



**Placement Empowerment Program**

***Cloud Computing and DevOps Centre***

***Implement auto-scaling in the cloud:set up an auto-scaling group for multiple vm’s to handle variable workloads.***

Name:M Anumitha Department:AML



**INTRODUCTION:**

In modern cloud environments, handling dynamic workloads efficiently is crucial for optimizing both performance and cost. One of the key strategies to address fluctuating demand is auto-scaling. Auto-scaling is the process of automatically adjusting the number of computing resources (such as Virtual Machines, or VMs) available in response to changing workload conditions.

This Proof of Concept (PoC) aims to demonstrate the implementation of an auto-scaling group in the cloud to handle variable workloads seamlessly. The objective is to show how resources can be dynamically allocated and deallocated based on real-time metrics like CPU utilization, memory usage, and network traffic. The result is a more efficient and responsive infrastructure that can scale up during periods of high demand and scale down when traffic is low, minimizing both operational costs and resource wastage.

**OVERVIEW: Top of FormBottom of Form**

This Proof of Concept (PoC) aims to demonstrate the implementation of an auto-scaling group in the cloud, where Virtual Machines (VMs) automatically adjust to handle variable workloads. Key components include an auto-scaling group that manages the number of VMs, a launch configuration that defines VM specifications, scaling policies to trigger automatic scaling actions based on metrics like CPU and memory usage, and health checks to ensure only healthy VMs are running. The benefits of this setup include cost efficiency by scaling down during low-demand periods, high availability during traffic spikes, and resource optimization by adjusting infrastructure in real-time. The expected outcome is a dynamic environment where VMs scale up or down based on workload, ensuring performance and minimizing costs. The PoC will be tested by simulating varying workloads to verify the system's scalability and responsiveness.

**Top of Form**

**Bottom of Form**

**OBJECTIVE:**

1. Implement an auto-scaling group to automatically adjust the number of VMs based on workload fluctuations.

2. Demonstrate dynamic resource scaling (up and down) based on real-time metrics like CPU utilization, memory usage, and network traffic.

3. Optimize performance and cost efficiency by automatically provisioning more resources during high demand and reducing resources when demand is low.

4. Ensure high availability by maintaining the necessary infrastructure to handle traffic spikes without manual intervention.

5. Validate the scalability of the cloud environment in response to variable workloads under different traffic conditions.

**Importance of Setting Up a Local Repository:**Top of FormBottom of Form

1.Cost Efficiency:  
Automatically scales down resources during periods of low demand, reducing costs by only using the resources needed at any given time.

2. Improved Performance:  
Ensures that the application can handle increased traffic during peak times by automatically scaling up resources to maintain optimal performance.

3. High Availability:  
Guarantees that sufficient resources are always available to handle unpredictable traffic spikes, ensuring minimal downtime or service disruptions.

4. Resource Optimization:  
Dynamically adjusts cloud resources to meet demand in real-time, preventing over-provisioning (which wastes resources) or under-provisioning (which affects performance).

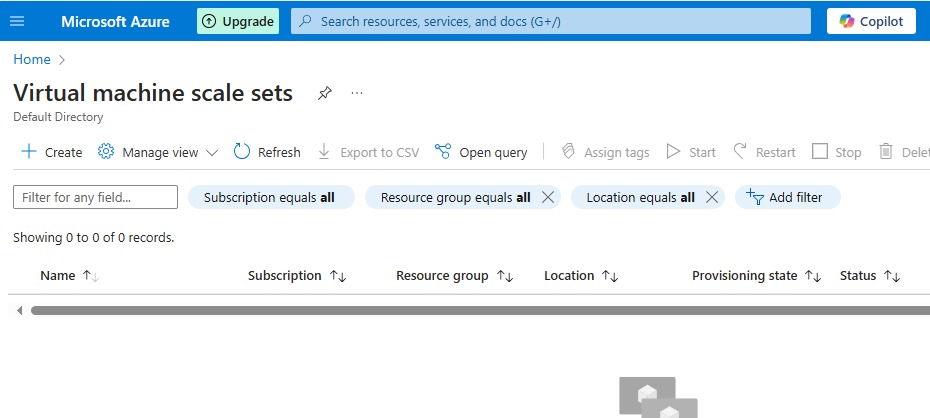
5. Operational Efficiency:  
Reduces manual intervention and operational overhead by automating the scaling process, allowing teams to focus on other critical tasks.

6. Scalability for Growth:  
Prepares the infrastructure to scale effortlessly as the workload grows, supporting the long-term growth and success of the business.

**STEP BY STEP OVERVIEW:**

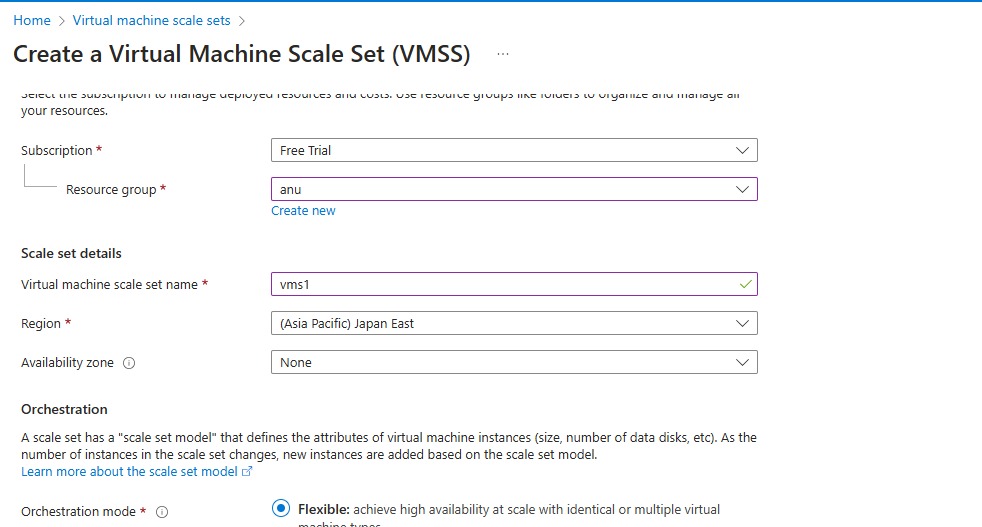
**STEP 1:**

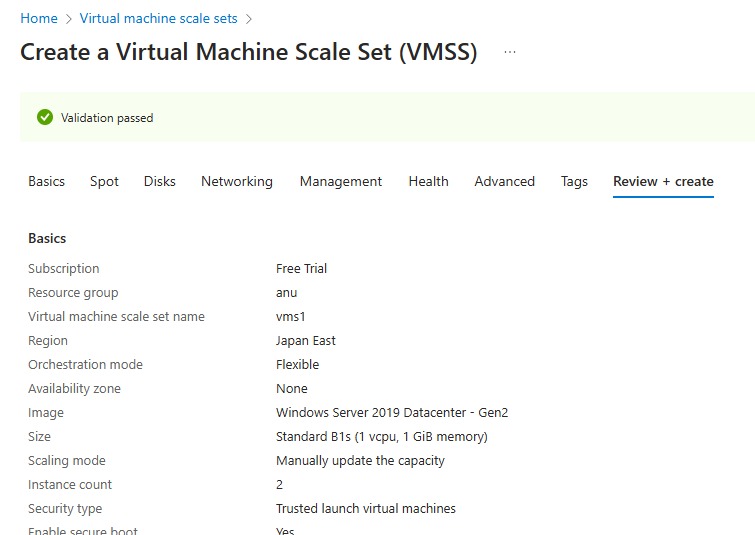
**Click virtual machine scale sets in left-bar menu:**



**STEP 2:**

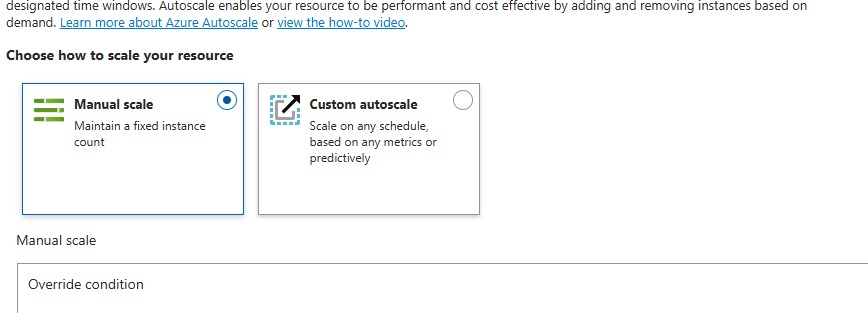
**Specify the required details to filled for virtual machine scale sets:**





**STEP 3:**

**Choose how to scale your resource like manual scale or custom scale:**

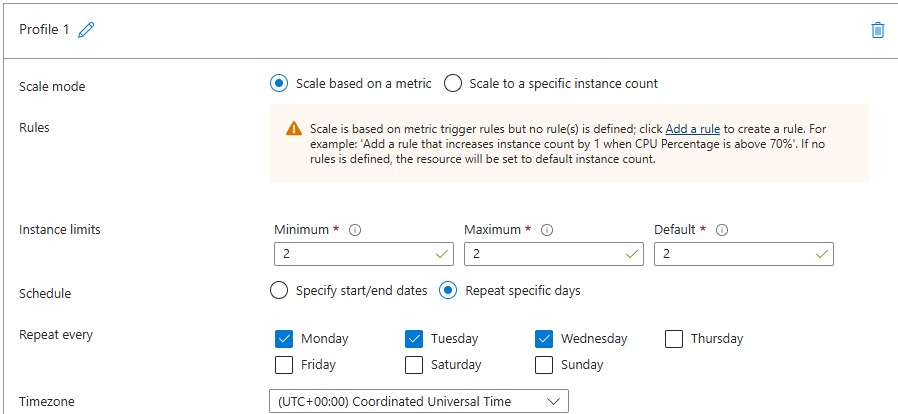


**STEP 4:**

**On a scale based metric method,**

**Give the details need to filled under this method:**

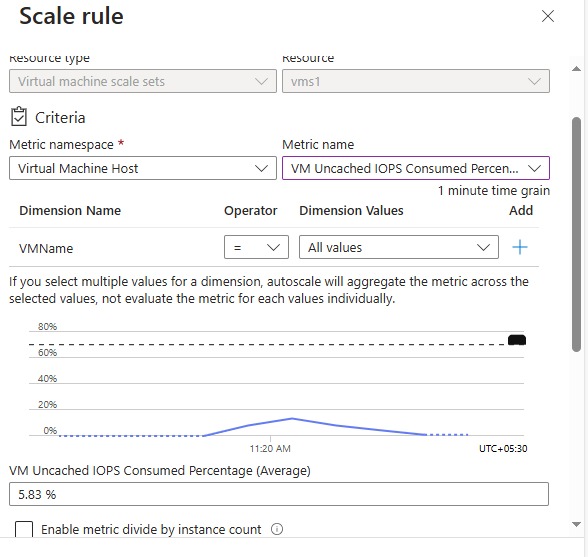
**Specify the instance limits,schedule for the cpu to run.**

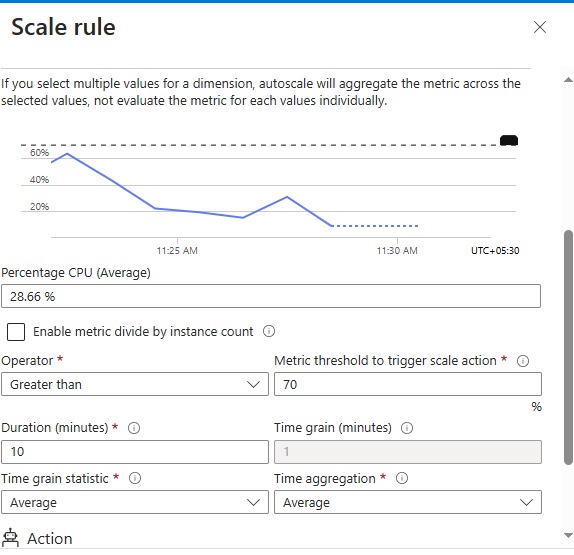


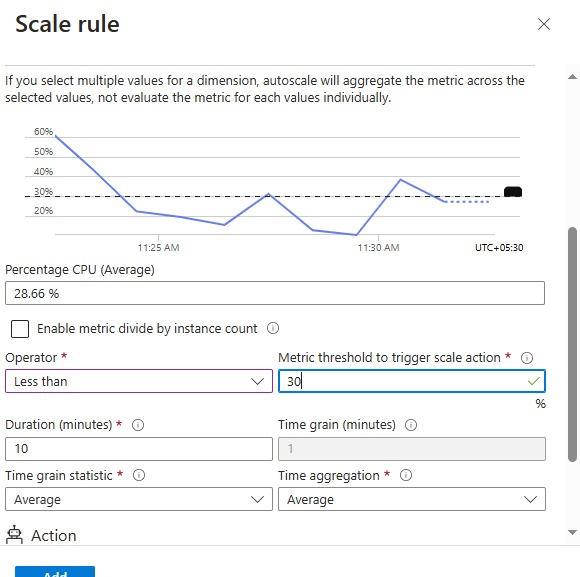
**Step 5:**

**Click add rule to scale up or scale down the vm:**

**To scale in give the operator to be greater than 70**







**EXPECTED OUTCOME:**

1. Dynamic Resource Scaling:  
VMs will automatically scale up during high traffic and scale down when demand is low, ensuring resources align with real-time workload needs.

2. Cost Savings:  
The system will reduce infrastructure costs by minimizing the number of active VMs during periods of low demand.

3. Improved Application Availability:  
Resources will be scaled to meet traffic spikes, ensuring continuous application availability and performance even under heavy load.

4. Optimized Performance:  
The cloud environment will efficiently handle varying workloads, maintaining high performance by provisioning the appropriate number of VMs.

5. Reduced Manual Intervention:  
Scaling will be fully automated, eliminating the need for manual adjustments and allowing the team to focus on other tasks.

6. Seamless Handling of Traffic Fluctuations:  
The infrastructure will respond seamlessly to changes in traffic, maintaining stability without disruptions.