Disaster Recovery Guide

Version 4.2

February 2017

Contents

Chapter 1: About disaster recovery

Topics:

Disaster recovery is the ability to recover from a hardware or software failure or a catastrophic event. ThoughtSpot protects you from data loss in the event of a hardware or software failure or a catastrophic event.

ThoughtSpot takes snapshots of itself automatically at periodic intervals. These can be pulled out as backups at intervals or manually as needed. See the ThoughtSpot Administrator Guide for details on backups, snapshots and restore operations.

The information here addresses disaster recovery specifically. These are some potential types of failure, listed in increasing order of severity:

- Disk failure
- Node failure
- Cluster replacement

ThoughtSpot supports recovery from disk or node failure within each appliance. You can also architect your system to support loss of an entire appliance, which is the highest level of disaster recovery.

Disk failure

ThoughtSpot uses replication of stored data. When a disk goes bad, ThoughtSpot continues to operate.

Replacement of a bad disk should be initiated through ThoughtSpot Support in this event, at your earliest convenience.

Symptoms

You should suspect disk failure if you observe these symptoms:

- Performance degrades significantly.
- You receive alert emails beginning with WARNING or CRITICAL that contain DISK_ERROR in the subject.

If you notice these symptoms, contact ThoughtSpot Support.

Disk replacement

The guidelines for disk replacement are:

- Losing one or two disks: The cluster continues to operate, but you should replace the disk(s) at the earliest convenience.
- Losing more than two disks: The cluster continues to operate, but the application may be inaccessible. Replace the disks to restore original operation.

Disk replacement is done on site by ThoughtSpot Support. Disks can be replaced while ThoughtSpot is running. However the disk replacement procedure involves a node restart, so a disruption of up to five minutes can happen, depending on what services are running on that node.

Node failure

ThoughtSpot uses replication of stored data. When a disk goes bad, ThoughtSpot continues to operate.

To support high availability, your ThoughtSpot instance must have at least three nodes. In a three or more node system, if one node fails, its services will be distributed to the other nodes. The failover is automatic. However, when a node fails, you should contact ThoughtSpot Support about replacing the node when possible.

A node is considered to have failed when one or more of these conditions occur:

- Two or more disks have failed.
- SSD has failed.
- Memory failure.
- Another hardware component has failed (networking, motherboard, power supplies).

Symptoms

You should suspect node failure if you observe these symptoms:

- Performance degrades significantly.
- You receive alert emails beginning with WARNING or CRITICAL, that describe problems with one of the nodes not running.
- A node does not come up upon booting or rebooting the system.

If you notice these symptoms, contact ThoughtSpot Support.

Node replacement

Node replacement is done on site by ThoughtSpot Support. You will need to schedule a maintenance window, since some downtime is required. For more information, please contact ThoughtSpot Support.

Cluster replacement

Cluster replacement can be achieved using a mirrored system architecture. This allows you to recover an entire system very quickly without data loss.

You have the option of architecting your system for fast recovery from a disaster in which you lose an entire ThoughtSpot instance. This involves running two ThoughtSpot appliances in a mirrored configuration. This configuration is used in mission critical systems or for business processes in which ThoughtSpot data has been operationalized.

The two ThoughtSpot instances are called:

- Primary: The production ThoughtSpot instance.
- Mirror: A standby instance that can be placed into service in the event that the primary fails.

In this configuration, the primary initiates periodic full backups of itself. It pushes the backups to a shared NAS (network attached storage). The mirror instance pulls the backups from the shared NAS at defined intervals. It uses each new backup to restore itself to match the production cluster.

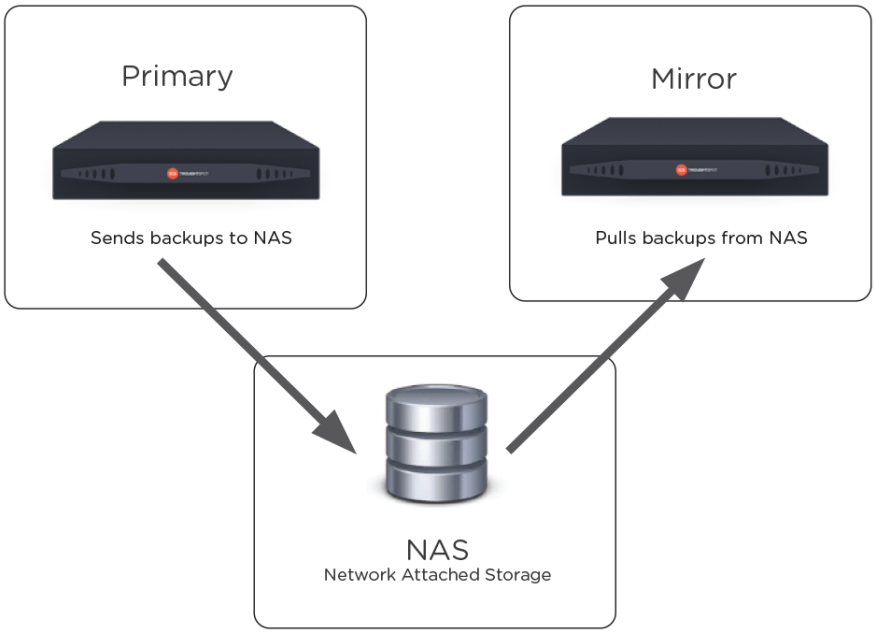

Figure 1: A ThoughtSpot disaster recovery configuration

Mount a NAS file system

Some operations, like backup/restore and data loading, require you to either read or write large files. You can mount a NAS (network attached storage) file system for these operations.

This procedure shows you how to mount a NAS file system for storing or accessing large files. The file system will be mounted at the same location on each node in the cluster automatically. When any node is restarted, the file system will be mounted again automatically, if it can be found.

When supplying a directory for writing or reading a backup, you can specify the mountpoint as the directory to use. Likewise, you can stage data there for loading.

Note that backups are written by the Linux user "admin". If that user does not have permission to write to the NAS file system, you could write the backups to

disk (for example /export/sdc1, /export/sdd1, /export/sde1, or /export/sdf1) and then set up a cron job that executes as root user and copies the backup to the NAS device every night, then deletes it from the directory.

Do not send the periodic backups or stage files on /export/sdb1 since it is a name node. It is used internally by Hadoop Distributed File System (HDFS) and if this drive fills up, it can cause serious problems. Do not allow backups or data files to accumulate on ThoughtSpot. If disk space becomes limited, the system will not function normally.

1. Log in to the Linux shell using SSH.
2. Mount the directory to the file system, by issuing the appropriate command:
   - For an NFS (Network File System) directory:

     ```
     tscli nas mount-nfs
        --server <server_NFS_address>
        --path_on_server <path>
        --mount_point <target>
     ```

   - For a CIFS (Common Internet File System) directory:

     ```
     tscli nas mount-cifs
        --server <server_CIFS_address>
        --path_on_server <path>
        --mount_point <target>
        --username <user>
        --password <password>
        --uid <uid>
        --gid <gid>
     ```

3. Use the mounted file system as you wish, specifying it by referring to its mount point.
4. When you are finished with it, you may optionally unmount the NAS file system:

   ```
   tscli nas unmount --dir <directory>
   ```

Set up a disaster recovery configuration

Use this procedure to set up a disaster recovery configuration with a primary and a mirror instance.

The disaster recovery setup uses periodic backups from the primary to a shared storage (NFS mounted drive or NAS). Note, if you do not use the tscli command to mount NAS, please make sure the NAS is mounted on all nodes of both clusters. When choosing times and frequencies for periodic backups, you should choose a reasonable frequency. Do not schedule backups too close together, since a backup cannot start when another backup is still running. Avoid backing up when the system is experiencing a heavy load, such as peak usage or a large data load.

This is the procedure for designating a primary and a mirror ThoughtSpot instance.

1. Ensure you have tscli on the target appliance. If not, please contact ThoughtSpot Support. In addition, the appliance should not be running a cluster, so if one exists, please contact ThoughtSpot Support to delete the cluster. A ThoughtSpot cluster should be up and running on the source appliance.
2. Log in to the primary appliance, and set it up to take periodic backups and write them to the shared backup directory (in a SAN or shared NFS-mounted drive):

   ```
   $ tscli backup set-periodic --at <hour1,hour2,...> --directory
    <shared_backup_directory> [--num_backups <num_backups>]
   ```

   For example:

   ```
   $ tscli backup set-periodic --at 01,17 --directory /mnt/thoughtspot_backups
    --num_backups 5
   ```

3. Run `tscli backup periodic-config` to check whether the periodic backup is set correctly.
4. Designate the mirror appliance cluster to act as a mirror:

   ```
   $ tscli backup start-mirror <shared_backup_directory>
    <mirror_node1_IP,mirror_node2_IP,...> <mirror_cluster_name>
    <mirror_cluster_id>
   ```

For example:

```
$ tscli backup start-mirror /mnt/thoughtspot_backups 192.168.2.111,
 192.168.2.112,192.168.2.113 thoughtspot_mirror thoughtspot-mirror-29543
```

5. It may take some time for the cluster to begin acting as a mirror. Issue the command to verify that the cluster has started running in mirror mode:

   ```
   $ tscli backup mirror-status
   ```

   You can use this command in the future to check whether the mirror cluster is up to date.

Recover a cluster from a disaster

If the primary cluster fails, the mirror cluster can take over its operations after a small manual intervention. The manual procedure makes the mirror instance into the primary.

In the event that the production cluster is destroyed, monitoring and alerting will notify the administrator. The administrator can then make the mirror into the new primary, by stopping it from pulling backups generated by the old primary. These steps define the mirror cluster as the new primary. Then a new mirror can be deployed.

1. If it is still running, disconnect the primary from the network.
2. Log in to the mirror, and issue the command to stop it from acting as a mirror:

   ```
   $ tscli backup stop-mirror
   ```

3. It may take some time for the mirror to finish recovering the last backup from the primary. Issue the command to verify that the cluster has stopped running in mirror mode before continuing:

   ```
   $ tscli backup mirror-status
   ```

4. Start up the periodic backups on the mirror (which is now the new primary):

   ```
   $ tscli backup set-periodic --at <hour1,hour2, ...> --directory
    <shared_backup_directory> [--num_backups <num_backups>]
   ```

5. Deploy a new mirror appliance when possible.