Final Project - REPORT

Azure Batch

Preamble:

We begin our installation and configuration of Azure Batch with the aid or either a MacBook or Linux machine, that I used during my work. A windows machine will also suffice; I do not have one at my disposal to reproduce the required steps.

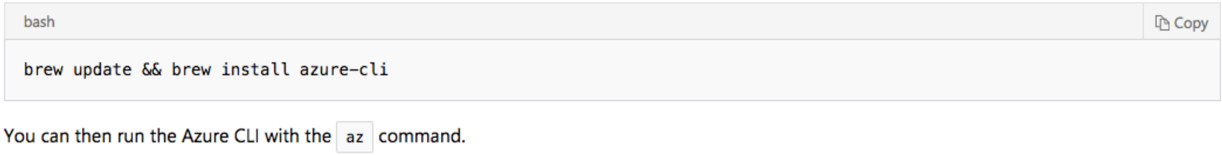
Problem Statement:

We have a need to deploy a number of commute nodes within the cloud to perform a series of jobs and tasks. We can scale these commute nodes up and down as needed. We can also easily destroy the environment and replace it through automation via scripts and the Azure CLI if needed. This project assumes that we have a need to deploy a few extra Apache webservers running on Ubuntu and host a website comprised of static html.

Pre-requisites:

We begin our installation by installing a few required elements. The first is the Azure CLI itself. This is located for download at https://docs.microsoft.com/en-us/cli/azure/install-azure-cli?view=azure-cli-latest

I already have Homerrew installed on my Mac to allow me to easily install via a single command.



```
brew update && brew install azure-cli
```

You can then run the Azure CLI with the `az` command.

Next, we will need to install the Azure Batch CLI Extensions with this command:

```
az extension add --source https://github.com/Azure/azure-batch-cli-extensions/releases/download/azure-batch-cli-extensions-2.0.1/azure_batch_cli_extensions-2.0.1-py2.py3-none-any.whl
```

Next we will run a shell(bash) script that I created that will completely create all the require items that are needed in <3 minutes. It will prompt for 2 questions:

1) How many nodes
2) Batch pool name (id)

IMPORTANT: I must make note that I coded the JSON files for use with a set poolID I called, "webservers" and I had to code my vnet/subnet and subscriptionID. You would need to make updates to these scripts as needed for your environment.

CREATING AZURE BATCH ACCOUNT SCRIPT

Create the files I have below with the same names. As noted earlier, updates to the scripts would be needed for your subscription name, resource group name, storage account names, etc.

SCRIPT NAME: create_project.sh

```
#!/bin/bash
# Feb 8, 2018
# Kirk Dahl
#
# Azure Deep Dive Training
# Final Project - Microsoft Azure Batch

clear
echo "STOP - have you logged into Azure yet? (y/n)"
read answer
case $answer in
   y) ;;
   n) exit;;
   *) exit;;
esac

##########################################
########## PREREQUISITES #################
##########################################


#CREATE RESOURCE GROUP
echo
echo
echo ======================================
echo "Creating Resource Group"
echo ======================================
az group create \
    --name rg-kirkdahl \
    --location eastus

#CREATE STORAGE ACCOUNT
echo
echo
echo ======================================
echo "Creating Storage Account"
echo ======================================
az storage account create \
    --resource-group rg-kirkdahl \
    --name sakirkdahl \
    --location eastus \
    --sku Standard_LRS

#CREATE VNET
echo
echo
echo ======================================
echo "Creating VNET"
echo ======================================
az network vnet create \
    --name vnet-kirkdahl \
    --resource-group rg-kirkdahl \
    --subnet-name subnet-kirkdahl


################################################################
########## BEGIN THE MICROSOFT BATCH INSTALLATION ##############
################################################################

#CREATE BATCH ACCOUNT
echo
echo
echo ======================================
echo "Creating Batch Account"
echo ======================================
az batch account create \
    --name batchkirkdahl \
    --storage-account sakirkdahl \
    --resource-group rg-kirkdahl \
    --location eastus

#UPLOAD APPLICATION
echo
echo
echo ======================================
echo "Creating Application Package"
echo ======================================
az batch application package create \
  --resource-group rg-kirkdahl \
  --name batchkirkdahl \
  --application-id PartyApp \
  --package-file ./18.zip \
  --version 1.0

#CREATE JOB
echo
echo
echo ======================================
echo "Creating Job"
echo ======================================
az batch job create \
    --id PartyApp \
    --account-name batchkirkdahl \
    --account-endpoint https://batchkirkdahl.eastus.batch.azure.com \
    --pool-id webservers
```

```
#CREATE TASKS – using json files
echo
echo
echo ======================================
echo "Creating Job Tasks"
echo ======================================
# FIRST TASK UNZIPS WEBSITE CODE
az batch task create \
--job-id PartyApp \
--account-name batchkirkdahl \
--account-endpoint https://batchkirkdahl.eastus.batch.azure.com \
--json-file ./batch-task.json


# SECOND TASK RESTART APACHE SERVER
az batch task create \
--job-id PartyApp \
--account-name batchkirkdahl \
--account-endpoint https://batchkirkdahl.eastus.batch.azure.com \
--json-file ./batch-task2.json

#CREATE BATCH POOL
echo
echo
echo ======================================
echo "Creating Batch Pool"
echo ======================================
az batch pool create \
  --template batch-pool.json \
  --account-name batchkirkdahl \
  --account-endpoint https://batchkirkdahl.eastus.batch.azure.com
```

SUPPORTING FILES

FILENAME: batch-pool.json

```
{
    "parameters": {
        "nodeCount": {
            "type": "int",
            "metadata": {
                "description": "The number of pool nodes"
            }
        },
        "poolId": {
            "type": "string",
            "metadata": {
                "description": "The pool ID "
            }
        }
    },
    "pool": {
        "type": "Microsoft.Batch/batchAccounts/pools",
        "apiVersion": "2016-12-01",
        "properties": {
            "id": "[parameters('poolId')]",
            "virtualMachineConfiguration": {
                "imageReference": {
                    "publisher": "Canonical",
                    "offer": "UbuntuServer",
                    "sku": "16.04.0-LTS",
                    "version": "latest"
                },
                "nodeAgentSKUId": "batch.node.ubuntu 16.04"
            },
            "vmSize": "STANDARD_A1",
            "enableInterNodeCommunication": true,
            "targetDedicatedNodes": "[parameters('nodeCount')]",
            "enableAutoScale": false,
            "maxTasksPerNode": 1,
            "taskSchedulingPolic": {
                "nodeFillType": "pack"
            },
            "url":
"https://batchkirkdahl.eastus.batch.azure.com/pools/webservers",
            "networkConfiguration": {
                "poolendpointConfiguration": {
                    "inboundNatPools": [
                        {
                        "backendPort": 80,
                        "frontendPortRangeEnd": 4000,
                        "frontendPortRangeStart": 3000,
                        "name": "http.0",
                        "networkSecurityGroupRules": [
                            {
                                "access": "allow",
                                "priority": 150,
                                "sourceAddressPrefix": "*"
                            }
                        ],
                        "protocol": "tcp"
                        }
                    ]

                },
                "subnetId": "/subscriptions/21ad651f-2ff2-49d2-87b2-
50c56efef22c/resourceGroups/rg-
kirkdahl/providers/Microsoft.Network/virtualNetworks/vnet-
kirkdahl/subnets/subnet-kirkdahl"
            },
            "applicationPackageReferences": [
                {
                    "applicationId": "PartyApp",
                    "version": 1.0
                }
            ],
            "packageReferences": [
                {
                    "type": "aptPackage",
                    "id": "apache2"
                },
                {
                    "type": "aptPackage",
                    "id": "unzip"
                }

            ]
        }
    }
}
```

FILENAME: batch-task.json

```
{
  "applicationPackageReferences": [
    {
      "applicationId": "partyapp",
      "version": "1.0"
    }
  ],
  "commandLine": "sudo -S unzip -u /mnt/batch/tasks/apppackages/partyapp*.zip
-d /var/www/html",
  "constraints": {
    "maxTaskRetryCount": 1
  },
  "containerSettings": null,
  "dependsOn": null,
  "displayName": "unzip",
  "id": "unzipSite",
  "userIdentity": {
    "autoUser": {
      "elevationLevel": "admin",
      "scope": "task"
    },
    "userName": null
  }

}
```

FILENAME: batch-task2.json

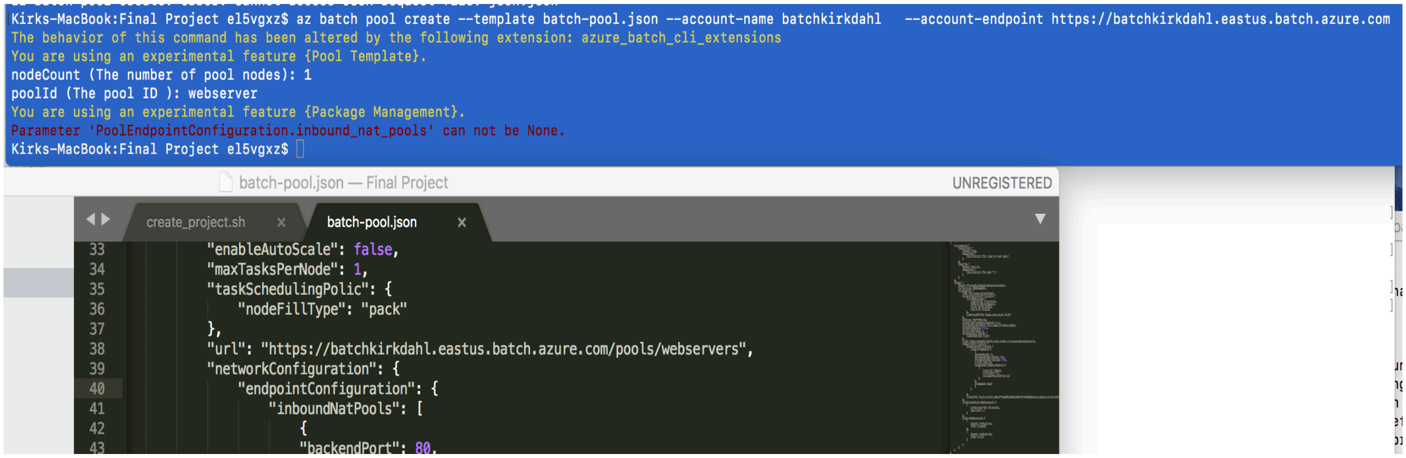
```
{
  "applicationPackageReferences": [
    {
      "applicationId": "partyapp",
      "version": "1.0"
    }
  ],
  "commandLine": "sudo -S systemctl restart apache2",
  "constraints": {
    "maxTaskRetryCount": 1
  },
  "containerSettings": null,
  "dependsOn": null,
  "displayName": "unzip",
  "id": "restartApache",
  "userIdentity": {
    "autoUser": {
      "elevationLevel": "admin",
      "scope": "task"
    },
    "userName": null
  }

}
```

Drawback – Bug - Unknown

During my working with these scripts, I ran across something that would prohibit me from viewing the websites from the internet. During installation of the batch pool, I use this NetworkConfiguration class. However, the use of “poolEndpointConfiguration is required by the Azure CLI, yet it creates NO inbound_nat_pools. The Azure Batch application does not yet allow for manual addition or manipulation of the auto-built load-balancers it creates as of yet. So it must be added during the installation in the GUI or by script. When I created the pool via the GUI and then performed a **`az batch pool show`** command, the output showed the json as “endpointConfiguration” – not poolEndpointConfiguration. I was caught in a catch-22. So to make my website viewable from the internet, I was forced to create the Batch Pool via the GUI.

```
"networkConfiguration": {
    "poolendpointConfiguration": {
        "inboundNatPools": [
            {
            "backendPort": 80,
            "frontendPortRangeEnd": 4000,
            "frontendPortRangeStart": 3000,
            "name": "http.0",
            "networkSecurityGroupRules": [
                {
                    "access": "allow",
                    "priority": 150,
                    "sourceAddressPrefix": "*"
                }
            ],
            "protocol": "tcp"
            }
        ]

    },
                    "subnetId": "/subscriptions/21ad651f-2ff2-49d2-87b2-50c56efef22c/resourceGroups/rg-kirkdahl/providers/Microsoft.Network/virtualNetworks/vnet-kirkdahl/subnets/subnet-kirkdahl"
               },
```

You can see the az command complains about missing inboundNatPools in “poolEndpointConfiguration”. So the code must have “pool”. However, the GUI creates the json with no “pool”



```
Kirks-MacBook:Final Project el5vgxz$ az batch pool create --template batch-pool.json --account-name batchkirkdahl   --account-endpoint https://batchkirkdahl.eastus.batch.azure.com
The behavior of this command has been altered by the following extension: azure_batch_cli_extensions
You are using an experimental feature {Pool Template}.
nodeCount (The number of pool nodes): 1
poolId (The pool ID ): webserver
You are using an experimental feature {Package Management}.
Parameter 'PoolEndpointConfiguration.inbound_nat_pools' can not be None.
Kirks-MacBook:Final Project el5vgxz$
```

```
        "enableAutoScale": false,
        "maxTasksPerNode": 1,
        "taskSchedulingPolic": {
            "nodeFillType": "pack"
        },
        "url": "https://batchkirkdahl.eastus.batch.azure.com/pools/webservers",
        "networkConfiguration": {
            "endpointConfiguration": {
                "inboundNatPools": [
                    {
                    "backendPort": 80,
```

OUTPUT FROM SCRIPTS

My scripts. The main create_project.sh uses the 3 json packages and the 18.zip is a copy of the free demo website that I will upload.

```
Kirks-MacBook:code el5vgxz$ ls -l
total 6240
-rw-r--r--@ 1 el5vgxz  staff  2181249 Feb  7 12:24 18.zip
-rw-r--r--@ 1 el5vgxz  staff     2736 Feb  8 20:15 batch-pool.json
-rw-r--r--@ 1 el5vgxz  staff      540 Feb  8 17:43 batch-task.json
-rw-r--r--@ 1 el5vgxz  staff      447 Feb  8 11:25 batch-task2.json
-rwxr-xr-x@ 1 el5vgxz  staff     3128 Feb  8 15:47 create_project.sh
```

I will execute the script with the time command to illustrate the speed at which this can be created. No barring the issue with the pool create issue noted above.

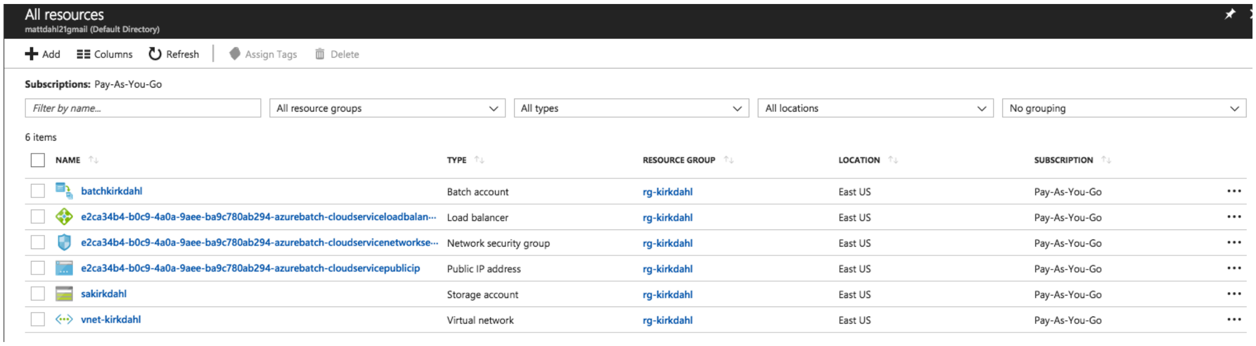
```
time ./create_project.sh
```

```
=====================================
Creating Resource Group
=====================================
{
  "id": "/subscriptions/21ad651f-2ff2-49d2-87b2-50c56efef22c/resourceGroups/rg-kirkdahl",
  "location": "eastus",
  "managedBy": null,
  "name": "rg-kirkdahl",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null
}


======================================
Creating Storage Account
======================================
{
  "accessTier": null,
  "creationTime": "2018-02-09T03:31:16.690234+00:00",
  "customDomain": null,
  "enableHttpsTrafficOnly": false,
  "encryption": {
    "keySource": "Microsoft.Storage",
    "keyVaultProperties": null,
    "services": {
      "blob": {
        "enabled": true,
        "lastEnabledTime": "2018-02-09T03:31:16.721500+00:00"
      },
      "file": {
        "enabled": true,
        "lastEnabledTime": "2018-02-09T03:31:16.721500+00:00"
      },
      "queue": null,
      "table": null
    }
  },
  "id": "/subscriptions/21ad651f-2ff2-49d2-87b2-50c56efef22c/resourceGroups/rg-
kirkdahl/providers/Microsoft.Storage/storageAccounts/sakirkdahl",
  "identity": null,
  "kind": "Storage",
  "lastGeoFailoverTime": null,
  "location": "eastus",
  "name": "sakirkdahl",
  "networkAcls": {
    "bypass": "AzureServices",
    "defaultAction": "Allow",
    "ipRules": [],
    "virtualNetworkRules": []
  },
  "primaryEndpoints": {
    "blob": "https://sakirkdahl.blob.core.windows.net/",
    "file": "https://sakirkdahl.file.core.windows.net/",
    "queue": "https://sakirkdahl.queue.core.windows.net/",
    "table": "https://sakirkdahl.table.core.windows.net/"
  },
  "primaryLocation": "eastus",
  "provisioningState": "Succeeded",
  "resourceGroup": "rg-kirkdahl",
  "secondaryEndpoints": null,
  "secondaryLocation": null,
  "sku": {
    "name": "Standard_LRS",
    "tier": "Standard"
  },
  "statusOfPrimary": "available",
  "statusOfSecondary": null,
  "tags": {},
  "type": "Microsoft.Storage/storageAccounts"
}


=====================================
Creating VNET
=====================================
{
  "newVNet": {
    "addressSpace": {
      "addressPrefixes": [
        "10.0.0.0/16"
      ]
    },
    "dhcpOptions": {
      "dnsServers": []
    },
    "enableDdosProtection": false,
    "enableVmProtection": false,
    "etag": "W/\"549d3b42-0a87-4891-a7ec-d857ee305ea9\"",
    "id": "/subscriptions/21ad651f-2ff2-49d2-87b2-50c56efef22c/resourceGroups/rg-
kirkdahl/providers/Microsoft.Network/virtualNetworks/vnet-kirkdahl",
    "location": "eastus",
    "name": "vnet-kirkdahl",
    "provisioningState": "Succeeded",
    "resourceGroup": "rg-kirkdahl",
    "resourceGuid": "dc03c0e8-6906-47f5-975c-9a004bbe0f83",
    "subnets": [
      {
        "addressPrefix": "10.0.0.0/24",
        "etag": "W/\"549d3b42-0a87-4891-a7ec-d857ee305ea9\"",
        "id": "/subscriptions/21ad651f-2ff2-49d2-87b2-50c56efef22c/resourceGroups/rg-
kirkdahl/providers/Microsoft.Network/virtualNetworks/vnet-kirkdahl/subnets/subnet-
kirkdahl",
        "ipConfigurations": null,
        "name": "subnet-kirkdahl",
        "networkSecurityGroup": null,
        "provisioningState": "Succeeded",
        "resourceGroup": "rg-kirkdahl",
        "resourceNavigationLinks": null,
        "routeTable": null,
        "serviceEndpoints": null
      }
    ],
    "tags": {},
    "type": "Microsoft.Network/virtualNetworks",
    "virtualNetworkPeerings": []
  }
}


=====================================
Creating Batch Account
=====================================
{
  "accountEndpoint": "batchkirkdahl.eastus.batch.azure.com",
  "activeJobAndJobScheduleQuota": 20,
  "autoStorage": {
    "lastKeySync": "2018-02-09T03:32:02.293335+00:00",
    "storageAccountId": "/subscriptions/21ad651f-2ff2-49d2-87b2-
50c56efef22c/resourceGroups/rg-
kirkdahl/providers/Microsoft.Storage/storageAccounts/sakirkdahl"
  },
  "dedicatedCoreQuota": 20,
  "id": "/subscriptions/21ad651f-2ff2-49d2-87b2-50c56efef22c/resourceGroups/rg-
kirkdahl/providers/Microsoft.Batch/batchAccounts/batchkirkdahl",
  "keyVaultReference": null,
  "location": "eastus",
  "lowPriorityCoreQuota": 20,
  "name": "batchkirkdahl",
  "poolAllocationMode": "BatchService",
  "poolQuota": 20,
  "provisioningState": "Succeeded",
  "resourceGroup": "rg-kirkdahl",
  "tags": null,
  "type": "Microsoft.Batch/batchAccounts"
}


======================================
Creating Application Package
======================================
{
  "format": "zip",
  "id": "PartyApp",
  "lastActivationTime": "2018-02-09T03:32:26.179577+00:00",
  "state": "active",
  "storageUrl": "https://sakirkdahl.blob.core.windows.net/app-partyapp-
627b1b2b504c2257c681dab6e3ba291e70663afc/partyapp-1.0-5380fd75-9aa1-468e-a88b-
b0d1ddaeaa08?sv=2015-04-
05&sr=b&sig=LUiGrxpeA9FZVEDg%2FDLiIgMFhJuBV%2BLVUdYy%2FqAjEk0%3D&st=2018-02-
09T03%3A27%3A26Z&se=2018-02-09T07%3A32%3A26Z&sp=rw",
  "storageUrlExpiry": "2018-02-09T07:32:26.903629+00:00",
  "version": "1.0"
}


======================================
Creating Job
======================================
The behavior of this command has been altered by the following extension:
azure_batch_cli_extensions


======================================
Creating Job Tasks
======================================
{
  "affinityInfo": null,
  "applicationPackageReferences": [
    {
      "applicationId": "partyapp",
      "version": "1.0"
    }
  ],
  "authenticationTokenSettings": null,
  "commandLine": "sudo -S unzip -u /mnt/batch/tasks/apppackages/partyapp*.zip -d
/var/www/html; cp /var/www/html/partytime/index.html /var/www/html",
  "constraints": {
    "maxTaskRetryCount": 1,
    "maxWallClockTime": "10675199 days, 2:48:05.477581",
    "retentionTime": "10675199 days, 2:48:05.477581"
  },
  "containerSettings": null,
  "creationTime": "2018-02-09T03:32:32.515388+00:00",
  "dependsOn": null,
  "displayName": "unzip",
  "eTag": "0x8D56F6DC1222C5D",
  "environmentSettings": null,
  "executionInfo": {
    "containerInfo": null,
    "endTime": null,
    "exitCode": null,
    "failureInfo": null,
    "lastRequeueTime": null,
    "lastRetryTime": null,
    "requeueCount": 0,
    "result": null,
    "retryCount": 0,
    "startTime": null
  },
  "exitConditions": null,
  "id": "unzipSite",
  "lastModified": "2018-02-09T03:32:32.515388+00:00",
  "multiInstanceSettings": null,
  "nodeInfo": null,
  "outputFiles": null,
  "previousState": null,
  "previousStateTransitionTime": null,
  "resourceFiles": null,
  "state": "active",
  "stateTransitionTime": "2018-02-09T03:32:32.515388+00:00",
  "stats": null,
  "url": "https://batchkirkdahl.eastus.batch.azure.com/jobs/PartyApp/tasks/unzipSite",
  "userIdentity": {
    "autoUser": {
      "elevationLevel": "admin",
      "scope": "task"
    },
    "userName": null
  }
}
{
  "affinityInfo": null,
  "applicationPackageReferences": [
    {
      "applicationId": "partyapp",
      "version": "1.0"
    }
  ],
  "authenticationTokenSettings": null,
  "commandLine": "sudo -S systemctl restart apache2",
  "constraints": {
    "maxTaskRetryCount": 1,
    "maxWallClockTime": "10675199 days, 2:48:05.477581",
    "retentionTime": "10675199 days, 2:48:05.477581"
  },
  "containerSettings": null,
  "creationTime": "2018-02-09T03:32:35.129339+00:00",
  "dependsOn": null,
  "displayName": "unzip",
  "eTag": "0x8D56F6DC2B107D5",
  "environmentSettings": null,
  "executionInfo": {
    "containerInfo": null,
    "endTime": null,
    "exitCode": null,
    "failureInfo": null,
    "lastRequeueTime": null,
    "lastRetryTime": null,
    "requeueCount": 0,
    "result": null,
    "retryCount": 0,
    "startTime": null
  },
  "exitConditions": null,
  "id": "restartApache",
  "lastModified": "2018-02-09T03:32:35.129339+00:00",
  "multiInstanceSettings": null,
  "nodeInfo": null,
  "outputFiles": null,
  "previousState": null,
  "previousStateTransitionTime": null,
  "resourceFiles": null,
  "state": "active",
  "stateTransitionTime": "2018-02-09T03:32:35.129339+00:00",
  "stats": null,
  "url": "https://batchkirkdahl.eastus.batch.azure.com/jobs/PartyApp/tasks/restartApache",
  "userIdentity": {
    "autoUser": {
      "elevationLevel": "admin",
      "scope": "task"
    },
    "userName": null
  }
}


======================================
Creating Batch Pool
======================================
The behavior of this command has been altered by the following extension:
azure_batch_cli_extensions
You are using an experimental feature {Pool Template}.
nodeCount (The number of pool nodes): 2
poolId (The pool ID ): webservers
You are using an experimental feature {Package Management}.

real    1m45.783s
user    0m4.814s
sys     0m2.816s
Kirks-MacBook:code el5vgxz$
```

SCREEN SHOTS ILLUSTRATING PORTAL VIA OF OUTPUT

I will not show screenshots that show the various items that were created in the portal via the script above that took just under 2 minutes to execute.

All resources



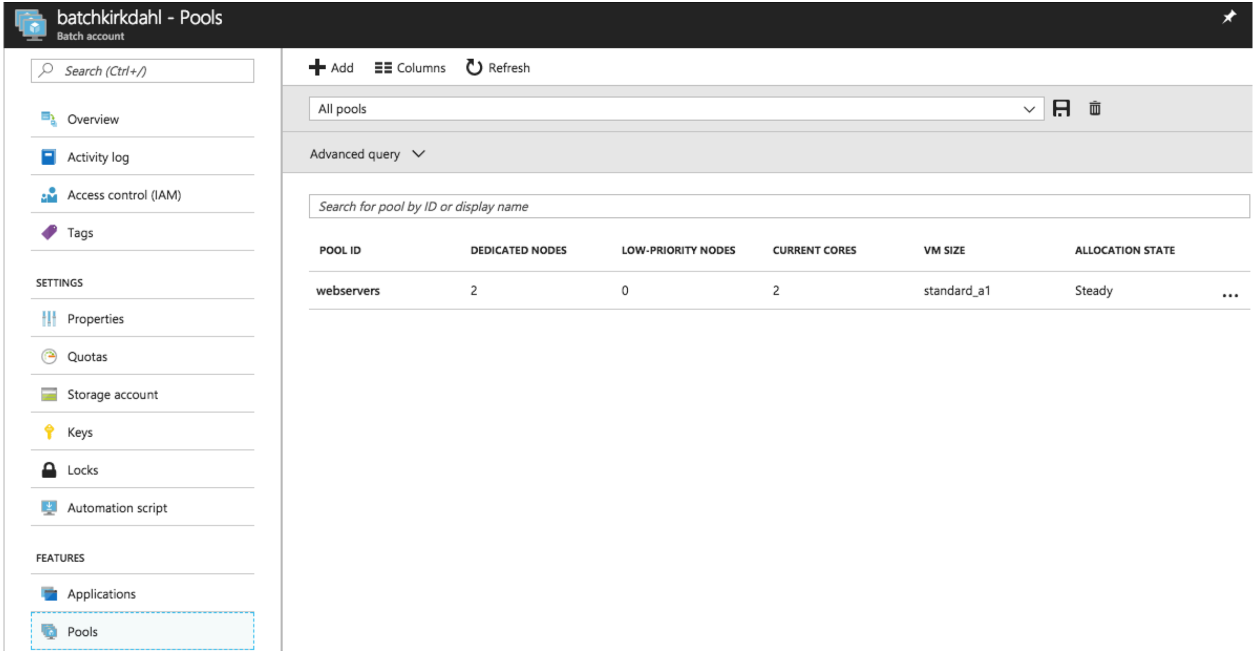
Subscriptions: Pay-As-You-Go

Filter by name... | All resource groups | All types | All locations | No grouping

6 items

NAME	TYPE	RESOURCE GROUP	LOCATION	SUBSCRIPTION
batchkirkdahl	Batch account	rg-kirkdahl	East US	Pay-As-You-Go
e2ca34b4-b0c9-4a0a-9aee-ba9c780ab294-azurebatch-cloudserviceloadbalan···	Load balancer	rg-kirkdahl	East US	Pay-As-You-Go
e2ca34b4-b0c9-4a0a-9aee-ba9c780ab294-azurebatch-cloudservicenetworkse···	Network security group	rg-kirkdahl	East US	Pay-As-You-Go
e2ca34b4-b0c9-4a0a-9aee-ba9c780ab294-azurebatch-cloudservicepublicip	Public IP address	rg-kirkdahl	East US	Pay-As-You-Go
sakirkdahl	Storage account	rg-kirkdahl	East US	Pay-As-You-Go
vnet-kirkdahl	Virtual network	rg-kirkdahl	East US	Pay-As-You-Go

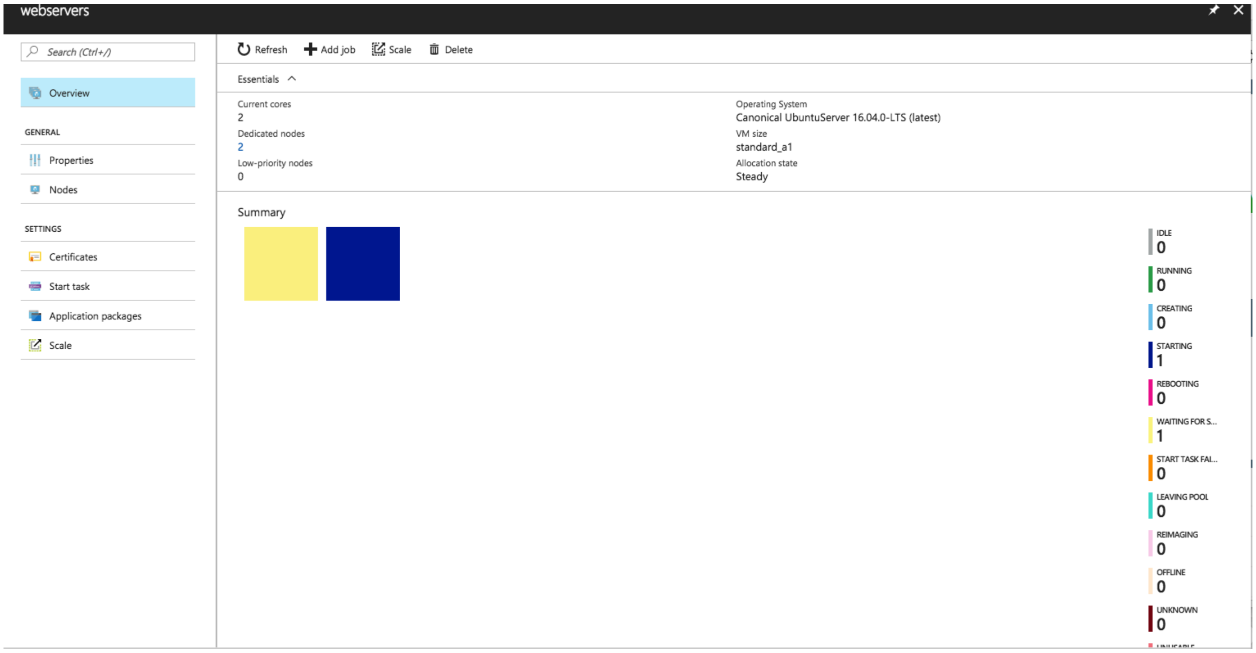
Batch Pool (webservers) showing 2 nodes



All pools

POOL ID	DEDICATED NODES	LOW-PRIORITY NODES	CURRENT CORES	VM SIZE	ALLOCATION STATE
webservers	2	0	2	standard_a1	Steady

One Node is starting and the other has started and is now waiting on startup tasks (install unzip and apache)

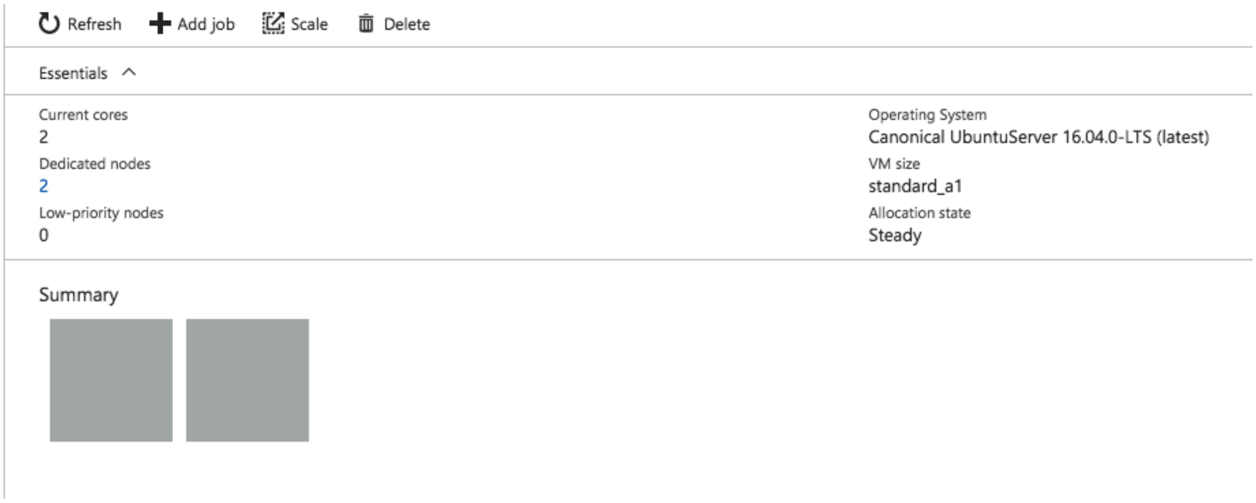


Current cores
2
Dedicated nodes
2
Low-priority nodes
0

Operating System
Canonical UbuntuServer 16.04.0-LTS (latest)
VM size
standard_a1
Allocation state
Steady

IDLE 0
RUNNING 0
CREATING 0
STARTING 1
REBOOTING 0
WAITING FOR S... 1
START TASK FAI... 0
LEAVING POOL 0
REIMAGING 0
OFFLINE 0
UNKNOWN 0

Both nodes have completed startup tasks and are idle



Current cores
2

Dedicated nodes
2

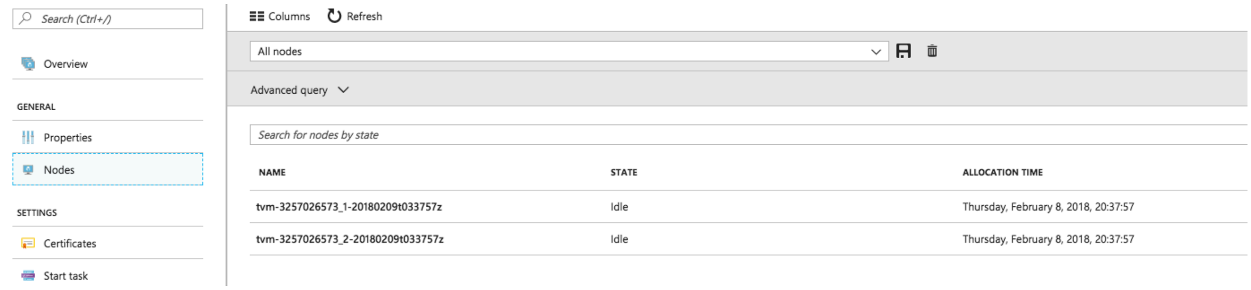
Low-priority nodes
0

Operating System
Canonical UbuntuServer 16.04.0-LTS (latest)

VM size
standard_a1

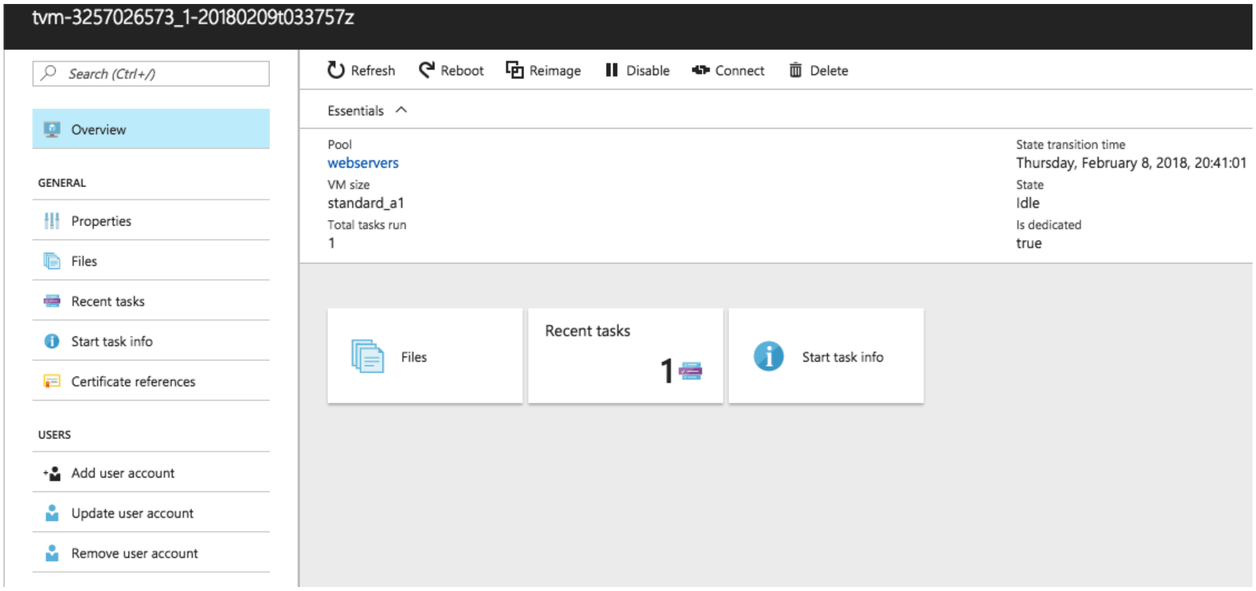
Allocation state
Steady

Another view of both nodes



NAME	STATE	ALLOCATION TIME
tvm-3257026573_1-20180209t033757z	Idle	Thursday, February 8, 2018, 20:37:57
tvm-3257026573_2-20180209t033757z	Idle	Thursday, February 8, 2018, 20:37:57

Detail of a single node showing the single (startup task) that was run

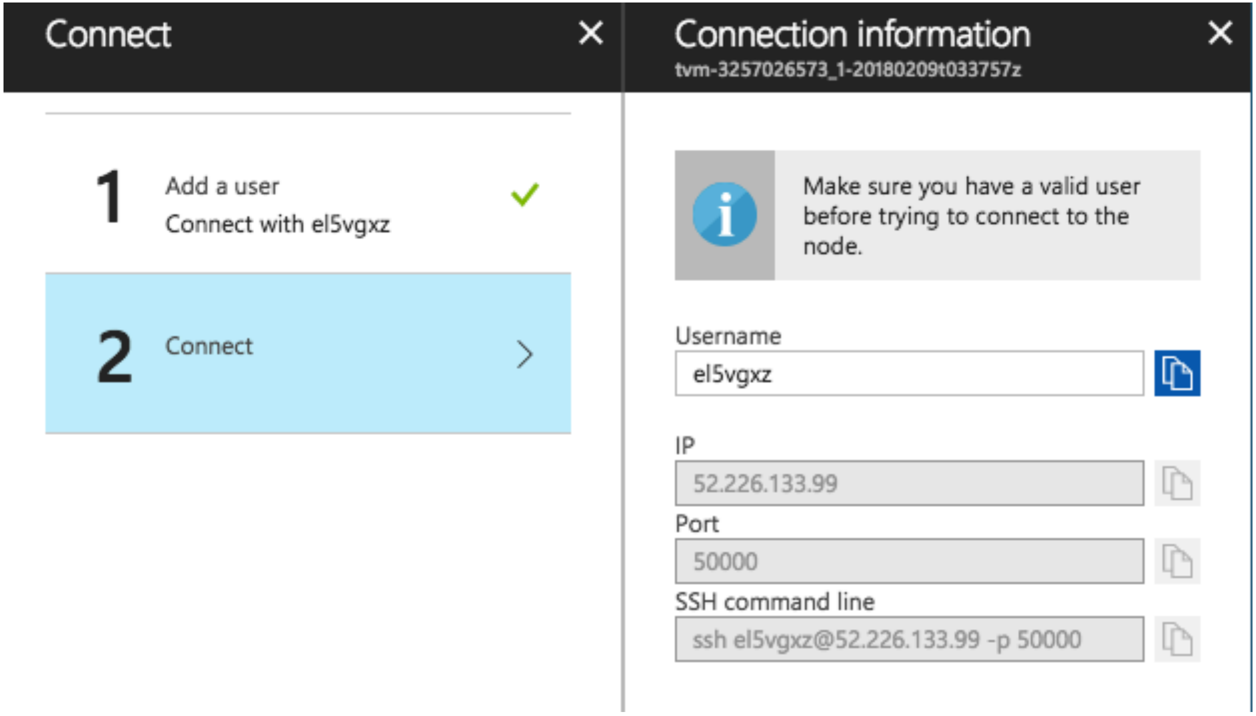


tvm-3257026573_1-20180209t033757z

Pool
webservers
VM size
standard_a1
Total tasks run
1

State transition time
Thursday, February 8, 2018, 20:41:01
State
Idle
Is dedicated
true

Adding a user so that I can SSH into one of the nodes. When adding user, change the expire time to a day ahead. It runs GMT and after a certain in the day, your account will be expired even after creating it.



Connect

1 Add a user
Connect with el5vgxz ✓

2 Connect

Connection information
tvm-3257026573_1-20180209t033757z

Make sure you have a valid user before trying to connect to the node.

Username
el5vgxz

IP
52.226.133.99

Port
50000

SSH command line
ssh el5vgxz@52.226.133.99 -p 50000

```
Kirks-MacBook:code el5vgxz$ ssh el5vgxz@52.226.133.99 -p 50000
Password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-47-generic x86_64)

 * Documentation:  https://help.ubuntu.com
```

```
  * Management:     https://landscape.canonical.com
  * Support:        https://ubuntu.com/advantage

   Get cloud support with Ubuntu Advantage Cloud Guest:
     http://www.ubuntu.com/business/services/cloud

 219 packages can be updated.
 123 updates are security updates.



 The programs included with the Ubuntu system are free software;
 the exact distribution terms for each program are described in the
 individual files in /usr/share/doc/*/copyright.

 Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
 applicable law.

 el5vgxz@faa1f864394442779b572ada4c66da1a000000:~$ sudo -i
 root@faa1f864394442779b572ada4c66da1a000000:~#
```

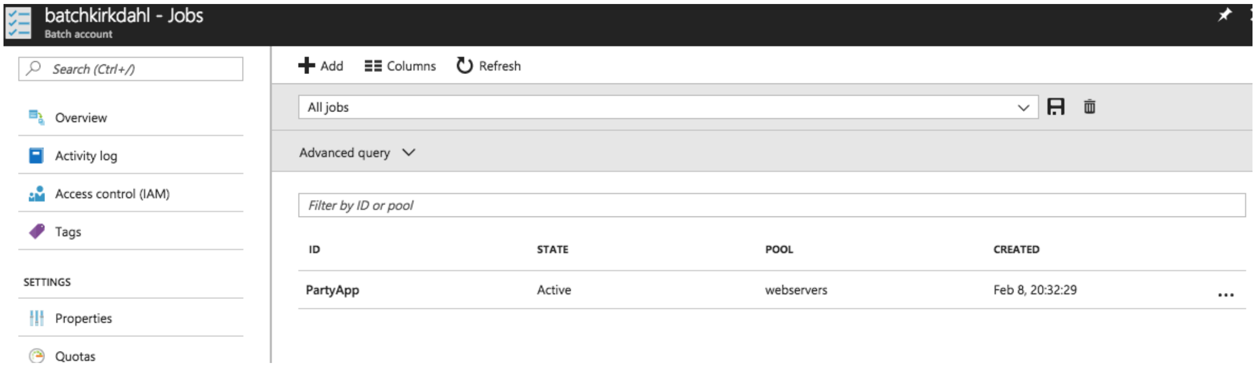
This shows unzip was installed

```
 root@faa1f864394442779b572ada4c66da1a000000:~# unzip
 UnZip 6.00 of 20 April 2009, by Debian. Original by Info-ZIP.
```

This shows Apache2 was installed

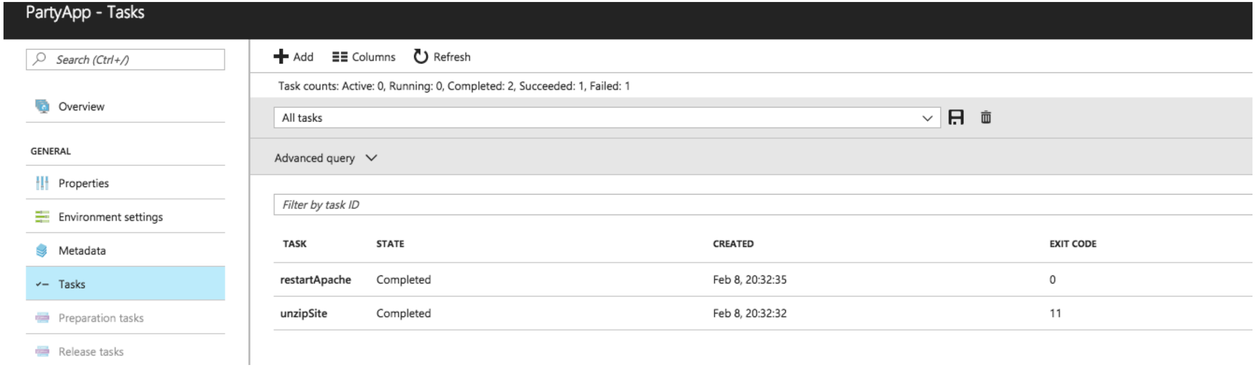
```
 root@faa1f864394442779b572ada4c66da1a000000:~# /etc/init.d/apache2 start
 [ ok ] Starting apache2 (via systemctl): apache2.service.
 root@faa1f864394442779b572ada4c66da1a000000:~#
```

This shows the “PartyApp” job



ID	STATE	POOL	CREATED
PartyApp	Active	webservers	Feb 8, 20:32:29

This shows TASKS in the PartyApp JOB



Task counts: Active: 0, Running: 0, Completed: 2, Succeeded: 1, Failed: 1

TASK	STATE	CREATED	EXIT CODE
restartApache	Completed	Feb 8, 20:32:35	0
unzipSite	Completed	Feb 8, 20:32:32	11

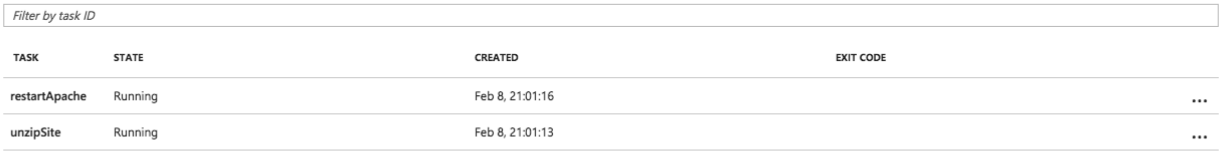
Now, the tasks show completed. But they never ran on the nodes, because it took too long to create the nodes while the script was creating everything. To get the tasks to run on the nodes now. We have two options.

1) Reboot the nodes, they will go through startup tasks and then job/tasks
2) Delete the tasks and recreate with command line from the script.

You can see on the node that the PartyApp website does not exist yet in the /var/www/html. This is done bu the unzipSite task

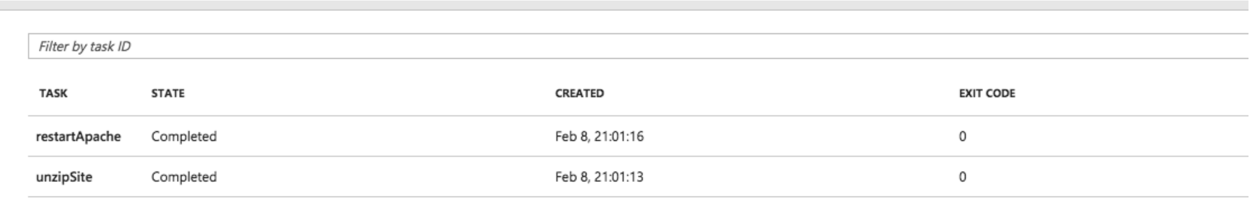
```
root@faa1f864394442779b572ada4c66da1a000000:/var/www/html# ll
total 20
drwxr-xr-x 2 root root  4096 Feb  9 03:40 ./
drwxr-xr-x 3 root root  4096 Feb  9 03:40 ../
-rw-r--r-- 1 root root 11321 Feb  9 03:40 index.html
```

I have deleted the task and recreated by just running the az batch task create from my script.



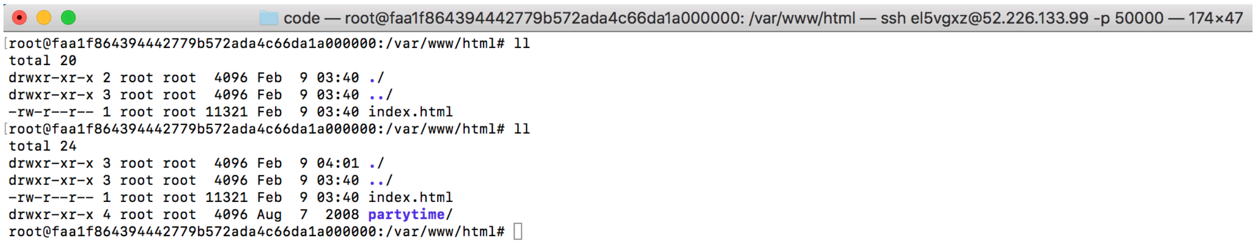
TASK	STATE	CREATED	EXIT CODE
restartApache	Running	Feb 8, 21:01:16	
unzipSite	Running	Feb 8, 21:01:13	

The tasks will immediately run and then completed with exit code of 0 (success). If the codes were different, then we could review the stdout and stderr files from the portal.



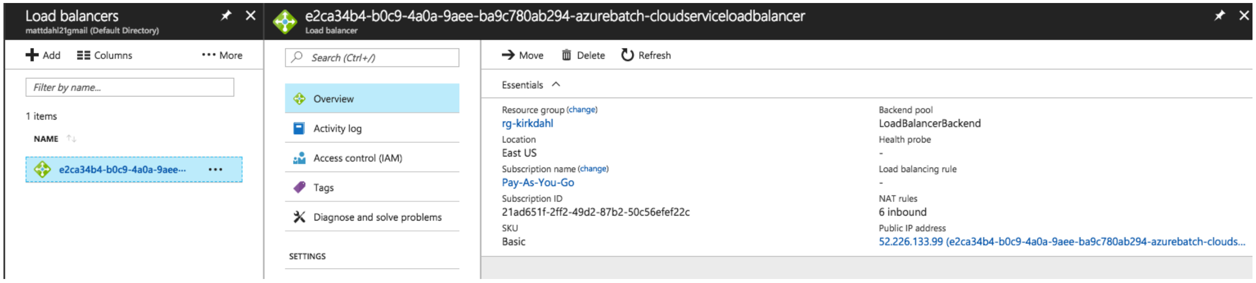
TASK	STATE	CREATED	EXIT CODE
restartApache	Completed	Feb 8, 21:01:16	0
unzipSite	Completed	Feb 8, 21:01:13	0

You can see that now I perform a listing and the website was unzipped into the apache folder


code — root@faa1f864394442779b572ada4c66da1a000000: /var/www/html — ssh el5vgxz@52.226.133.99 -p 50000 — 174×47

```
root@faa1f864394442779b572ada4c66da1a000000:/var/www/html# ll
total 20
drwxr-xr-x 2 root root  4096 Feb  9 03:40 ./
drwxr-xr-x 3 root root  4096 Feb  9 03:40 ../
-rw-r--r-- 1 root root 11321 Feb  9 03:40 index.html
root@faa1f864394442779b572ada4c66da1a000000:/var/www/html# ll
total 24
drwxr-xr-x 3 root root  4096 Feb  9 04:01 ./
drwxr-xr-x 3 root root  4096 Feb  9 03:40 ../
-rw-r--r-- 1 root root 11321 Feb  9 03:40 index.html
drwxr-xr-x 4 root root  4096 Aug  7  2008 partytime/
root@faa1f864394442779b572ada4c66da1a000000:/var/www/html# 
```

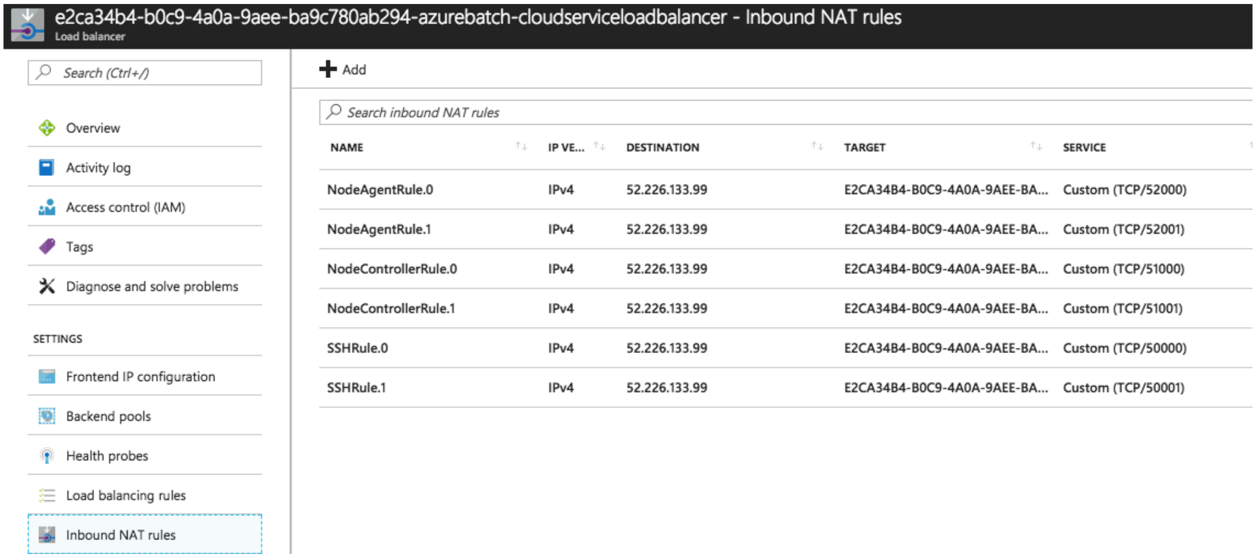
`The last part is to review the autocreated Load Balancer.`



Resource group (change)
rg-kirkdahl
Location
East US
Subscription name (change)
Pay-As-You-Go
Subscription ID
21ad651f-2ff2-49d2-87b2-50c56efef22c
SKU
Basic

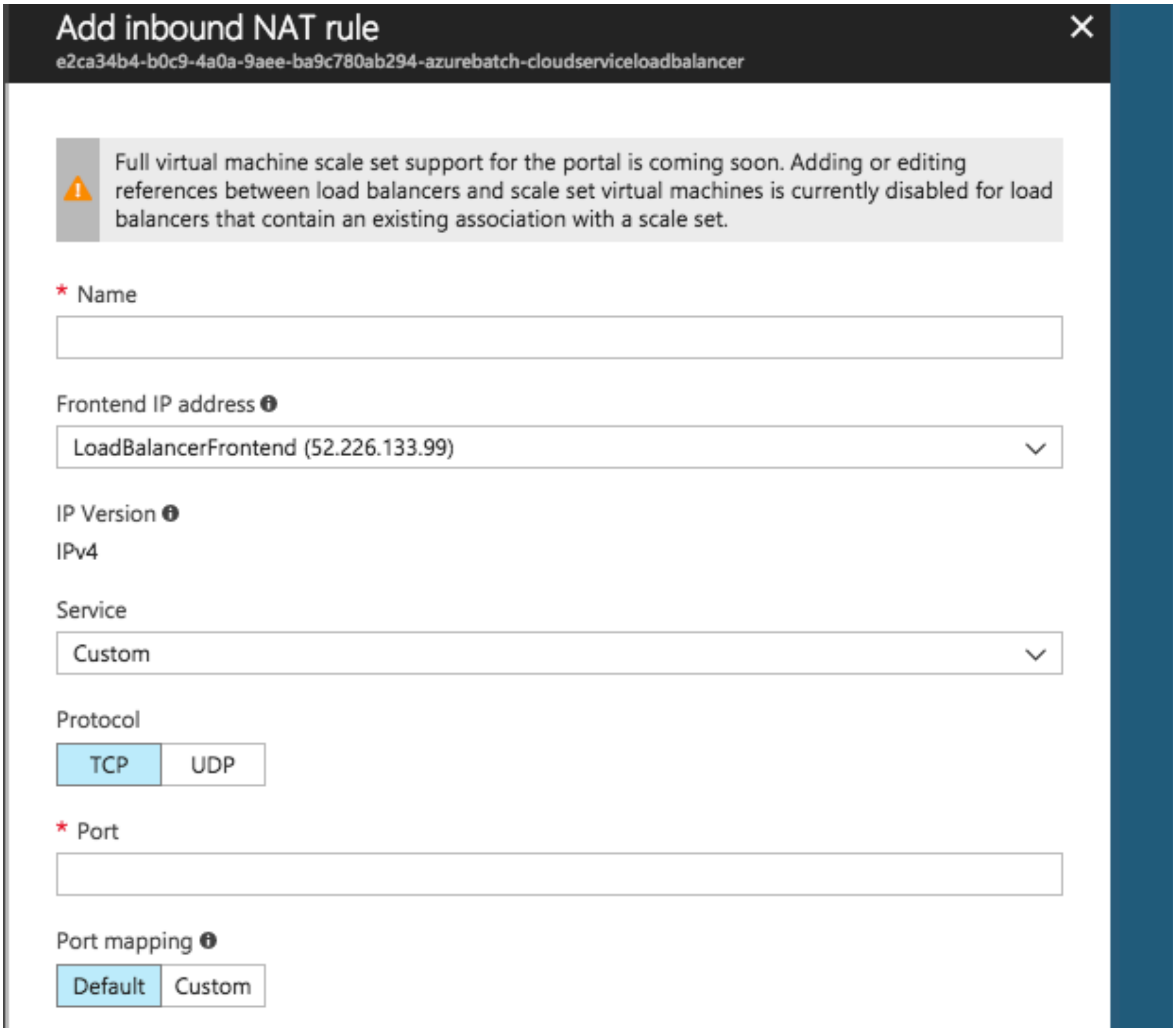
Backend pool
LoadBalancerBackend
Health probe
-
Load balancing rule
-
NAT rules
6 inbound
Public IP address
52.226.133.99 (e2ca34b4-b0c9-4a0a-9aee-ba9c780ab294-azurebatch-clouds...

We will see that there are 6 inbound nat rules. 3 for each node by default. However, my rule for port 80 was never created as I mentioned earlier when using the script due to what I believe is bug in the CLI.



NAME	IP VE...	DESTINATION	TARGET	SERVICE
NodeAgentRule.0	IPv4	52.226.133.99	E2CA34B4-B0C9-4A0A-9AEE-BA...	Custom (TCP/52000)
NodeAgentRule.1	IPv4	52.226.133.99	E2CA34B4-B0C9-4A0A-9AEE-BA...	Custom (TCP/52001)
NodeControllerRule.0	IPv4	52.226.133.99	E2CA34B4-B0C9-4A0A-9AEE-BA...	Custom (TCP/51000)
NodeControllerRule.1	IPv4	52.226.133.99	E2CA34B4-B0C9-4A0A-9AEE-BA...	Custom (TCP/51001)
SSHRule.0	IPv4	52.226.133.99	E2CA34B4-B0C9-4A0A-9AEE-BA...	Custom (TCP/50000)
SSHRule.1	IPv4	52.226.133.99	E2CA34B4-B0C9-4A0A-9AEE-BA...	Custom (TCP/50001)

And we can not manually create one or edit these as of yet per the warning. So we can not see the webservers from their public IP on the loadbalancer. **The work around is to create the Batch Pool in the GUI**



Add inbound NAT rule

e2ca34b4-b0c9-4a0a-9aee-ba9c780ab294-azurebatch-cloudserviceloadbalancer

Full virtual machine scale set support for the portal is coming soon. Adding or editing references between load balancers and scale set virtual machines is currently disabled for load balancers that contain an existing association with a scale set.

* Name

Frontend IP address: LoadBalancerFrontend (52.226.133.99)

IP Version: IPv4

Service: Custom

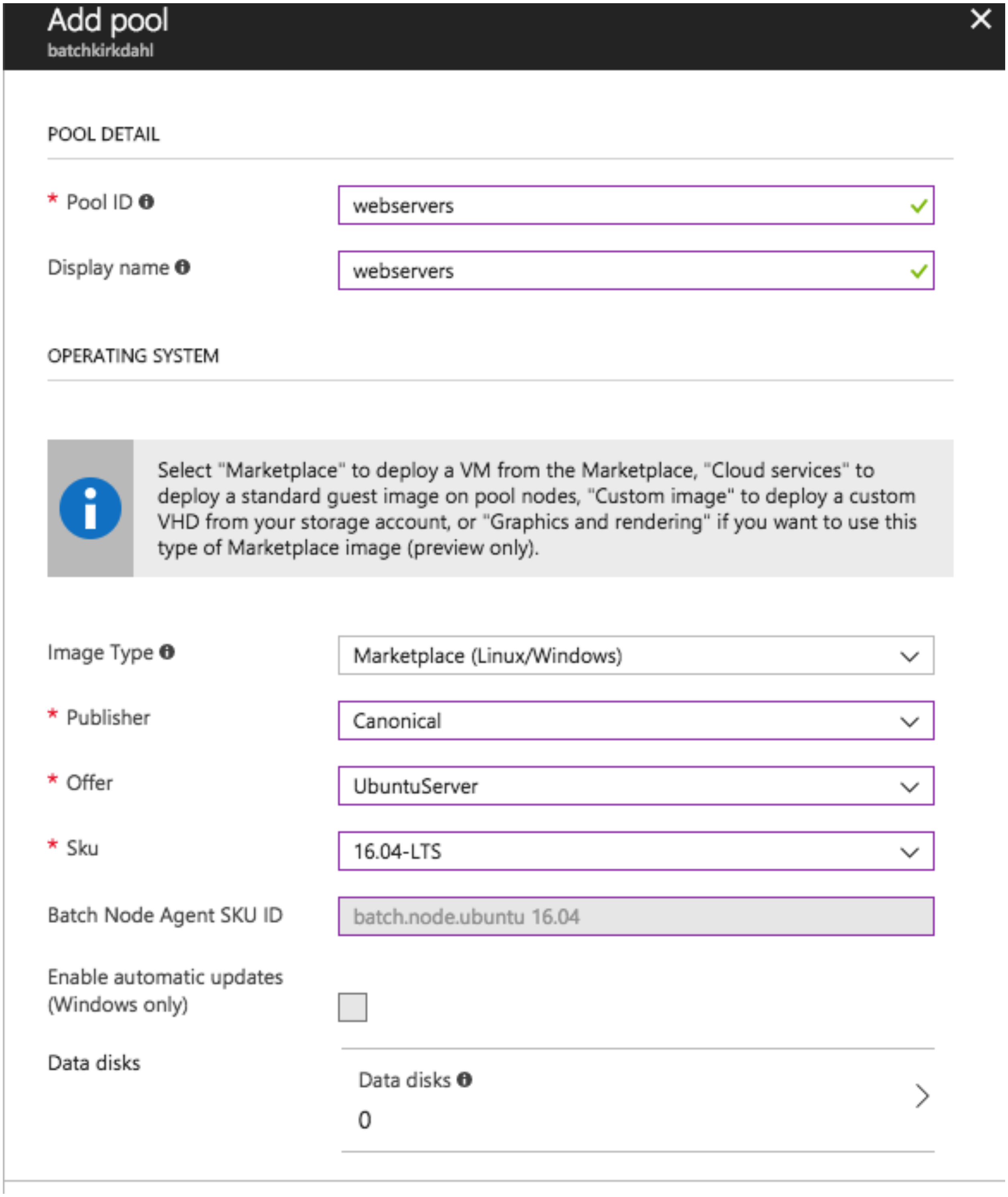
Protocol: [✓ TCP] UDP

* Port

Port mapping: [✓ Default] Custom

CREATING THE BATCH POOL IN THE GUI

We start with giving a name and picking vm os tuype



Add pool
batchkirkdahl

POOL DETAIL

* Pool ID: webservers ✓

Display name: webservers ✓

OPERATING SYSTEM

Select "Marketplace" to deploy a VM from the Marketplace, "Cloud services" to deploy a standard guest image on pool nodes, "Custom image" to deploy a custom VHD from your storage account, or "Graphics and rendering" if you want to use this type of Marketplace image (preview only).

Image Type: Marketplace (Linux/Windows)

* Publisher: Canonical

* Offer: UbuntuServer

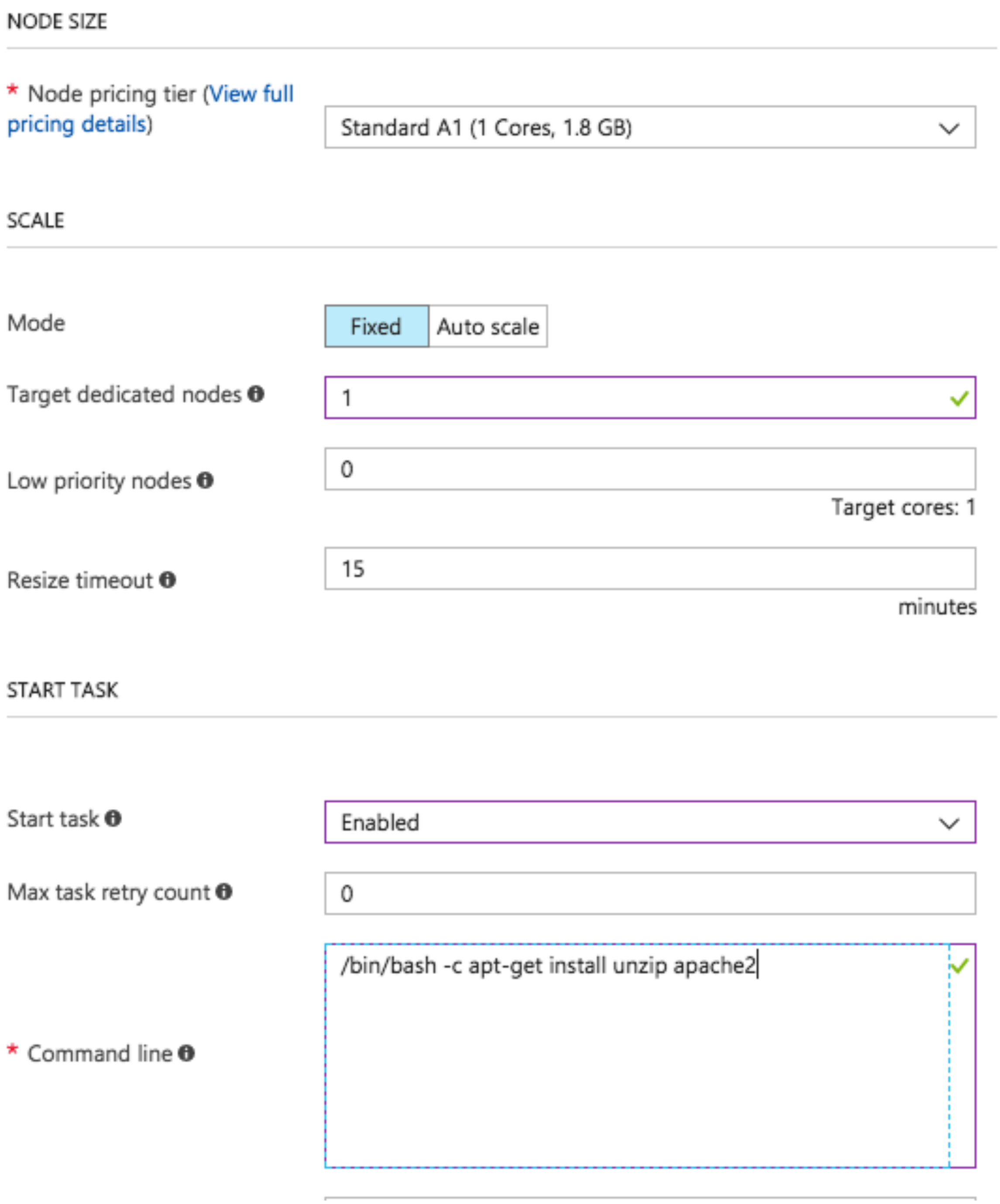
* Sku: 16.04-LTS

Batch Node Agent SKU ID: batch.node.ubuntu 16.04

Enable automatic updates (Windows only): □

Data disks: Data disks 0

We then select the VM size, the number of nodes and a startup command



NODE SIZE

* Node pricing tier (View full pricing details): Standard A1 (1 Cores, 1.8 GB)

SCALE

Mode: [✓ Fixed] Auto scale

Target dedicated nodes: 1

Low priority nodes: 0

Target cores: 1

Resize timeout: 15 minutes

START TASK

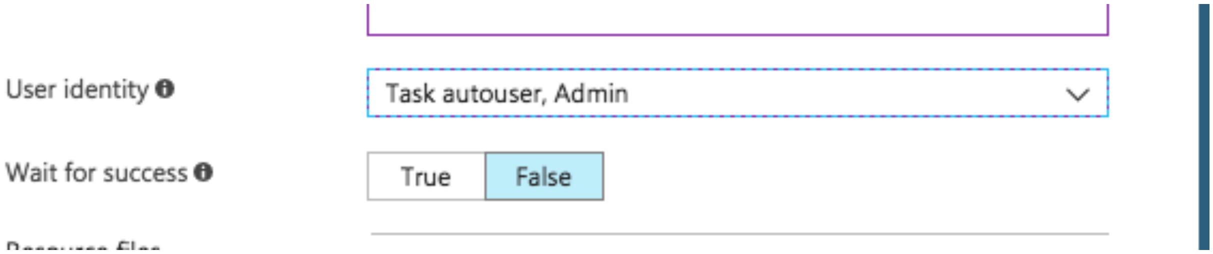
Start task: Enabled

Max task retry count: 0

* Command line:

```
/bin/bash -c apt-get install unzip apache2
```

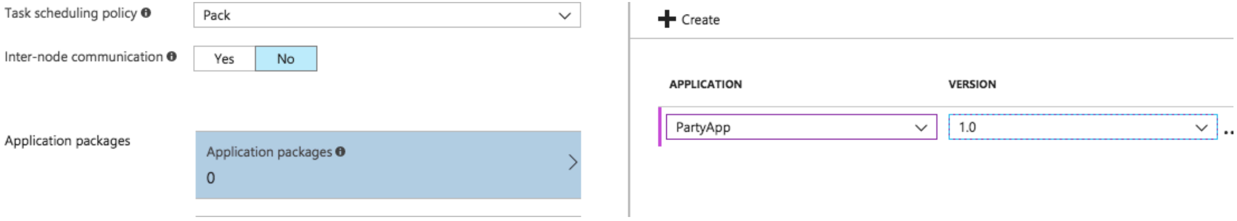
We then select to run task as admin



User identity: Task autouser, Admin

Wait for success: True [✓ False]

Then select the Application package we want deployed to the nodes.



Task scheduling policy: Pack

Inter-node communication: Yes [✓ No]

Application packages

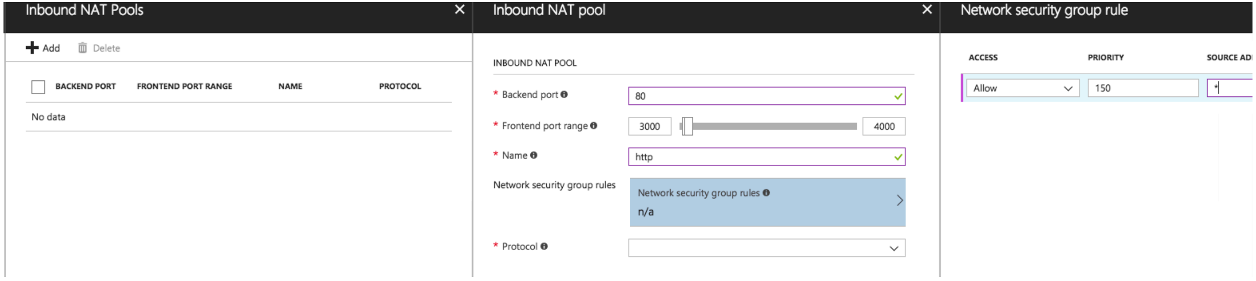
Application packages
0

APPLICATION	VERSION
PartyApp	1.0

We then add the inbound Nat rule

We create the backup port as 80 for default on apache, then provie name of “http”. We define a port range for the NAT. We cannot use 50000-52000 reserved by Azure. We add an NSG to allow * for source addresses.

Important to hit the Select on NSG, then provide the protocol of TCP. Then hit Ok.



Inbound NAT Pools

□ BACKEND PORT	FRONTEND PORT RANGE	NAME	PROTOCOL
No data			

Inbound NAT pool

INBOUND NAT POOL

* Backend port: 80

* Frontend port range: 3000 – 4000

* Name: http

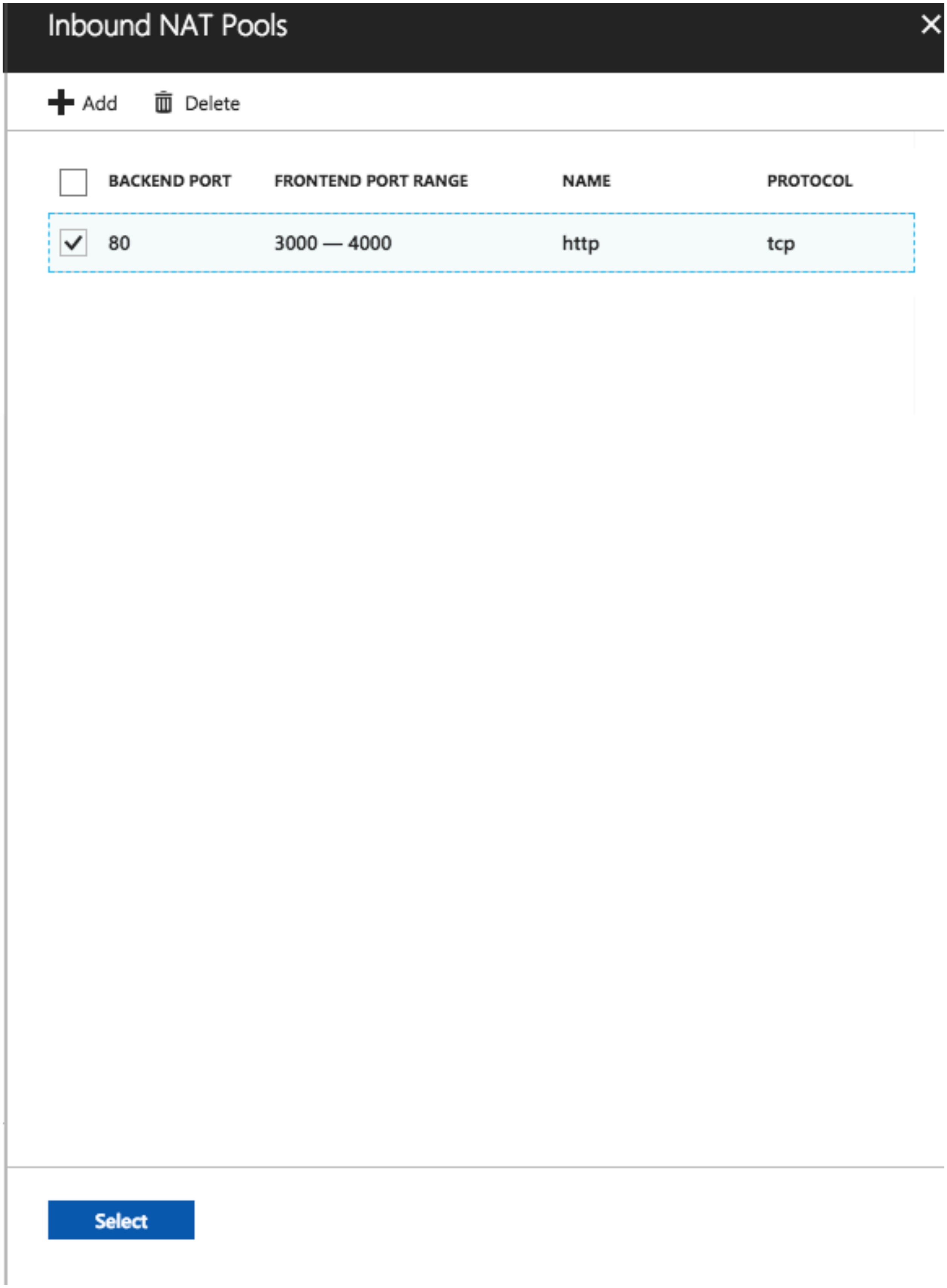
Network security group rules: Network security group rules n/a

* Protocol:

Network security group rule

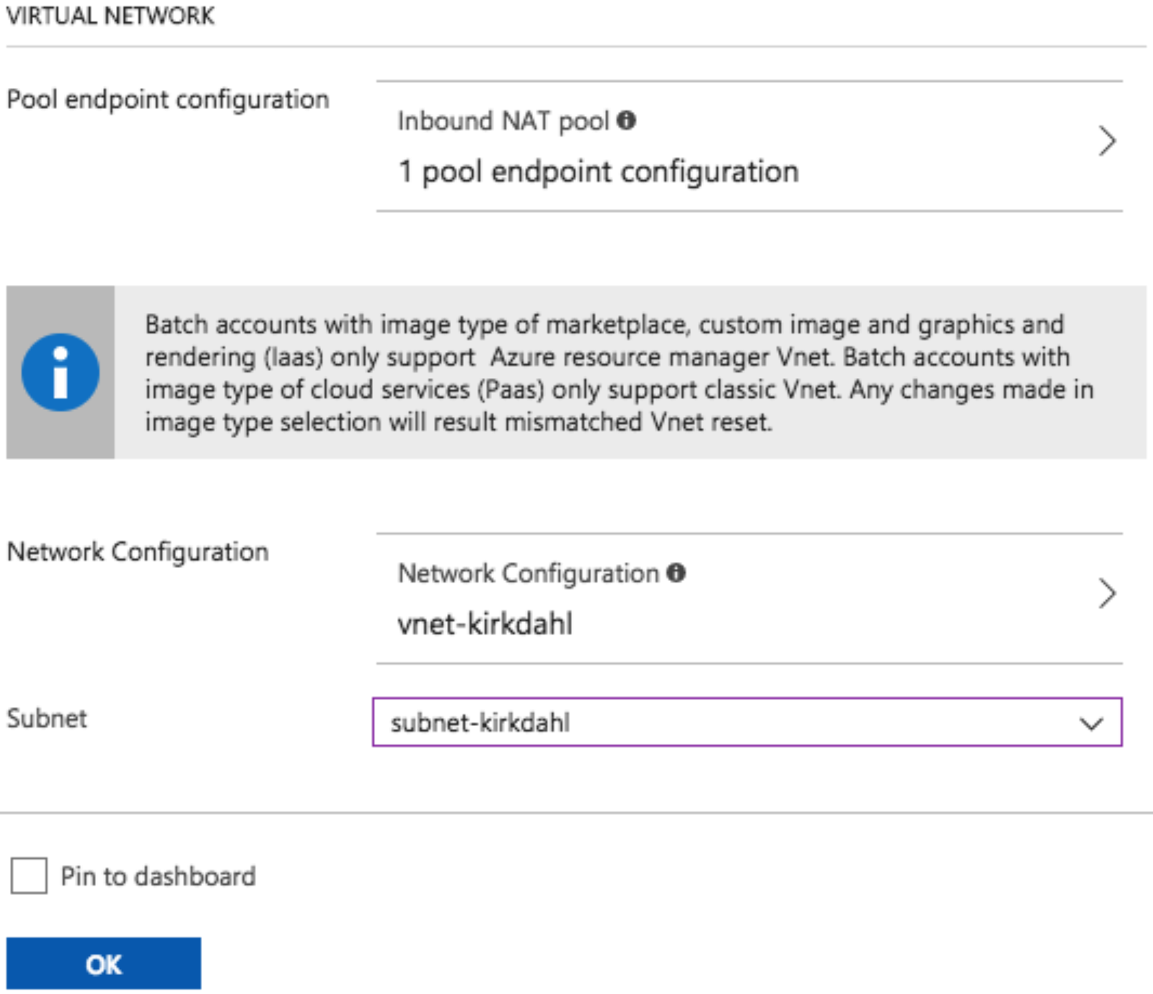
ACCESS	PRIORITY	SOURCE AD
Allow	150	*

Once Create, its <u>important</u> to check-box the new 80 inbound rule AND hit the Select button



□	BACKEND PORT	FRONTEND PORT RANGE	NAME	PROTOCOL
☑	80	3000 — 4000	http	tcp

This adds our inbound nat rule, after which we select the VNET and subnet which must be created in pre-requisite time. I have them included in my shell script. Then click OK



VIRTUAL NETWORK

Pool endpoint configuration

Inbound NAT pool
1 pool endpoint configuration

Batch accounts with image type of marketplace, custom image and graphics and rendering (laas) only support Azure resource manager Vnet. Batch accounts with image type of cloud services (Paas) only support classic Vnet. Any changes made in image type selection will result mismatched Vnet reset.

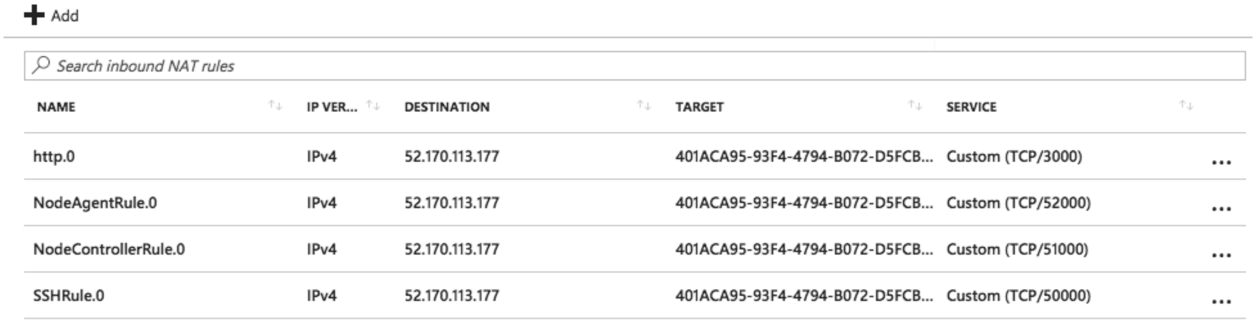
Network Configuration

Network Configuration
vnet-kirkdahl

Subnet: subnet-kirkdahl

□ Pin to dashboard

After creating the Pool manually with the GUI. We can see in the new load balancer the default 3 inbound nat rules and the additional one that we added for http

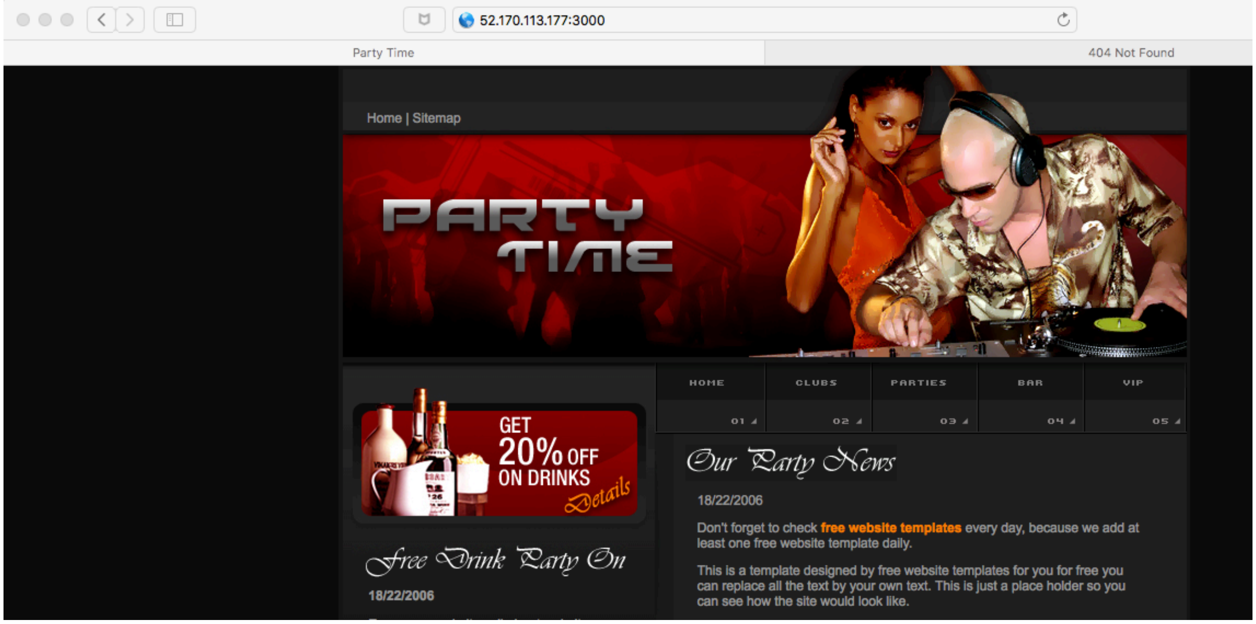


NAME	IP VER...	DESTINATION	TARGET	SERVICE
http.0	IPv4	52.170.113.177	401ACA95-93F4-4794-B072-D5FCB...	Custom (TCP/3000)
NodeAgentRule.0	IPv4	52.170.113.177	401ACA95-93F4-4794-B072-D5FCB...	Custom (TCP/52000)
NodeControllerRule.0	IPv4	52.170.113.177	401ACA95-93F4-4794-B072-D5FCB...	Custom (TCP/51000)
SSHRule.0	IPv4	52.170.113.177	401ACA95-93F4-4794-B072-D5FCB...	Custom (TCP/50000)

We delete the tasks and recreate them and they complete the installation of the website on the new node in the new pool we just created.

I did have to add the end update my code as I unzipped the partime app into its own directory. And the website could not find the links. I had to ssh onto the node and run `cp /var/www/html/partytime/* -R /var/www/html`

You can see that the public ip from the screenshot before with nat rules and port 3000 line up to display the website on the internet.



18/22/2006

18/22/2006

Don't forget to check **free website templates** every day, because we add at least one free website template daily.

This is a template designed by free website templates for you for free you can replace all the text by your own text. This is just a place holder so you can see how the site would look like.

CONCLUSSION:

In Conclusion, I found some of the JSON using the classes outside my background. The GUI is extremely intuitive to have a site up and running. Or any jobs for that matter that you wish to executes via an app with input files and deliever output and then spoin down the machines.