**Windows – TRANSPORT**  
Test FTPS Server  
Microsoft Windows Server 2016 Standard  
VMware Virtual Platform  
Intel(R) Xeon(R) CPU E7-4830 v2 @ 2.20GHz  
6GB RAM  
IP 142.34.2.4  
Subnet: 142.34.2.0/24 VLAN 108

**LINUX – SPAGHETTINI**  
Delivery SMT Tomcat Java Application Server  
Red Hat Enterprise Linux 7.8 (Maipo)  
VMware Virtual Platform  
Intel(R) Xeon(R) CPU E7-4830 v2 @ 2.20GHz  
27GB RAM  
IP 142.34.84.64  
Subnet: 142.34.84.0/24 VLAN 319

**Test Plan**

Create a folder for test data and scripts on bitbucket. <https://apps.nrs.gov.bc.ca/int/stash/projects/OPTIMIZE/repos/optimize-azure-api/browse>

Test functionality on the windows (SAN) file share vs the linux mounted share.  
Run script, have timestamp for each directory, log the time it takes to push/pull different file types (2 big, 1 small). Data types: Images, files, video/MP3/MP4, PDF, PPT, Docx, .pst. Could use Robocopy, or Rsync i.e. syncing log directories. Try different levels of verbosity. Use same script on each environment.

Admin challenge: Azure file share within a private endpoint = we can add, but not delete (we should discuss with Brian Price at weekly check-in).

**Testing Oct 1, 2020 - Linux**

* Log onto **REPO.IDIR.BCGOV** and create a folder called AZURE\_TEST\_DATA on E:\!Source. Create 2 subfolders for text and binary data.
* Upload the executable found here:  
  <https://bcgov-my.sharepoint.com/:u:/g/personal/brian_price_gov_bc_ca/EasbJ-K90IdHp7afZZmQrw0Bs-f8DQJ5GRULHpRBaVXRDA?e=V7N3ip>
* Run the executable using this command:  
  *create-files <Output Folder> <# of files> <minimum size (MB)> <maximum size (MB)> <T or B>*  
  This will generate the requested number of files, at a random size, in either Text or Binary format.  It does accept decimal places for file sizes (so .1 is 100kb).  The content of these files is random. For this test, we created 50 files each of binary & text that were between 1MB and 1000MB.
* Once the random test data has been generated, move the AZURE\_TEST\_DATA folder into E:\ProgramData\Dell GeoDrive Data\APP\_LOGS\_DLVR. This is an Object Storage share that is mounted on **SPAGHETTINI.BCGOV** as /shares/applogs. (The reason it was done this way is because the executable does not like the spaces in *Dell GeoDrive Data* and would not work properly)
* Log onto SPAGHETTINI and move the subfolders from /shares/applogs/AZURE\_TEST\_DATA to /fs/u02/azure\_files.
* Check that the script DataLoadTest.sh is set to do a dry run (-r option) on the Azure folders
* Run the following command from /home/<IDIR\_A>/scripts/DataLoadTest.sh :  
  *./DataLoadTest.sh /fs/u02/azure\_santest/ /shares/AzurePOC/TransferTest/ /fs/u02/azure\_files /home/<IDIR\_A>/logs/dataload\_test\_log.log*
* **RESULT**: severe latency during copy. Subsequent tests on Oct 9th (not using dry run) showed after the first hour only 4.6 GB had been moved.

**Testing Oct 13, 2020 - Windows**

* Log onto **TRANSPORT.IDIR.BCGOV** and create a folder called TestData on E:\!Source.
* Copy the folder ObjectStoreTestUtility from E:\!Source on REPO to E:\!Source on TRANSPORT
* Run the executable in ObjectStoreTestUtility using this command:  
  *create-files <Output Folder> <# of files> <minimum size (MB)> <maximum size (MB)> <T or B>*  
  The content of these files is random. For this test, we created 50 text files that were between 1MB and 1000MB for a total size of 24.5GB.
* Map a drive letter on TRANSPORT to [\\bcgovclnpd1danrmazfgrs01.file.core.windows.net\bcgovclnpd1danrmazfgrstestshare](file:///\\bcgovclnpd1danrmazfgrs01.file.core.windows.net\bcgovclnpd1danrmazfgrstestshare) and select “use different credentials”. Username is azure\bcgovclnpd1danrmazfgrs01 and the password is the secret key for the storage account.
* Copy the randomly-generated text files on E:\!Source\TestData and paste them into the newly-mapped azure drive
* **RESULT:** Severe latency during copy. 6.5GB copied in the first 90 minutes, overall it took 8 hours to complete.  
    
  Graphical user interface, application

  Description automatically generated