

Patch Installation Guide

MapR patches are version-specific and cumulative. Each patch contains the code fixes that were included in the previous patch for that MapR version.

MapR version-specific patches are available here: <http://package.mapr.com/patches/releases/>

Applying a patch is a 3 step process:

- [Step 1: Verify Cluster Readiness for a Patch](#)
- [Step 2: Apply the Patch to Data Nodes](#)
- [Step 3: Apply the Patch to CLDB Nodes](#)

When you apply a patch to the cluster, the patched files along with original files (non-patched) are copied to the `/opt/mapr/.patch` folder. The file ending with `.O` is the original file (non-patched), and the file ending with `.<patch_number>` is the patched version. For example, if there is a file under `/opt/mapr/.patch/lib/`, you can compare that with the corresponding file under `/opt/mapr/lib/` by using the `md5sum` command to verify that the patch was successfully deployed.

Contact support@mapr.com if you need more information or if you face any problems with patch installation.



A patch for a given software version can be removed, and an older patch for the same MapR software version can be installed. However, rolling back a cluster from a newer MapR version to an older version is not supported.

For information on the bugs fixed in the patch:

- Go to the MapR Customer Support web portal at www.mapr.com/support. After logging into the portal, click the "Patches" tab and choose the appropriate MapR version to view the list of bug fixes. You will then see the patch name for the fixes displayed.
- View a copy of the latest [Patch Release Notes](#).

Step 1: Verify Cluster Readiness for a Patch

Before you apply a patch, check that the cluster is ready for a patch to be applied. In addition to the prerequisites, consider verifying that the cluster utilizes best practices which will facilitate a more optimal patch installation.

Patch Installation Prerequisites

Before you apply a patch on the cluster, verify that all CLDB nodes are running and that container 1 is fully replicated on each CLDB node.

```
Run maprccli dump containerinfo -ids 1 -json
```

In the output, all CLDBs should be listed under `ActiveServers` and each node should report a `VALID` state.

For example:

```

...
"data":[
  {
    "ContainerId":1,
    "Epoch":3,
    "Master":"<masterCLDB_IP>:5660--3-VALID",
    "ActiveServers":{
      "IP:Port":[
        "<masterCLDB_IP>:5660--3-VALID",
        "<slaveCLDB_IP>:5660--3-VALID",
        "<slaveCLDB_IP>:5660--3-VALID"
      ]
    },
    "InactiveServers":{
    },
    "UnusedServers":{
    },
  },
],
...

```

 RESYNC state will display when container 1 is not fully replicated on that node. You must wait until each CLDB node has a VALID state for container 1 before proceeding with the patch installation.

For more information, see [dump containerinfo](#).

Best Practices for Patch Installation

Failure to follow the best practices may, in some cases, impact the speed in which the patch installation completes. Check to see if your cluster abides by the following best practices:

- The volume min replication setting should be greater than or equal to 2 for CLDB volume.**
 This ensures that container 1 always has at least two valid copies.
 Run `maprcli dump volumeinfo -volumename mapr.cldb.internal -json`
 In the output, the "VolumeMinReplication" parameter lists the current replication setting for the named volume. For more information, see [maprcli dump volumeinfo](#).
- No under replicated volumes should exist on the cluster.**
 Run the following command to check for under-replicated volumes: `maprcli alarm list`
 For more information, see [alarm list](#).
- Each CLDB node should be configured to have a minimum of 3 disks in its storage pool.**
 Run the following command on each CLDB node to get a list of the disks configured for each storage pool:
`mrconfig sp list [-v]`
 In this example output, there are three disks associated with SP1:

```

[root@atsqa6c51 ~]# /opt/mapr/server/mrconfig sp list -v
ListSPs resp: status 0:2
No. of SPs (2), totalsize 4562260 MB, totalfree 4537550 MB
SP 0: name SP1, Online, size 2736933 MB, free 2724749 MB, path /dev/sdb, log 200
MB, port 5660, guid a3055a6db41f285b005883bbd701c1e5, clusterUuid
-5009075714600063565-100367519220387605, disks /dev/sdb /dev/sdd /dev/sde

```

For more information, see [mrconfig sp list](#).

Step 2: Apply the Patch to Data Nodes

Apply the patch to nodes dedicated to storing and processing data prior to applying the patch on nodes that run the CLDB. This includes nodes that run the Fileserver for storage and processing components such as the Node Manager and the HBase client.

For clusters with more than 100 data nodes, it is a best practice to apply the patch in batches. Also, wait a few minutes before proceeding to the next batch of nodes.

Complete the following steps on each data node:

1. Stop the MapR Warden and ZooKeeper (if installed) services:
 - a. To stop MapR Warden, run the following command:

```
sudo serice mapr-warden stop
```

- b. If ZooKeeper is installed on the node, run the following command:

```
sudo service mapr-zookeeper stop
```

2. If there is already a patch installed on the cluster, run the following commands to uninstall it:

CentOS/Redhat

```
sudo rpm -e mapr-patch
```

SUSE

```
sudo zypper remove mapr-patch
```

Ubuntu

```
sudo apt-get -y remove mapr-patch
```

3. Install the patch using one of the following commands:

CentOS/Redhat

```
sudo rpm -ivh mapr-patch-<new_patch_number>.rpm
```

SUSE

```
sudo zypper install mapr-patch-<new_patch_number>.rpm
```

Ubuntu

```
sudo dpkg -i mapr-patch<new_patch_number>.deb
```

4. Start the MapR Warden and ZooKeeper (if installed) services:
 - a. If ZooKeeper is installed on the node, run this command to start ZooKeeper:

```
sudo service mapr-zookeeper start
```

- b. To start Warden, use this command:

```
sudo service mapr-warden start
```

5. To verify that the patch was installed successfully, run the following commands:

CentOS/Redhat or SUSE

```
sudo rpm -ql mapr-patch-<new_patch_number>
```

Ubuntu

```
sudo dpkg -l | grep mapr-patch-<new_patch_number>
```

Step 3: Apply the Patch to CLDB Nodes

Apply the patch to CLDB slave nodes prior to applying the patch on the master CLDB node. After you apply a patch to a CLDB node you must verify that container 1 is fully replicated before proceeding to apply the patch to the next CLDB node.

For large clusters with many containers, when you do not patch CLDB nodes in the prescribed order, there may be a considerable delay before the cluster can process client operations. For smaller clusters, this is not as critical as the cluster can generally start accepting client operations in about 5 minutes.

Complete the following steps of each CLDB slave node and then on the CLDB master node:

1. Stop the MapR Warden and ZooKeeper (if installed) services:
 - a. To stop MapR Warden, run the following command:

```
sudo service mapr-warden stop
```

- b. If ZooKeeper is installed on the node, run the following command:

```
sudo service mapr-zookeeper stop
```

2. If there is already a patch installed on the cluster, run the following commands to uninstall it:

CentOS/Redhat

```
sudo rpm -e mapr-patch
```

SUSE

```
sudo zypper remove mapr-patch
```

Ubuntu

```
sudo apt-get -y remove mapr-patch
```

3. Install the patch using one of the following commands:

CentOS/Redhat

```
sudo rpm -ivh mapr-patch-<new_patch_number>.rpm
```

SUSE

```
sudo zypper install mapr-patch-<new_patch_number>.rpm
```

Ubuntu

```
sudo dpkg -i mapr-patch<new_patch_number>.deb
```

4. Start the MapR Warden and ZooKeeper (if installed) services:
a. If ZooKeeper is installed on the node, run this command to start ZooKeeper:

```
sudo service mapr-zookeeper start
```

- b. To start Warden, use this command:

```
sudo service mapr-warden start
```

5. To verify that the patch was installed successfully, run the following commands:

CentOS/Redhat or SUSE

```
sudo rpm -ql mapr-patch-<new_patch_number>
```

Ubuntu

```
sudo dpkg -l | grep mapr-patch-<new_patch_number>
```

6. Verify that the CLDB node that you patched is running and that container 1 on that node is fully replicated.

Run `maprccli dump containerinfo -ids 1 -json`

In the output, the CLDB node that you just patched should be listed under `ActiveServers` and it should report a `VALID` state for container 1.

For example:

```
...
  "data": [
    {
      "ContainerId": 1,
      "Epoch": 3,
      "Master": "<masterCLDB_IP>:5660--3-VALID",
      "ActiveServers": {
        "IP:Port": [
          "<masterCLDB_IP>:5660--3-VALID",
          "<slaveCLDB_IP>:5660--3-VALID",
          "<slaveCLDB_IP>:5660--3-VALID"
        ]
      },
      "InactiveServers": {
      },
      "UnusedServers": {
      },
    }
  ],
  ...
```



The `RESYNC` state will display when container 1 is not fully replicated on that node. You must wait until the CLDB node that you just patched has a `VALID` state for container 1.

For more information, see [dump containerinfo](#).