Trust Separation on the XC40 using PBS Pro

Sam Clarke

May 2017

Overview

- About the Met Office workload
- Trust zone design
- Node configuration
- Lustre implementation
- PBS Implementation
  - Use of hooks
  - Placement Sets
  - Cgroups
- Conclusions

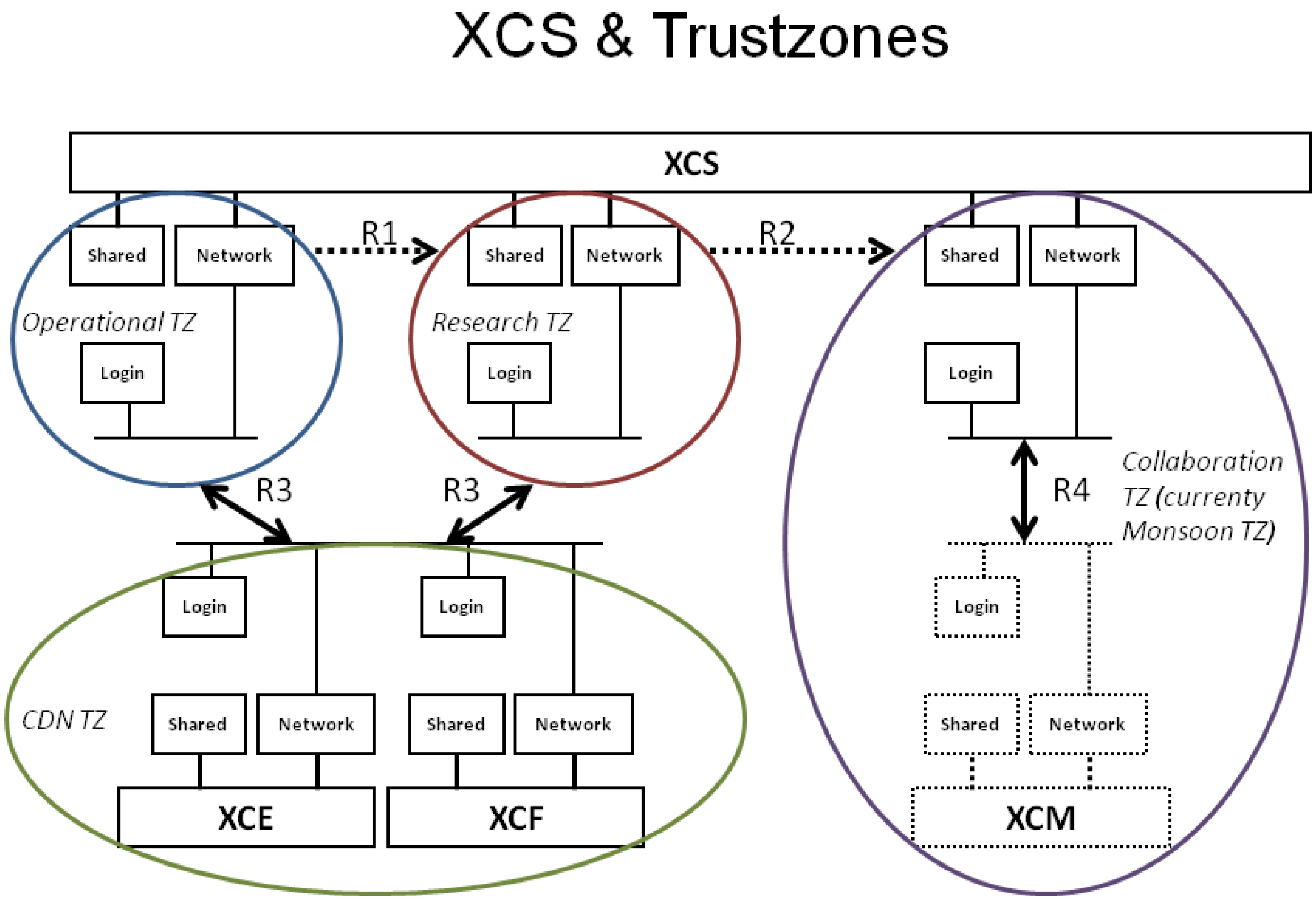
The Met Office Workload

- Three principal groups
  - Operational weather forecasting
  - Internal research
  - Collaborative research
- Previously groups were kept separate
  - Operational and research work on two internal XC40s
  - Collaboration on a smaller dedicated XC40
- Different levels of trust
  - Operational work is sensitive and time critical
  - Internal users are security cleared and trusted
  - External users access the systems from the internet

Trust Zone Design: Towards a Shared System

- Create a single system with three trust levels
- Limit file system access to specific levels
- Prevent traffic between levels
- Prevent tasks at different levels from interfering with each other
- Prevent tasks within a level from interfering with each other
- Provide a flexible configuration that allows resources to be dynamically moved between groups

Trust Zone Layout

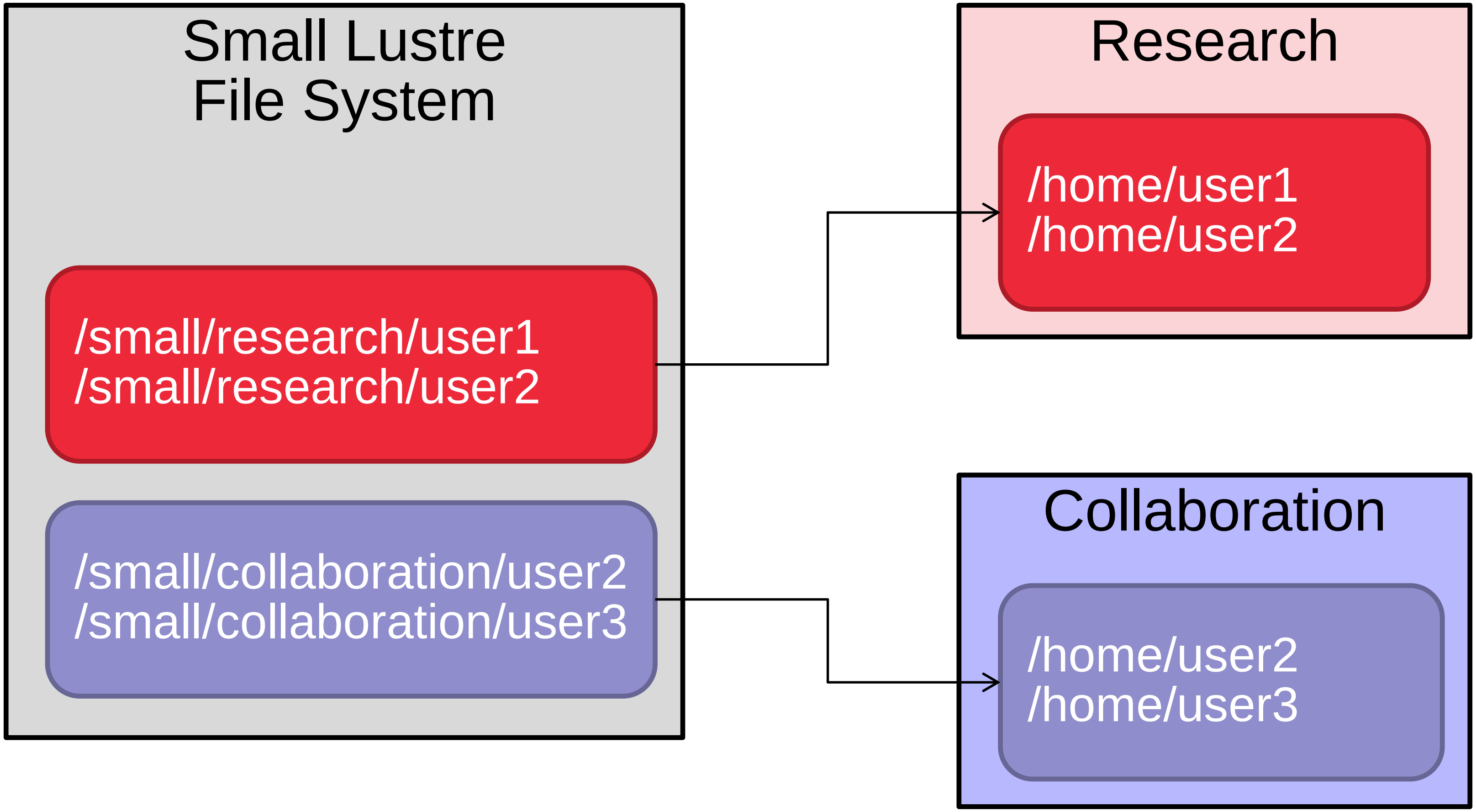


Lustre File Systems

- Two Sonexion 2000 Lustre appliances
  - Small one is 450TB and used for configuration files
  - Large one is 14PB and used for data and scratch files
- Partition using Unix ACLs
  - Create a restricted access directory tree for each group
  - Create a regular user file structure under each tree
- Create bind mounts into specific parts of the subtree
  - Bindings are static on eLogin, MOM and MAMU XC40 nodes
  - Bindings are dynamic on compute nodes

Lustre File Systems

Layout and Bindings



PBS Implementation

Job Submission

- Label the nodes
  - Create a string resource called trustzone
  - Set the trustzone resource on MOM and MAMU nodes
- Trust zone submit hook
  - When a new job is submitted on an eLogin
    - Examine the identity of the requestor host on the SDB
    - Use a static map to set the trustzone resource on the job
  - When a new job is submitted on a MOM or MAMU node
    - Examine the resource setting of the current node
    - Set the trustzone resource on the job to match the resource on the current node
  - Allocate a MOM node chunk to all compute jobs and set the trustzone resource

PBS Implementation

Job Scheduling

- Scheduler has been configured to use the trustzone resource
- Job waits for sufficient MOM or MAMU nodes with the trustzone resource to become free
- Job is dispatched to nodes where the trustzone resource matches

PBS Implementation

Job Execution

- On a MAMU node
  - Node has been statically configured to use a trust zone
  - No further action is required
- On a MOM node at start-up
  - Determine the set of compute nodes allocated to the job
  - Use pcmd to bind Lustre directories on each compute node
  - Configure any additional network routing
- On a MOM node at shutdown
  - Unbind the Lustre directories
  - Remove any network settings

PBS Implementation

Trust zone conclusions

- Scalability of compute job startup
  - Initially poor with machine-sized jobs taking many minutes to start
  - Resolved by merging pcmd mounts into a single command
- Placement sets and calendaring errors
  - Prevented anything other than large jobs from running
  - Caused by lack of placement resources on MOM nodes
  - Resolved by replacing placement set strings with string arrays and adding definitions to every MOM node

PBS Implementation

Cgroups

- Implemented in an internal hook provided by Altair
  - Initially tested cgroups with PBS 12
  - Started using in earnest on XCS at 13.0.401
- Initial problems involved stale cgroups
  - Clean-up hook did not run successfully because of a race condition between processes ending and the cgroup being removed
  - Resolved when Altair supplied an asynchronous version of the hook

PBS Implementation

Cgroup vnode timeouts

- Compute node vnode type timeouts
  - A test node was created with a non-standard vntype
  - All other vntypes were excluded using the cgroup configuration file
  - But hook ran unpredictably on other nodes in the system
- Caused by server load
  - Every cgroup invocation was requesting current vnode type from the PBS server
  - Server load caused the call to intermittently timeout
  - Timeout returned a None value which failed to match our exclude list
  - Workaround was to run on XCS with cgroups enabled on all vnodes
  - Resolved by Altair in PBS 13.0.406

PBS Implementation

Cgroup memory limits and file cache

- These constrain a node to a subset of the available memory
  - Memory limit applies to both compute memory and file cache
  - Causes IO-intensive operations to fail if there is insufficient memory to buffer the operation
  - Can be workaround around using direct IO but this is slow
- Impacted users doing recursive copies on MOM nodes
  - Workaround involved raising the standard memory limit on MOM chunk assigned to the job
- This is a standard feature of cgroups on Linux
  - There don’t seem to be any kernel tunables to change this behaviour

PBS Implementation

Cgroup runtime error handling

- Prior to PBS 13.0.406
  - Runtime failures resulted in jobs being placed on hold
  - Scheduler had no awareness of the problem so new jobs often assigned to same set of failing nodes
  - Often caused large numbers of jobs to go into limbo
- In 13.0.406
  - Node is taken offline when an error occurs
  - Periodic hook runs, attempts to fix problems with stuck cgroups, and puts node back online once problem has been resolved
  - Works well for MAMU nodes
  - But when an error occurs on a MOM node, it results in many compute vnodes being taken offline causing the system to drain

PBS Implementation

Cgroup conclusions

- Performance is improving
  - Hook is much better than the original version
  - Altair have worked hard to fix bugs and have been very responsive to problems
- Many edge conditions
  - Race conditions waiting for processes to end
  - Linux cgroup memory limits including file cache
  - Compute nodes accidentally taken offline
- Not ready for full production use yet
  - There is too much potential for forecast disruption
  - But plan to keep on trying on our XCS system

Conclusions

- Trust separation working well
  - Very few problems seen following implementation
  - System successfully accredited by external auditors
  - Collaboration users have access to a system capable of running jobs over 50 times larger than the original MONSooN XC40
- Lustre file system separation has been trouble-free
- Cgroups remain problematic
  - The hook seems to improve with every PBS release
  - We are confident the problems will be fixed or workaround eventually

Acknowledgements

- The Met Office HPC Systems Group
  - Helen Fairhurst, Doug Hayman, Joe Heaton, David Moore, Robin Pallister, and Will Wishart
- Cray on-site support staff
  - George Brown, Laurence Baldwin, and Maurizio Ianniciello
- Cray and Altair software support

Questions?