National College of Ireland

Configuration Manual

MSc Research Project
MSc Cloud Computing

Tanya Chopra
Student ID: x18177271

School of Computing
National College of Ireland

Supervisor: Vikas Sahni

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Tanya Chopra
Student ID:	x18177271
Programme:	MSc Cloud Computing
Year:	2020
Module:	MSc Research Project
Supervisor:	Vikas Sahni
Submission Due Date:	17/08/2020
Project Title:	Configuration Manual
Word Count:	1211
Page Count:	14

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

I agree to an electronic copy of my thesis being made publicly available on TRAP the National College of Ireland's Institutional Repository for consultation.

Signature:	
Date:	14th August 2020

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Tanya Chopra
x18177271

1 Methodology

This project uses an Agile Methodology. In this method, only the particular sprint that requires change is taken into account, and the desired change is done in that particular sprint alone, rather than changing every step of the project. This makes the whole process effective from the very beginning. The agile methodology used in this project is explained as below:

• The first sprint creates a network interface. It checks if the network interface is working.

• Then, it checks if the application is present in the cloned VM. For a new VM to be created, a request is sent to Microsoft. It takes up to 6 minutes for Microsoft to accept the request and create a VM. During this process, the request waits in a loop while checking if the VM is available.

• When this process is completed, an IP address is allotted to the VM.

• Once this IP address is created, the same IP address becomes available in the endpoint in the traffic manager.

• The traffic manager acts as a load balancer here.

• Once this process is completed, the old VM and the Disk of the VM, network interface are deleted.

• This ensures that the cloning process of the VM is completed.

• Then, a new set of alerts are created for the new VM that is cloned.

• The usual process is that when there is a change in the load given to the CPU, an alert is automatically triggered in the Azure platform so that the machine is upgraded or downgraded. But as this project is using agile methodology, even before the actual process starts, a manual trigger is created once, to check if the process is working.

Now, the step by step explanation of this research project is given below.

2 Authentication

First step is Authentication for this Research project. In Azure active directory, app registration is done. The name of the app used here is MakeshiftCrossScaleApp.

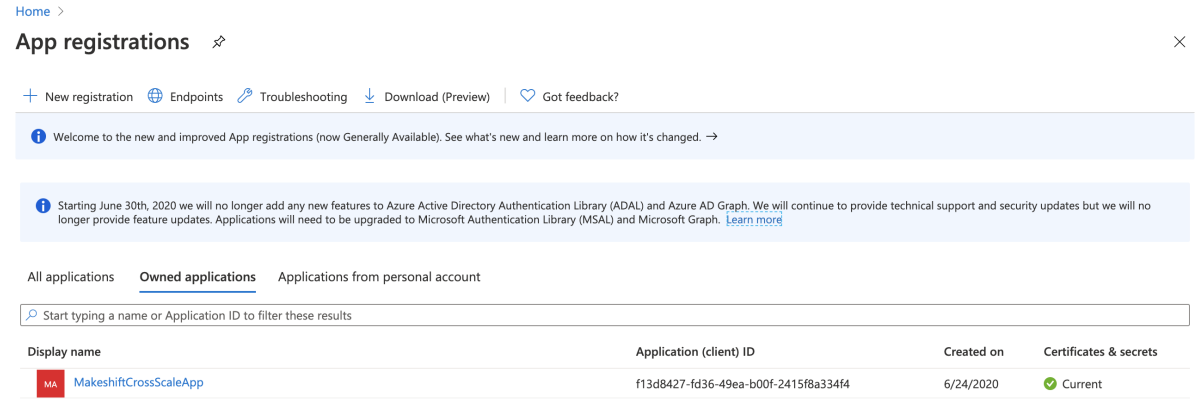

Figure 1: App Registration

This app consists of Application ID and Directory ID.

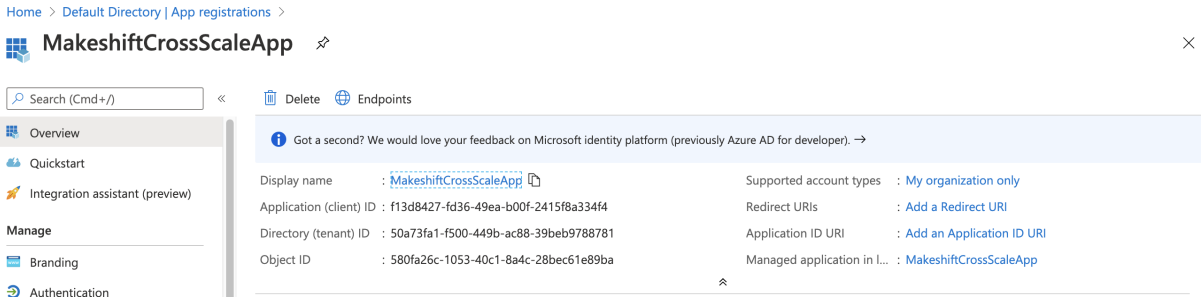

Figure 2: MakeshiftCrossScaleApp

In this application, an application secret key is created. This key is used for authentication process in the code level.

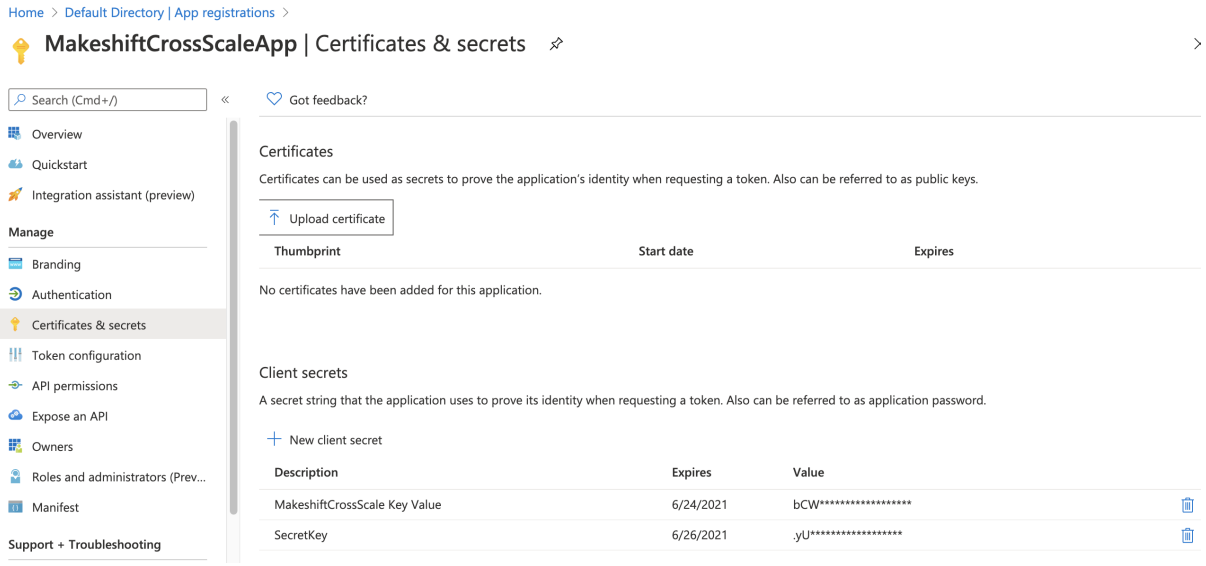

Figure 3: MakeshiftCrossScaleApp Secret Key

3 Configuration Setup

Traffic manager is also known as Load balancer, which acts as a gateway to manage the varying workloads.

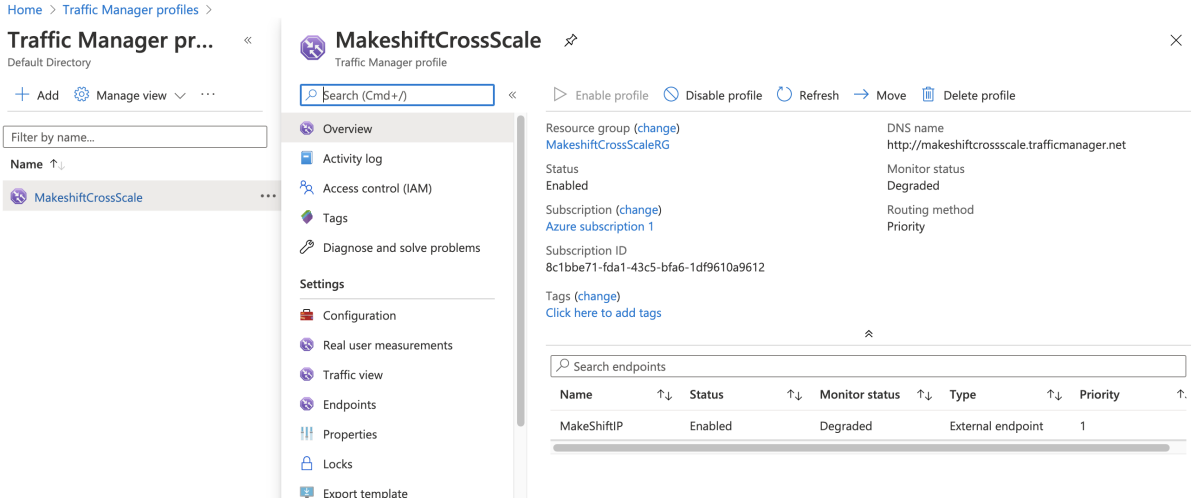

Figure 4: Traffic Manager Profile

Traffic manager is pointing to the static IP address which is 52.152.142.112. Whenever there is an IP change, traffic manager updates the IP according to the priority.

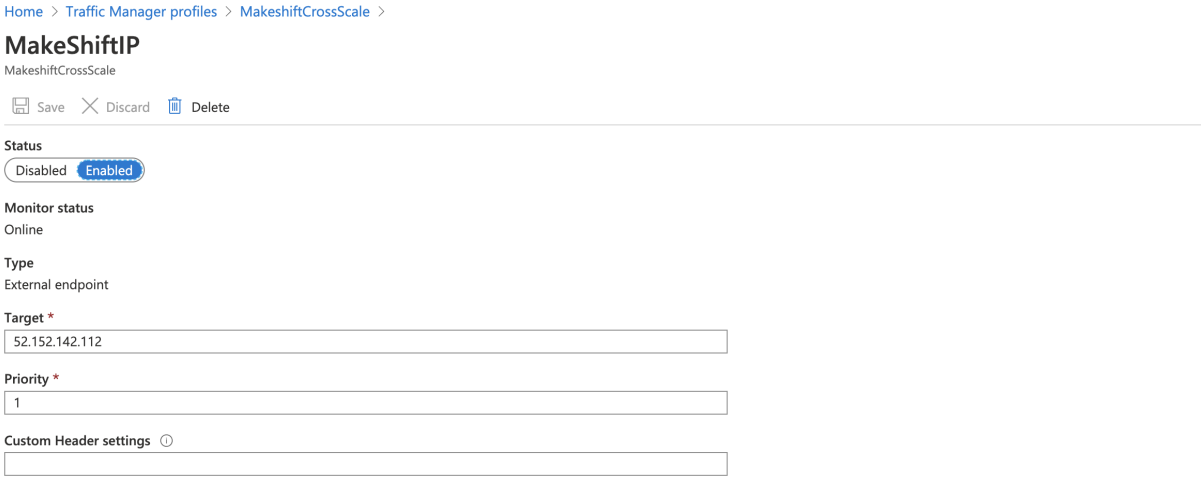

Figure 5: Traffic manager endpoint

After the creation of the traffic manager, this is the only resource in the resource group. [1] (Rohinkoul (n.d.))

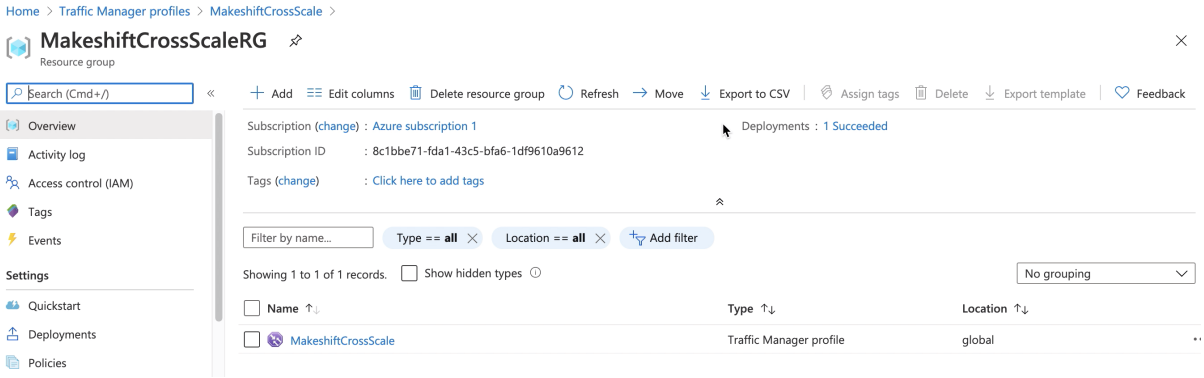

Figure 6: MakeshiftCrossScaleRG

[1] https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview

These are static IP addresses which are being used by the VM, whenever there is a cloning process.

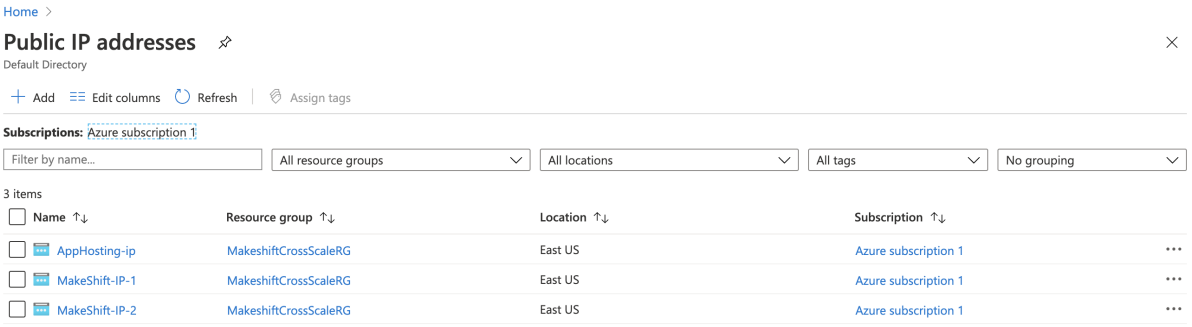

Figure 7: Public IP addresses

52.188.168.121 is the first static IP address.

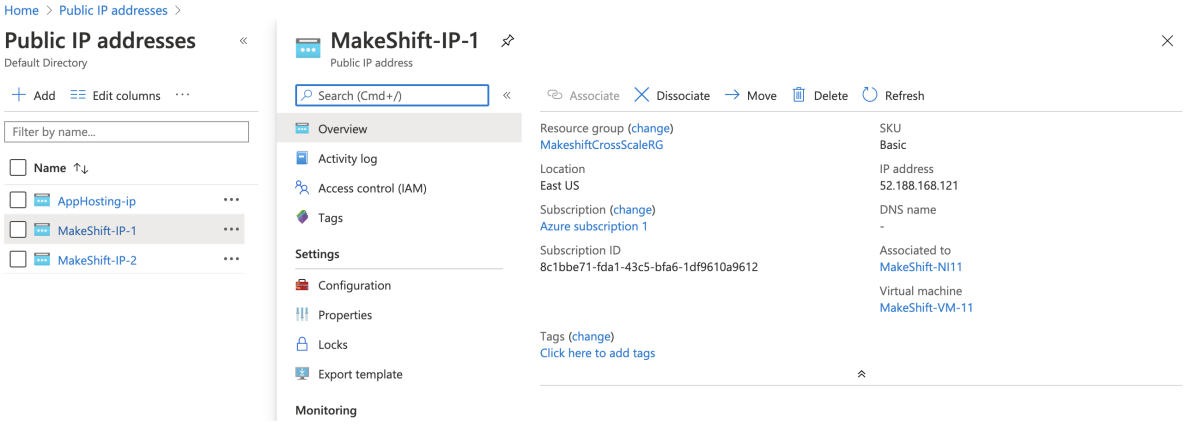

Figure 8: MakeShift-IP-1

52.152.142.112 is the second static IP address.

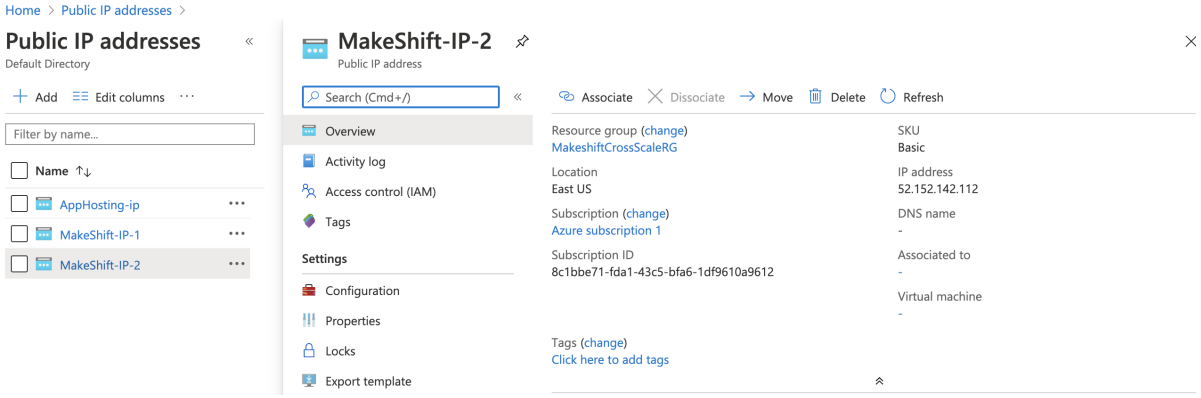

Figure 9: MakeShift-IP-2

4 Creation of VM and its image

This is the initial VM where the test application is hosted for checking downtime while up-scaling or down-scaling the VM. Azure subscription 1 and MakeshiftCrossScaleRG is selected as the Subscription and Resource group in the below image respectively. These are the mandatory fields that are marked in red asterick.

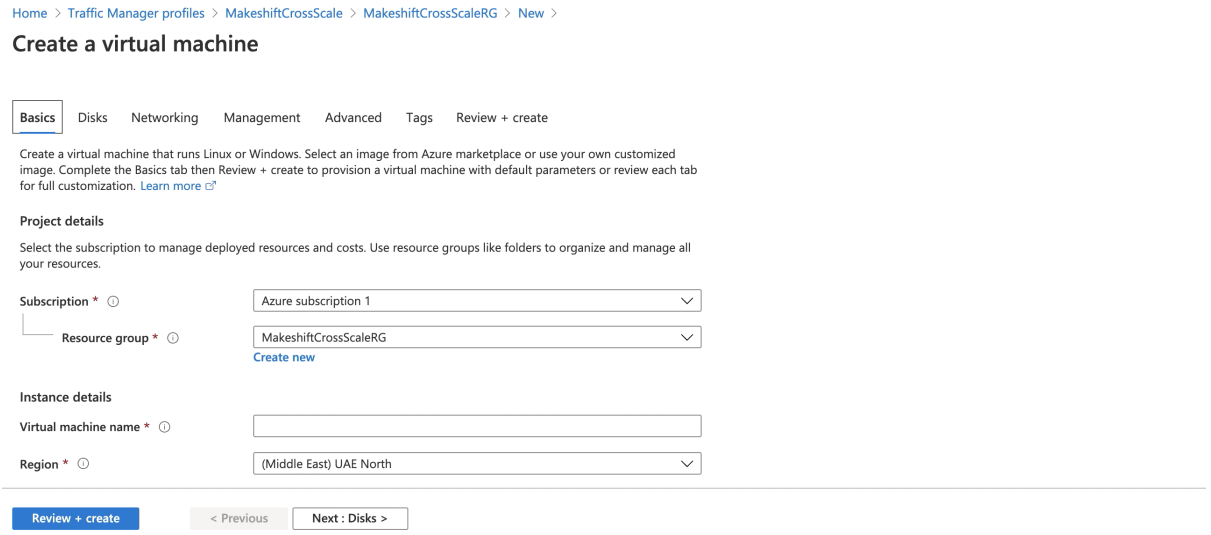

Figure 10: VM Creation

MakeShiftVM is selected as the name of the VM. Any region can be selected for the Region column. Windows Server 2019 Datacenter is the image of the VM. Any size can be chosen for VM. Then, Username and Password is created for the user to access the VM.

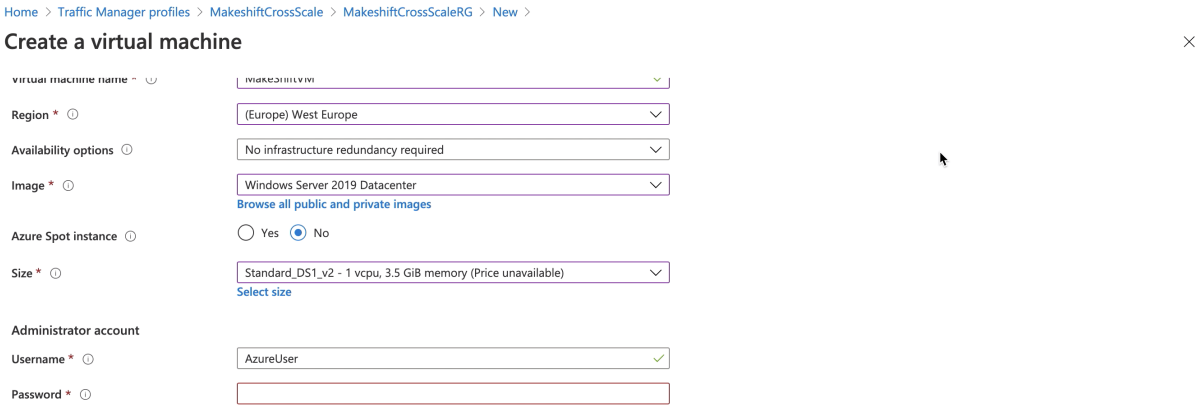

Figure 11: VM Creation

Below image consists of all the properties of the existing VM and acts as a backup for the cloning process in the future.

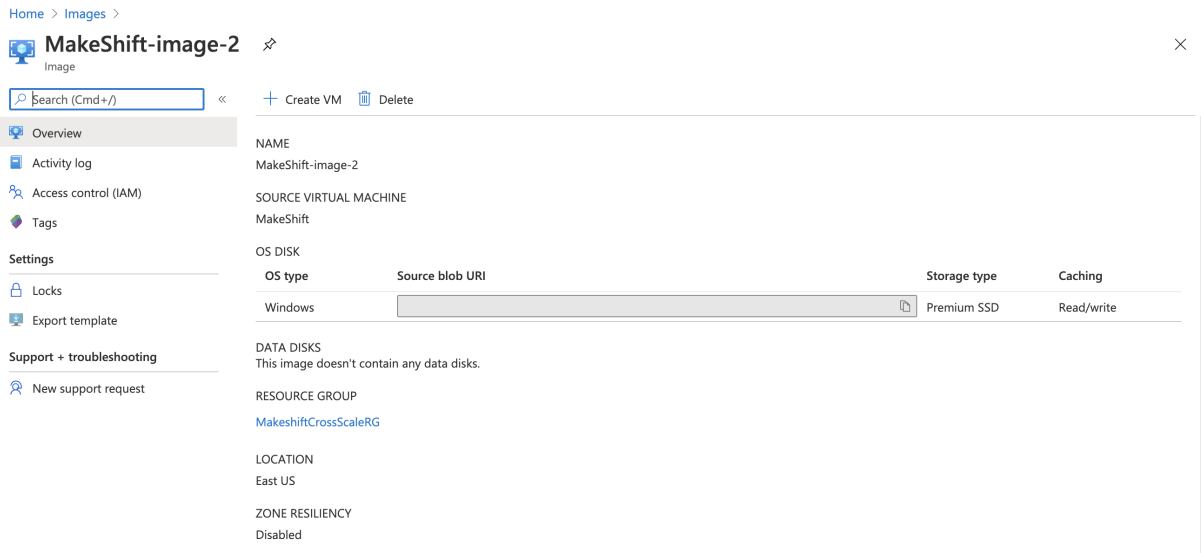

Figure 12: Image of the VM

5 Scaling up process: Cloning and Deletion of VMs

This is the sample web application hosted inside the VM and the traffic manager points to this application. Whenever there is a cloning process, this application is used to check the downtime. To simulate the workload, there is a functionality in the application which gets activated by clicking the button (Increase CPU % to 80%) in the front-end.

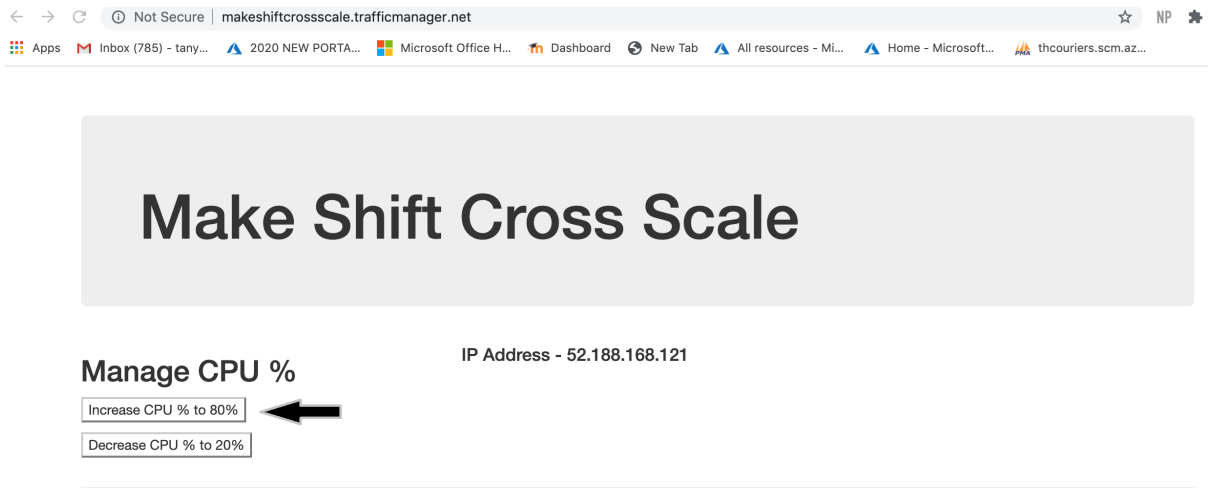

Figure 13: Sample web application

The 'Makeshift Cross Scale Algorithm' is hosted in the App Service which is the PaaS solution. It consists of two end-points such as Scale up and Scale down. Whenever the CPU percentage goes more than 80% or less than 20%, respective alerts are triggered and the algorithm is called.

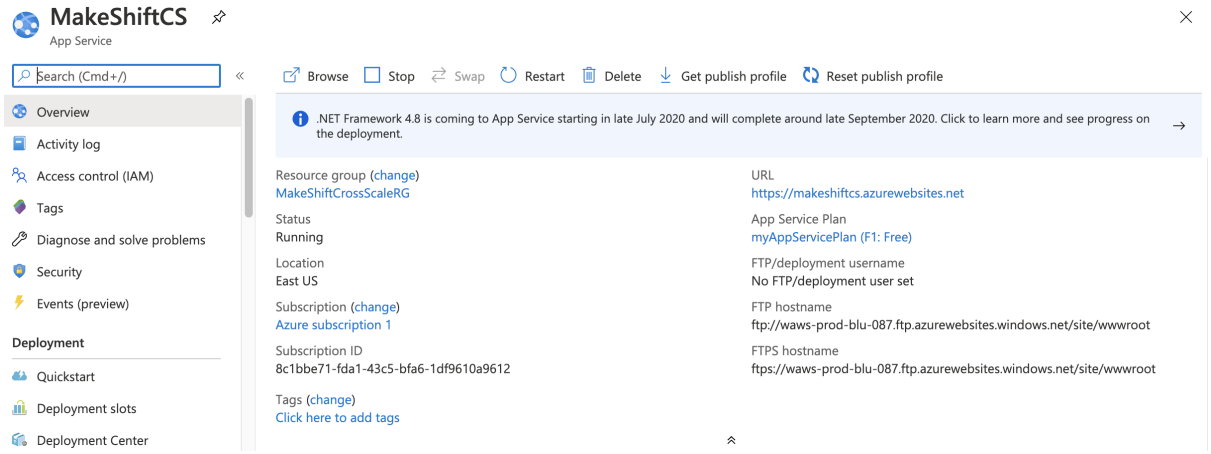

Figure 14: App Service: MakeShiftCS

When the workload in the CPU reaches more than 80%, and remains for 1 minute. There is an alert for scaling up and it calls the API endpoint.

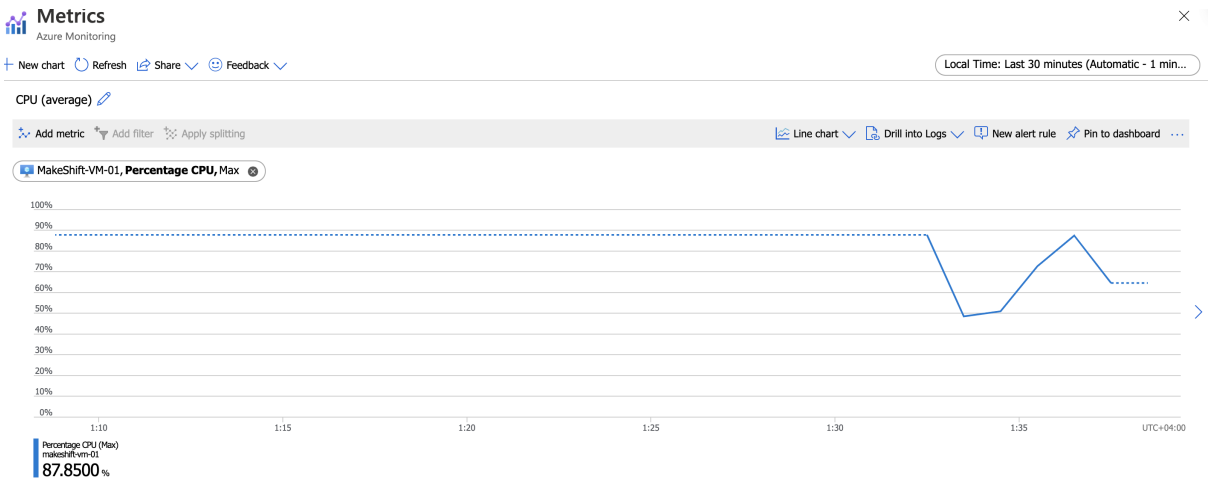

Figure 15: CPU Percentage

The alert 'MakeShift-ScaleUp01' gets triggered when the CPU % goes greater than 80%.

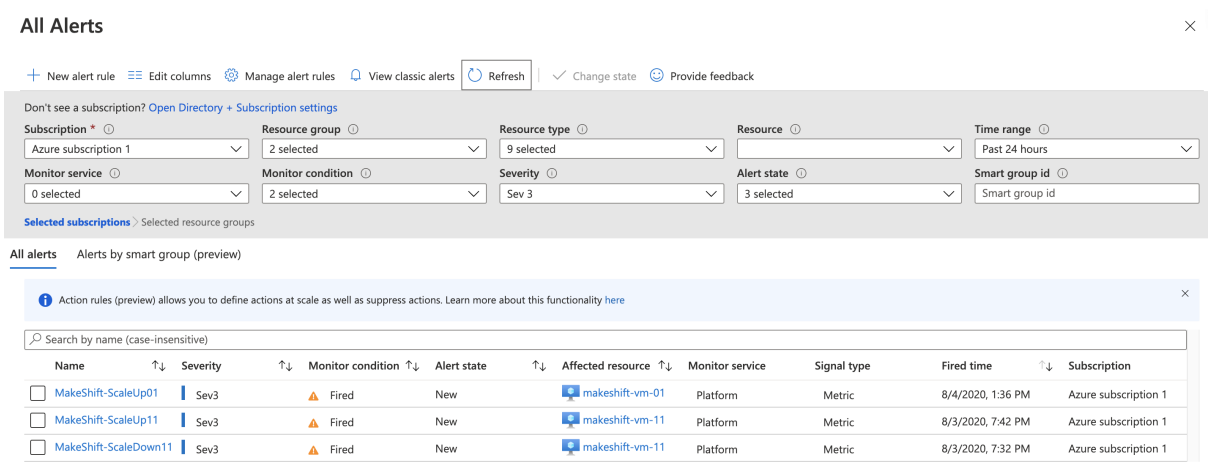

Figure 16: Alerts Triggered

This is the resource group named 'MakeshiftCrossScaleRG'. As the cloning process starts, the name of the new VM and its resources namely network interface and disk are added in the resource group.

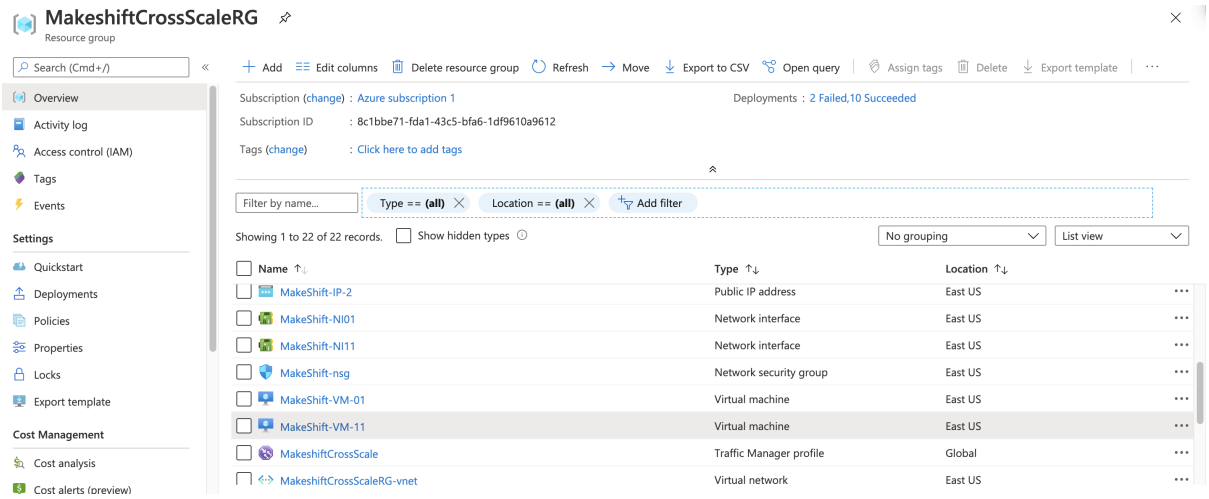

Figure 17: MakeshiftCrossScaleRG during cloning

The below image contains the information of the old VM, while the cloning process is taking place.

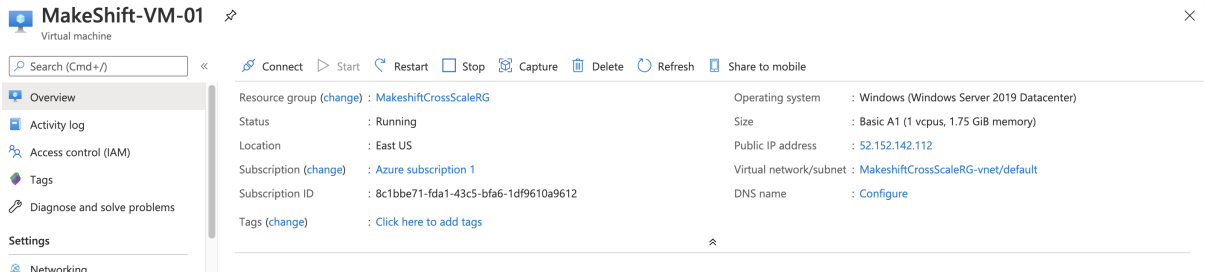

Figure 18: Old VM during cloning

The below image shows the creation of the new VM, while the cloning process is going on. [2] (Rloutlaw1 (n.d.))

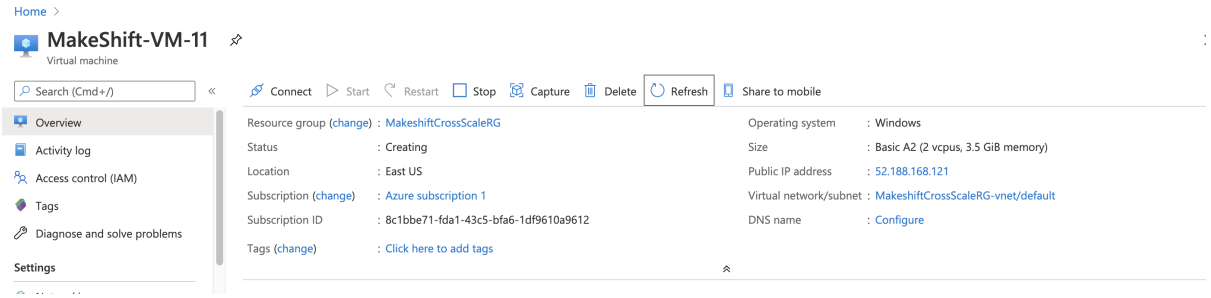

Figure 19: Creation of new VM during cloning

This is the sample application which points to the IP address 52.152.142.112, and while the cloning process is going on the sample web application is up and running without any downtime. This confirms that the VM being cloned is running.

Figure 20: Sample application during cloning

[2] https://docs.microsoft.com/en-us/rest/api/compute/virtualmachines/createorupdate

The cloning of the MakeShift-VM-11 is completed with the new scaled up size of the VM .i.e., Basic A2 and it's in the running mode.

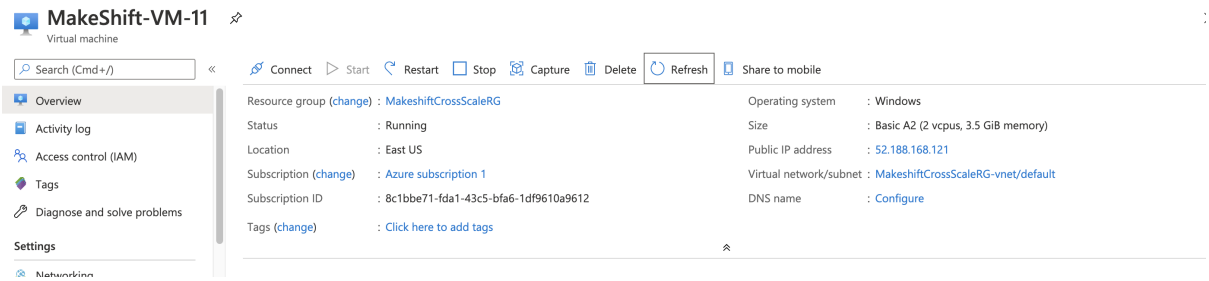

Figure 21: New cloned VM

When the old VM gets deleted, the resources associated with it such as network interface and disk are also deleted. This is done in order to avoid any unnecessary utilization of resources. [3] (Rloutlaw (n.d.)).

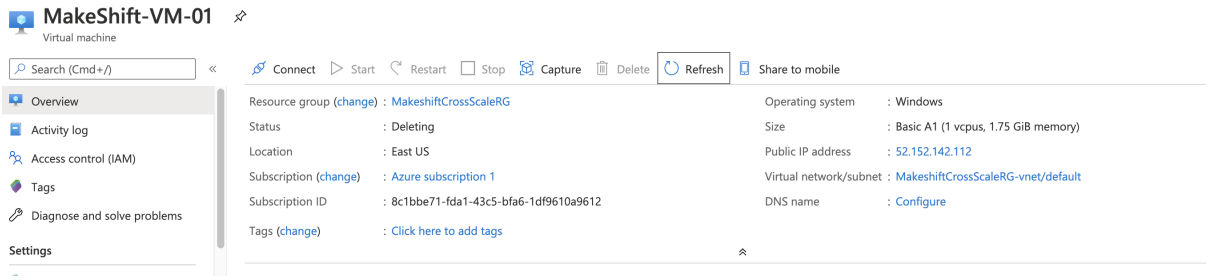

Figure 22: Deletion of the Old VM

[3] https://docs.microsoft.com/en-us/rest/api/compute/virtualmachines/delete

The cloning of the VM is completed and now the downtime of the application is also eliminated. The traffic manager points out to the new cloned VM's IP address .i.e., 52.188.168.121. The sample application is hosted in the new cloned VM.

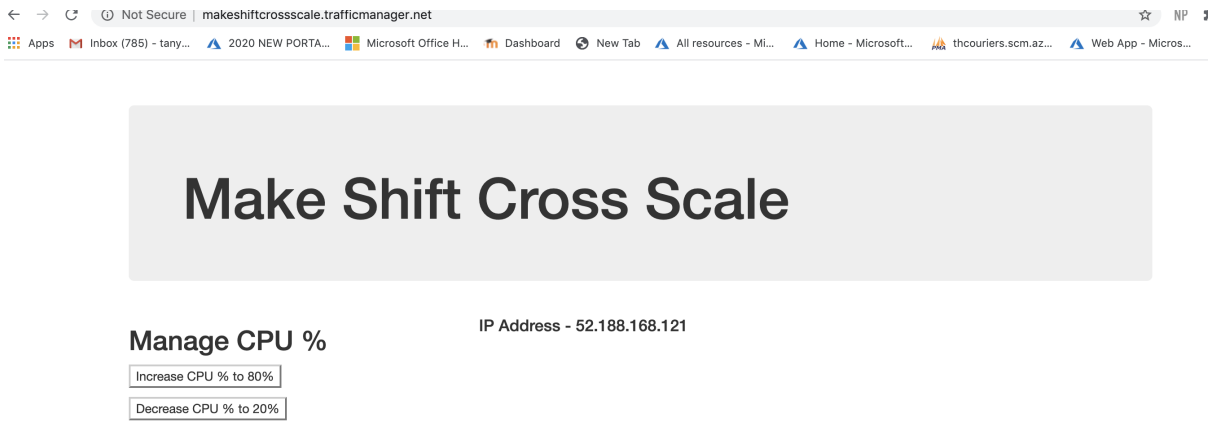

Figure 23: New IP address of the cloned VM

Lastly, after the cloning of the new VM and the old VM's deletion with its resources, this is how the resource group looks like.

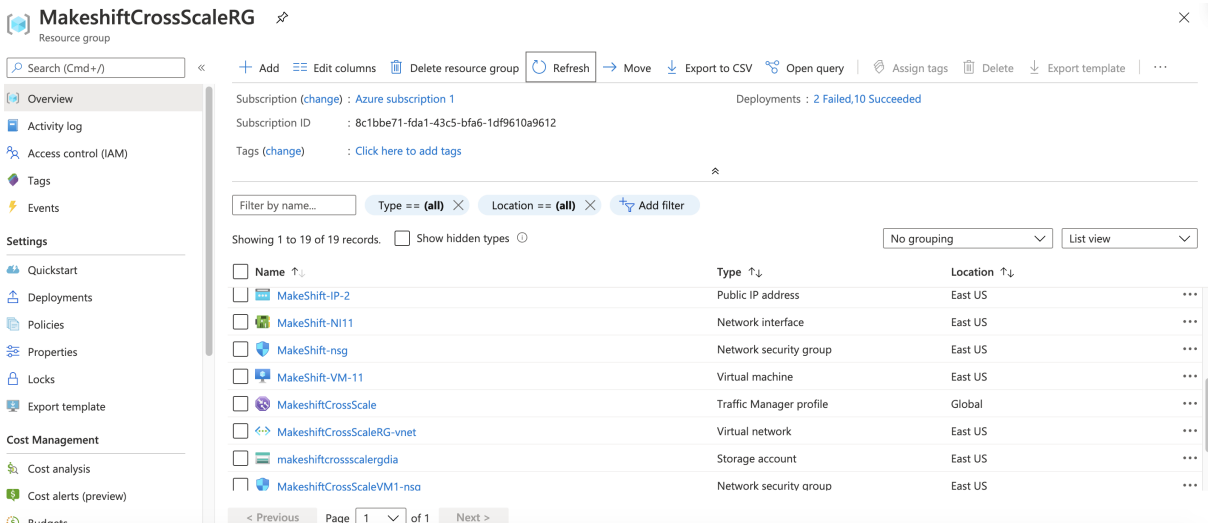

Figure 24: Updated Resource Group

The results obtained from this project are:
- Downtime is eliminated.
- Cloning time of the VM in average is 8.8 minutes approximately.
- Deletion of the old VM and its resources takes place in 4.5 minutes approximately.

References

Rloutlaw (n.d.). Virtual machines - delete (azure compute).
URL: *https://docs.microsoft.com/en-us/rest/api/compute/virtualmachines/delete*

Rloutlaw1 (n.d.). Virtual machines - create or update (azure compute).
URL: *https://docs.microsoft.com/en-us/rest/api/compute/virtualmachines/createorupdate*

Rohinkoul (n.d.). Azure traffic manager.
URL: *https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview*