Tungsten Cluster Master Class

Intermediate: Provisioning Datasources

Matthew Lang

VP of Customer Success, Americas

Topics

In this short course, we will discuss:

- The `tprovision` tool
- Methods for the backup/restore process needed for provisioning
- Provisioning from a working replica
- Considerations when provisioning from a primary datasource
- Rebuild a site in Composite Active/Active (CAA) topology

tprovision

tprovision Overview

- tprovision allows you to easily provision a node from a remote host.
- Is run on the host to be provisioned.
- Runs the backup on the remote hosts specified by the source option.
- Copies the backup to the local server and restores it.
- Will recover replication services and have the newly provisioned node (re)join the cluster.
- Defaults to mysqldump, but can use xtrabackup, rsync, and mysqlclone.

```
shell@host3>  tprovision --source host2
```

- When run on host3, will provision from host host2.
- Best to run in a screen session!

`tprovision` Process

- Shut down target database and SHUN
- Backup any MySQL SSL certs if found in the MySQL data dir
- Remove target database (except when using `rsync` method)
- If using `msyqldump`, create empty database
- If using `mysqlclone`, verify/install clone plugin
- Perform the backup and streaming restore
- Restore SSL certs, if required
- Start MySQL
- Remove old THL

Backup Methods

`mysqldump`
The standard MySQL backup tool. `mysqldump` will be initiated on the source and streamed to the target.

`xtrabackup`
Online backup tool from Percona. The backup is initiated on the source and streamed to the target.

`mysqlclone`
MySQL v8+, the source is the “donor”, and the clone is steamed to the target. Requires plugin, `tproivsion` will install plugins if not already installed.

`rsync`
Initiated on the source. Sends changed database blocks at the filesystem level. Source database is quiesced, resulting in a viable database copy.

Considerations When Provisioning From a Primary Datasource

- The following backup methods WILL LOCK the database and can cause downtime:
  - `rsync`
  - `mysqldump`
- `xtrabackup` and `mysqlclone` will keep the database online and serving queries.
- The process of backing up the primary database will put additional load on the source host.
- Provision from a replica whenever possible.

A Closer Look at the `rsync` Method

- `rsync` takes advantage of the existing database on the target, even if it is corrupt.
- `rsync` does not remove the target database.
- `rsync` in done in 2 passes:
  - Pass 1:
    - Do an initial sync, in place, copying the changes at the block.
    - This ”seeds” the database files, so that pass 2 (below) is quicker.
    - Resulting database is not useable until Pass 2 is done.
  - Pass 2:
    - Quiesce the source database (now database is offline).
    - Do final sync. Now the target database is useable.
- Pass 1 can be done in advance to limit downtime during maintenance window specify `--seed`.
- Not recommended to use a primary as a `source`, to override, use `--i-am-sure`.

Main Options

`-s	--source`	Source database to provision from.
`-c	--create-master`	Create a new primary in CAA topology. Used when `reprovisioing` an entire site, to indicate the target node will be the primary in the service. Appropriate cross-site replication services will be reset.
`-t	--threads`	Specify `xtrabackup` threads OR simultaneous `rysnc` threads.
`-m	--method`	Backup method: `mysqldump`, `xtrabackup`, `mysqlclone`, `rsync`.
`-p	--port`	SSH port to source host, if not 22.
`-n	--dryrun`	Do not execute, just show what would be done.
`-h	--help`	Show help.

Rebuilding an Entire Dataservice in CAA Topology

- We can rebuild an entire dataservice using `tprovision`.
- First we choose which host will the primary, and provision it using the `-c` option.

```
shell@east1>  tprovision --source west2 -c
```

- The example above will create a new primary on host `east1` using the database on `west2` as the source.
- When creating a new primary, the local dataservice needs to be reset (`east` in the example), as well as the cross-site replicator services (`west_from_east` in the example). `tprovision` does all of this for you.
- After the primary has been provisioned, provision the replicas as usual.

Demo

Summary

- `tprovision` allows you to easily provision failed databases in the cluster with a single command.
- Various backup/restore methods are available.
- The `rsync` method can be potentially fast but not recommended when the source is the primary.
- We should use a utility like `screen` since the provision process may take a while.
- Lots of useful information is printed on the screen, all info is logged into `~CONTINUENT_ROOT/service_logs/tprovision.log`
- An entire site can be reprovisioned in CAA topology.

Thank you for listening

continuent.com

Matthew Lang

VP of Customer Success, Americas