

Comparison of Storage1 with Storage2

The primary difference between accessing Storage1 and Storage2 is their name. For example, a Storage1 allocation may be accessed at via the SMB protocol by mounting `//storage1.ris.wustl.edu/example1`. A similar Storage2 allocation could by accessed by the same protocol by mounting `//storage2.ris.wustl.edu/example2`. From an RIS Compute1 node, the same allocations might be accessed using the Linux VFS interface at `/storage1/fs1/example1` and `/storage2/fs1/example2`.

The second most important difference between the two platforms is the mechanisms for “Discretionary Access Control” (DAC). While Storage1 leverages NFSv4-style **GPFS** ACLs, Storage2 uses NFSv4-compatible ACLs. While they are effectively equivalent in their functionality, they have varying degrees of idiosyncrasy in their usage.

	Storage1	Storage2
Interfaces	SMB, VFS/“POSIX”, NFS, Globus	SMB, VFS/“POSIX”, NFS, Globus
Platform	IBM’s Scale Storage, “GPFS”	Qumulo Core
DAC Mechanisms	NFS4-style GPFS ACLs, “POSIX”	NFS4-compatible ACLs, “POSIX”
User Identifier	Unix UID	Windows SID

Be sure to distinguish between **Storage2** and Storage1, **fs2**, which are distinct platforms.

Viewing an ACL on Storage2

NFS Protocol

Use the `nfs4_getfacl` command:

SMB Protocol

Using the Windows Explorer GUI:

- Select a file with the cursor and open its context menu (this is usually done with a “right-click” of the pointing device over the respective icon or list item)
- Select “Properties...”. This should open a new window with an interface to file properties.
- Click on the “Sharing and Permissions” tab in the file property window.
- Inspect the displayed ACL.

Using `smbclient`:

```
showacls smb: \> ls /storage2-dev/fs1/ohids FILENAME:ohids MODE:D SIZE:0 MTIME:Tue Jul 2 15:32:46 2024 revision: 1 type: 0x8404:
SEC_DESC_DACL_PRESENT SEC_DESC_DACL_AUTO_INHERITED SEC_DESC_SELF_RELATIVE DACL ACL Num ACEs: 46 revision: 2 --- ACE type:
ACCESS ALLOWED (0) flags: 0x00 Specific bits: 0x1ff Permissions: 0x1201ff: SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID:
S-1-5-21-3742419866-1697586437-2279049763-500 ACE type: ACCESS ALLOWED (0) flags: 0x00 Specific bits: 0x1ff Permissions: 0x1201ff:
SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3742419866-1697586437-2279049763-513 ACE type: ACCESS ALLOWED (0) flags:
0x12 SEC_ACE_FLAG_CONTAINER_INHERIT SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0x1ff Permissions: 0x1201ff: SYNCHRONIZE_ACCESS
READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1625061 ACE type: ACCESS ALLOWED (0) flags: 0x19
SEC_ACE_FLAG_OBJECT_INHERIT SEC_ACE_FLAG_INHERIT_ONLY SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0x1df Permissions: 0x1201df:
SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1625061 ACE type: ACCESS ALLOWED (0)
flags: 0x12 SEC_ACE_FLAG_CONTAINER_INHERIT SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0xa9 Permissions: 0x1200a9:
SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1625062 ACE type: ACCESS ALLOWED (0)
flags: 0x19 SEC_ACE_FLAG_OBJECT_INHERIT SEC_ACE_FLAG_INHERIT_ONLY SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0x89 Permissions:
0x120089: SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1625062 ACE type: ACCESS
ALLOWED (0) flags: 0x12 SEC_ACE_FLAG_CONTAINER_INHERIT SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0x1ff Permissions: 0x1201ff:
SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1982696 ACE type: ACCESS ALLOWED (0)
flags: 0x19 SEC_ACE_FLAG_OBJECT_INHERIT SEC_ACE_FLAG_INHERIT_ONLY SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0x1df Permissions:
0x1201df: SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1982696 ACE type: ACCESS
ALLOWED (0) flags: 0x12 SEC_ACE_FLAG_CONTAINER_INHERIT SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0xa9 Permissions: 0x1200a9:
SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1982697 ACE type: ACCESS ALLOWED (0)
flags: 0x19 SEC_ACE_FLAG_OBJECT_INHERIT SEC_ACE_FLAG_INHERIT_ONLY SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0x89 Permissions:
0x120089: SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1982697 ACE type: ACCESS
ALLOWED (0) flags: 0x12 SEC_ACE_FLAG_CONTAINER_INHERIT SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0x1ff Permissions: 0x1201ff:
SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1980602 ACE type: ACCESS ALLOWED (0)
flags: 0x19 SEC_ACE_FLAG_OBJECT_INHERIT SEC_ACE_FLAG_INHERIT_ONLY SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0x1df Permissions:
0x1201df: SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1980602 ACE type: ACCESS
ALLOWED (0) flags: 0x12 SEC_ACE_FLAG_CONTAINER_INHERIT SEC_ACE_FLAG_INHERITED_ACE Specific bits: 0xa9 Permissions: 0x1200a9:
SYNCHRONIZE_ACCESS READ_CONTROL_ACCESS SID: S-1-5-21-3579272529-3368358661-2280984729-1980603 ACE type: ACCESS ALLOWED (0)
```

[illegible]

Using `smbcals`:

Viewing Share-level Quota Usage on Storage2

SMB Protocol

Using the Windows Explorer GUI:

Select the shared drive for the allocation and open its context menu (this is usually done with a “right-click” of the pointing device over the respective icon or list item)

Select “Properties...”. This should open a new window with an interface the share properties.

Inspect the used and available space.

Use `smbclient` to get raw values, optionally calculating the desired measurements:

```
ls .snapshot DHSRn 0 Tue Jul 2 15:32:46 2024 . D 0 Tue Jul 2 15:32:46 2024 .. D 0 Thu Sep 5 11:07:44 2024 10737418240 blocks of size 512. 10737418232
blocks available smb: \> ^d $ bc -q # KB used (10737418240-10737418232)*512/1024 4 # TB available scale=3 (10737418240-8)*512/1024^4 4.999]]>
```

VFS/”POSIX” Protocol

Use the `df` command with the path to a locally mounted share:

Transferring Data Between Storage1 and Storage2 with Globus

Globus is an application that serves to move data into the storage service.

Storage2 allocations simply appear as another path on the existing [RIS Globus Collection](#). Simply select a Storage1 source and a Storage2 destination to move data from Storage1 to Storage2 using Globus.

For more information, see

<https://washu.atlassian.net/wiki/spaces/RUD/pages/1795588152/Moving+Data+With+Globus?atlOrigin=eyJpIjoiZWYyODM4YWRIYTExNDMzZTg5ODE2MTE1NDEzNzBk>

or

<https://washu.atlassian.net/wiki/spaces/RUD/pages/1796145237/Moving+Data+With+Globus+CLI?atlOrigin=eyJpIjoiODRiNzQ4OWQ5NjdhNDRIMWEzODRiYWFiY2M2Zj>

for more information on using Globus to transfer data.