Working with Volumes on a Windows Server 2012, 2012 R2, 2016 or 2019 Host

The first three workflow tasks should be completed before beginning the other items that this section is focused on configuring. There are two methods that will be shown, (1) Using Graphical User Interfaces (GUI) for the FlashArray and Windows Server Disk Management and (2) Using Windows PowerShell and Pure Storage PowerShell SDK.

Steps to Create, Connect and Configure

1. Installing Multipath-IO
2. Configuring Multipath-IO
3. Setting the MPIO Policy
4. Create FlashArray Volume(s)
5. Connect FlashArray Volume(s) to Host (or Host Group)
6. Manage FlashArray Volume(s) from Windows Server Host
7. Create Mounts Point(s) (Optional)

This section assumes that there is a Pure Storage FlashArray setup and configured with hosts and all SAN/Network fabric connectivity in place.

FlashArray Management GUI

Create FlashArray Volume(s)

The FlashArray Management interface is very simple to use. Follow the below steps to create a volume. Repeat the same actions to create a volume named ReFS (optional).

1. Create a new volume named NTFS.
Viewing the created NTFS volume.

Connect FlashArray Volume(s) to Host (or Host Group)

1. Connect the NTFS volume to SERVER08.

Selecting the NTFS volume to connect to SERVER08.
Viewing the NTFS volume that is now connected to SERVER08.

Windows Server Disk Management

Manage FlashArray Volume(s) from Windows Server Host

1. Start Disk Management using **Start > Run > diskmgmt.msc**
2. Perform **Rescan Disks**.

3. Initialize disks by right-clicking on the Disk # and selecting **Initialize Disk**.

4. The Initialize Disk dialog will open and indicate all of the disks that need to be initialized. For this example there is Disk 1 (NTFS) and Disk 2 (ReFS) that will be initialized using GPT (Guid Partition Table). GPT is used to support disks larger that 2 TB.
5. Create a volume by right-clicking on the disk partition and selecting **New Simple Volume**.

6. The **New Simple Volume Wizard** will open, click **Next >**
7. **Specify Volume Size**, the default is used for use maximum size.

8. **Assign Drive Letter or Path**, the default is used of selecting the next available driver letter.
9. **Format Partition**, choose **NTFS** as the File system, **64K** Allocation unit size (cluster size) and set the **Volume label**.

(Optional) To create an ReFS volume choose **ReFS** as the File system.
10. Accept the settings from the wizard and click **Finish**.

The new volume(s) are now ready for use.

---

**Create Mounts Point(s) (Optional)**

1. Right-click on the disk partition and select **Change Drive Letter and Paths**...
2. The Change Drive Letter and Paths for the selected disk will be opened. Click **Add**...

3. **Add Drive Letter or Path** is opened. Since there is already a drive letter assigned, the **Mount in the following empty NTFS folder** is selected. Click **Browse**...
4. Navigate using the **Browse for Drive Path** dialog and either select and existing folder or create a **New Folder**... In the below example a new folder and sub-folder have been created named `\FlashArrayMounts\NTFS`. Click **OK**.

5. Click **OK**.
6. Now the newly assigned mount point can be seen in the Change Drive Letter and Paths for the selected disk.

The volume(s) now have an assigned mount point.

Windows PowerShell

Create FlashArray Volume(s)

The volumes that will be used in the examples are based on the NTFS and ReFS file systems. The host that will be used is named SERVER01 and is running Windows Server 2016 with Windows PowerShell 5.0.
1. Start a new Windows PowerShell session or open the PowerShell Integrated Scripting Environment (ISE).

2. Create a connection to the FlashArray.

   ```powershell
   $FlashArray = New-PfaArray -EndPoint 10.1.1.1 -Credentials (Get-Credential) -IgnoreCertificateError
   PS C:\> $FlashArray
   Disposed   : False
  EndPoint   : 10.1.1.1
   UserName   : pureuser
   ApiVersion : 1.7
   Role       : ArrayAdmin
   ApiToken   : 58db6abf-1933-553f-1628-0c6e207aba1e
   ```

3. Create two new volumes named ReFS and NTFS.

   ```powershell
   New-PfaVolume -Array $FlashArray -VolumeName 'ReFS' -Unit T -Size 1
   New-PfaVolume -Array $FlashArray -VolumeName 'NTFS' -Unit T -Size 1
   ```

   ```plaintext
   source  : serial  : created : name    size    
   serial  : 45084F3508BF461400011ACB 2017-05-23T21:46:04Z ReFS 1099511627776
   serial  : 45084F3508BF461400011ACC 2017-05-23T21:46:04Z NTFS 1099511627776
   ```

---

**Connect FlashArray Volume to Host (or Host Group)**

1. Connect ReFS and NTFS volumes to SERVER01.

   ```powershell
   PS C:\> New-PfaHostVolumeConnection -Array $FlashArray -VolumeName 'ReFS' -HostName 'SERVER08'
   PS C:\> New-PfaHostVolumeConnection -Array $FlashArray -VolumeName 'NTFS' -HostName 'SERVER08'
   ```

   ```plaintext
   vol name     lun
   ---  ----     ---
   ReFS Server08  1
   NTFS Server08  2
   ```

©2020 Copyright Pure Storage. All rights reserved.
If the volumes need to be visible to a cluster of Windows Server hosts then they need to be connected to a Host Group. The following steps show how to create a Host Group, add a Host and then connect the volume to the Host Group.

```powershell
PS C:\> New-PfaHostGroup -Array $f -Hosts 'SERVER08' -Name 'HOSTGROUP1'
PS C:\> New-PfaHostGroupVolumeConnection -Array $f -VolumeName 'ReFS1' -HostGroupName 'HOSTGROUP1'
PS C:\> New-PfaHostGroupVolumeConnection -Array $f -VolumeName 'NTFS1' -HostGroupName 'HOSTGROUP1'
```

<table>
<thead>
<tr>
<th>hosts</th>
<th>name</th>
</tr>
</thead>
<tbody>
<tr>
<td>{SERVER08}</td>
<td>HOSTGROUP1</td>
</tr>
<tr>
<td></td>
<td>HOSTGROUP1</td>
</tr>
<tr>
<td></td>
<td>HOSTGROUP1</td>
</tr>
</tbody>
</table>

```powershell
PS C:\> Get-PfaHostGroupVolumeConnections -Array $f -HostGroupName 'HOSTGROUP1' | Format-Table -AutoSize
```

<table>
<thead>
<tr>
<th>vol</th>
<th>name</th>
<th>lun</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReFS1</td>
<td>HOSTGROUP1</td>
<td>254</td>
</tr>
<tr>
<td>NTFS1</td>
<td>HOSTGROUP1</td>
<td>253</td>
</tr>
</tbody>
</table>

Manage FlashArray Volume(s) from Windows Server Host

1. Rescan/update the Windows Server host, SERVER01 to see the new volumes.

   ```powershell
   PS C:\> Update-HostStorageCache
   ```

2. View the currently connected volumes to the Windows Server host.

   ```powershell
   PS C:\> Get-Disk
   ```

<table>
<thead>
<tr>
<th>Number</th>
<th>Friendly Name</th>
<th>Serial Number</th>
<th>HealthStatus</th>
<th>OperationalStatus</th>
<th>Total Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PURE FlashArray</td>
<td>45084F3508BF461400011ACB</td>
<td>Healthy</td>
<td>Online</td>
<td>1 TB RAW</td>
</tr>
<tr>
<td>2</td>
<td>PURE FlashArray</td>
<td>45084F3508BF461400011ACC</td>
<td>Healthy</td>
<td>Online</td>
<td>1 TB RAW</td>
</tr>
<tr>
<td>0</td>
<td>PURE FlashArray</td>
<td>73E940225A2A52BB0003AE86</td>
<td>Healthy</td>
<td>Online</td>
<td>150 GB</td>
</tr>
</tbody>
</table>

   If the SAN Policy of the Windows Server host is kept at the default of OfflineShared when the volumes are connected to the host they will not come online automatically. Performing a Get-Disk will indicate that they have an OperationalStatus of Offline.

   ```powershell
   PS C:\> Get-Disk
   ```
3. Initialize the new volumes.

Set the \-Number parameter from the output of Get-Disk. The Partition Style has been updated from RAW to GPT.

```
PS C:\> Initialize-Disk -Number 1 -PartitionStyle GPT
PS C:\> Initialize-Disk -Number 2 -PartitionStyle GPT
PS C:\> Get-Disk
```

4. Create a new partition.

```
PS C:\> New-Partition -DiskNumber 1 -UseMaximumSize -AssignDriveLetter
```

©2020 Copyright Pure Storage. All rights reserved.
PS C:\> New-Partition -DiskNumber 2 -UseMaximumSize -AssignDriveLetter

    DiskPath: \??\mpio\disk&ven_pure&prod_flasharray&rev_8888#1\&f6ac24&0&36323413933730343530384633353038424634636313430303031414343#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}

    PartitionNumber DriveLetter Offset Size Type
    -------------- ----------- ------ ---- ----
    2              E          135266304 1023.87 GB Basic

PS C:\> Get-Volume

    DriveLetter FileSystemLabel FileSystem DriveType HealthStatus OperationalStatus SizeRemaining       Size
    ------------ --------------- ---------- --------- ------------ ----------------- -------------       ----
    D           NTFS            NTFS       Fixed     Healthy      Unknown                     0 B       0 B
    System Reserved NTFS       Fixed     Healthy      OK                       169 MB     500 MB
    C                           NTFS       Fixed     Healthy      OK                    117.59 GB 149.51 GB
    E                                      Fixed     Healthy     Unknown                     0 B       0 B

5. Format volumes as NTFS and ReFS.

The below examples set the AllocationUnitSize (cluster size) to 64KB.

PS C:\> Format-Volume -DriveLetter D -FileSystem NTFS -NewFileSystemLabel 'NTFS' -AllocationUnitSize 64KB

    DriveLetter FileSystemLabel FileSystem DriveType HealthStatus OperationalStatus SizeRemaining       Size
    ------------ --------------- ---------- --------- ------------ ----------------- -------------       ----
    D               NTFS            NTFS       Fixed     Healthy      OK                    1023.7 GB 1023.87 GB

PS C:\> Format-Volume -DriveLetter E -FileSystem ReFS -NewFileSystemLabel 'ReFS' -AllocationUnitSize 64KB

    DriveLetter FileSystemLabel FileSystem DriveType HealthStatus OperationalStatus SizeRemaining       Size
    ------------ --------------- ---------- --------- ------------ ----------------- -------------       ----
    E               ReFS            ReFS       Fixed     Healthy      OK                   1018.03 GB 1023.81 GB

PS C:\> Get-Volume

    DriveLetter FileSystemLabel FileSystem DriveType HealthStatus OperationalStatus SizeRemaining       Size
    ------------ --------------- ---------- --------- ------------ ----------------- -------------       ----
    D               NTFS            NTFS       Fixed     Healthy      OK                    1023.7 GB 1023.87 GB
    System Reserved NTFS       Fixed     Healthy      OK                       169 MB     500 MB
    C                           NTFS       Fixed     Healthy      OK                    117.59 GB 149.51 GB
    E               ReFS            ReFS       Fixed     Healthy      OK                   1018.03 GB 1023.81 GB
Create Mounts Point(s) (Optional)

This is an optional step and shows how to create Partition Access Paths (mount points) for volumes.

1. Retrieve the partition details in order to see the **PartitionNumber** which is required for creating a **PartitionAccessPath**.

   ```powershell
   PS C:\> Get-Partition
   DiskPath: \?\mpio\#disk\&ven_pure\&prod_flasharray\&rev_8888\&1\&7f6ac24\&0\&3632344139333730343530383446333530384246346363134303033131414342#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}
   PartitionNumber  DriveLetter  Offset                      Size Type
   ---------------  -----------  ---------------          ---- ----
   1                17408        128 MB Reserved
   2                D           135266304                 1023.87 GB Basic
   DiskPath: \?\mpio\#disk\&ven_pure\&prod_flasharray\&rev_8888\&1\&7f6ac24\&0\&3632344139333730343530383446333530384246346363134303033131414343#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}
   PartitionNumber  DriveLetter  Offset                      Size Type
   ---------------  -----------  ---------------          ---- ----
   1                17408        128 MB Reserved
   2                E           135266304                 1023.87 GB Basic
   DiskPath: \?\mpio\#disk\&ven_pure\&prod_flasharray\&rev_8888\&1\&7f6ac24\&0\&3632344139333730343530383446333530384246346363134303033131414343#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}
   PartitionNumber  DriveLetter  Offset                      Size Type
   ---------------  -----------  ---------------          ---- ----
   1                1048576      500 MB IFS
   2                C           525336576                 149.51 GB IFS
   ```

2. Create directory that will be assigned to the new volume.

   ```powershell
   PS C:\> New-Item -Path 'C:\FlashArrayMounts\NTFS' -ItemType Directory
   Directory: C:\FlashArrayMounts\NTFS
   Mode LastWriteTime Length Name
   ----- ----------- ------ ----
   d---- 5/23/2017 3:50 PM   NTFS
   PS C:\> New-Item -Path 'C:\FlashArrayMounts\ReFS' -ItemType Directory
   Directory: C:\FlashArrayMounts\ReFS
   Mode LastWriteTime Length Name
   ----- ----------- ------ ----
   d---- 5/23/2017 3:50 PM   ReFS
   ```
3. Add the mount points for the individual volumes. Use the PartitionNumber retrieved from Step 1 for the new drives (D and E).

```powershell
PS C:\> Add-PartitionAccessPath -DiskNumber 1 -AccessPath 'C:\FlashArrayMounts\NTFS' -PartitionNumber 2
PS C:\> Add-PartitionAccessPath -DiskNumber 2 -AccessPath 'C:\FlashArrayMounts\ReFS' -PartitionNumber 2
```

4. View the new mount points.

```powershell
PS C:\> cd C:\FlashArrayMounts\ 
PS C:\FlashArrayMounts> ls 
Directory: C:\FlashArrayMounts

<table>
<thead>
<tr>
<th>Mode</th>
<th>LastWriteTime</th>
<th>Length</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>d----l</td>
<td>5/23/2017 3:50 PM</td>
<td>NTFS</td>
<td></td>
</tr>
<tr>
<td>d----l</td>
<td>5/23/2017 3:50 PM</td>
<td>ReFS</td>
<td></td>
</tr>
</tbody>
</table>
```

Below is the view of the mount points from Windows Explorer.