

Step 1: Restart and Check Cluster Services

After upgrading MapR Core using either a manual offline or rolling upgrade method (not upgrading with the MapR Installer) and upgrading your Ecosystem components, configure and restart the cluster and services. This procedure is not applicable to the scripted rolling upgrade method.

This task is applicable only to manual offline and rolling upgrade methods. This procedure is done automatically with manual scripted upgrades.

Before restarting cluster services, upgrade any existing ecosystem packages to versions compatible with the upgraded MapR. See the [MapR Ecosystem Pack Matrix](http://maprdocs.mapr.com/home/InteropMatrix/r_MEP_52.html) (http://maprdocs.mapr.com/home/InteropMatrix/r_MEP_52.html) for more information.

This procedure configures and restarts the cluster and services, including Ecosystem components, remounts the NFS share, and checks that all packages have been upgraded on all nodes.

After finishing this procedure, run non-trivial health checks, such as performance benchmarks relevant the cluster's typical workload or a suite of common jobs. It is a good idea to run these types of checks when the cluster is idle. In this procedure, you configure each node in the cluster without changing the list of services that will run on the node. If you want to change the list of services, do so after completing the upgrade. After you have upgraded packages on all nodes, perform this procedure on all nodes to restart the cluster. Upon completion of this procedure MapR core services are running on all nodes.

1. On each node in the cluster, run `configure.sh` ([../ReferenceGuide/configure.sh.html#configure.sh](#)) with the `-R` option

```
# /opt/mapr/server/configure.sh -R
```
2. If ZooKeeper is installed on the node, start it.

```
# service mapr-zookeeper start
```
3. Start Warden.

```
# service mapr-warden start
```
4. Run a simple health-check targeting the file system and MapReduce services only. Address any issues or alerts that might have come up at this point.
5. Set the new cluster version in the `/opt/mapr/MapRBuildVersion` file by running the following command on any node in the cluster.

```
# maprcli config save -values {mapr.targetversion:"`cat /opt/mapr/MapRBuildVersion`"}
```

6. Verify the new cluster version.

For example:

```
# maprcli config load -keys mapr.targetversion
mapr.targetversion
5.2.0.xxxxx.GA
```

7. Remount the MapR NFS share.

The following example assumes that the cluster is mounted at `/mapr`:

```
# mount -o hard,nolock <hostname>:/mapr /mapr
```

8. Run commands, as shown in the example, to check that the packages have been upgraded successfully.

Check the following things:

- All expected nodes show up in a cluster node list, and the expected services are configured on each node.

- A master CLDB is active, and all nodes return the same result.
- Only one ZooKeeper service claims to be the ZooKeeper leader, and all other ZooKeepers are followers.

For example:

```
# maprcli node list -columns hostname,csvc
hostname configuredservice ip
centos55 tasktracker,hbmaster,cldb,fileserver,hoststats 10.10.82.55
centos56 tasktracker,hbregionserver,cldb,fileserver,hoststats 10.10.82.56
centos57 fileserver,tasktracker,hbregionserver,hoststats,jobtracker 10.10.82.57
centos58 fileserver,tasktracker,hbregionserver,webserver,nfs,hoststats,jobtracker 10.10.82.58
...more nodes...

# maprcli node cldbmaster
cldbmaster
ServerID: 8851109109619685455 HostName: centos56

# service mapr-zookeeper qstatus
JMX enabled by default
Using config: /opt/mapr/zookeeper/zookeeper-3.4.5/conf/zoo.cfg
Mode: follower
```