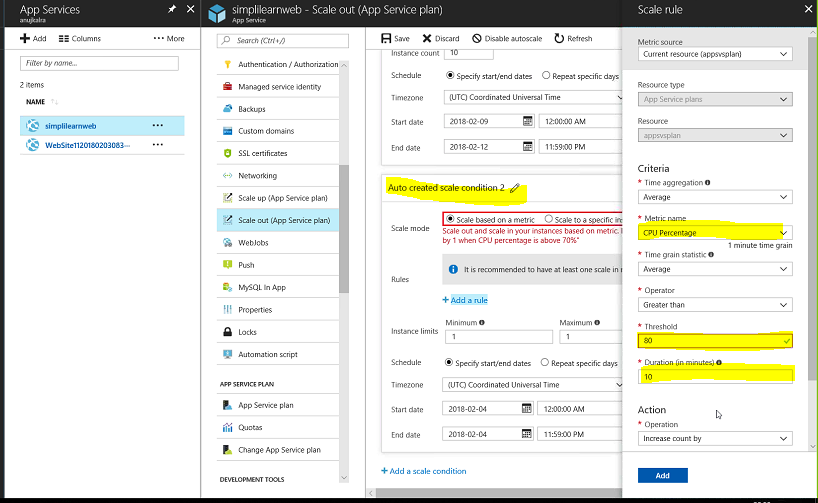
**Auto Scaling**



Auto scaling was done using two thing

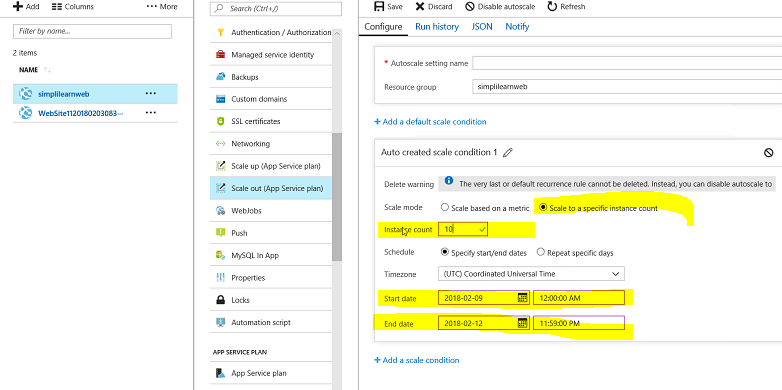
* Performance, if CPU reaches 80 %it will go ahead and scale an instance

e.g.



* For a specific time period.

e.g.



For a specific time period, with instance count = 10, it will start adding instances.

**Traffic Manager**

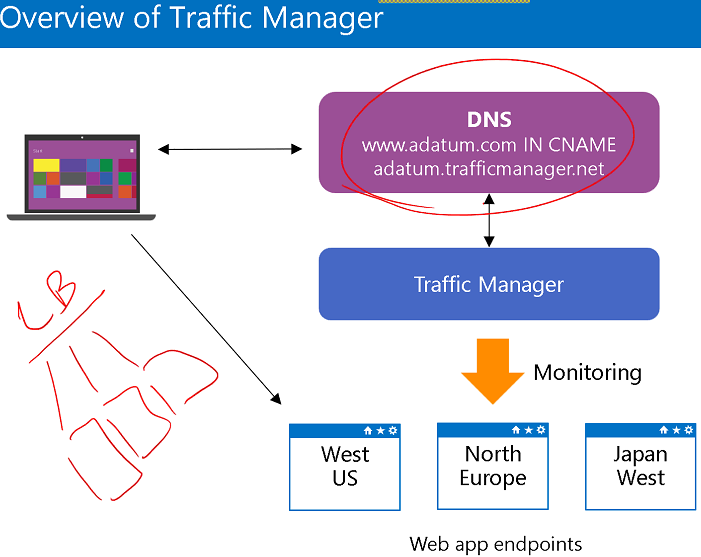
What is Traffic Manager?

How to configure Traffic Manager and

Best settings for Traffic Manager

Monitoring

It’s a kind of load balancer (Load Balancer we can use with app service instances), it works with DNS



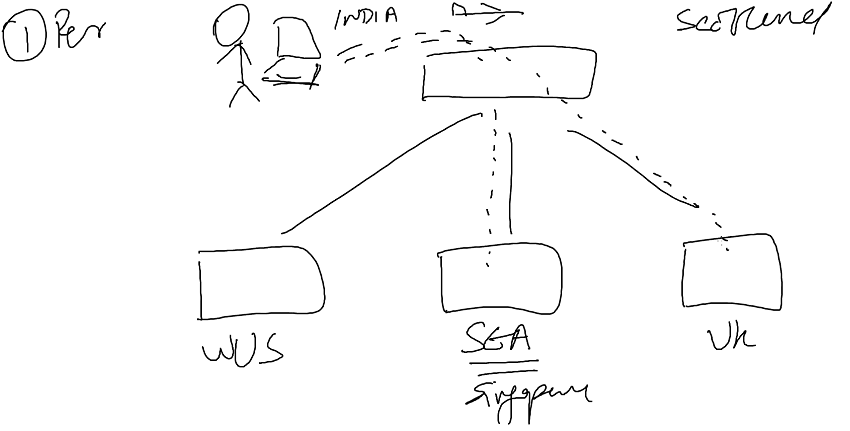
Traffic Manger depends on few things:

1. Performance

Suppose a user (India) tries to connect to different networks in different regions suppose (UK, South East Asia, United States etc.)

Then the traffic manager will take the route of SEA (Singapore) as it is nearest to India.

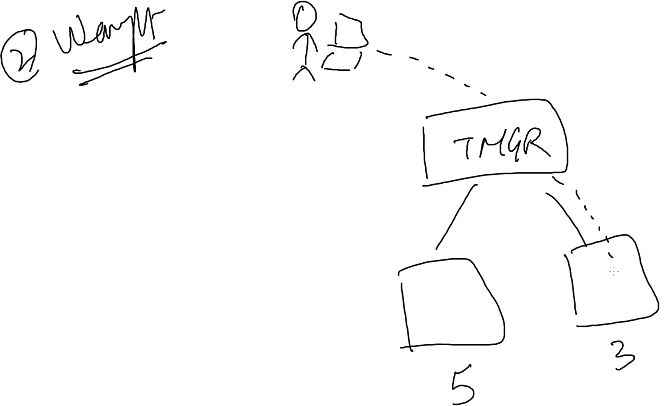
This depends on the performance of link.



1. Weighted: it depends on how many links are there in a particular location.

Suppose a user trying to access application using TM which has two links,

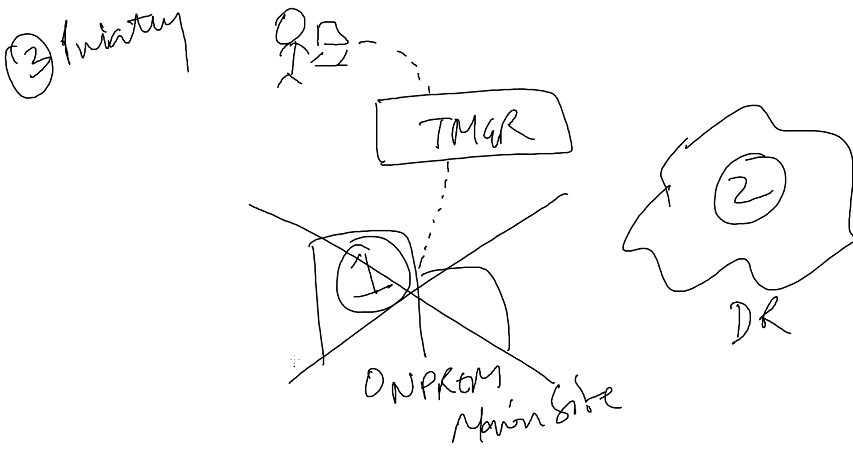
Both these links has 5 connection and 3 connections respectively. So TM route the connection where there is less weight so that both have the same weight.



1. Priority:

Suppose I am using my TM and trying to connect to an application.

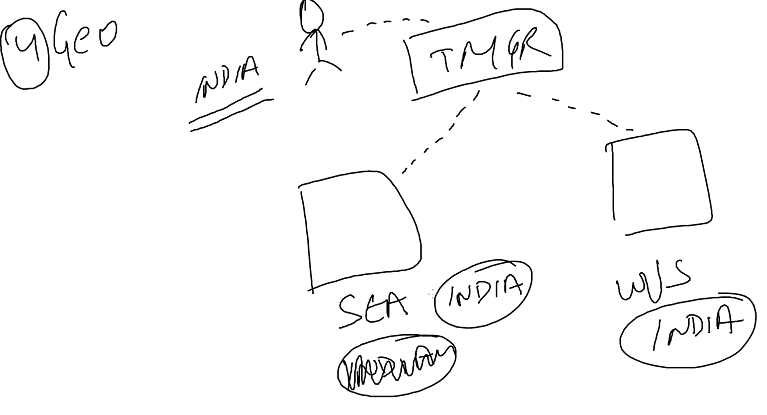
In this case I have two locations ONPREM (priority =1) i.e. Main site and other is Cloud - DR (Disaster resource) (priority = 2). So whenever I try to connect from anywhere I will be redirect to site one. But if site 1 is down for some reason then TM redirects to cloud.



1. Geo:

Suppose a user from India trying to connect to some other country using his TM

Let one country is South East Asia and other West US, then TM will redirect it to SEA only if India Site is tagged with SEA otherwise it will redirect to WUS. it will give 404 error for SEA.



DEMO Traffic Manager (Checking for Geo)

1. Create two Web Sites

* WebAppUStm – upload some template using filezilla
* WebAppSEAtm

1. Create Traffic Manager profile – Add two end points

* Click on Endpoints – Add

Type: Azure Endpoint

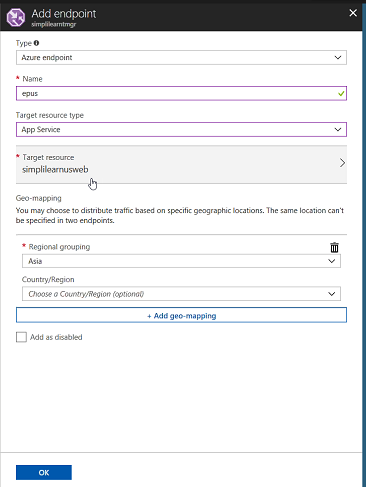
Name: endpointUS

Target resource type: AppService

Target resource: WebAppUStm

Region Grouping: US

* Create



So, suppose I am in India then it will redirect me to server in US but technically it should redirect SEA.

In Short, if I am in Asia and I want to access WebApp which is in US server then Asia region should be added in US Endpoint and vice versa.

Select configuration and changed it to **Performance**

* Remove previous end points
* Add new End points
* Access the DNS of traficManagerprofile, it will redirect us to the nearest server.

**Monitoring WebApp**

Troubleshoot WepApp /restart IIS

There is a tool KUDU and we can use kudu to restart or kill the app. In order to run kudu, we can

Browser App and scm to the url and it opens kudu.

