TORQUE Resource Manager 4.0.1 Release Notes

May 2012

The release notes file contains the following sections:

Overview

TORQUE scales to newer, larger systems and can handle larger numbers of jobs, nodes, and commands per second. Some of the features TORQUE 4.0.1 offers include the following: pbs_server has improved request handling and is now multi-threaded, RPP/UDP is no longer used, mom reporting can be organized into hierarchies, job reporting can be divided into a tree structure, and many other smaller yet significant changes. The following two sections (New Feature Highlights and New Feature Details) provide more information about what TORQUE 4.0.1 has to offer.

New Feature Highlights

The following is a summary of key new features in TORQUE 4.0.1:

- Improved request handling in pbs_server
  - Dramatically increased number of requests TORQUE can handle
- Multi-threaded pbs_server
  - Faster response to client commands; no more waiting for someone else's call to qstat to finish before your request can run
  - Higher job throughput
  - More robust
- New trqauthd client daemon
  - trqauthd replaces pbs_iff
  - trqauthd is multi-threaded and able to reliably handle multiple simultaneous client requests
  - Run as root, giving administrators more control
- Communication
  - RPP (reliable packet protocol) removed
  - UDP removed

    - All communication done using TCP/IP
- Mom Hierarchy
    - Reduces overall compute node update traffic to pbs_server
    - Allows administrators to manage network communications
- Scalability
    - Ability to support over 15,000 compute nodes in a cluster
    - Job radix allows users to submit jobs with very large node needs

New Feature Details

This section contains detailed information pertaining to specific new features.

Improved request handling in pbs_server

Previous versions of TORQUE handled new requests on port 15001 in pbs_server one at a time. TORQUE 4.0.1 manages new requests using a listening thread which accepts each new connection as it happens and creates a new thread to process the request.

Multi-threaded pbs_server

pbs_server is multi-threaded starting with TORQUE 4.0. By default, the number of available threads is 2*(number of cores)+1. This value can be modified using a set of three server parameters: (1) min_threads, (2) max_threads, and (3) thread_idle_seconds. Each of these parameters take an integer as an argument and can be set using qmgr.

- min_threads: Sets the minimum number of threads that will be available in pbs_server
- max_threads: Sets the maximum number of threads that can be on pbs_server
- thread_idle_seconds: The number of seconds a thread can be idle before it is removed from the system threadpool. Note the number of threads will never fall below the minimum. thread_idle_seconds by default is -1. This indicates to TORQUE to never remove a thread from the threadpool.

By setting max_threads greater than min_threads, pbs_server is able to dynamically add threads as the server load increases. The thread_idle_seconds parameter is used to detect a drop in the server load and removes threads as they are idle for the number of seconds given in the parameter.

Setting max_threads equal to min_threads will keep the number of threads static.

For more information, see the PBS Server documentation.

Mom Hierarchy

The Mom Hierarchy is a new feature in TORQUE 4.0 which is designed to improve the efficiency of communications between pbs_server and the compute nodes (pbs_mom). By default, all compute nodes send status updates directly to pbs_server. As cluster sizes increase, the need to reduce the traffic and time required to keep the cluster status up-to-date becomes more important. The Mom Hierarchy allows administrators to configure compute nodes in a way where each node sends its status update information to another compute node. The compute nodes pass the information up a tree or hierarchy until eventually the information reaches a node that will pass the information directly to pbs_server.

Setting up the Mom Hierarchy

The name of the file that contains the configuration information is named mom_hierarchy. By default, it is located in the */var/spool/torque/server_priv* directory. The file uses an XML-like syntax as follows:

```
<path attr=val>
  <level attr=val> comma separated node list </level>
  <level attr=val> comma separated node list </level>
  ...
</path attr=val>
<path attr=val>
  <level attr=val> comma separated node list </level>
  ...
</path attr=val>
...
```

The `<path> </path>` tag pair identifies a group of compute nodes. The `<level> </level>` tag pair contains a comma-separated list of compute node names. Multiple paths can be defined with multiple levels within each path.

Within a `<path>` tag pair, the levels define the hierarchy. All nodes in the top level communicate directly with the server. All nodes in lower levels communicate to the first available node in the level directly above it. If the first node in the upper level goes down, the nodes in the subordinate level will then communicate to the next node in the upper level. If no nodes are available in an upper level, then the node communicates directly to the server.

If an upper level node has fallen out and then is back in again, the lower level nodes eventually find that the node is available and send their updates to that node.

For more information, see Setting up the MOM hierarchy.

trqauthd

trqauthd is a new daemon starting in TORQUE 4.0. It replaces pbs_iff which is used by TORQUE client utilities to authorize user connections to pbs_server. Unlike pbs_iff, which is executed by the TORQUE client utilities each time the utility is run, trqauthd is started once and remains resident. TORQUE client utilities then communicate with trqauthd on port 15005 on the loopback interface. Unlike pbs_iff, trqauthd is multi-threaded and is able to successfully handle a greater volume of simultaneous requests than pbs_iff.

Running trqauthd

trqauthd MUST be run as root. It must be running on any host where TORQUE client commands will execute.

By default, trqauthd is installed to */usr/local/sbin*.

trqauthd can be invoked directly from the command line or by using init.d scripts which are located in the *contrib/init.d* directory of the TORQUE source.

There are three init.d scripts for trqauthd in the *contrib/init.d* directory of the TORQUE source tree:

- debian.trqauthd: Used for the apt-based systems (Debian and Ubuntu are the most common variations of this)
- suse.trqauthd: Used for the rpm-based systems (Red Hat, SUSE, Scientific, CentOS, and Fedora are some common examples)
- rqauthd: An example for other packages' managers (anything that doesn't use rpm or apt)

Inside each of the scripts are the variables PBS_DAEMON and PBS_HOME. These two variables should be updated to match your TORQUE installation. PBS_DAEMON needs to point to the location of trqauthd. PBS_HOME needs to match your TORQUE installation. For more information about PBS_HOME, please see the TORQUE PBS_HOME documentation.

Choose the script that matches your dist system and copy it to */etc/init.d*. If needed, rename it to trqauthd.

To start the daemon, type: `/etc/init.d/trqauthd start`

To stop the daemon, type: `/etc/init.d/trqauthd stop`

You can also use the following: `service trqauthd start/stop`

For more information, see Configuring trqauthd for client commands.

Scalability

job_radix is a new option for submitting large jobs in TORQUE 4.0. It is an attribute of qsub -W.

```
qsub -l nodes=500 -W job_radix=3 job.sh
```

The purpose is to reduce network traffic between a job's sister nodes and its Mother Superior. By default, a Mother Superior manages a multi-node job by receiving all updates and stdout/stderr directly from each sister node and updating pbs_server. Job radix creates a maximum number of nodes with which the Mother Superior and resulting intermediate MOMs directly communicate. This means that only the specified number of nodes may send information directly to the Mother Superior and that only that same number of nodes may send information directly to each of those nodes. The tree continues across all of the job's nodes so that no single node communicates with more sister nodes than were specified by job_radix.

For more information, see Managing multi-node jobs.

Upgrading to TORQUE 4.0.1 and Backward Compatibility

TORQUE 4.0 (and 4.0.1) is not backward compatible with previous versions of TORQUE. When you upgrade to TORQUE 4.0[.1], all MOM and server daemons must be upgraded at the same time.

The job format is compatible between 4.0[.1] and previous versions of TORQUE. Any queued jobs will upgrade to the new version with the exception of job arrays in TORQUE 2.4 and earlier. It is not recommended to upgrade TORQUE while jobs are in a running state.

Because TORQUE 4.0[.1] has removed all use of UDP/IP and moved all communication to use TCP/IP, previous versions of TORQUE will not be able to communicate with the components of TORQUE 4.0[.1]. However, all files in the */var/spool/torque* ($TORQUE_HOME) directory and all subdirectories are forwardly compatible.

Job Arrays

Job arrays from TORQUE version 2.5 and 3.0 are compatible with TORQUE 4.0[.1]. Job arrays were introduced in TORQUE version 2.4 but modified in 2.5. If upgrading from TORQUE 2.4, you will need to make sure all job arrays are complete before upgrading.

serverdb

The pbs_server configuration is saved in the file *$TORQUE_HOME/server_priv/serverdb*. When TORQUE 4.0[.1] is run for the first time, this file will be converted from a binary file to an XML-like format. This format can be used by TORQUE versions 2.5 and 3.0, but not earlier versions. It is recommended that this file be backed up before moving to TORQUE 4.0[.1].

Upgrading

Because TORQUE 4.0[.1] will not communicate with previous versions of TORQUE, it is not possible to upgrade one component and not upgrade the others. Rolling upgrades will not work.

Before upgrading the system, all running jobs must complete. To prevent queued jobs from starting, nodes can

be set to offline or all queues can be disabled. Once all running jobs are complete, the upgrade can be made. Remember to allow any job arrays in version 2.4 to complete before upgrading. Queued array jobs will be lost.

System Requirements

The following software is required to run TORQUE 4.0.1:

- ANSI C compiler. The native C compiler is recommended if it is ANSI; otherwise use gcc.
- A fully POSIX make. If you are unable to "make" PBS with your make, we suggest using gmake from GNU.
- Tcl/Tk version 8 or higher if you plan to build the GUI portion of TORQUE or use a Tcl-based scheduler.

Installation Information

The directions to install and configure TORQUE are in chapter 1 of the TORQUE 4.0.1 Administrator Guide. Also note additional instructions in the PBS Administrators Guide and README.building_40.

Note that you may need to install libssl-dev in order for the source to make successfully. Specifically, the build system is looking for libssl.so and libcrypto.so. For non-RPM setups, you may need to make a symbolic link from the ssl and crypto libraries to the respective .so names.

Documentation

Technical Documentation

The online help for TORQUE 4.0.1 is available in HTML and PDF format.
