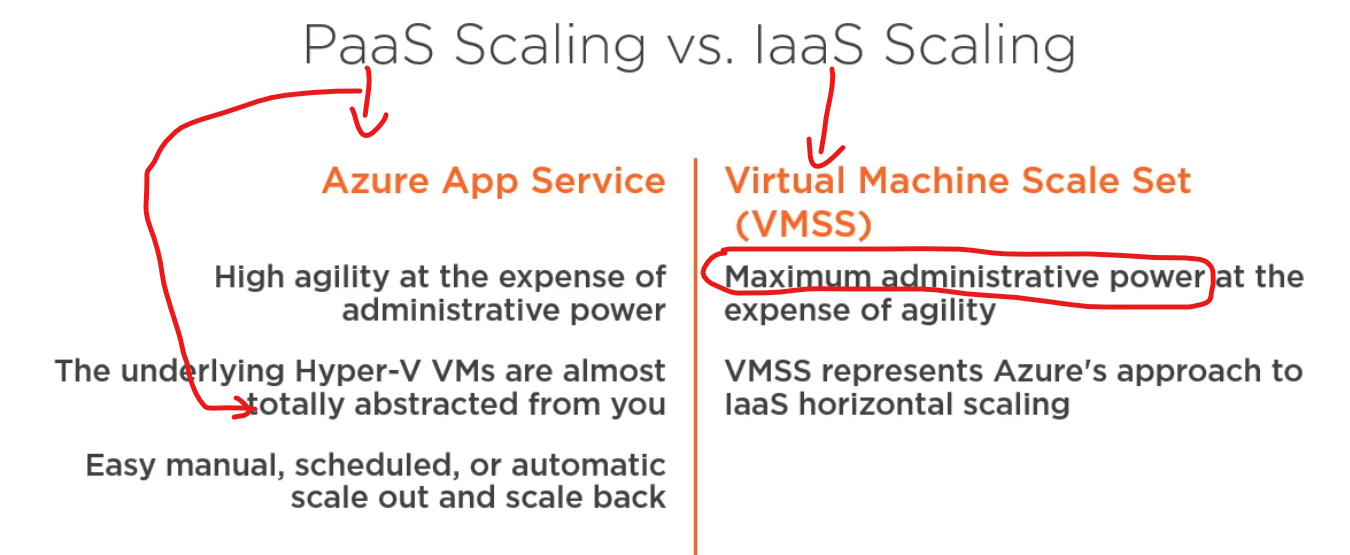
**VMSS:** Virtual Machine Scale Sets

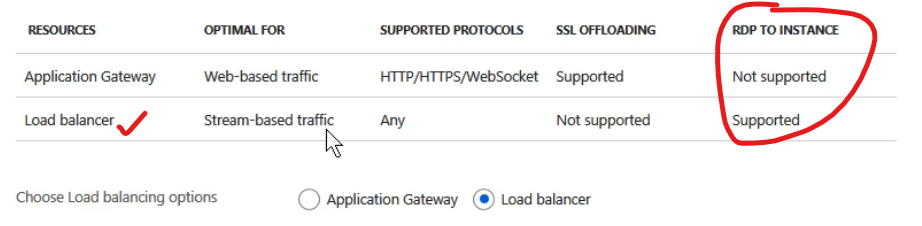
AZURE BATCH USES THIS TECHNOLOGY.

**Scale set:** Group of identically configured VMs (represents option to horizontally scale in Azure)

* Need to create to manage multiple VMs
* High availability and app resiliency
  + Eg. We have 2 VMs for frontend, but Black Friday is coming up, we know there’s gonna be a spike in incoming traffic: schedule an automatic deployment to scale horizontally (then scale back down after)
* Need for large scale (eg. 1000 VMs)
* Need for IaaS autoscale
  + Metric based autoscale threshold



**Low priority deployment:** Can save cost (up to 80%), eviction policy has to be set (when unused, Azure needs the computational power back: stop/deallocate or delete VM)  
not available for every instance size, though



(usually load balancer is desired here)

**NAT: Network Address Translation:** If we can’t connect to the horde of VMs but still want to administer: Azure starts a TCP port at 50000, increment if necessary (per VM)



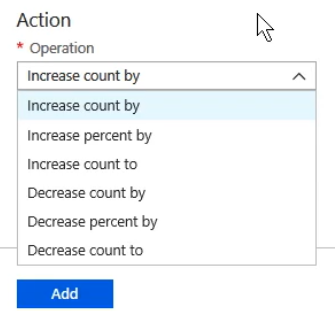
So backend load gets distributed (based on rules) to these VMs (each corresponding to a TCP service)

**Remote Desktop into single VM**: paste in load balancer’s IP:50000 🡪 boom, we are in VM (one of the horde, especially if low prio deployment is set, these VMs can suddenly disappear)

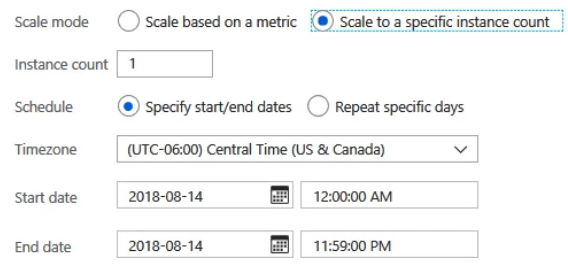
But these an anonymus clones of each other, you shouldn’t need to log into to a single one of them

* **Autoscale**
  + Manual: diy by CLI or Powershell, call a command when scaling is necessary, not automated
  + Scheduled
  + Metrics
    - Eg CPU usage on the average in the last 10 mins were above 70%, scale out

With AutoScale, set min instances, max instances and a default.



Cooldown period: Can bet set, eg. 5 mins to scale back down.



* **Low priority VMs (see low prio deployment)**
  + Take advantage of unutilized capacity of Azure (idling power from other customers/Azure)
  + Good for a one time gig, where there’s heavy computation
  + Saves cost, but:
    - INTERRUPTIONS WILL HAPPEN: workload has to be OK with that
    - VMs can be evicted AT ANY TIME (reclaimed by Azure if other customer needs)
    - Eviction policy can be deallocate only, so VM can spin up again (or Deleted, when VM gets completely removed when capacity is reclaimed by Azure – this is good for stateless operations)

When load balancing VMs, actually their network interfaces are being balanced, not the VMs themselves.