ITAS 161: HYPER-V VM MOBILITY WITH DOBBY

By: Ethan Holmes

Contents

Introduction

In this project, we are asked to create a failover cluster amongst our machines that is able to create a highly available Virtual Machine for us to use across our Dobby the house elf domain, the machine that we are replicating is our domain controller that will be up through the domain, which we will use disk2vhd to make the disk that we will use to create these highly available virtual machines. To accomplish all of this we created 3 new Hyper-V machines, 2 of which will be apart of a cluster with a file server. And a 3rd Hyper-V server outside the domain that we can move the machine to for additional availability.

Step 1: Creating the Domain and networking.

1. After creating our domain controller, 2 Hyper-V machines and a file server, we can join them all to our domain named “dobbythehouseelf.local” and give them IP Addresses that are within our given ranges of 192.168.8.x and the Hyper-V and File Server machines getting a second address of 172.16.8.x (where X represents the Host ID) to help create redundancy within the cluster.

2. Once all machines are domain joined and ready to go, install all appropriate roles on the machines, Hyper-V machines receiving the Hyper-V role as well as Failover Cluster Manager, our Domain Controller getting the Active Directory services and lastly the File Server receiving File Services as well as iSCSI services with those file services.

Step 2: Failover Cluster configuration and iSCSI setup

1. From within Server Manager on our File Server machine, on the left side we can access File and Storage services and click the “Disks” tab, here we can see the disks on the top of the page, we should have an uninitiated disk that is currently offline, this disk will be an extra disk that we added to the virtual machine while creating it. Bring the disk online and create a new volume with it (in our case we will use the E: letter drive).

2. Follow the on-screen instructions. Use the full size of the disk, select the NTFS volume format, give the volume a name and complete the setup of the volume.

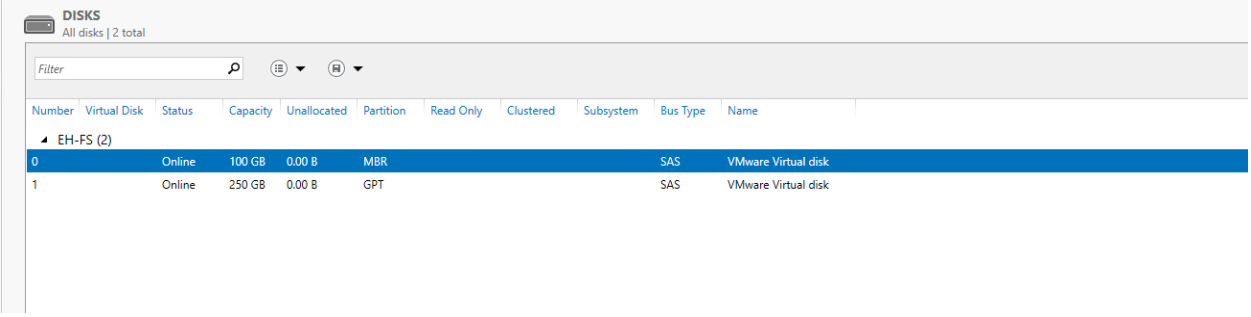

Figure 1: 2 Initialized disks to be used, Disk 1 to be used in the iSCSI

3. After the disk is created, move to the "iSCSI" tab on the left side of the screen, and we can now use the disk we just made to create an iSCSI virtual disk

4. Select the E: drive, give the iSCSI target an appropriate name to help identify the disk, then tell the iSCSI to use all the space on the disk and to dynamically expand.

5. Give the iSCSI target a name, click next and on "Access Servers" we can add both of our Hyper-V machines to be used by the iSCSI target by selecting "Enter a Value for the selected value" entering the IP address of the Hyper-V machine we want to add, that we just create

6. We can leave the types of Authentication blank as we are working within a testing environment, however in production please choose the appropriate type of authentication, CHAP or other. Click create to finish the creation of your iSCSI device.

Step 3: Adding iSCSI to Hyper-V servers

1. For this part, what we do will be mirrored on both our Hyper-V servers. Open the iSCSI initiator, click yes if it asks you to start the service. Click the "Target" tab and add the IP address of our file server hosting the iSCSI we just created, click quick connect and the server should add itself.

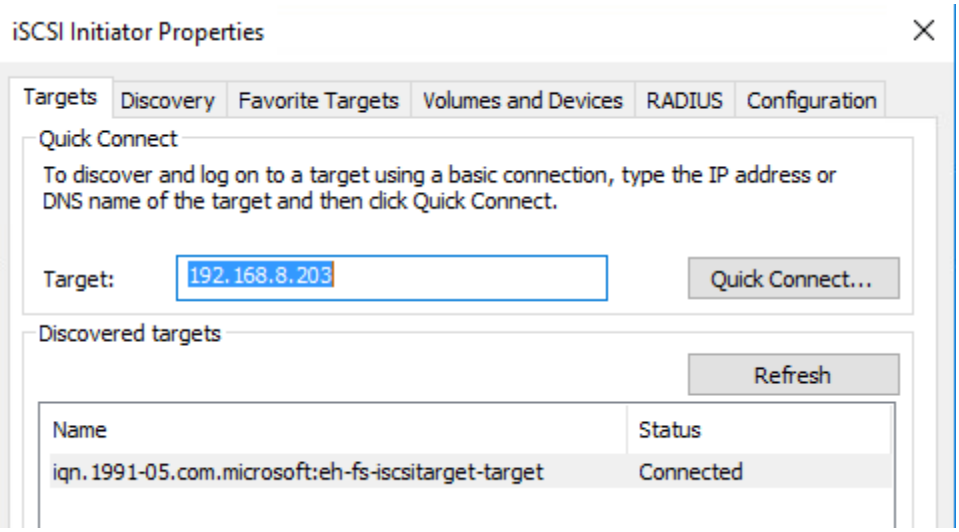

Figure 2: Using quick connect to connect to the iSCSI

2. Click the "Volumes and Devices" tab and click "Auto Configure" where we should then find a new volume which will be the volume that we created previously.

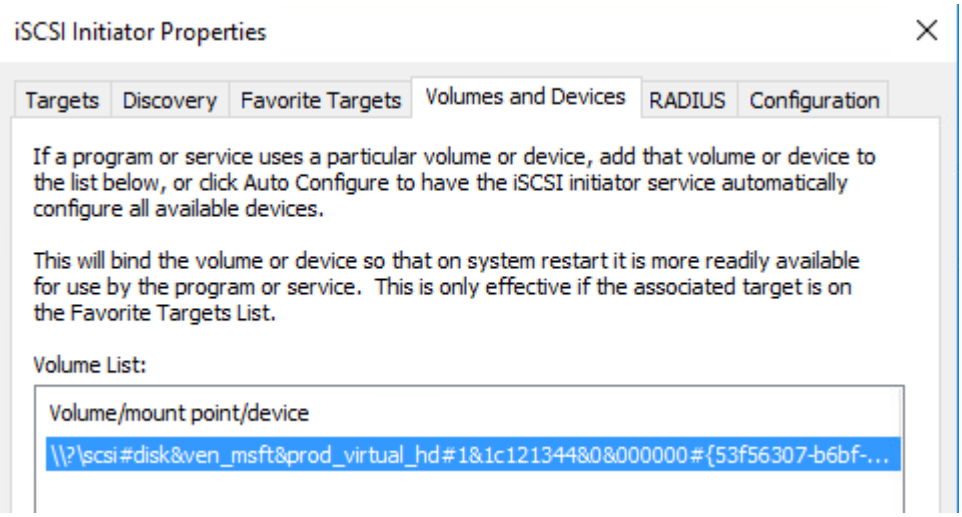

Figure 3: Successfully added iSCSI Volume

3. Back in server manager, under the disks tab we will see a disk that is offline, if not, refresh server manager. Once you see the disk, we can go through the same process with the disks that we did previously. Bring the disk online and give it the maximum space allowed and assign a drive letter to it. Choose the NTFS file system and give a name to the disk. We now have a usable disk across network storage.

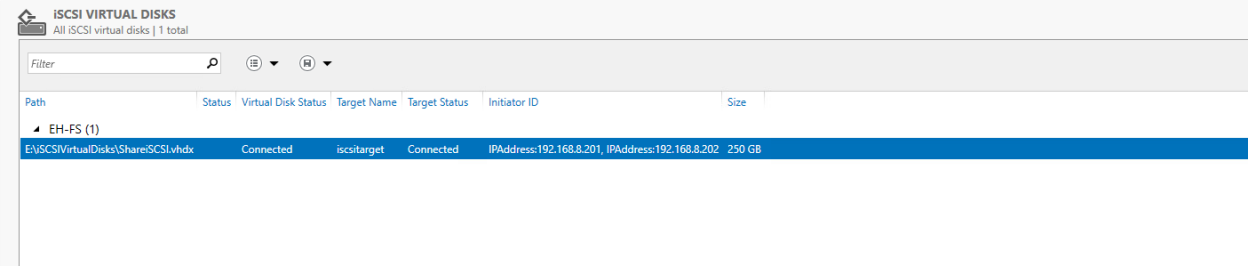

Figure 4: The Initialized disk

Step 4: Failover Cluster Creation

1. In Server Manager on both the Hyper-V machines, go to “Failover Cluster Manager” and open the manager. From the screen we are greeted with, click “Validate Cluster” and add the 2 Hyper-V machines by name into the field and complete the validation process.

2. Once the validation process has finished, we can then create our cluster. Click “Create a cluster” option and add the 2 Hyper-V servers by name, as well as the 192 and 172 IPs that are within the range we used earlier, on the next page, make sure that the “All available storage” option is *not* selected. Finish the creation of the cluster and make your way to the “Disks” tab on the left pane.

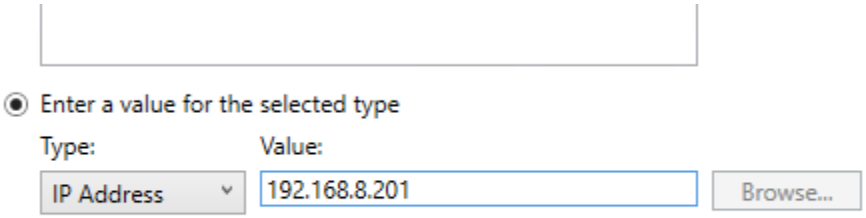

Figure 5: Adding the Hyper-V servers to the iSCSI cluster

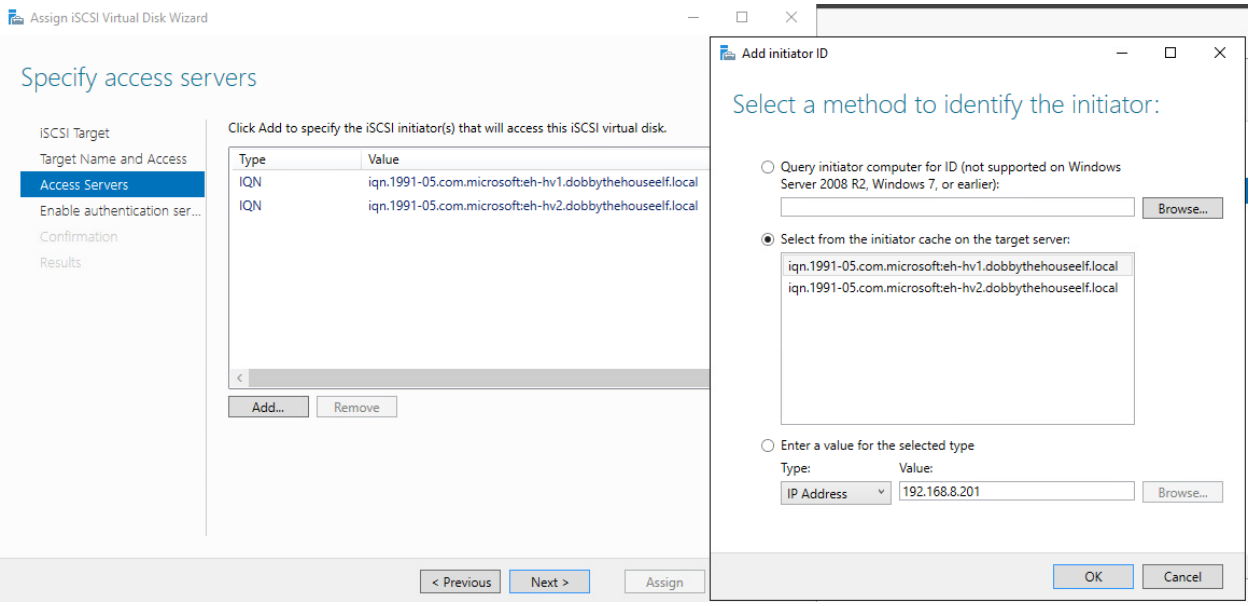

Figure 6: Both servers added and in use

3. In the “Disk” menu, we can then click “Add Disk” from the right pane, from the following screen we should see an available 200gb volume, we can right click this and select the “Add to cluster shared volumes option. To verify the addition of this shared volume, we can then move to our Hyper-V machine where in our C: drive, we should see a folder labelled as “Cluster Storage” which will be the folder along with our shared storage that we can use for high availability.

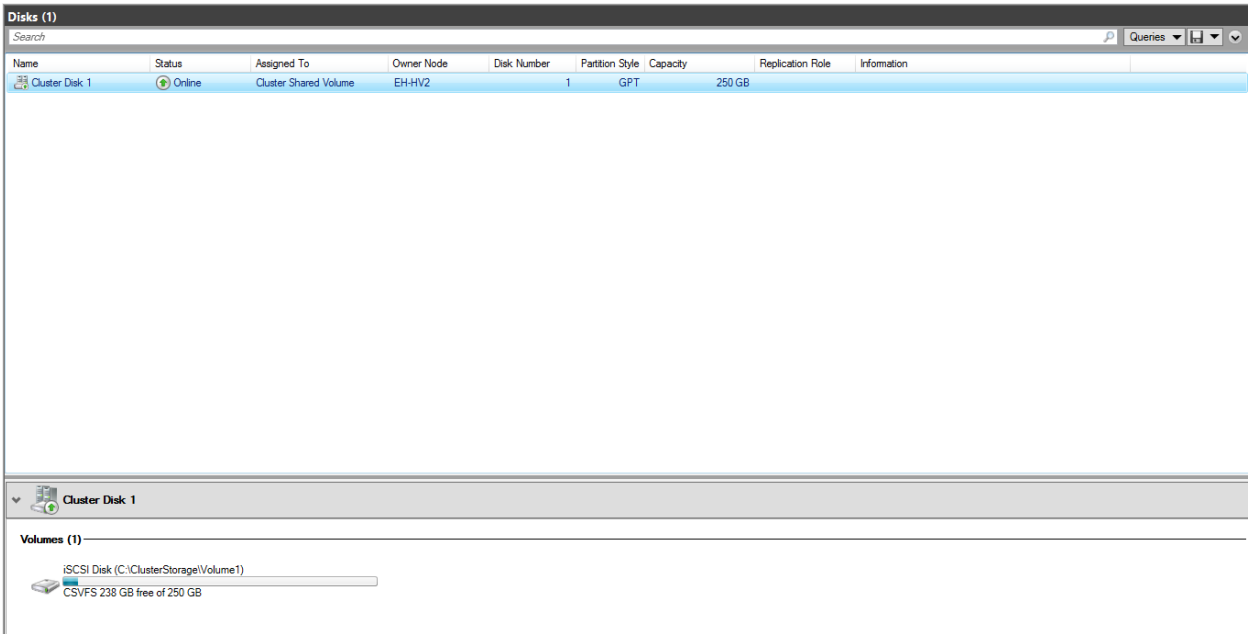

Figure 7: Fully configured disk with the shared path in the Volumes section

Part 5: Moving the Domain Controller Virtual Machine inside the Cluster

1. Using your preferred program, we will need to create a VHD out of our Dobby domain controller that we can use to create a highly available virtual machine.

2. Using Disk2VHD on our Domain Controller we will create a VHD of the disk, once completed I move the disk to the “Cluster Storage” folder on the C: drive, this is because the disks need to be in the shared folder for the virtual machines to be configured for high availability. Then switch to the Hyper-V machine where we can then click on “Roles” and in the right pane select the “New Virtual Machine” option

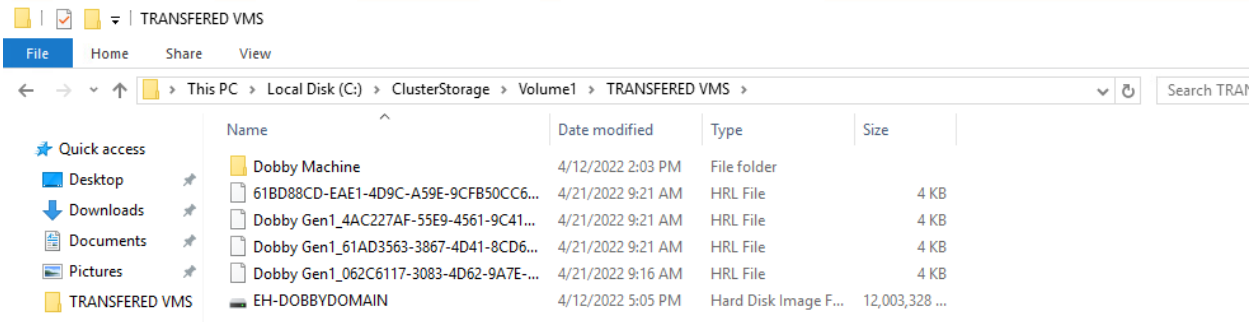

Figure 8: DobbyDomain disk in the Shared folder

3. We can now configure our virtual machine, this machine will get 2GB of RAM with 1 CPU, the important feature here is that we set the virtual machine to a *GEN 1 VM* otherwise the machine will fail to boot after creation due to differences between VMware and Hyper-V

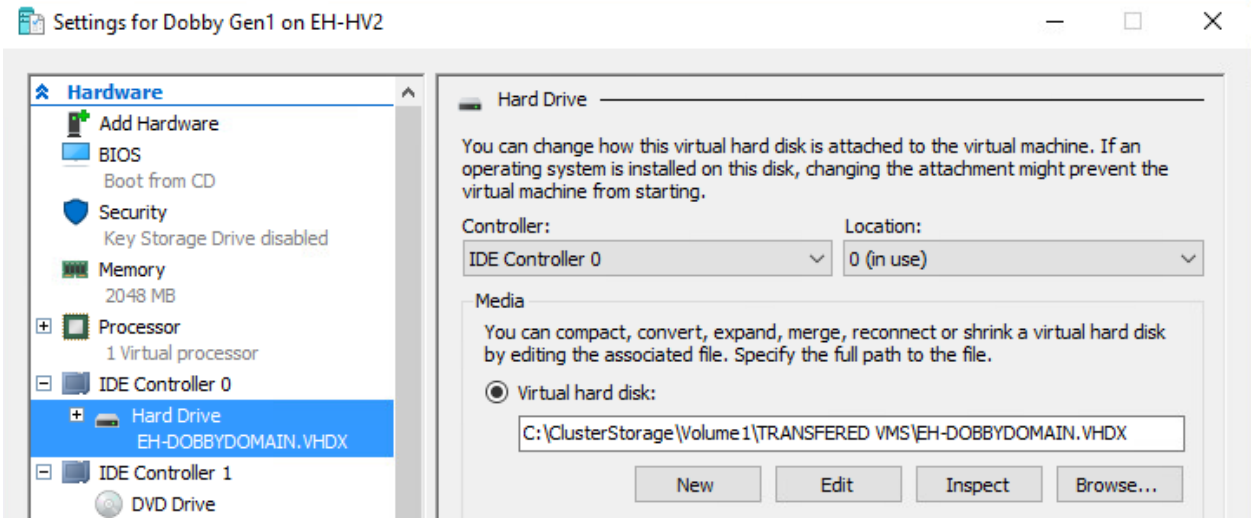

Figure 9: Location of Machine in a highly-available location

4. Boot the VM, login to make sure it works. We can then go to our second Hyper-V machine, where after a refresh, we can see the Virtual machine show up since it is in the same Failover Cluster

Figure 10: Working Dobby Domain Machine on the Hyper-V cluster

Part 6: Moving the Virtual Machine outside the Cluster

1. We now need to move the virtual machine to the Hyper-V server that is not located in our Failover cluster, but is domain joined.

2. We can start by adding the “Hyper-V Replica Broker” to either one of our failover cluster Hyper-V machines. This will allow us to configure replication settings for high availability on our machines. Give a name to the role and then an IP in our primary range (192.168.8.x) while disabling the secondary IP address. Click next and then finish.

3. We now need to configure replication on our Server Hyper-V machine, to do this we can go to our Server and then navigate to Hyper-V settings, from here we can then go down to Replication Configuration” and then click the checkbox to enable the computer as a replication server. Below it we will make sure that the “Use Kerberos (HTTP)” on Port 80 is selected and in the Authorization and Storage below it, make sure that “Allow replication from any authenticated server” is selected.

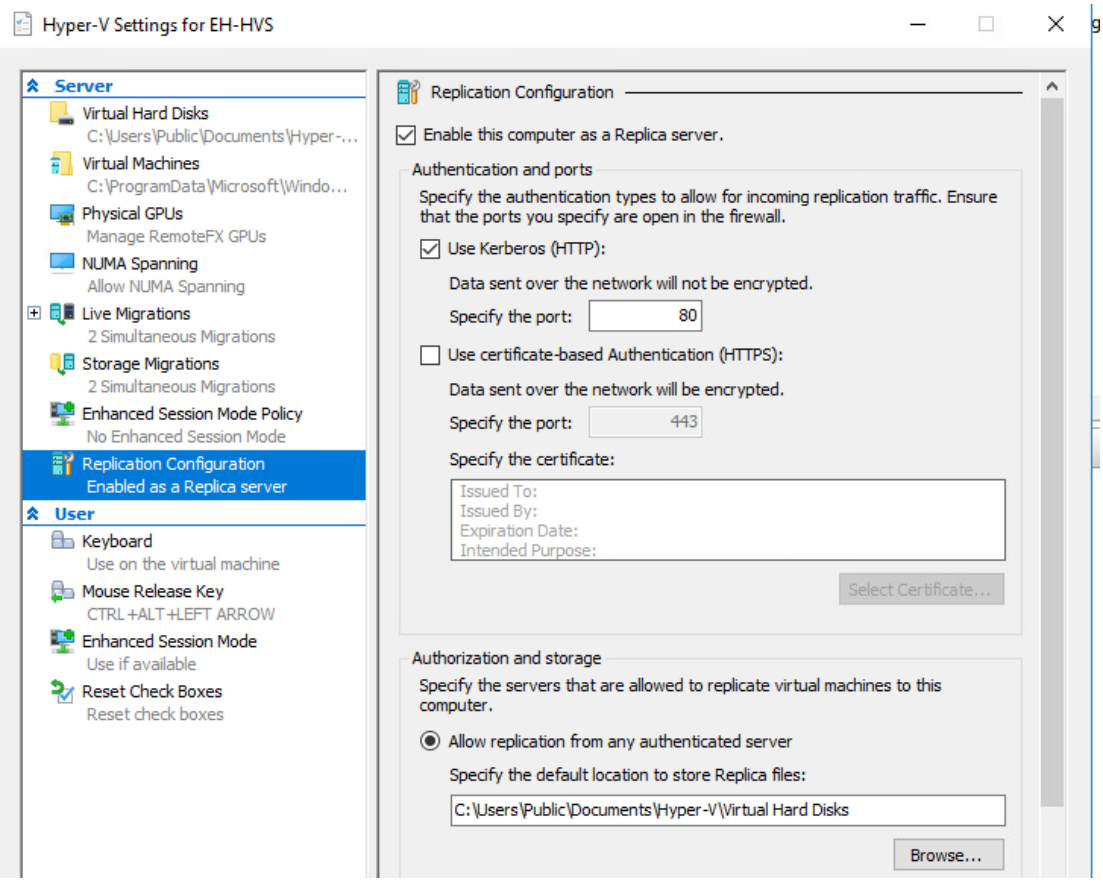

Figure 11: Enabling replication on the Hyper-V server

4. Now that configuration of the Server is done, we need to enable replication for the Virtual Machine itself. Go back to whatever Hyper-V machine we added the replica broker on, and right click the Dobby Virtual Machine that we configured, click the “Enable Replication” option.

5. The following window will ask you to specify a replica server, here we can type in the name of our Server Hyper-V that we want to add the virtual machine to.

6. On the following window, we can then leave the defaults, making sure that the port is equal to the port from the server that we configured in step 3 and click next

7. Here, we can choose our virtual machine file from the clustered storage and then click next. The following page allows us to configure how often replications happen, for now we can leave it at the default of every 5 minutes, however change it to as often as needed by your organization. After, it will then ask you which replications that you would like to save on the machine, for storage reasons we will go with the option to keep the last replication.

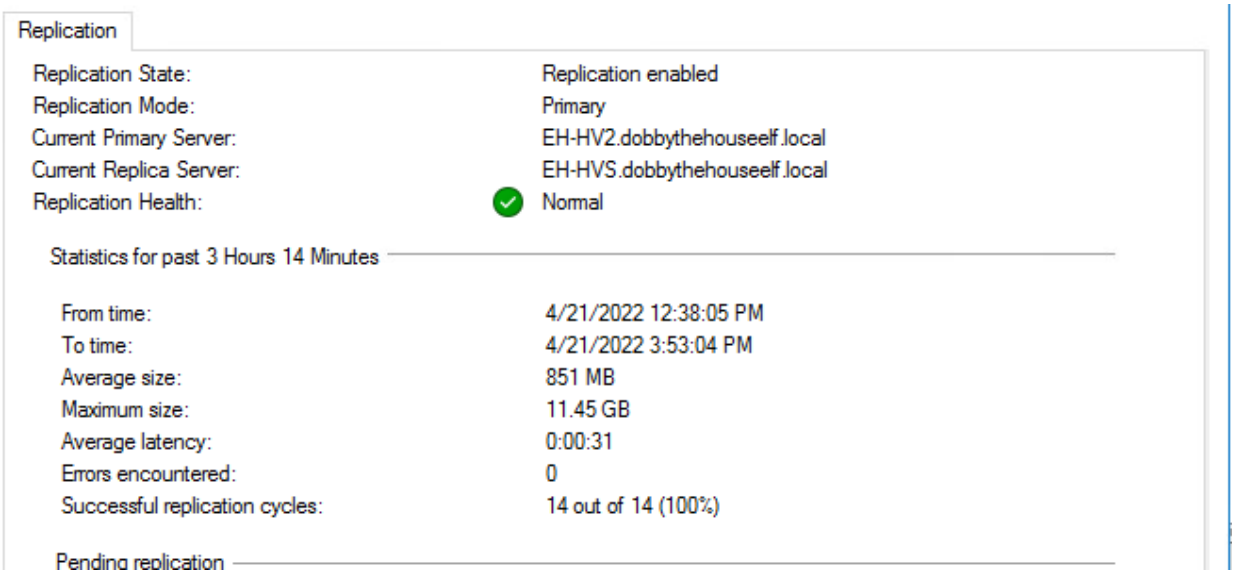

Figure 12: Size of Replications and where the server replicates to

8. The replication process will start, click ok and then finish in the summary tab, from here you can switch back to your Hyper-V machines and see the machines doing their replication process, and you can see on your Server Hyper-V the machine being replicated into your Hyper-V manager.

9. Finally, to test that the replicated virtual machine will work, we can then perform a failover from the Hyper-V machine, when we switch back to the Server Hyper-V, we can see that the machine has a checkpoint loaded and is online, we can click connect and see that the machine has been successfully gone through its failover and made it to the new machine.

Summary

I found this project to have plenty of similarities to FreeNAS, though that may be because of the setting up of failover clusters and working with in the manager. Beyond that, learning about machine replication and how to preform replication outside of a server was a fun experience.

Link to Youtube Video

https://www.youtube.com/watch?v=BbolSD2OtE0

Sources

https://docs.microsoft.com/en-us/windows-server/virtualization/hyper-v/manage/set-up-hyper-v-replica

https://www.veeam.com/blog/what-is-hyper-v-failover-cluster-windows-server.html