

XCache at Birmingham

Atlas DDM Ops Meeting, 10th June, 2019
Mark Slater, Birmingham University

I have now managed to setup Xcache at Bham as has been running on and off in production for a couple of weeks as the external config was tweaked

The hardware involved is:

- Basic server (16C, 24GB) to act as redirector
- Two 40TB pool nodes
- ~400 cores for Atlas → 50 simultaneous 8-core jobs
- Pool nodes are connected at 10Gb/s, workers at 1Gb/s.

Please note: Everything presented here is preliminary and may contain errors!

Two(+?) options for setting it up:

Using SLATE:

This means allowing DDM to run Docker containers on the cluster. Pros being you don't have to setup it up or manage it yourself, Cons that you have to allow an outside source root access to machines

Manual install and configuration:

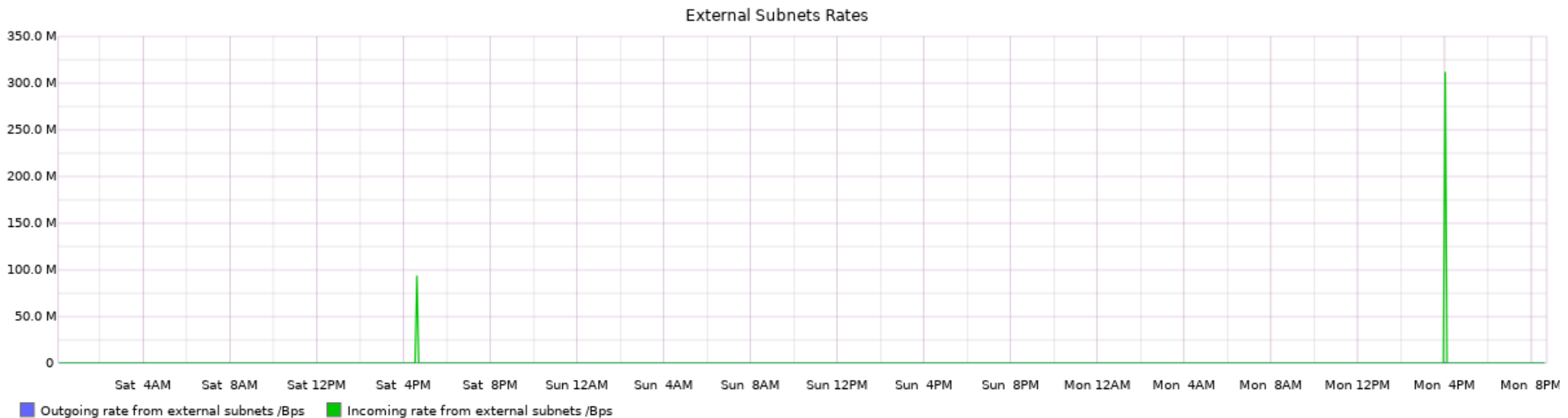
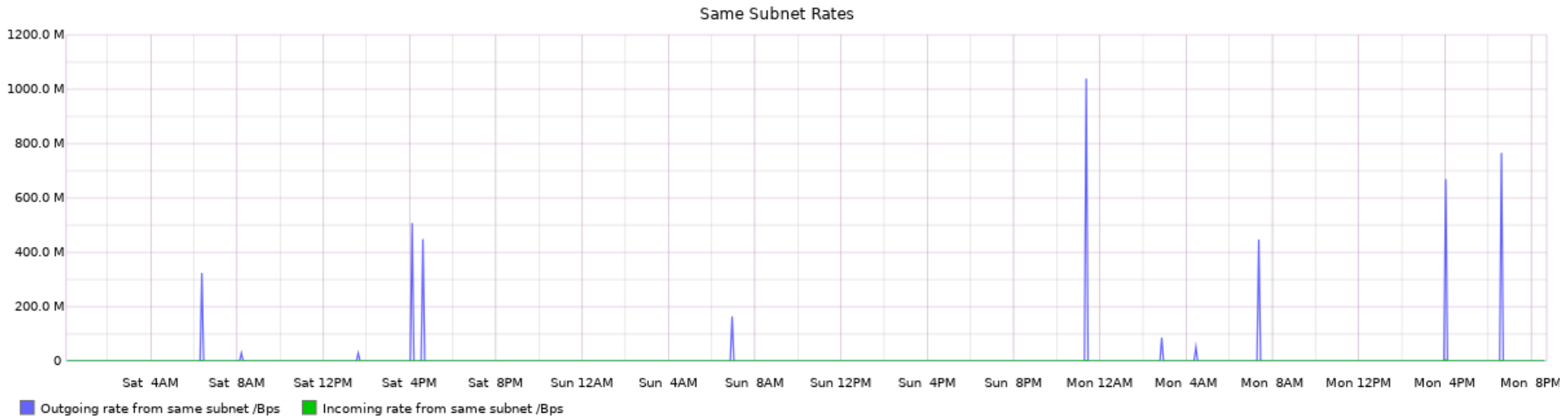
All setup and config is done by sys-admin but full control is maintained

I chose the latter and using a singularity definition I got from here:

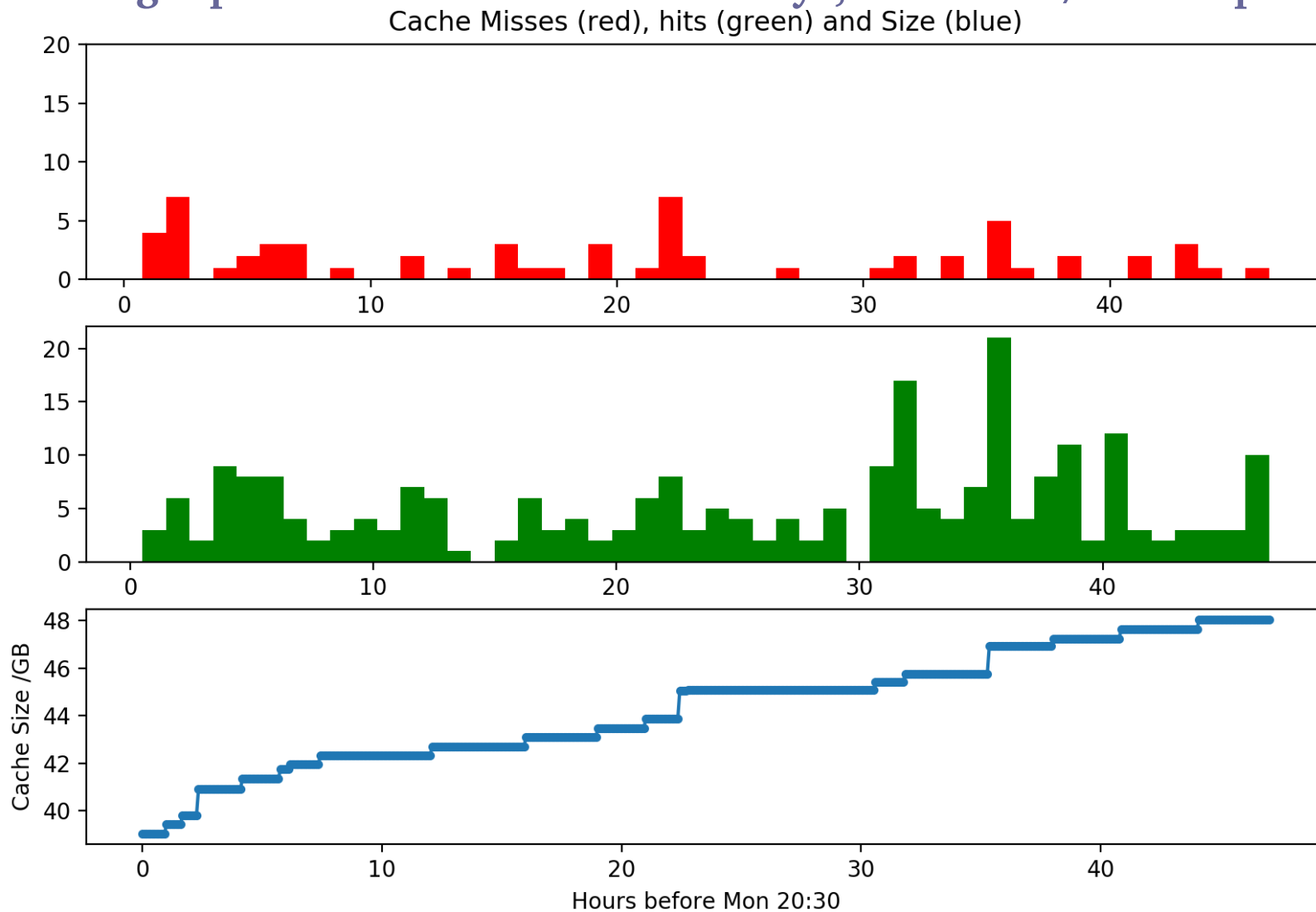
<https://raw.githubusercontent.com/wyang007/rucioN2N-for-Xcache/master/xcache.singularity.def>

The only issue I had (beyond my own stupidity) was that I needed to use my own grid certificate to grab files. It would be nice to have an approved Xcache or UK Robot cert to do this though I appreciate the security implications here

Network Access from a single pool node as recorded in 10s windows every 5mins:



Data from single pool node from the last 2 days, cache hits/misses per hour:



From this data I have drawn the following (possibly erroneous) conclusions:

- The caching mechanism seems to work as intended
- New files are cached for arriving jobs then access ~3-5 times soon after
- Network load is very bursty, probably because external access lasts 10-15s
- Do not see any extended access either internally or externally
- Disk being filled at ~4GB/day so not likely to run out of space any time soon!

This system has been pretty easy to setup and run – many thanks for the help from the GridPP and Atlas teams for making this relatively painless!

Overall I can't comment on how useful it is as an actual cache at the moment as I'm seeing about an 50-80% cache hit rate and these hits are very soon after the first access (though this will depend on the flavour of jobs being run)

Having said that, at the very least it is a very easy way to fill a small amount of storage with useful files and therefore allow (almost) diskless sites to run without overloading the larger ones.