Tutorial S3: Batch Pool

This tutorial walks through how to connect all the resources you've created, including your Docker image and the storage container, to the creation of a Batch Pool of virtual machines.

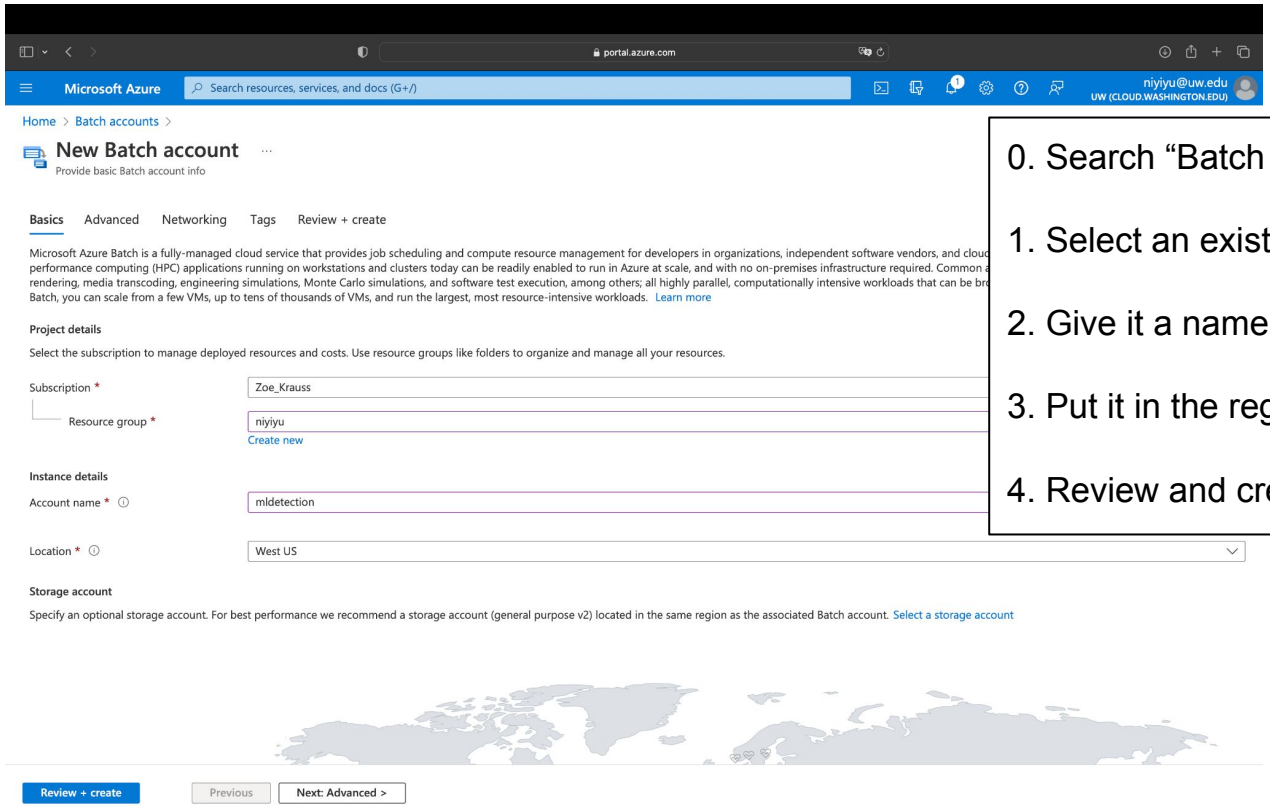
This tutorial covers how to do the following steps through the Azure desktop portal:

1. Create a **Batch account**
2. Create a **Batch Pool** within the Batch account
3. Check on the nodes within the created Batch Pool
4. Get Batch account name and key (for NotebookS4)

When done correctly, each node of the pool you create will have read/write access to everything in your storage container, and each node will be capable of running your Docker image, including its computing environment and all script directories.

1. Creating a Batch account

Start by creating a Batch account.



0. Search “Batch Accounts”

1. Select an existing resource group

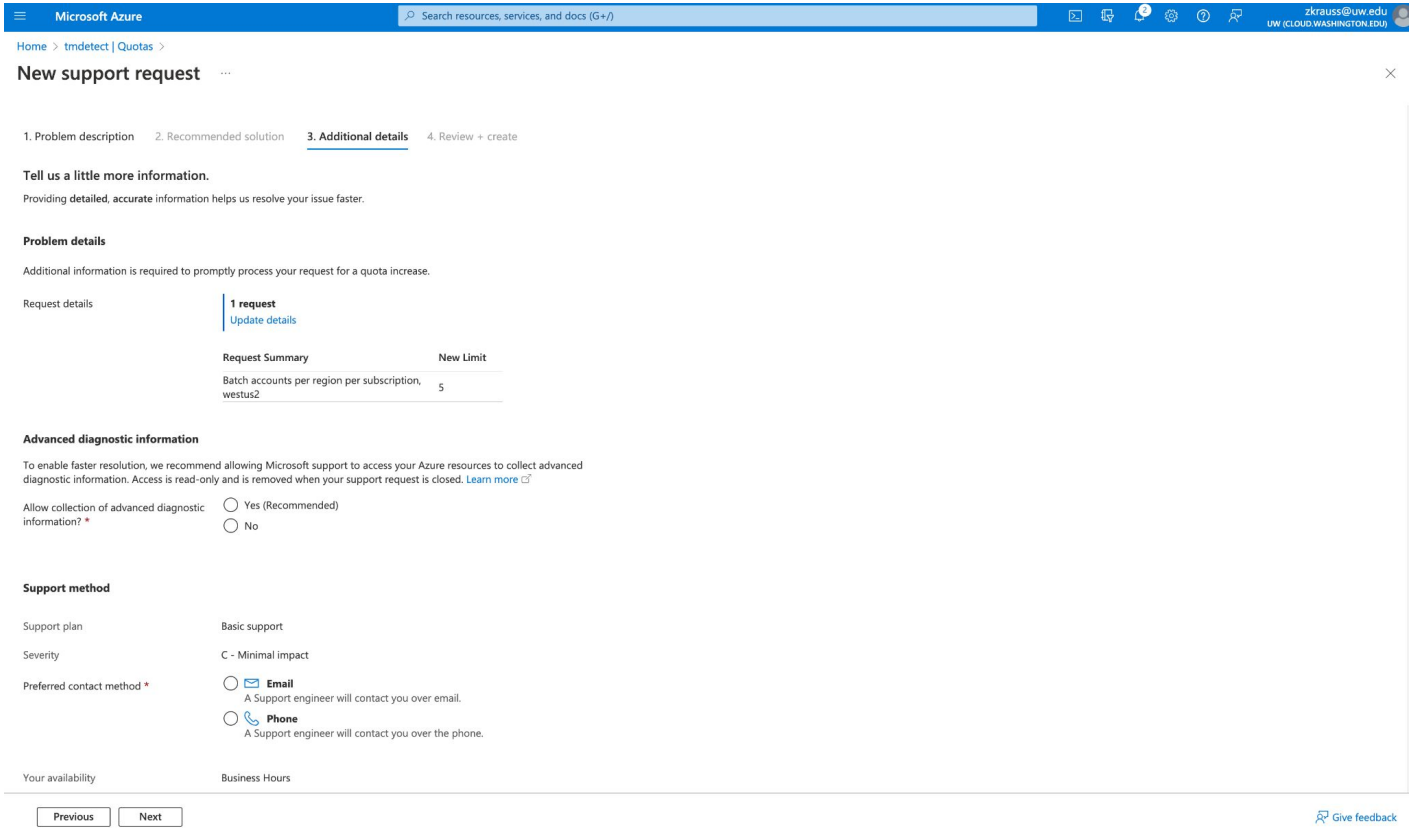
2. Give it a name

3. Put it in the region your resource group is in

4. Review and create

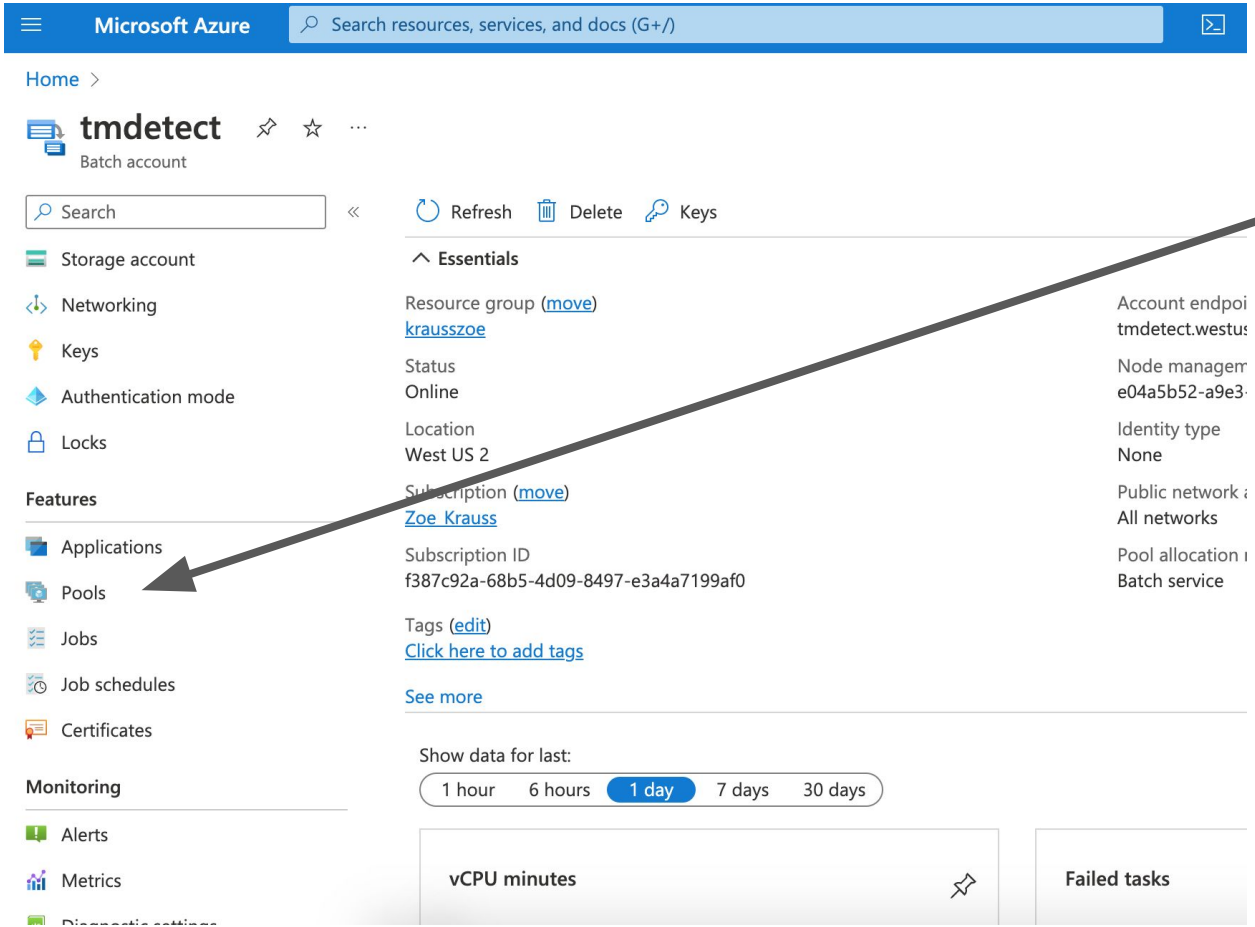
Quota errors can arise from the number of batch accounts allowed per region, or the number of nodes of a certain machine size allowed. You can request to be allowed more of these by going to your batch account page and following “Quotas → Request Quota increase”

See the following: https://learn.microsoft.com/en-us/azure/batch/batch-quota-limit

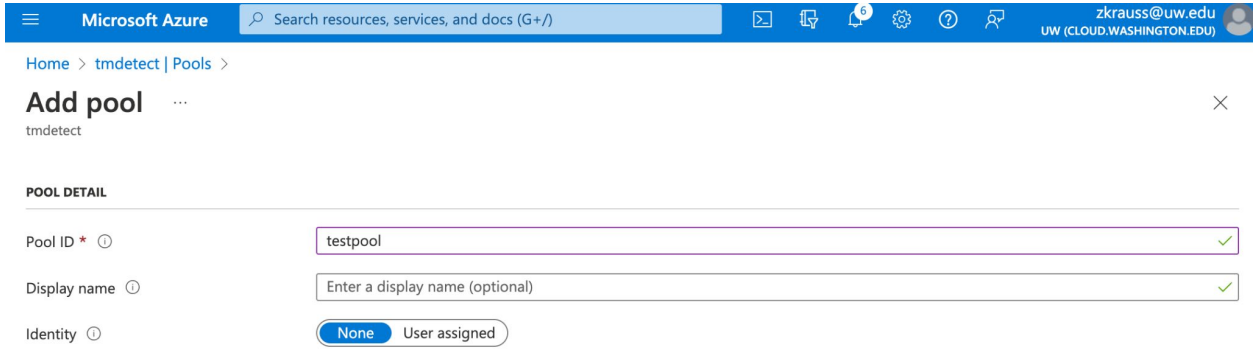


2. Create a Batch Pool

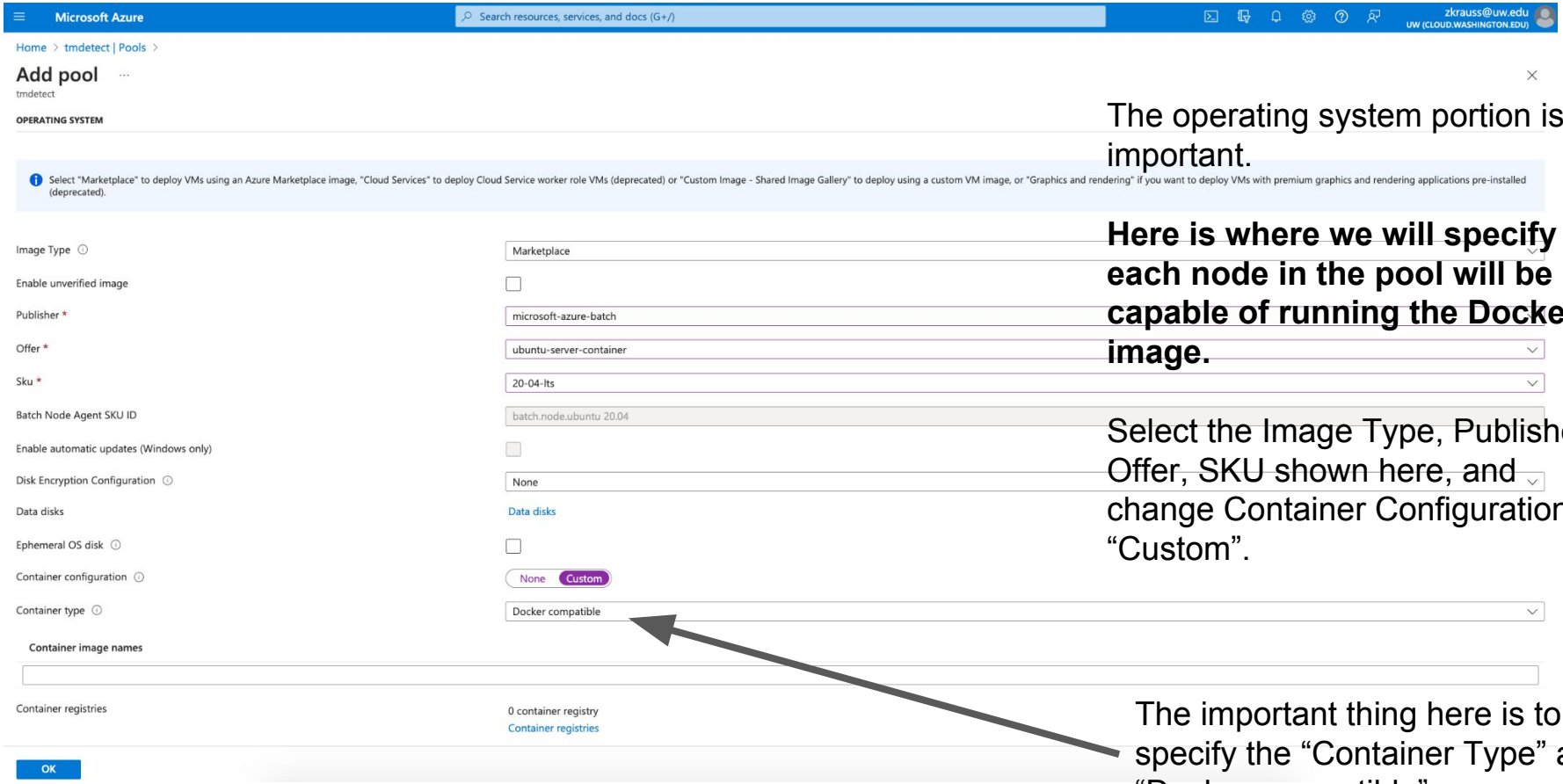
Now you can create a Pool.



Within your batch account, navigate to “Pools” on the sidebar.



Name the pool whatever you'd like. The Display name and Identity are unimportant.

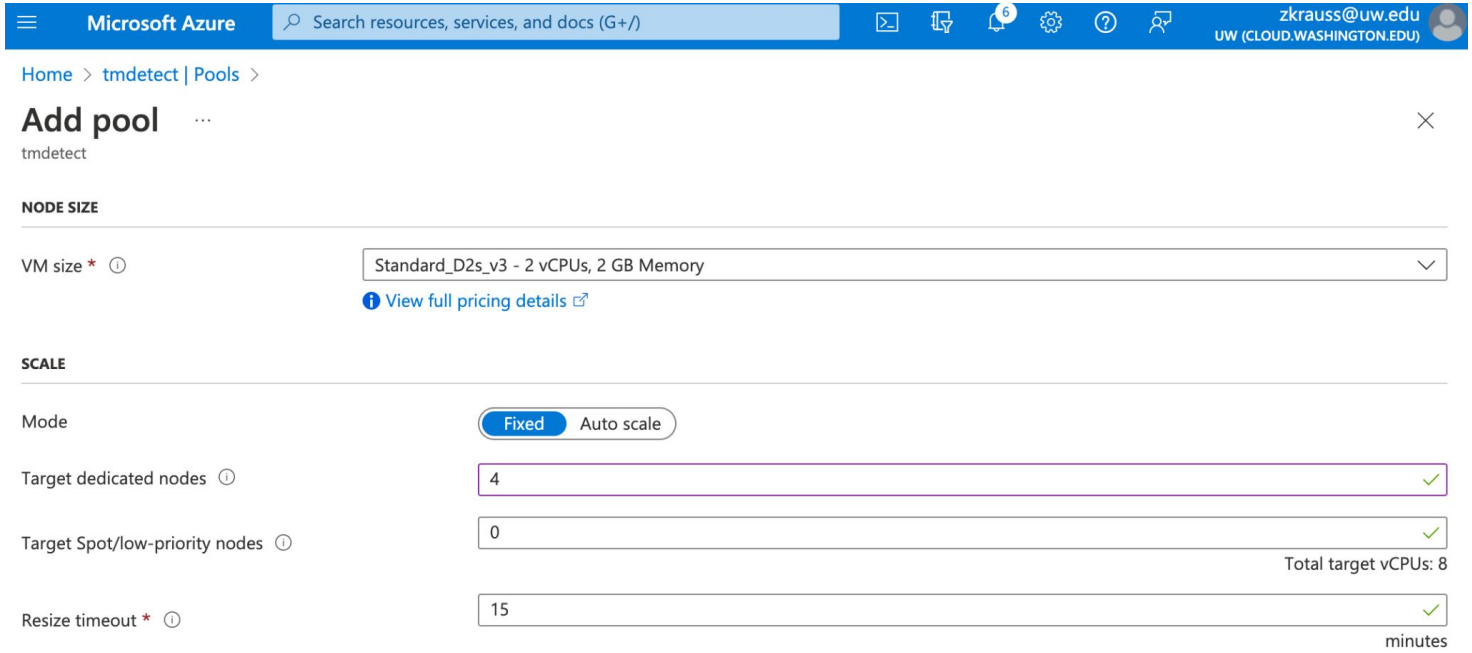


The operating system portion is important.

Here is where we will specify that each node in the pool will be capable of running the Docker image.

Select the Image Type, Publisher, Offer, SKU shown here, and change Container Configuration to “Custom”.

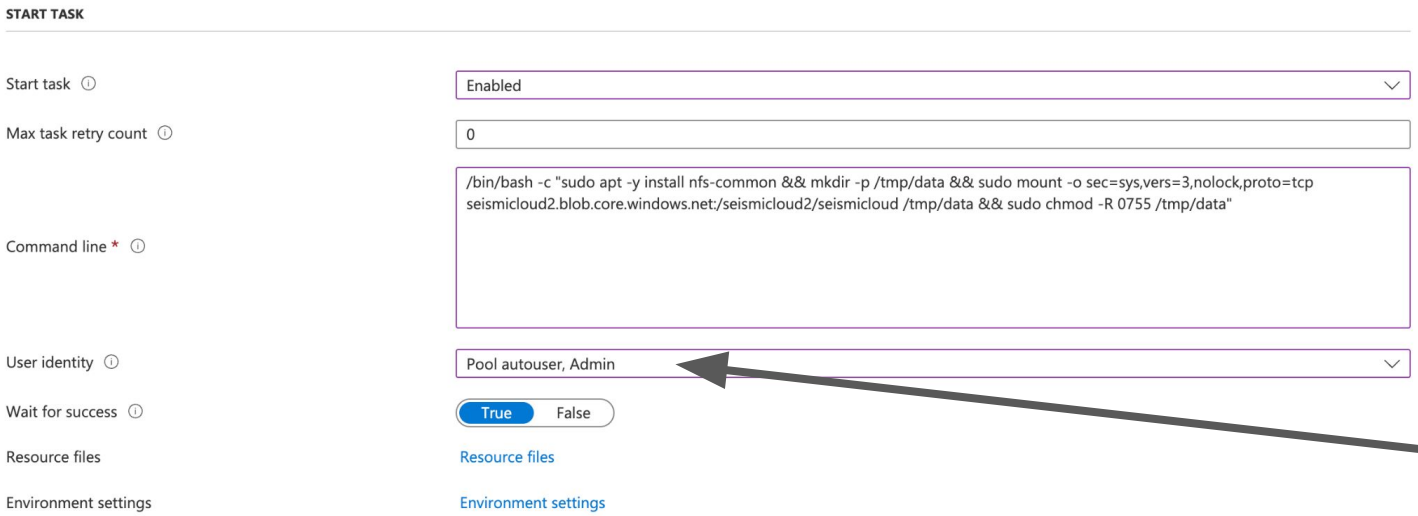
The important thing here is to specify the “Container Type” as “Docker compatible”.



Next you decide the size of your pool. The pool will be made up of N nodes which are each an individual virtual machine (VM), the size of which you choose in VM size under Node Size.

The Target dedicated nodes describe how many of the VMs of the size you chose will be in the pool. All other default inputs are fine.

We will connect each node in the pool to the storage container by running a Start Task on each node.



We'll walk through what this command contains on the next slide.

Make sure your User identity is set to Admin. You'll need this for permissions.

Start-up command:

```
/bin/bash -c "sudo apt -y install nfs-common && mkdir -p /tmp/data && sudo mount -o
sec=sys,vers=3,nolock,proto=tcp seismicloud.blob.core.windows.net:/seismicloud/seismicloud /tmp/data &&
sudo chmod -R 0755 /tmp/data"
```

Check the following link for details on the construction of this command:
https://learn.microsoft.com/en-us/azure/storage/blobs/network-file-system-protocol-support-how-to

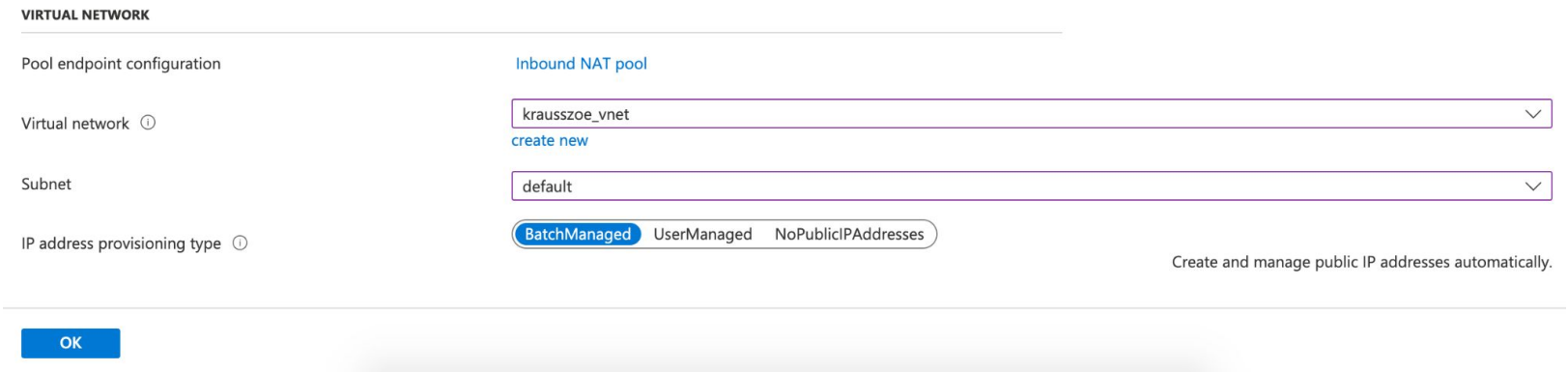
The && symbols represent separate bash commands.

>> sudo apt -y install nfs-common

>> mkdir -p /tmp/data
This creates an empty directory to mount the storage container within.

>> sudo mount -o sec=sys,vers=3,nlock,proto=tcp <accountname.blob.core.windows.net:/accountname/containername> /tmp/data
This is the command that actually mounts the storage container to the directory you made above. Within the <>, the names refer to the storage account and the storage container within the account.

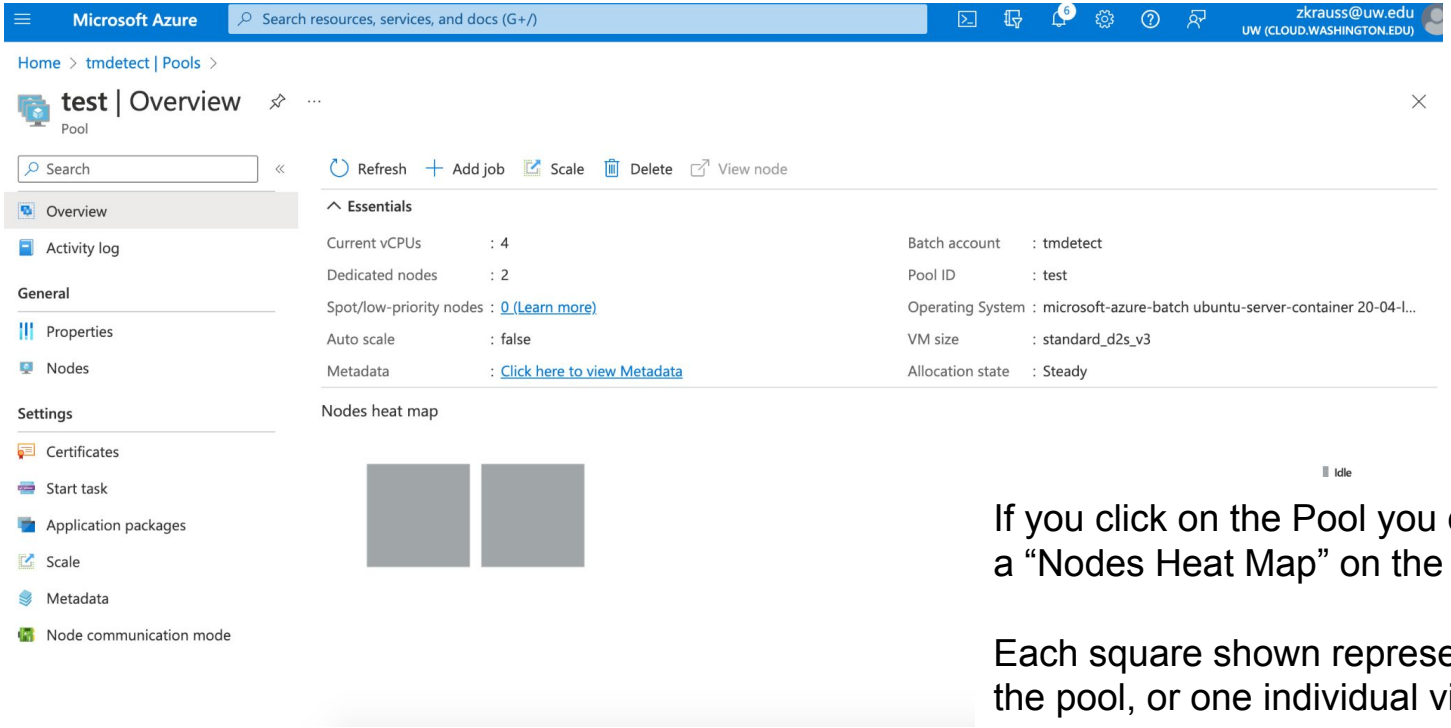
>> sudo chmod -R 0755 /tmp/data
And then this changes the permissions so you can write to it.



The last thing you need to specify outside of default settings is the virtual network that all of these resources are in.

Go ahead and create your pool!

3. Check on node status within the created Pool

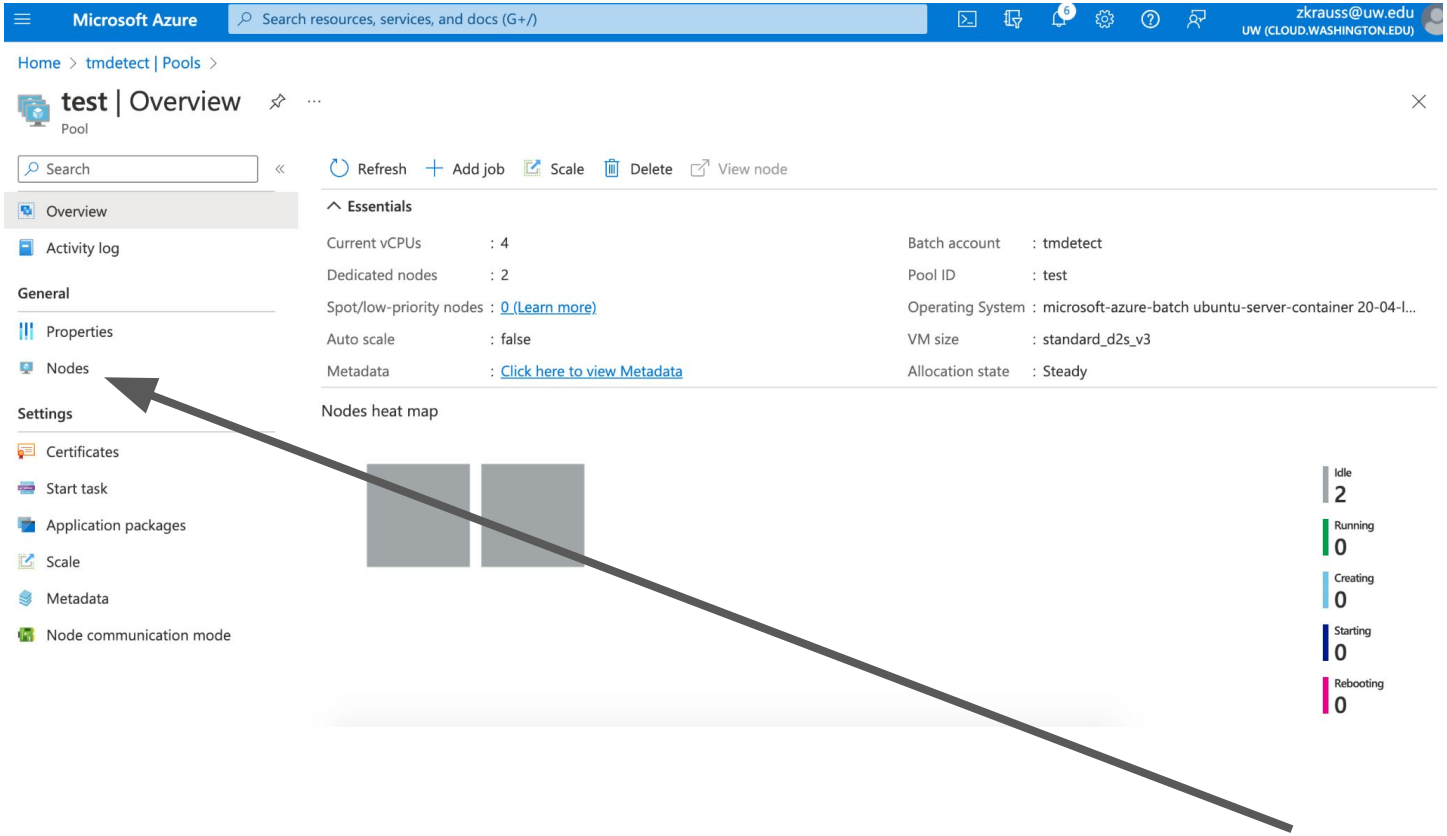


If you click on the Pool you created, you'll see a "Nodes Heat Map" on the Overview Screen.

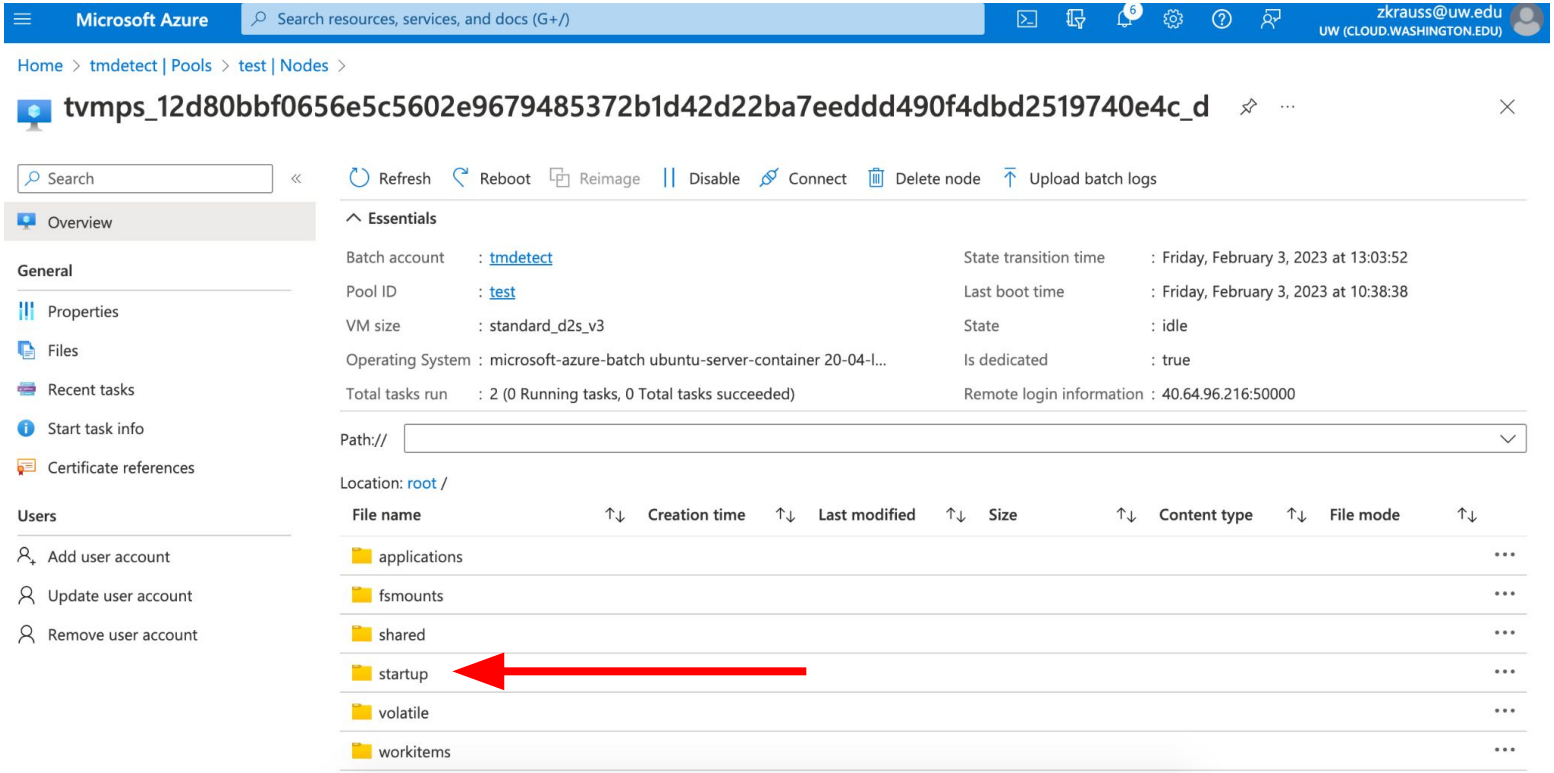
Each square shown represents one node in the pool, or one individual virtual machine.

The color of the square tells you if the node is running a job, starting up, etc. They will be green if running!

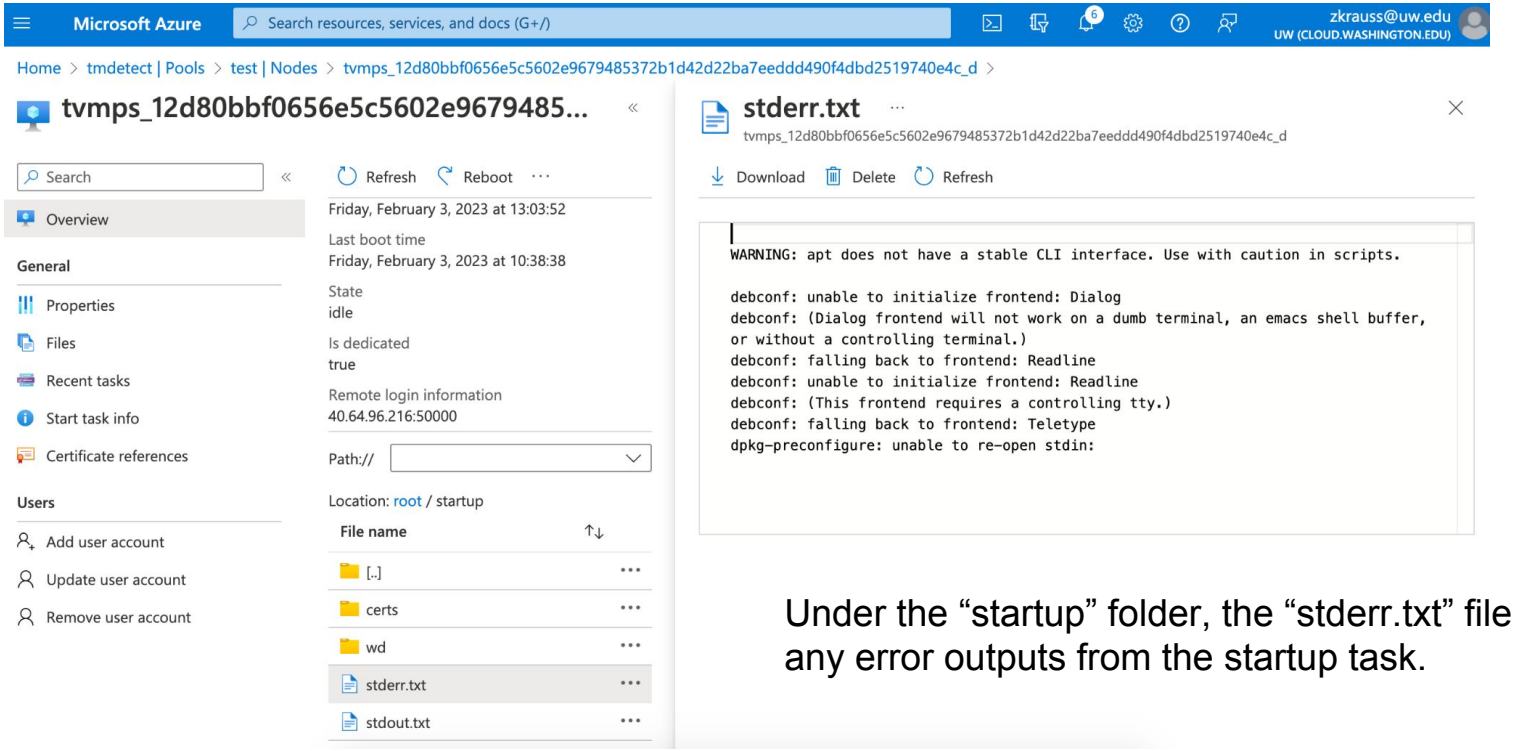
If the nodes appear to have a start task failed, there is a way to look into this…



To look into what's happening on each Node, click "Nodes" on the sidebar.



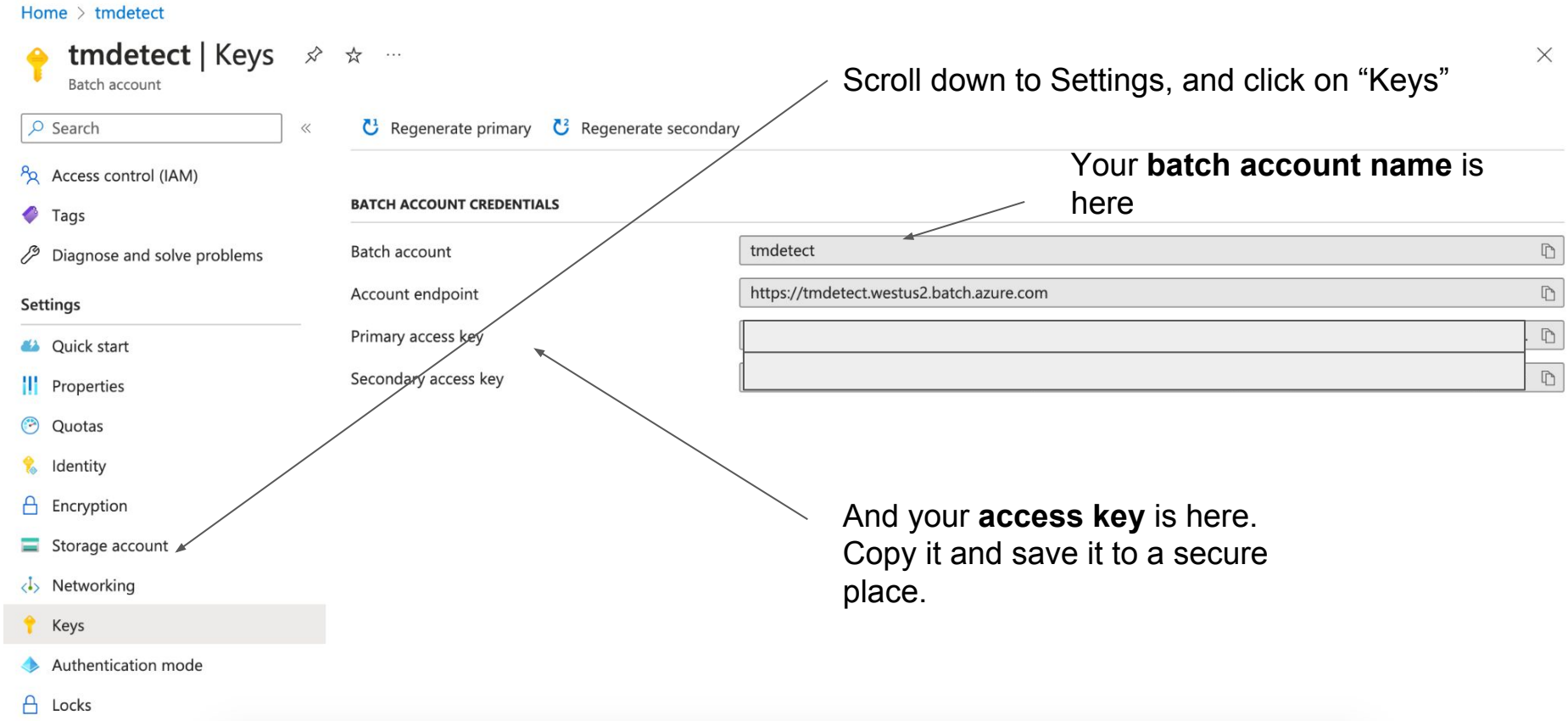
Click on one of the displayed Nodes and you will see the file system of the virtual machine displayed.
Many of these folders are just default on each Azure virtual machine.
The “startup” folder, though, can give us some insight if the startup task failed.



Under the “startup” folder, the “stderr.txt” file will contain any error outputs from the startup task.

4. Get Batch account name and key

Navigate to your Batch account



Now you have a pool all set-up with the ability to run the Docker image, and read and write to your storage container!

Next up: create jobs and send them to this pool. See Jupyter Notebook tutorial NotebookS4.