Journal of Physics: Conference Series **396** (2012) 052064 doi:10.1088/1742-6596/396/5/052064

Testing performance of Standards-based protocols in DPM

Samuel Skipsey[1], Wahid Bhimji[2], Ricardo Rocha[3]

[1] Department of Physics and Astronomy, University of Glasgow, G12 8QQ
[2] University of Edinburgh, School of Physics & Astronomy, James Clerk Maxwell Building, The Kings Buildings, Mayfield Road, Edinburgh EH9 3JZ, UK
[3] European Organization for Nuclear Research (CERN), CH-1211 Genve, Switzerland

E-mail: Sam.Skipsey@glasgow.ac.uk

Abstract. In the interests of the promotion of the increased use of non-proprietary protocols in grid storage systems, we perform tests on the performance of WebDAV and pNFS transport with the DPM storage solution. We find that the standards-based protocols behave similarly to the proprietary standards currently in use, despite encountering some issues with the state of the implementation itself. We thus conclude that there is no performance-based reason to avoid using such protocols for data management in future.

1. Introduction

The history of grid storage systems has been a process long associated with the creation of bespoke protocols and paradigms, rather than the adoption of wider standards. This culture has its roots in the early history of the project, where there simply were no other software packages designed to solve the problems that grid storage aimed to; however, the world has now moved on significantly. While Cloud Computing and related paradigms are not Grid (they solve the simpler problem of dynamic resource allocation in a spatially constrained hosting environment with abundant resources) they do share some of the same data management requirements, as does the increasing amount of work done over the World Wide Web via web services. With corporate backing, and the work of the open source community, many standards have become available which cover the same use cases as our fragmented proprietary protocols; HTTPS/WebDAV[1], NFS4.1/pNFS[2] (and many others) vs CASTOR rfio[3], DPM rfio[4] (not the same as CASTOR rfio, thanks to project drift all too common in Grid software), dCache dCap[5] and even gridFTP[6].

Luckily, there is an awareness that the use of open standards can be of benefit to the WLCG community (as well as to the vastly larger pool of potential users of the technology and infrastructure developed for that community), and the EMI Data Management track explicitly includes targets for supporting pNFS and WebDAV as transfer protocols. dCache's support for pNFS has already been well demonstrated (it was one of the first software packages to actually support the standard, even before clients were easily available), but support in DPM has been slower to come. We tested the currently available implementations of the WebDAV and pNFS transfer protocols in DPM release 1.8.3 for their performance against the default rfio transport mechanism.

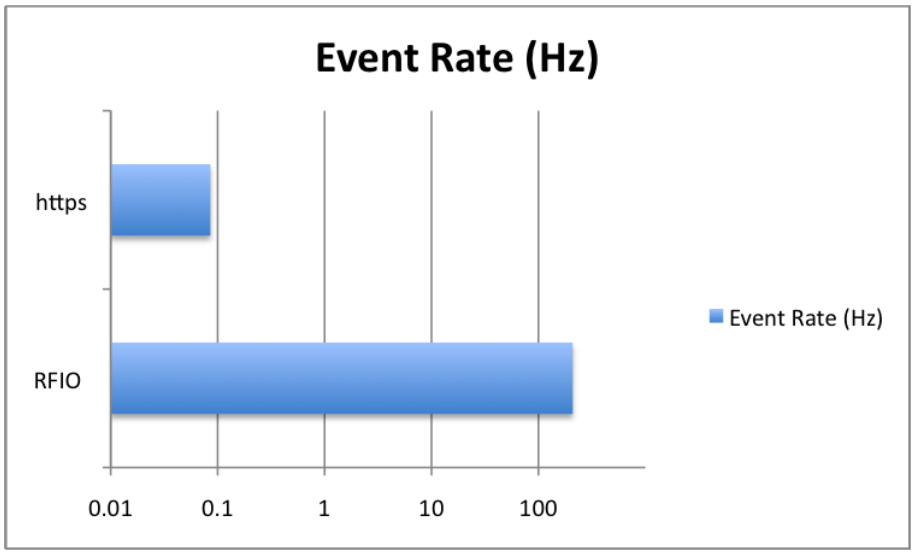

Figure 1. Comparison of ROOT analysis performance when streaming data over WebDAV/https vs rfio.

2. HTTP(s)/WebDAV

The WebDAV standard (RFC2518) is an extension to the HTTP(S) standard intended to allow collaborative document authoring over the Web. As a consequence of the requirements for this, a WebDAV server can be mounted as a filesystem on a client (this is possible with a pure HTTP server, but not without significant compromises, as HTTP is a purely file-centred protocol). Being an extension to HTTP(S), WebDAV servers can fall back to basic HTTP if the client does not support their extensions, so file-transfer is possible even with webbrowser, or simple command-line tools such as curl or wget. Federation of resources is possible with WebDAV (and pure HTTP) via the request redirection mechanism defined in the relevant RFCs.

2.1. Testing

While DPM has supported HTTP as a transfer protocol for several years, WebDAV support is a new feature, provided as stable in the 1.8.2 release and later. We tested the performance of the WebDAV plugin on the test DPM instance, `svr025.gla.scotgrid.ac.uk` at Glasgow, and the test DPM at Edinburgh. Some testing was performed using the testing framework developed to measure ROOT I/O performance, by Wahid Bhimji and Illya Vukotic[7].

Figure 1 shows the performance of ROOT-based analysis using the above cited framework when streaming input AODs (derived physics data files produced in ROOT format) over WebDAV, compared to streaming over rfio. As the WebDAV implementation must authenticate the user, https is required for security. At the point of testing, https was "all or nothing" in the implementation, meaning that encryption was also applied to the transferred data after redirection to the disk server. This explains the orders of magnitude reduction in performance relative to rfio.

It is possible to manually manipulate the URL passed as the redirect to the data source in order to access it over an unencrypted http connection, if the server is configured to not insist on security. Unfortunately, ROOT was unable to algorithmically achieve the same result (it provides no way to intercept the redirect location and modify it before following the redirect) , but figure 2 shows the performance difference when using curl to directly copy files with the "default" https URL and the manipulated http URL, against an rfio copy. As can be seen,

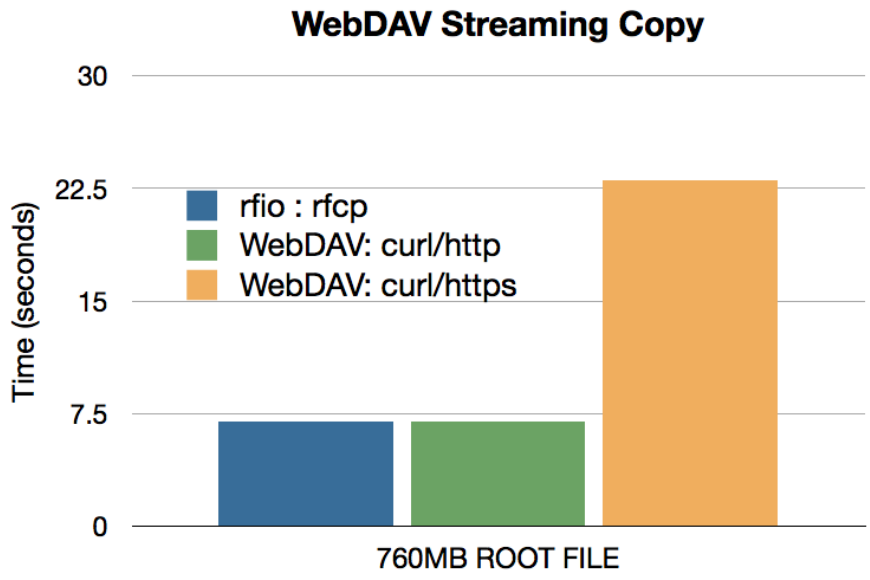

Figure 2. Testing with curl shows http rates equiv to rfio. ROOT is unable to perform the URL manipulation necessary to perform the same test with an analysis workload.

the entire overhead for the transfer operation is in the additional security layer, with bare http being almost indistinguishable from rfio. We note that a significant fraction of interactions with a local storage element are of the form of streaming copies (the evidence for higher efficiency in remote I/O v copy-to-worker-node methods has historically never been consistent, especially with rfio), and so this does represent a real use of rfio by users.

As an additional example of the issue with https, figure 3 shows the performance in time expired and average transfer speed for xrootd, rfio and https for a 1 GB file. The large variance in time for xrdcp seems to result from unpredictable initialisation delays; once the copy starts, xrootd shows comparable performance to rfio on the link available (in fact, it saturates the link available for this test).

3. NFS4.1/pNFS

NFS4.1 is the most recent release of the popular Network File System. As well as introducing multiple modernisations, including unicode support, it also provides a standard for accessing filesystems distributed over multiple filesystems, via the "parallel NFS" extension, or pNFS. pNFS requires the server to distribute "layout files" to the client, indicating the location of a file (or parts of a file in the case that data is striped across multiple servers) within the set of file servers behind it. pNFS has the advantage of support in the linux kernel, full support being available in the 3.x.x series, but backported into earlier kernels by Red Hat.

3.1. Testing

pNFS support was not yet fully stable at the time we conducted our tests. We installed the NFS4.1/pNFS transport plugins from the development repository on the test DPM instance at Glasgow.

As SL5 does not have a new enough kernel to support the kernel pNFS client, we could not evaluate the performance of the module against the standard worker nodes at either site. Some limited testing was performed against a client installed with SL6.2, which includes a "technology

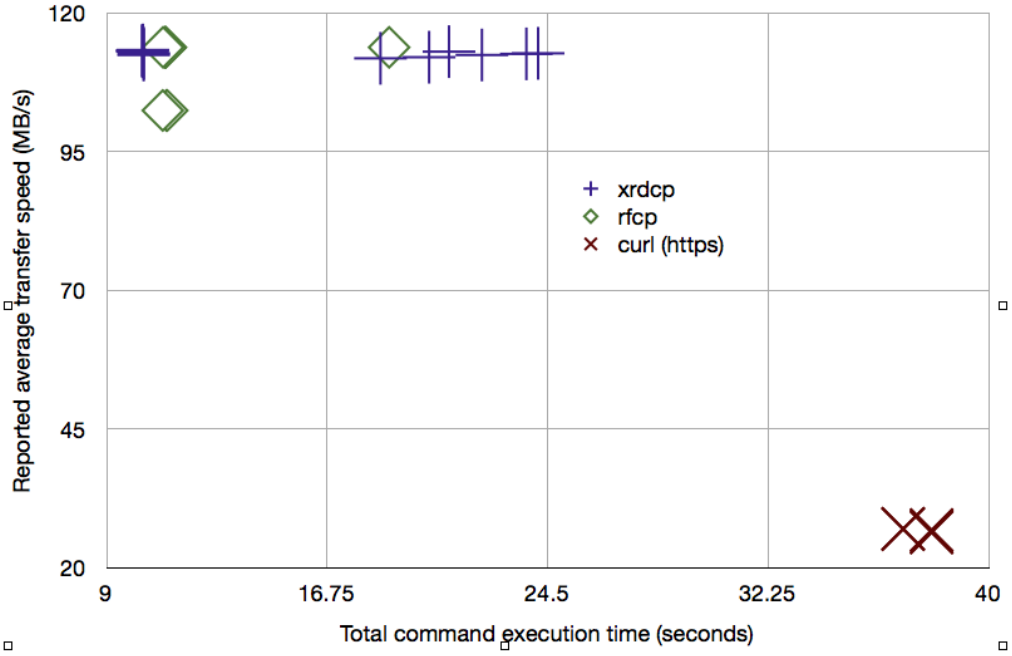

Figure 3. Command completion time v actual reported transfer speed for the rfcp (rfio), xrdcp (xrootd) and curl (https) commands against the same 1 GB file. 10 attempts per protocol are shown. Once again, https is consistent in performance, but the encryption overhead caps maximum speed far below the equally matched xrootd and rfio.

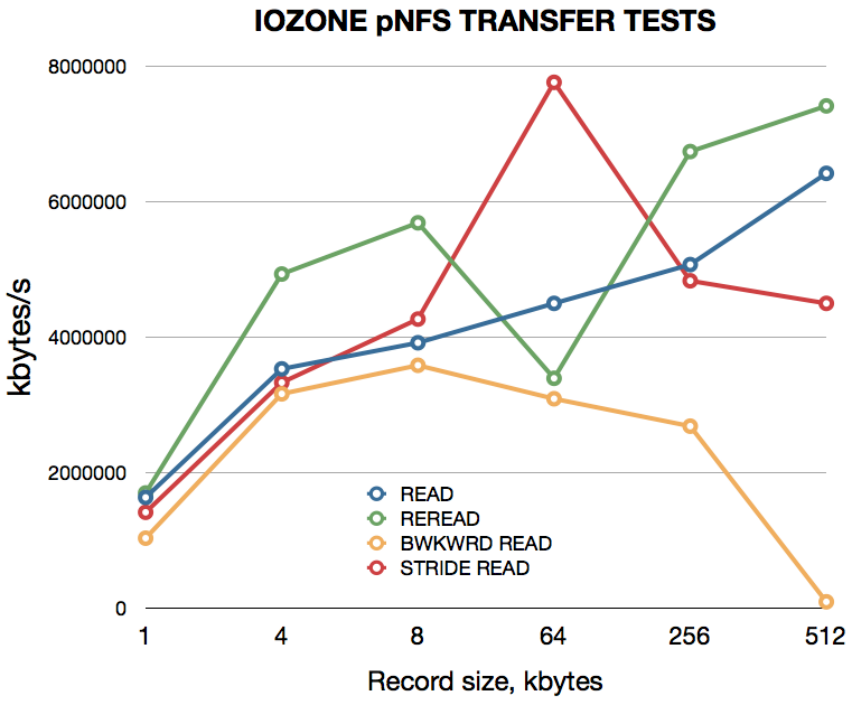

Figure 4. Results of read-only testing of pNFS via iozone.

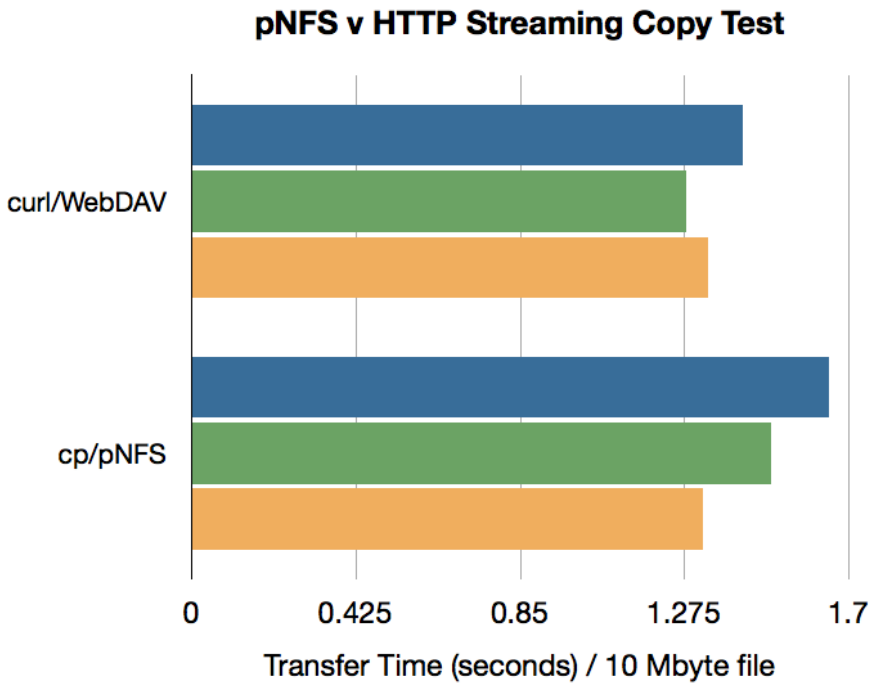

Figure 5. Results of comparison between WebDAV and pNFS streaming copy performance. The multiple bars show performance on the first (top) and second copies (middle) and the average performance on multiple copies after the first (bottom). pNFS appears to show better caching performance after the first copy completes.

preview" of pNFS client support.

Testing of pNFS was problematic, as the available plugin proved to be exceptionally unstable. After some patches to the release, some tests were eventually performed, but the system was not sufficiently reliable to perform a full ROOT-based analysis. Figure 4 shows the result of running the popular benchmarking tool "iozone" on a pre-copied file on the test DPM. Filesystem writes over pNFS were not reliable, so iozone was limited to read-only testing. As can be seen, scaling performance improves with the record size on the whole. We cannot explain the poor performance with backward reads and large record sizes, except to note that DPM attempts to perform some read-ahead caching, which would not interface well with a large stride read in the opposite direction, causing significantly more data to be transferred.

Figure 5 shows the results of a streaming copy test, comparing multiple copies of a 10 MB test file across a WAN link via curl (http) and posix cp (pNFS). As can be seen, with files this small, there is no significant difference in performance between the two protocols. This is notable mainly as http was designed as a WAN protocol, while NFS has traditionally been seen as inefficient outside the LAN. That pNFS can perform well on a WAN link suggests that it might be useable as the basis for a wider area federated storage, with POSIX semantics.

4. Conclusions

Although it is clear from the caveats in our testing procedure that there are some remaining issues with the use of both protocols in production work, the raw performance of copies with pNFS, in particular, is very promising. Considering the relative newness of the pNFS and WebDAV interfaces to DPM, that they can both meet the performance of the established standard protocol is helpful. In the case of WebDAV support, the deficiencies experienced were predominantly on

the side of the client (in this case, ROOT), while it is admittedly the case that our experience of the pNFS plugin is that it is inherently unreliable. Since the tests performed in this paper, both implementations have undergone significant development, and so this should not be seen as a reflection of any deficiencies in the current versions. In particular, the WebDAV implementation in SVN for DPM now supports redirection from an https connection for authentication to an http connection for data transfer, resolving the performance issues of http in a manner compatible with ROOT. Future testing with the 1.8.4 release of DPM should be performed to monitor the development of the protocols and their efficiency.

References

[1] http://www.ietf.org/rfc/rfc2518.txt
[2] http://www.ietf.org/rfc/rfc5661.txt
[3] Giuseppe Lo Presti et al , *24th IEEE Conference on Mass Storage Systems and Technologies* (MSST 2007), 2007, pp.275-280
[4] Badino, P, et al. *24th IEEE Conference on Mass Storage Systems and Technologies* (MSST 2007). 2007 pp.60-71
[5] Fuhrmann, P. *Twelfth NASA Goddard and Twenty First IEEE Conference on Mass Storage Systems and Technologies* 2004
[6] Allcock, William et al. *Proceedings of the 2005 ACM/IEEE conference on Supercomputing* 2005 pp54-
[7] W. Bhimji et al *J. Phys.: Conf. Ser.* THIS PROCEEDINGS CHEP 2012