Dell EMC PowerScale: Non-Disruptive Upgrade Best Practices

Abstract
This white paper provides configuration considerations for Dell EMC™ PowerScale™ OneFS™ Non-Disruptive Upgrade (NDU) features including OneFS upgrade and patch upgrade, and covers how NDU can impact different workloads including SMB, NFS, HDFS, FTP, and HTTP.

January 2020
Revisions

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>August 2018</td>
<td>Initial release</td>
</tr>
<tr>
<td>April 2019</td>
<td>Update to reflect the improvements in OneFS 8.2.0</td>
</tr>
<tr>
<td>August 2019</td>
<td>Update to reflect the improvements in OneFS 8.2.1 – simplified patch installation and multi-patches installation during OneFS upgrade.</td>
</tr>
<tr>
<td>January 2020</td>
<td>Update to reflect parallel upgrade in OneFS 8.2.2</td>
</tr>
<tr>
<td>May 2020</td>
<td>Update to reflect rebranding in OneFS 9.0.0</td>
</tr>
</tbody>
</table>

Acknowledgements

This paper was produced by the following members of the Dell EMC storage engineering team:

Author: Vincent Shen (Vincent.Shen@dell.com)

The information in this publication is provided "as is." Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.
Acknowledgements

Use, copying, and distribution of any software described in this publication requires an applicable software license.

Copyright © 2018–2019 Dell Inc. or its subsidiaries. All Rights Reserved. Dell, EMC, Dell EMC and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be trademarks of their respective owners. [5/18/2020] [Best Practices] [H17459.2]
Table of contents

Revisions ........................................................................................................................................... 2
Acknowledgements .......................................................................................................................... 2
Table of contents .............................................................................................................................. 4
Executive summary ........................................................................................................................... 6
Audience ............................................................................................................................................ 6
What’s new for OneFS 8.2. .............................................................................................................. 7
We value your feedback .................................................................................................................... 7
1 OneFS upgrade basics .................................................................................................................... 8
   1.1 What is NDU ............................................................................................................................ 8
   1.2 Upgrade types ....................................................................................................................... 8
   1.2.1 Simultaneous upgrade ................................................................................................... 9
   1.2.2 Rolling upgrade ............................................................................................................... 9
   1.2.3 Parallel upgrade ............................................................................................................. 10
   1.3 Supported upgrade path ..................................................................................................... 11
   1.4 Pre-upgrade check ............................................................................................................... 12
   1.5 Upgrade state and transition ............................................................................................ 12
   1.6 Pause and resume ................................................................................................................ 15
2 Client behavior in an upgrade ....................................................................................................... 16
   2.1 NFS behavior and configuration consideration ..................................................................... 16
   2.1.1 PowerScale dynamic IP pool for NFS workloads ............................................................. 16
   2.1.2 NFS recovery/retry mechanism ..................................................................................... 18
   2.1.3 Performance impact ....................................................................................................... 21
   2.1.4 NDU best practices concluded for NFSv3/v4 ................................................................. 22
   2.2 SMB behavior and configuration consideration ................................................................. 22
   2.2.1 SMB1 and SMB2: always disruptive ............................................................................ 22
   2.2.2 SMB3 CA ....................................................................................................................... 23
   2.2.3 NDU best practices concluded for SMB1/SMB2/SMB3 ................................................. 25
   2.3 HDFS behavior and configuration consideration ............................................................... 25
   2.4 FTP behavior and configuration consideration .................................................................. 25
   2.5 HTTP behavior and configuration consideration ............................................................... 26
3 Patch upgrade ............................................................................................................................... 27
   3.1 Roll-Up Patches overview ................................................................................................ 27
   3.2 General best practices ........................................................................................................ 27
   3.3 Rolling and simultaneous patch upgrade .......................................................................... 28
Table of contents

3.4 Installation of patches during a OneFS upgrade ........................................................................29
3.5 Simplified patch installation process ......................................................................................30

4 Firmware upgrade ..........................................................................................................................33
4.1 Least Disruptive Firmware Upgrade Recommendation ..............................................................33
4.2 Fastest Firmware Upgrade Recommendation ..............................................................................33
4.3 Balanced Firmware Upgrade Recommendation ...........................................................................34
4.4 Workflow .......................................................................................................................................35
4.5 General recommendations ............................................................................................................36

5 Troubleshooting ................................................................................................................................37
5.1 OneFS upgrade .............................................................................................................................37
5.1.1 Monitor the OneFS upgrade ....................................................................................................37
5.1.2 OneFS upgrade logs .................................................................................................................37
5.1.3 Failure handling .......................................................................................................................38
5.2 Patch upgrade .............................................................................................................................38
5.2.1 Monitor the patch upgrade ......................................................................................................39
5.2.2 Failure handling ......................................................................................................................39
5.3 Firmware upgrade .......................................................................................................................40

A Technical support and resources .....................................................................................................41
A.1 Related resources .........................................................................................................................41
Executive summary

This white paper provides configuration considerations and best practices of the Dell EMC™ PowerScale™ OneFS™ Non-Disruptive Upgrade (NDU) including the following:

- Explanation of OneFS NDU mechanism and its general configuration considerations
- Explanation of how OneFS upgrade can impact the client workloads and the best practices, discussing the following workloads:
  - SMB: including SMB1, SMB2 and SMB3 CA
  - NFS: including NFSv3 and NFSv4
  - HDFS
  - FTP
  - HTTP
- Patch upgrade consideration

Audience

This guide is intended for experienced system and storage administrators who are familiar with file services and network storage administration.

This guide assumes the reader has a working knowledge of the following:

- Network-attached storage (NAS) systems
- The PowerScale scale-out storage architecture and the PowerScale OneFS operating system

The reader should also be familiar with PowerScale documentation resources, including:

- Dell EMC Community Network info hubs
- Dell EMC OneFS release notes, which are available on the Dell EMC support network and contain important information about resolved and known issues.
- Dell EMC PowerScale OneFS Best Practices
What’s new for OneFS 8.2.

In this white paper, some key NDU improvements and new features are introduced with details. Click the links in Table 1 for more details.

Table 1  NDU improvements and new features in OneFS 8.2.0

<table>
<thead>
<tr>
<th>Improvements/New Features (click links to go to sections)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pause and resume</td>
<td>Pause and Resume during OneFS upgrade</td>
</tr>
<tr>
<td>Installation of patches during a OneFS upgrade</td>
<td>In OneFS 8.2.0, PowerScale supports installation of patches during a OneFS upgrade</td>
</tr>
<tr>
<td>Simultaneous Firmware upgrade</td>
<td>Starting with OneFS 8.2.0, PowerScale supports simultaneous firmware upgrade</td>
</tr>
<tr>
<td>OneFS upgrade logs</td>
<td>OneFS 8.2.0 supports filters to extract upgrade logs.</td>
</tr>
<tr>
<td>Monitor the OneFS upgrade hangs</td>
<td>OneFS 8.2.0 supports to detect upgrade hangs.</td>
</tr>
</tbody>
</table>

We value your feedback

Dell EMC and the authors of this document welcome your feedback on the white paper.

Authors: Vincent Shen (Vincent.shen@dell.com)
OneFS upgrade basics

1 OneFS upgrade basics

This section explains the following topics:

- NDU introduction
- Upgrade types including simultaneous upgrade and rolling upgrade
- Supported upgrade path
- Upgrade states and transition

1.1 What is NDU

NDU is a framework which is introduced in OneFS 8.0 to provide better control and predictability to the upgrade process.

From a user perspective, the goal is to provide a seamless and robust upgrade process which is non-disruptive to normal business workflows.

At the very high level, the NDU framework implements a state machine, which means at any given time, the PowerScale cluster is at one of the following states:

- Upgrading
- Upgraded
- Rollback
- Committed

NDU supports the state’s transition through some user actions. For details of the NDU state machine and its transition, refer to section 1.4.

At the low level, the NDU framework is implemented by a controlling Supervisor process and a servant Agent daemon process on every PowerScale node.

- The Supervisor is a short-lived process and it is to assess the current state of the PowerScale cluster and take the appropriate action.
- The Agent is a daemon process on every PowerScale node. It is used to take actions on the local node based on received messages from the Supervisor process.

Note: NDU framework is not only used for OneFS upgrade, it is applied to patch and node firmware upgrade as well. This chapter will only focus on OneFS upgrade. For patch upgrade, refer to section 3, Patch upgrade.

1.2 Upgrade types

There are three options available for upgrading OneFS:

- Simultaneous upgrade
- Rolling upgrade
- Parallel upgrade

The details of each upgrade type will be explained in the following sections.
1.2.1 Simultaneous upgrade
A simultaneous upgrade installs the new operating system and restarts all nodes in the PowerScale cluster concurrently.

Simultaneous upgrades are faster than rolling upgrades but require a temporary interruption of service during the upgrade process. All client connections to the cluster must be terminated prior to initiating the upgrade and data is inaccessible until the installation of the new OneFS operating system is complete and the cluster is back online. Based on this, OneFS simultaneous upgrade is a disruptive upgrade path as all of the cluster services will be offline during the upgrade process.

1.2.2 Rolling upgrade
A rolling upgrade individually upgrades and restarts each node in the PowerScale cluster so that only one node is offline at a time.

A rolling upgrade takes longer to complete than a simultaneous upgrade. You can specify the order in which nodes are upgraded by using the --nodes parameter of the isi upgrade cluster start command. The --nodes parameter can also be used in the scenario to upgrade a specific subset of nodes. The following example command starts a rolling upgrade on logical node numbers (LNN) 1, 3, and 5 in that order:

```bash
isi upgrade cluster start <install-image-path> --nodes 1,3,5
```

The following example commands use a dash-separated range to upgrade LNN 1 to node 5:

```bash
isi upgrade cluster start <install-image-path> --nodes 1-5
```

It is required to upgrade all the nodes in order to install a patch, do a node firmware upgrade and do the next OneFS upgrade. If you only upgrade several nodes in the cluster, a weekly alert is sent to confirm that the upgrade is making progress if you have subscribed to the corresponding alert channel. Do not leave the cluster in a partially upgraded state for a prolonged period. Some new features in the upgrade might not be available until all the nodes in the cluster have been upgraded and the upgrade is committed. Refer to the release notes for the OneFS version that you are upgrading to for information about features that require the cluster to be committed to the upgraded version of OneFS.

To add new nodes to a running upgrade process, use the following command:

```bash
isi upgrade cluster add-nodes -nodes=2,4,6
```

To add all the remaining nodes to an upgrade process, use the following command:

```bash
isi upgrade cluster add-remaining-nodes
```

If you do not specify an order, nodes are upgraded in ascending order from the node with the lowest Array ID to the node with the highest Array ID. Because Array ID’s are never reused, a node’s Array ID might not be the same as the node’s logical node number (LNN). To check each node’s Array ID, run the following command:

```bash
isi_nodes "%{name}: LNN %{lnn}, Array ID %{id}"
```

A typical outcome of the above command is as below, and in this case, LNN matches Array ID.

```
tme-sandbox-1: LNN 1, Array ID 1
tme-sandbox-2: LNN 2, Array ID 2
tme-sandbox-3: LNN 3, Array ID 3
```
Important: During a rolling upgrade, nodes that are not actively being upgraded remain online and can continue serving clients. However, clients that are connected to a restarting node are disconnected and reconnected. How the client connection behaves when a node is restarted depends on several factors including client workload type, client configuration (mount type, timeout settings), IP allocation method, and how the client connected to the cluster. Usually, NDU requires specific configurations on either the PowerScale side or the client-side. For detailed client behavior and the recommended configurations, refer to the section Client behavior in an upgrade.

Rolling upgrades are not available between all OneFS versions. Refer to the next section Supported upgrade path for information about which types of upgrades are supported between OneFS versions.

1.2.3 Parallel upgrade
The parallel upgrade is introduced in OneFS 8.2.2. It provides some extent of parallelism which is to upgrade at most one node per neighborhood at any time. By doing that, it can reduce upgrade duration and ensure that the end-user can still continue to have access to their data.

As shown in Figure 1, this feature can be enabled through WebUI.

![Upgrade OneFS](image)

Figure 1 OneFS parallel upgrade

You can also leverage CLI to enable this feature through OneFS upgrade:

```
# isi upgrade start --parallel /ifs/install.tar.gz
```

The parallel upgrade can dramatically improve the OneFS upgrade efficiency without impacting the data availability. You can use the following formula to estimate the duration of the parallel upgrade:

\[
\text{Estimation time} = (\text{per node upgrade duration}) \times (\text{highest number of nodes per neighborhood})
\]

In the above formula:

- The first parameter – **per node upgrade duration**, it’s 20 minutes on average.
- The second parameter – **the highest number of nodes per neighborhood**, you can get the number by running the following command:

```
# sysctl efs.lin.lock.initiator.coordinator_weights
```
The following is an example:

In a 150 node PowerScale cluster, ideally, there are 15 neighborhoods with 10 PowerScale nodes each. Neighborhood 1st with node number 1 to 10 and Neighborhood 2nd with node number 11 to 20 and etc.

During the parallel upgrade, the upgrade framework will pick at most one node from each neighborhood, to run the upgrading job simultaneously. So in this case, node 1 from neighborhood 1st, node 11 from neighborhood 2nd, node 21 from neighborhood 3rd and etc will be upgraded at the same time. Considering, they are all in different neighborhoods or failure domain, it will not impact the current running workload. After the first pass completes, it will go to the 2nd pass and then 3rd and etc.

So, in this example the estimated duration of the parallel upgrade is 200 minutes:

\[
Estimation \ time = (\text{per \ node \ upgrade \ duration}) \times (\text{highest \ number \ of \ nodes \ per \ neighborhood}) = 20 \times 10 = 200 \ minutes
\]

1.3 Supported upgrade path

The supported upgrade path is a matrix by which it is easy to see which OneFS releases can be upgraded to from a given source release. This matrix applies to both simultaneous and rolling upgrades - in other words, an upgrade is either possible or not, regardless of the upgrade method chosen. Table 2 is an example of a high-level matrix of the supported upgrade path from 8.0.0.

Starting from 8.0.0, the supported upgrade path follows the rule below:

Consider a source release version in the format of w.x.y.z. The rule is that a simultaneous or rolling upgrade is supported to any version up to and including x+2, where the w digit either remains the same or increases by one step only.

**Note:** starting from OneFS 8.2.0, the release version is in a 3-digit format such as w.x.y.

This document will only focus on the upgrade path starting from OneFS 8.0.0. For a detailed supported upgrade path for all OneFS releases, refer to [OneFS Upgrade Paths](#).

Table 2  The supported upgraded path from OneFS 8.0.0

<table>
<thead>
<tr>
<th>From/To</th>
<th>8.0.0.z₂</th>
<th>8.0.1.z₂</th>
<th>8.1.0.z₂</th>
<th>8.2.0</th>
<th>8.3.0 (assumption)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0.0.z₁</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Not supported</td>
</tr>
<tr>
<td>8.0.1.z₁</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Not supported</td>
</tr>
<tr>
<td>8.1.0.z₁</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td></td>
</tr>
<tr>
<td>8.2.0</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td></td>
</tr>
<tr>
<td>8.3.0 (assumption)</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td>Simultaneous or Rolling</td>
<td></td>
</tr>
</tbody>
</table>
OneFS upgrade basics

**Note:** At the time of publication, 8.3.0 is a hypothetical release and only for demonstration purposes.

There are some special upgrade support matrix and path defined. For a complete list of OneFS upgrade paths, refer to [OneFS Upgrade Paths](#).

### 1.4 Pre-upgrade check

It is recommended to have pre-upgrade checks before performing an actual upgrade job. At the time of writing, there are three tools for this purpose:

- **IOCA**
- **CLI for upgrade assessment** (`isi upgrade cluster assess`)
- **Health Check Framework**

For details of each tool and the recommendations, refer to [OneFS 8.0.0 - 8.2.0 Upgrade Planning and Process Guide](#).

### 1.5 Upgrade state and transition

From OneFS 8.0.0, the entire NDU process can be described by an PowerScale cluster upgrade state machine as demonstrated in Figure 2. This means an PowerScale OneFS cluster is in exactly one of the following upgrade state at any given time:

- Committed
- Upgrading
- Upgraded
- Rollback

The state will change as it moves through the NDU cycle based on a set of allowed transitions. In Figure 2, transitions are the blue arrows between two upgrade states:

- Upgrade
- Upgrade complete
- Rollback
- Rollback complete
- Commit

Table 3 lists all the details of each OneFS upgrade states and how they can switch from one to another through transitions.
OneFS upgrade basics

![Diagram of PowerScale cluster upgrade states and transition paths](image)

**Figure 2** PowerScale cluster upgrade state and transition paths

<table>
<thead>
<tr>
<th>PowerScale cluster status</th>
<th>Description</th>
</tr>
</thead>
</table>
| Committed                 | • A previous upgrade operation has been completed and committed.  
                            • All nodes are running the same version of OneFS and all features of that version are available. Rollback to the previously installed version is not available.  
                            • The cluster is ready to start another OneFS upgrade when required.  
                            • A cluster remains in this state until another upgrade is initiated.  
                            • This is considered the steady state of a cluster, and it is expected that a cluster over its lifecycle will spend the majority of its operational time in this state. |
| Upgrading                 | • At least one PowerScale node has started upgrading to the target release version.  
                            • The required information to roll back to the source release is maintained while the cluster is in Upgrading state.  
                            • A cluster remains in Upgrading state until either all nodes are upgraded to the target release, or a rollback is initiated.  
                            • In Upgrading state, the cluster is running in mixed mode, as there are now two versions of OneFS present in the cluster.  
                            • Nodes which have already upgraded may be able to access some of the functionality of the new release.  
                            • Nodes which have not been upgraded cannot access any new release functionality. |
OneFS upgrade basics

<table>
<thead>
<tr>
<th>PowerScale cluster status</th>
<th>Description</th>
</tr>
</thead>
</table>
| Upgraded                  | - All nodes are now running the target release version; however, the upgrade has not been Committed.  
- The required configuration to roll back to the source release is maintained while the cluster is in an Upgraded state.  
- A cluster in the Upgraded state can run any new functionality of the target release. |
| Rollback                   | - The cluster is in the process of rolling back a OneFS upgrade.  
- Rollback can be initiated by the administrator on a cluster in either the Upgrading or Upgraded state.  
- Once the upgrade is committed, rollback is no longer available.  
- In Rollback state, the cluster restores the saved information associated with the source release and prepares the nodes to reboot to the original source release version. Once the nodes have rebooted, the cluster transitions automatically to the Committed state.  
- Rollback is available for both rolling and simultaneous upgrades. A cluster can be rolled back only to the previously installed release.  
- This state should be considered a transition state. Clusters should not be run in this state for extended periods of time. |

To check out the current cluster upgrade state and activity, use the following CLI command:

`isi upgrade cluster view`

or

`isi upgrade view`

An example of the output from the command above is as below, which indicates the PowerScale cluster is in the committed states:

**Upgrade Status:**

Cluster Upgrade State: committed  
Current Upgrade Activity: -  
- Upgrade Start Time: 2018-08-09T07:22:15  
- Upgrade Finished Time: 2018-08-14T06:09:35  
- Current OS Version: 8.1.0.4_build(57)style(5)  
- Upgrade OS Version: N/A  
- Percent Complete: 0%

**Nodes Progress:**

Total Cluster Nodes: 3  
Nodes On Older OS: 3  
Nodes Upgraded: 0  
Nodes Transitioning/Down: 0  

LNN  Progress  Version  Status
### 1.6 Pause and resume

Starting from OneFS 8.2.0, a OneFS upgrade can be paused and resume. This usually happens when customers reach the end of a maintenance window and they can pause the upgrade and resume in a later window. To pause a running OneFS upgrade process, run the following command:

```bash
isi upgrade pause
```

After this command is triggered, upgrade status will be in a **Pausing** status until the current upgrading node is completed. The remaining nodes will not be upgraded until the upgrade process is resumed.

To resume a paused OneFS upgrade process, run the following command:

```bash
isi upgrade resume
```

To check the **Pausing/Paused** status, use the following CLI command:

```
isi upgrade view
```

Or, to view the **PAUSE** file data by using the following command:

```
cat /ifs/.ifsvar/upgrade/processes/upgrade/PAUSE
```

A typical output which indicates the PowerScale cluster is still in pausing status is as below:

```json
{
  "PauseState": "Pausing"
}
```
2 Client behavior in an upgrade

This section explains the behavior of different workloads during a OneFS rolling upgrade including the following workloads:

- NFS
- SMB
- HDFS
- HTTP
- FTP

For each workload, this section includes best practices and configurations for NDU consideration.

2.1 NFS behavior and configuration consideration

This section explains how PowerScale OneFS upgrade process can impact the NFS workloads including both NFSv3 and NFSv4.

Note: NFS version 2 is not supported in OneFS 7.2.0 and above. Due to this reason, it is not included in this white paper.

Before explaining how the PowerScale OneFS NDU process can impact the NFS workloads, it is very important to understand the following three points:

- PowerScale dynamic IP pool for NFS workloads
- NFS recovery or retry mechanism
- Performance impact

Best practices are included in the conclusion of this section.

2.1.1 PowerScale dynamic IP pool for NFS workloads

2.1.1.1 NFSv3 with dynamic IP pool

Dynamic IP pools assign out all the IP addresses within a given range to all the available NICs across the entire PowerScale cluster. Dynamic IP addresses can move from one NIC to another, when a node goes to an unhealthy state. This ensures that dynamic IP addresses are always available during failover and failback. For a stateless protocol like NFSv3, the best practice is to use a dynamic IP pool for business continuity.

During the OneFS NDU process, if the rolling upgrade is selected, it will individually upgrade and restart each node in the PowerScale cluster so that only one node is offline at a time. Once a node is offline, the IP address of this node will move to one of the remaining available nodes by using the dynamic IP pool.

As shown in 0, in a 4 nodes PowerScale cluster, once node 1 is offline, both of the dynamic IPs on node 1 will move to the remaining nodes to ensure the business continuity. If the NFS clients use 192.168.200.241 as the NFS server IP to mount NFS exports, during the node 1 offline, it is actually accessing node 2 in the PowerScale cluster and this is transparent to the NFS clients.
**Important:** This will introduce a noticeable pause of the NFS workload. Usually, it only takes less than 20 seconds, which is the amount of time that it usually takes the network ARP cache to flush. This NFS workload pause only happens in the clients which connect to the PowerScale node being rebooted. The other clients will not be affected. In this period of time, you will see the throughput between the NFS client and the NFS server is 0. And after that, it will restore automatically.

![Dynamic IP example](image)

**Figure 3  Dynamic IP example**

Figure 4 is an example of how it works in the rolling upgrade: NFS is mounted at 192.168.200.241. After initiating the OneFS rolling upgrade, node 1 reboots first and causes the NFS mount IP to move from Node 1 to Node 2, which will introduce a noticeable pause of NFS workload. Then, Node 2 starts to upgrade and follows a reboot. This will introduce another IP reallocation from Node 2 to Node 3 and a second short pause. In this case, during the rolling upgrade process, there will be 4 short pauses in total. This is for a 4 nodes PowerScale cluster. If it is a large cluster, the interruption will be much more frequent.

![Worst case of OneFS rolling upgrade](image)

**Figure 4  Worst case of OneFS rolling upgrade**

The above example is the worst case which tends to be unlikely to happen. This is because from OneFS 8.0, when performing NFS failover using SmartConnect dynamic IP, we tend to favor nodes that are already upgraded. For example, on a three-node PowerScale cluster with 8 IPs in a dynamic pool, if PowerScale node 3 has been upgraded and node 1 and node 2 have not, OneFS will rebalance IPs so that node 3 will...
have 4 IPs and at the same time node 1 and node 2 only have two IPs each. This awareness reduces the overall rolling upgrade process to the overall services.

On the other hand, we have multiple dynamic IPs per pool to spread the load across multiple nodes to mitigate the impact. Determining the number of IP addresses within a dynamic allocation pool varies depends on the PowerScale node count, the estimated number of clients that would be in a failover event and so on. For detailed best practices, refer to PowerScale Network Design Considerations.

2.1.1.2 **NFSv4 with dynamic IP pool**

NFSv4 is a stateful protocol and in this case, it expects the NFS server to maintain session state information. This means each PowerScale node runs its own NFS daemon and the session information is unique per node. For this reason, we usually recommended using PowerScale static IP pool for NFSv4. However, beginning in PowerScale OneFS 8.0, the NFSv4 session state information is kept in sync across multiple nodes. In the OneFS 8.0 and later, it is recommended to use a dynamic IP pool for NFSv4 connections.

In the case where a static IP pool is applied for NFSv4 workloads, there will be a much longer time during which NFS clients will not receive any response from an PowerScale node. This is because static IP’s cannot move amongst the interfaces. Clients that cannot communicate to the specific PowerScale node may receive an “NFS server not responding” message until the PowerScale node comes back online. In some cases, the NFS client may timeout.

2.1.2 **NFS recovery/retry mechanism**

The behavior of NFS recovery is determined by several NFS mount options as below. These mount options apply to both NFSv3 and NFSv4.

**timeo=n**

The timeo is measured in deciseconds (tenths of a second) and it means how long the NFS client waits for a response before it retries an NFS request. In this period of time, NFS clients will see an “NFS server not responding” response.

For NFS over TCP the default timeo value is 600 (60 seconds). As shown in Figure 5, the NFS client performs linear backoff algorithm for timeout value, which means after each retransmission the timeout is increased by timeo up to the maximum of 600 seconds. Figure 5 shows an example where timeo equals to 600.

![Figure 5](image)

**Figure 5** An example of an NFS timeout linear backoff algorithm (timeo=600)

**retrans=n**
The `retrans` is the number of times the NFS client retries a request before it attempts further recovery action. If the `retrans` option is not specified, the NFS client tries each request 3 times. Figure 6 shows an example of `retrans` equal to 2 and `timeo` equaling to 600.

![Diagram of NFS retrans](image)

**Figure 6** An example of NFS `retrans` = 2

**Soft/hard mount**

The soft or hard mount option determines the recovery behavior of the NFS client after an NFS request times out as described in Table 4. For most clients, Dell EMC recommends using the hard mount option and avoid soft mount.

<table>
<thead>
<tr>
<th>Mount type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard (or not specify)</td>
<td>After an NFS request timeout, it will attempt to retry and NFS requests are retried indefinitely.</td>
</tr>
<tr>
<td>Soft</td>
<td>Once an NFS request timeout, it will attempt to retry. But after <code>retrans</code> retransmissions have been sent, the NFS client fails an NFS request, causing the NFS client to return an error to the calling application. For example if <code>retrans</code> equals to 2, the NFS client will return an error after 2 attempts to retry. This example is also shown in Figure 7.</td>
</tr>
</tbody>
</table>
Client behavior in an upgrade

**Figure 7** An example of soft mount failure (retrains = 2)

It is obvious to see how the client behaves during the noticeable pause in a rolling upgrade is determined by the above 3 mount options. The detailed explanation is as the following:

- In the case of a hard mount, due to the fact that the NFS client request will attempt to retry indefinitely, there will be no error message in the NFS layer during the noticeable pause in a rolling upgrade process.

**Note:** Although in the NFS layer, there will be no errors and NFS client will try to retry indefinitely for hard mount, some applications may still encounter errors and this depends on how the application is implemented. Consult your application vendors for this situation.

- In the case of a soft mount, if the noticeable pause ends in the green area as shown in 0, there will be no error message in the NFS client application. If the noticeable pause ends beyond the green area as shown in Figure 9, the NFS client will send an error message during the OneFS rolling upgrade process. In most cases, it recommends using hard mount instead of using soft mount.
2.1.3 Performance impact

Dell EMC recommends all non-disruptive upgrades be performed at a time of low I/O. This is identified as the target maintenance window. If you perform the OneFS rolling upgrade during the maintenance window, you will see minimal performance impact during the overall process.

In case OneFS rolling upgrade is initiated at a time the cluster is under heavy workload, you will see limited performance impact due to the PowerScale node reboots, since you now have \((n-1)\) PowerScale nodes in the cluster to serve the workload during the reboot time. Note that performance impacts will be lessened as PowerScale cluster size increases.
2.1.4 NDU best practices concluded for NFSv3/v4
With the knowledge of section 2.1.1, PowerScale dynamic IP pool, and section 2.1.2, NFS recovery/retry mechanism, we can conclude the following NDU best practices for NFSv3/v4:

- Use PowerScale dynamic IP pool for NFSv3.
- Use PowerScale dynamic IP pool for NFSv4, if the OneFS version is 8.0 and above.
- Leverage SmartConnect multiple dynamic IPs and SSIP to spread the load across multiple nodes to mitigate the impact of OneFS rolling upgrade process.
- Use NFS hard mount option and the default NFS mount option is good enough for NDU consideration.

2.2 SMB behavior and configuration consideration
This section explains how the PowerScale OneFS upgrade process can impact the SMB workloads including:

- SMB1 and SMB2
- SMB3 with continuous availability (CA)
- Best practices concluded

2.2.1 SMB1 and SMB2: always disruptive
SMB is a stateful protocol which means it maintains a session state for all the open files in the PowerScale node where the client connects to. This session state is not shared across the nodes. For a stateful protocol like SMB, it is recommended using OneFS static IP pools. But in certain workflows, SMB is preferred to use a dynamic IP pool. SMB preserves complex state information per session on the server side. If a connection is lost and a new connection is established with dynamic failover to another node, the new node may not be able to continue the session where the previous one had left off. If the SMB workflow is primarily reads, the impact of a dynamic failover will not be as drastic, as the client can re-open the file and continue reading. Conversely, if an SMB workflow is primarily writes, the state information is lost and the writes could be lost as well.

By using the static IP pool, IP addresses assigned to the node will not reallocate to other nodes in the event of hardware failure or reboot. The client behavior of SMB1 and SMB2 during rolling upgrade is listed in the following table:

<table>
<thead>
<tr>
<th>Access methodology</th>
<th>Client behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct IP access</td>
<td>• Connection will drop.</td>
</tr>
<tr>
<td>e.g. \langle PowerScale Node IP\rangle\langle share name\rangle</td>
<td>• The application may send an error message.</td>
</tr>
<tr>
<td></td>
<td>• The client will wait for the SMB service to resume on the node.</td>
</tr>
<tr>
<td>Access through SmartConnect zone</td>
<td>• Connection will drop.</td>
</tr>
<tr>
<td>e.g. \langle smartconnct zone name\rangle\langle share name\rangle</td>
<td>• The application may send an error message.</td>
</tr>
<tr>
<td></td>
<td>• The client can quickly re-establish the connection to another PowerScale node by leveraging the SmartConnect failover policy.</td>
</tr>
</tbody>
</table>
Therefore, the recommended configuration for SMB1 or SMB2 is to use SmartConnect with a SmartConnect Service IP and an IP failover policy to quickly re-establish the connection between the client and the PowerScale cluster.

When SmartConnect failover policy is used, the connection will drop and re-established to another PowerScale node in the cluster. In the rolling upgrade process, the node where the new connection is established will also have a chance to reboot later on. And the worst case is that in an n-node PowerScale cluster, this disruptive failover will happen n times. Although re-establishing the connection using SmartConnect is usually instantaneous, there is still a brief disruption of the client application, which means client application is aware of the disruption and will send an error message. In order to resume the client workload, the connection must be re-established.

### 2.2.2 SMB3 CA

In OneFS 8.0, PowerScale offers the SMB continuously available (CA) option. This allows SMB clients the ability to transparently fail over to another node in the event of a network or node failure. This feature applies to Microsoft Windows 8, Windows Server 2012 and later clients. This feature is part of PowerScale's non-disruptive operation initiative to give customers more options for continuous work and less downtime. The SMB CA option allows seamless movement from one node to another and no manual intervention on the client side. This enables a continuous workflow from the client side with no disruption error message to their working time.

Dell EMC recommends using static IP pool with SMB3 CA for transparent failover and NDU consideration. But using dynamic IP pool can also work, but there is a risk with SMB3 CA Witness sending confusing signals. The behavior really depends on the client implementation. For example, it probably causes just one failover to another IP, after which the client loses interest in the original address, but it could also potentially make the client jump around with reconnections for no good reason if its interest is not lost and it keeps watching.

The SMB CA feature needs to be enabled at share creation time. To enable SMB CA, the following preconditions need to be met:

- SMB3 is supported
- The cluster is running OneFS 8.0 or later
- Clients are running Windows® 8 or Windows Server® 2012 R2 or later

---

**Note:** It is recommended to enable SMB Witness feature for transparent failover, which can dramatically shorten the time to detect the failure. A common way to enable SMB Witness on PowerScale OneFS is to set the SmartConnect zone name and access the SMB share with the name. This is because SMB Witness can get the failure notification from SmartConnect and FlexNet.

If any precondition in the above list is not met, SMB3 CA will not function.

You can use the following command to create an SMB file share with CA enabled:

```bash
isi smb shares create <name> <path> --continuously-available=yes
```

In case the existing share is not SMB CA enabled, you can still enable it on an existing file share by using the following command:

```bash
isi_smb_ca_share --enable-ca --share=<the name of SMB share>
```
Client behavior in an upgrade

Making a change by the command above will actually delete and recreate the share without losing any data. But it will result in a quick disconnection for all current clients and this is a disruptive command. After the OneFS rolling upgrade is finished, if you want to revert it back, use the following command:

\texttt{isi\_smb\_ca\_share --disable-ca --share=<the name of SMB share>}

To verify the SMB CA and SMB Witness is enabled at the client level, check the Windows Event Log in the following path:

Applications and Services Logs, Microsoft, Windows, SMBCclient, Connectivity.

Figure 10 shows an example of the Windows Event Log message of successful Witness registration.

![Figure 10 SMB3 Witness registration](image)

During the rolling upgrade process, the PowerScale node reboots one by one. If SMB3 CA is enabled on a share, when the PowerScale node reboots, the connection to the share will not be disrupted and thus there will be no error message sent by the application. There will still be a very short period of time when all the workload on the share is paused and automatically resumed in several seconds.

For the performance impact, it is very similar to NFS workload. Refer to 2.1.3 Performance impact for more details.
### 2.2.3 NDU best practices concluded for SMB1/SMB2/SMB3

As a summary of 2.2.1 SMB1 and SMB2 and 2.2.2 SMB3 CA, we recommend the following NDU best practices for SMB1/SMB2/SMB3:

- Use PowerScale static IP pool for SMB1/SMB2/SMB3
- Access the SMB share through SmartConnect zone name
- Use SmartConnect failover policy and connect to SmartConnect zone for SMB1/SMB2
- Use SMB3 CA for SMB3 share

**Note:** Due to the nature of SMB CA, this feature will bring some performance impact especially on write I/O. The impact depends on the factors like the PowerScale node type, the configuration of PowerScale OneFS like endurant cache (EC), workload profile and etc.

### 2.3 HDFS behavior and configuration consideration

HDFS connections are unique in the fact that they are made up of two separate connects as listed below:

- A Name Node connection
- A Data Node connection

In comparison to Apache Hadoop, each OneFS node is a NameNode. Therefore, to ensure access during upgrade, the NameNode connection should be managed via SmartConnect, which will delegate each connection to an available node. SmartConnect requires that you add a new SmartConnect Service IPs (SSIP) record as a delegated DNS to the authoritative DNS zone that contains the cluster. All Hadoop clients should be configured to use a SmartConnect IP as the NameNode IP.

When the DataNode connection fails, the Hadoop JobTracker will restart failed jobs. This provides some protection against nodes going down for upgrade. However, some services use HDFS to write files outside of jobs, including Hbase’s Write Ahead Log. For those cases, OneFS introduced Pipeline Recovery in 8.0.1.0. With Pipeline Recovery, failed DataNode writes are automatically repeated on another working node. This includes when a DataNode is rebooted for upgrade. This allows the upgrading cluster to be used without interruption. No action is necessary to enable these recovery measures.

Refer to [EMC PowerScale Best Practices Guide for Hadoop Data Storage](#) for additional details and considerations with HDFS pool implementations.

### 2.4 FTP behavior and configuration consideration

FTP is a stateful protocol which means PowerScale should keep the session state between client and itself. Due to this reason, the recommendation is to use a static IP pool. The IP will not failover or failback during the PowerScale node reboot. The client has to wait for FTP serviceability to resume on the node that it is connected to. In this case, using SmartConnect Service IPs (SSIP) can help minimize the impact. SSIP is implemented by a way of DNS delegation and it can help to redirect the request to the right PowerScale nodes which are still alive. However, if the rolling upgrade reboot happens in the middle of file transmission, the transmission will get stopped with errors and need to be manually re-establish the connection. The recommendation is as below:

- Use static IP pool for FTP workload
- Use SSIP enabled subnet for FTP workload to minimize the impact
Note: With all the recommendations above, the OneFS upgrade process still provides a disruptive upgrade. But it can dramatically minimize the impact.

2.5 HTTP behavior and configuration consideration

PowerScale OneFS has a built-in web service and we can easily access the files by using the HTTP protocol. At the time of writing PowerScale only supports HTTP 1.1 which is a stateless protocol. Refer to RFC 7230 for details of HTTP 1.1.

Since it is a stateless protocol, it is recommended using dynamic IP pool to make sure all the IPs in the pool are accessible during the reboot of OneFS rolling upgrade process. However, if a file is in the transmission status, it will get disconnected by errors and you have to retry and re-establish the connection by manually refresh the page and reinitiate the file transfer. An alternative way is to use SmartConnect zone name to make sure the client HTTP request can always find the right PowerScale nodes which are still alive. However, it has the same side effect. Since SSIP is a way of a delegation of DNS, it will not support HTTP requests by directly accessing the IP address. Best practices include the following:

- Use dynamic IP pool for HTTP workload
- Use SSIP enabled subnet for HTTP workload, if all the HTTP request are through the zone name

Note: With all the recommendations above, the OneFS upgrade process still provides a disruptive upgrade. But it can dramatically minimize the impact.
### 3 Patch upgrade

The OneFS patch system provides a method to deploy a set of changes to all nodes in the PowerScale cluster in a simple and revisable manner, which is also under the control of OneFS NDU framework. It allows a user to apply a patch in a simultaneous way or a rolling sequence. The details of rolling and simultaneous patch upgrade will be discussed in 3.3.

#### 3.1 Roll-Up Patches overview

A monthly cadence for Roll-Up Patches (RUPs) has been established to deliver critical fixes to customers on the following releases:

- OneFS 8.1.0.4
- OneFS 8.1.2.0
- OneFS 8.2.0

In general, there are 3 kinds for RUPs delivered for each of the OneFS release listed above every month. They are shown in Table 6:

<table>
<thead>
<tr>
<th>RUPs category</th>
<th>Userspace/Kernel patch</th>
<th>Require reboot</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Userspace GA RUPs</td>
<td>Userspace patch</td>
<td>No</td>
<td>Highest priority fixes with minimum risk and maximum benefits</td>
</tr>
<tr>
<td>Userspace DA RUPs</td>
<td>Userspace patch</td>
<td>No</td>
<td>Broader fixes coverage</td>
</tr>
<tr>
<td>Kernel GA RUPs</td>
<td>Kernel patch</td>
<td>Yes</td>
<td>Fixes in kernel space, for example, drivers bug or security bug. It will not conflict with DA or GA Userspace RUPs.</td>
</tr>
</tbody>
</table>

The relationship among all the three RUPs categories is concluded as below:

- Each month’s Userspace GA RUP is a superset of the Userspace GA RUP for the previous month.
- Each month’s Userspace DA RUP is a superset of the Userspace DA RUP for the previous month.
- Each month’s Userspace DA RUP is a superset of the Userspace GA RUPs of the current month.
- Kernel GA RUPs will not conflict with either Userspace DA RUP or Userspace GA RUP of the same month

#### 3.2 General best practices

Here list several general best practices and configuration considerations for patch upgrade:

- The NDU framework was originally created to handle OneFS upgrades. The same framework is also used for the patch system and to deploy firmware packages. As designed for the 8.0 and later release, the NDU framework can only be used to perform one action at a time. This means that once a OneFS upgrade has been started, NDU will not be available to deploy patches or firmware packages until the upgrade has been committed.
• Beginning with the 8.0 release, the NDU framework creates the possibility to upgrade a cluster and then roll back to the previously installed version any time before the new version is committed. This is accomplished by creating a rollback image from one node in the cluster and then deploying that rollback image to all the nodes during the rollback operation. When the rollback file is created, it will include any patches installed on the node as well as the local patch databases. In the event that the OneFS upgrade is rolled back, the exact patches that were previously installed on each node will be replaced with the patches contained in the rollback file. This will only be an issue if there is a different patch set installed on different nodes in the cluster. Dell EMC recommends that you use consistent patch among all the PowerScale nodes in the cluster.

• Dell EMC recommends that you install the patch during an off-hours maintenance window to minimize the disruption of service to clients.

3.3 Rolling and simultaneous patch upgrade

Before OneFS 8.2.0, in the patch upgrade command `isi upgrade patch`, there is a parameter `--rolling=true/false` which controls how the patch is applied. With different purpose of the patch, this parameter behaves differently.

1. In the case of a patch which impacts certain services and only requires service to restart (user space patch), it follows the way as below:

When the parameter `--rolling=false` is set, a simultaneous patch request is made and a patch will be installed simultaneously across all nodes. The patch will typically run an `isi services` command to disable and then later to re-enable the affected services. Since services that are affected by the patch are simultaneously restarted on all the PowerScale nodes in the cluster, this will affect the specified services across the entire cluster causing temporary service disruption. For a simultaneous patch request, the PowerScale nodes will not be rebooted.

When the parameter `--rolling=true` is set, a rolling patch upgrade request is made. In this case, the patch will be installed and the node will be rebooted in succession. For rolling patch upgrade request, the specified services will not be restarted. Instead, NDU will migrate all user connections away from the nodes before starting the patch request. This migration process can be disruptive. NFS with dynamic pool and SMB CA can help to make this migration transparent to the client application. For other workloads, they will get disconnected and need to re-establish the connection when the node they are connected to reboots.

Simultaneous patch upgrade request can save time and act more efficient, but the specified service will be restarted which causes a temporary service disruption. At the same time, rolling patch upgrade request is the default setting and can take much longer especially when the PowerScale cluster is large. But, it can be less disruptive with the combination of NFS dynamic pool or SMB CA.

The guideline is for the specified service which will not impact the workload like WebUI, PAPI and etc, use the parameter `--rolling=false` for the patch upgrade. The will make the upgrade more efficient and will not impact the real workload. For the service which can impact the workload like NFS, SMB and etc, use the parameter `--rolling=true` with the combination of NFS dynamic IP pool or SMB CA to minimize the impact to the client application.
2. In the case of a patch requiring PowerScale node reboots (kernel patch), it follows the way as below:

When the parameter `--rolling=false` is set, a simultaneous patch request is made and a patch will be installed simultaneously across all nodes. And in this case, it will reboot all of the nodes in the cluster simultaneously.

When the parameter `--rolling=true` is set, a rolling patch upgrade request is made. In this case, it will install the patch and then reboot each node in succession.

The guideline is to use `--rolling=true` to have minimal impact on the workload. However, if customers are willing to have a maintenance window with the disconnected workload, use `--rolling=false` to make this patch upgrade more efficient.

Starting from OneFS 8.2.0, the parameter `--rolling` is no more and now it uses `--simultaneous` for the same purpose. It is very important to carefully read the Readme file for each patch, which explains the behavior of the patch installation process and its impact in details.

### 3.4 Installation of patches during a OneFS upgrade

Starting with version 8.2.0, OneFS supports automatically installing a patch during a OneFS upgrade. Use the newly added parameter `--patch-paths` of `isi upgrade start` to include a patch to install when staring a OneFS upgrade. The following is an example:

```
isi upgrade start --install-image-path=<OneFS image path> --patch-paths=<patch path>
```

There are some considerations to be aware of:

- This parameter only takes a single patch to be installed during the upgrade.
- The path of the patch should be within `/ifs`.
- The installation of the patch happens only after each node is reboot into a new version but before it goes to COMMIT states. This means it may require a second reboot for patch installation depending on which category the patch falls in. If it is a kernel patch, it then requires a second reboot.
- Once it is triggered, the patch will be listed in `isi upgrade patches list` with a partial status until fully installed on all nodes. See the example below:

```
tme-sandbox-2# isi upgrade patches list
Patch Name   Description         Status
----------------------------------------
hp-provision Halfpipe test files Partial
----------------------------------------
Total: 1
```

After the patch has been installed successfully on all nodes, the status will be changed to installed.

```
tme-sandbox-2# isi upgrade patches list
Patch Name   Description         Status
------------------------------------------
hp-provision Halfpipe test files Installed
------------------------------------------
```
Patch upgrade

- Before OneFS 8.2.1, the parameter `--patch-paths` can only take a single patch. If there is a requirement to install additional patches during the OneFS upgrade process, it is recommended to use `isi_patch register` command to achieve this.

**Note:** Dell EMC recommends adding additional patches be requested after upgrade has been started but before any nodes have been upgraded before commit. Otherwise it only applies to the remaining nodes. The nodes already upgraded will need to be reconciled manually on each node.

The following is an example:

Start an OneFS upgrade using the command below.

```
isi upgrade start --nodes=1 --install-image-path=/ifs/PipelineBeta/OneFS_v8.2.0.BETA.2_Install.tar.gz --skip-optional
```

Before any PowerScale node gets reboot, run the following two commands to add to additional patch during the OneFS upgrade process:

```
isi_patch register /ifs/PipelineBeta/hp-provision.pkg
isi_patch register /ifs/PipelineBeta/hp-base.pkg
```

The two patches are added to the patch list with partial status:

```
tme-sandbox-3# isi upgrade patches list
Patch Name   Description                    Status
---------------------------------------------------
hp-base       Patch interaction - Base patch Partial
hp-provision  Halfpipe test files            Partial
---------------------------------------------------
```

After the patches have been installed successfully on all PowerScale nodes, the status will be changed to `Installed`.

```
tme-sandbox-2# isi upgrade patches list
Patch Name   Description                    Status
---------------------------------------------------
hp-base       Patch interaction - Base patch Installed
hp-provision  Halfpipe test files            Installed
---------------------------------------------------
```

- Starting from OneFS 8.2.1, the parameter `--patch-paths` can take multiple patches and install them during the OneFS upgrade.
- After the rollback, the patches which have been installed during the OneFS upgrade will automatically be uninstalled.

### 3.5 Simplified patch installation process

Starting from OneFS 8.2.1, the patch installation process has been simplified and enhanced. Prior to OneFS 8.2.1, you will see the following behaviors when you apply a new RUP on top of an existing one:

- The previous patch has to be uninstalled first and then you can install the new patch no matter it's for kernel or userspace, DA or GA, RUP or singleton.
Patch upgrade

- Multiple reboots or service restart will be required during this process.
- There will be a less protected time window that in the middle of the process where existing security fixes in the current patch will be uninstalled.
- Patches are not able to patch patch-system.

In OneFS 8.2.1, the simplified patch installation process includes the followings:

- Support installing the patch without uninstalling the previous one.
- Reduce node reboot to only once.
- Reduce service restart to only once.
- In most cases, eliminate the less protected window in the upgrade process.
- Support patching patch system

**Note:** This process cannot apply to firmware or OneFS upgrades.

The following is an example to explain how this feature works:

In this example, the test patch, patch-1234, has been installed as shown below:

```bash
# isi upgrade patches list
Patch Name  Description                  Status
--------------------------------------------------
patch-1234  base patch to be deprecated. Installed
--------------------------------------------------
```

Install a new patch, test-deprecated-patch, which is to deprecate the previous one patch-1234. To do this, just leverage the command `isi upgrade patches install` as shown in the following:

```bash
# isi upgrade patches install /ifs/test-deprecated-patch.pkg
The below patches are deprecated by this patch and will be removed automatically:
- patch-1234
Would you like to proceed? (yes/[no]): yes
Requested install of patch test-deprecated-patch.
```

In this case, it will automatically detect an existing deprecated patch installed in the cluster and ask you if you want to proceed to remove it.

During the installation, the status of patch-1234 is **AdHoc** as shown below.

```bash
# isi upgrade patch list
Patch Name            Description                  Status
----------------------------------------------------------
patch-1234            base patch to be deprecated. AdHoc
----------------------
```

**AdHoc** status means:

- This patch is to be deprecated and removed by a new patch installation.
- This patch is still taking effect on the PowerScale cluster
After the new patch has been installed and the deprecated patch has been removed, the patch status will change to the followings:

```
# isi upgrade patch list
Patch Name          Description                   Status
---------------------------------------------------------
test-deprecated.patch patch with deprecated tag       Installed
```

This feature can dramatically simplify the upgrade process by removing unnecessary steps, node reboots, and service restart. As a result, it makes the patch upgrade process more efficient and secure. Based on this, Dell EMC recommends you to leverage this feature when it applies.
4 Firmware upgrade

Prior to OneFS 8.2.0, the only way to upgrade node firmware was through a rolling (serial) firmware upgrade process. Serial firmware upgrades have the least impact to business continuity, yet require extended maintenance windows that increase linearly with cluster size.

In OneFS 8.2.0, a new parameter --simultaneous is introduced for the CLI command isi upgrade firmware. This parameter along with -nodes-to-upgrade allows upgrading firmware on multiple PowerScale nodes simultaneously. OneFS will check if the simultaneous upgrade parameter is valid or not. The following is an example where node 1 and node 2 are PowerScale node pairs. OneFS will stop the upgrade process when there is an attempt to start a simultaneous firmware upgrade for these two nodes. This is to ensure data integrity and we do not update peer nodes simultaneously.

```
# isi upgrade cluster firmware start --no-verify --no-burn --simultaneous --nodes-to-upgrade=1,2
You are about to start a Simultaneous Firmware UPGRADE, are you sure? (yes/[no]): yes
Invalid nodes specified for simultaneous upgrade. Please run isi_upgrade_helper for possible valid commands
```

To support simultaneous firmware upgrades, a new tool, isi_upgrade_helper, is also included in the OneFS 8.2.0. This tool can help end users decide how to use the newly introduced simultaneous firmware upgrade mechanism to meet their business requirements. This tool will give 3 firmware upgrade recommendations as the followings:

- Least Disruptive Firmware Upgrade Recommendation
- Fastest Firmware Upgrade Recommendation
- Balanced Firmware Upgrade Recommendation

Within each recommendation, it also includes the corresponding CLI commands which can be used directly for firmware upgrade.

4.1 Least Disruptive Firmware Upgrade Recommendation

This option is to upgrade one PowerScale node at a time, which causes the lowest impact to availability and performance during the firmware upgrade process. It has no difference with the firmware upgrade mechanism prior to OneFS 8.2, which will take a long time for a large PowerScale cluster. The following command is used for this option:

```
isu upgrade cluster firmware start
```

4.2 Fastest Firmware Upgrade Recommendation

This recommendation which is also known as simultaneous firmware upgrade is to ensure the data integrity during the firmware upgrade process by preventing any Gen 6 Node Pairs from being upgraded concurrently. For example, the simultaneous firmware upgrade will be run in the following sequence:

```
isu upgrade cluster firmware start --simultaneous --nodes-to-upgrade <odd number of slots>
isu upgrade cluster firmware start --simultaneous --nodes-to-upgrade <even number of slots>
```
4.3 Balanced Firmware Upgrade Recommendation

This recommendation which is also known as intermediate firmware upgrade is a balance between simultaneous and rolling firmware upgrade. It runs faster than a rolling upgrade but is less intrusive than a simultaneous firmware upgrade. The recommendation adheres to the following restrictions:

- No more than 1 node per Disk Pool is added to the firmware upgrade list
- No more than 1 node per Gen 6 Node Pair is added to the firmware upgrade list
- PowerScale nodes without redundant power suppliers are not added to the firmware upgrade list

To check the power redundancy for all the PowerScale node in the cluster, use the following command:

`isi_for_array isi_hw_status | grep "Power Supp"`

An example is as below which indicates all the 4 nodes in the PowerScale cluster have redundant power supplies.

hop-isi-n-4: Power Supplies OK
hop-isi-n-4: Power Supply Slot3-PS0 good
hop-isi-n-4: Power Supply Slot4-PS1 good
hop-isi-n-2: Power Supplies OK
hop-isi-n-2: Power Supply Slot1-PS0 good
hop-isi-n-2: Power Supply Slot2-PS1 good
hop-isi-n-3: Power Supplies OK
hop-isi-n-3: Power Supply Slot3-PS0 good
hop-isi-n-3: Power Supply Slot4-PS1 good
hop-isi-n-1: Power Supplies OK
hop-isi-n-1: Power Supply Slot1-PS0 good
hop-isi-n-1: Power Supply Slot2-PS1 good

This recommendation will ensure there is minimal impact during the simultaneous upgrade process.
### 4.4 Workflow

The overall workflow to upgrade firmware for PowerScale is shown in Figure 11. The rolling firmware upgrade part is only for the PowerScale OneFS prior to 8.2.0. For OneFS 8.2.0, all the three options, rolling, simultaneous and intermediate firmware upgrade are available.

![Firmware upgrade workflows](image)

**Step 1:** It is required to install a firmware package prior to being able to upgrade. Starting from OneFS 8.0, use `isi upgrade patches` for this purpose. The following is an example:

```
isi upgrade patches install --simultaneous <firmware path>
```

**Note:** Since no reboot or service restart required for firmware patch install, Dell EMC recommends using simultaneous parameter to go faster.

**Step 2:** Prior to OneFS 8.2.0, the only option is to run a rolling firmware upgrade. In OneFS 8.2.0, a new tool, `isi_upgrade_helper` is introduced. You can use the following command to leverage to tool to give recommendations:

```
isi_upgrade_helper
```

The output of this tool includes three recommendations and the corresponding CLI command as introduced in the previous section.

- Least Disruptive Firmware Upgrade Recommendation
- Fastest Firmware Upgrade Recommendation
- Balanced Firmware Upgrade Recommendation

**Step 3:** Choose a recommendation from the helper tool and execute the firmware upgrade command according to the recommendation.
If executing a simultaneous firmware upgrade, repeat command: `isi_upgrade_helper` once firmware upgrade is complete to obtain next set of nodes to upgrade. Repeat until no firmware mismatches remain. To check the firmware mismatch, use the following commands and exam the Mismatch column.

```bash
isi upgrade firmware devices
```

<table>
<thead>
<tr>
<th>Device</th>
<th>Type</th>
<th>Firmware</th>
<th>Mismatch</th>
<th>Lnns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mellanox-EN1</td>
<td>40GigE</td>
<td>2.40.5030+EMC1090111023</td>
<td>No</td>
<td>1-4</td>
</tr>
<tr>
<td>Mellanox-ENO</td>
<td>40GigE</td>
<td>2.40.5030+EMC1090111023</td>
<td>No</td>
<td>1-4</td>
</tr>
<tr>
<td>EPspime_warthog</td>
<td>ePOST</td>
<td>07.00</td>
<td>No</td>
<td>1-4</td>
</tr>
<tr>
<td>EPadpt_moons</td>
<td>ePOST</td>
<td>01.50</td>
<td>No</td>
<td>1-4</td>
</tr>
<tr>
<td>EPbios_warthog</td>
<td>ePOST</td>
<td>28.14</td>
<td>No</td>
<td>1-4</td>
</tr>
<tr>
<td>EPps0_gen2_artesyn</td>
<td>ePOST</td>
<td>02.14</td>
<td>No</td>
<td>1,3</td>
</tr>
<tr>
<td>EPvrd0_warthog_stmicro</td>
<td>ePOST</td>
<td>01.01</td>
<td>No</td>
<td>1-4</td>
</tr>
<tr>
<td>EPbcc_infinity</td>
<td>ePOST</td>
<td>00.00</td>
<td>Yes</td>
<td>3</td>
</tr>
<tr>
<td>EPvrd1_warthog_stmicro</td>
<td>ePOST</td>
<td>01.01</td>
<td>No</td>
<td>1-4</td>
</tr>
</tbody>
</table>

**Note:** You may see Upgrade framework is re-gathering firmware status, please try again as the response of `isi upgrade firmware devices`. This is normal behavior.

### 4.5 General recommendations

Here list several general best practices and configuration considerations for PowerScale firmware upgrade:

- Always consult Dell EMC Remote Proactive for PowerScale node firmware upgrade.
- Perform initial simultaneous firmware upgrade on a subset of the PowerScale nodes before moving onto the remainder of the cluster.
- For **Fastest Firmware Upgrade Recommendation**, it is not recommended for a production environment with customer data.
- For **Balanced Firmware Upgrade Recommendation**, it is recommended for a production environment with customer data.
Troubleshooting

5 Troubleshooting

This section explains the troubleshooting methodologies for the following upgrades:

- OneFS upgrade
- Patch upgrade
- Firmware upgrade

5.1 OneFS upgrade

This section will focus on the guideline and methodologies of troubleshooting OneFS upgrade issues.

5.1.1 Monitor the OneFS upgrade

To monitor the high-level OneFS upgrade status use the following command

```shell
isi upgrade view -interactive
```

Starting from OneFS 8.2.0, it is possible to detect upgrade hangs. If an OneFS upgrade is not making any progress after 15 minutes, a notification will be sent by the CELOG. In general, CELOG Alerts will be generated for 3 causes of upgrade hangs:

- PowerScale node is down or not responding over backend network.
- Upgrade agent is not ready to execute the upgrade command.
- Upgrade hook takes long time to complete and can get stuck into start state for indefinite amount of time.

For the above 3 reasons, there are 3 newly introduced CELOG event as shown in the followings:

- **400150007**: Upgrade Hang – unable to communicate with Upgrade Agent on devids: {devids}
- **400150006**: Upgrade Hang – Upgrade Agent unable to make progress on devids: {devids}
- **400150008**: Upgrade Hook Hang – {hook} on devids: {devids}

Use the following command to check the event list for unresolved Upgrade Hang events:

```shell
isi event events list
```

5.1.2 OneFS upgrade logs

The location of the OneFS upgrade logs varies between MRs. They can be categorized into two classes:

**Pre 8.1.1.0**

- Error logs are at /var/log/upgrade.log
- Verbose logs are at /ifs/.ifsvar/upgrade/Agent-* and /ifs/.ifsvar/upgrade/Supervisor-
- Pre-upgrade logs and hook script logs can be found at /var/ifs/upgrade/logs/hook-logs
- Use `isi_upgrade_logs` to gather all the logs when an error is reported by `isi upgrade view`
**Troubleshooting**

### 8.1.1.0 and later

- All the logs are at /ifs/.ifsvar/upgrade/logs directory
- Use `isi_upgrade_logs` to display errors reported by `isi upgrade view`

Starting from OneFS 8.2.0, `isi_upgrade_logs` supports to filter and extract for specific upgrade logging information. The following list shows the supported filters. The filters can be used in combination.

- `--guid` - dump the logs for the node with the supplied guid
- `--devid` - dump the logs for the node/s with the supplied devid/s
- `--lnn` - dump the logs for the node/s with the supplied lnn/s
- `--process` - dump the logs for the node with the supplied process name
- `--level` - dump the logs for the supplied level range
- `--time` - dump the logs for the supplied time range
- `--metadata` - dump the logs matching the supplied regex

An example is shown below to extract all the upgrade log generated by the process `isi_upgrade_agent_d` on PowerScale node 3 and 4:

```
isi_upgrade_logs --lnn=3,4 --process=/usr/bin/isi_upgrade_agent_d
```

**Note:** When upgrading into 8.1.1.0 or later release from a pre 8.1.1.0 release, the logging will follow the old format.

### 5.1.3 Failure handling

In case you encounter some errors during the OneFS upgrade process, it is recommended to investigate the latest upgrade log first. After you determine the root cause and get the issue fixed, you can use the following command to retry the last upgrade action.

```
tme-sandbox-2# isi upgrade retry-last-action all
```

If it does not work, use the following command to roll back the upgrade:

```
tme-sandbox-2# isi upgrade rollback
```

**Note:** Rollback can lead to a simultaneous reboot process of all the nodes in the PowerScale cluster. Be careful to initiate OneFS rollback.

### 5.2 Patch upgrade

This section focuses on the guideline and methodologies of troubleshooting patch upgrade issues.
5.2.1 Monitor the patch upgrade

Since patch system activity is asynchronous, you will drop back to the command prompt immediately after issuing a patch system request. You can monitor the status of the patch installation or uninstallation by using the following commands:

```shell
 tme-sandbox-2# isi upgrade node list
 tme-sandbox-2# isi upgrade cluster view
```

You can also use the following commands to view the detailed messages of patch upgrade:

```shell
isi_upgrade_status
```

To collect the log bundles of patch upgrade, use the following command:

```shell
isi_upgrade_logs
```

5.2.2 Failure handling

It may be possible for a patch request to fail for some reasons. When this happens, both the patch system and the NDU framework will remain in use until either the patch request is completed or aborted. Both of these options can be completed using the following command:

1. The best course of action in case of failure is to examine the logs for both the NDU framework and the patch system to try and determine the root cause of the failure. If the issue can be corrected, the patch request can be restarted using the following command:

   a. To retry the last patch upgrade action on all the PowerScale nodes, use the following command:

      ```shell
      tme-sandbox-2# isi upgrade retry-last-action all
      ```

   b. To retry the last patch upgrade action on PowerScale node 2,4 and 6, use the following command:

      ```shell
      tme-sandbox-2# isi upgrade retry-last-action 2,4,6
      ```

2. In rare circumstances, it may not be possible to correct for the failure and the only option will be to abort. In this case, it is necessary to abort the patch request using the following command:

   ```shell
   tme-sandbox-2# isi upgrade patches abort
   ```

   Once this is completed, you can use the following command to archive and clean up the NDU status:

   ```shell
   tme-sandbox-2# isi upgrade cluster archive --clear
   ```

**Note:** After the commands in option 2 have completed, the patch may still be installed on some nodes. This is likely if you were doing a rolling patch install and it failed in the middle. If you wish to uninstall from the remaining nodes, just run a normal patch uninstall command: `isi upgrade patches uninstall <patch name>`. 
5.3 Firmware upgrade

This section focuses on the guideline and methodologies of troubleshooting firmware upgrade issues.

For isi_upgrade_helper, the logs are located on the PowerScale node where the tool is run at /var/log/isi_upgrade_helper.log. It is recommended to add --debug parameter for verbose log information.

To monitor the upgrade process, use the following command:

isi upgrade view

The behaviors are different between rolling firmware upgrade and simultaneous firmware upgrade. The details are as the followings:

- PowerScale nodes that are selected for rolling firmware upgrade will quickly proceed through:
  a. Committed status
  b. Upgrade Ready status
  c. Committed status

- PowerScale nodes that are selected for simultaneous firmware upgrade will proceed through the following stats and it will run much slower:
  a. Committed status
  b. Upgrade Ready status
  c. Non-responsive (Rebooting)
  d. Upgrade Ready status
  e. Committed status
A  Technical support and resources

Dell.com/support is focused on meeting customer needs with proven services and support.

Storage technical documents and videos provide expertise that helps to ensure customer success on Dell EMC storage platforms.

A.1  Related resources

OneFS Technical Overview
OneFS 8.1.0 CLI Administration Guide
OneFS 8.1.0 CLI Command Reference
OneFS 8.1.0 Web Administration Guide