EPJ Web of Conferences **245**, 04025 (2020) https://doi.org/10.1051/epjconf/202024504025
CHEP 2019

Testing the limits of HTTPS single point third party copy transfer over the WAN

Edgar Fajardo[1,*], *Brian* Bockelman[2,**], and *Frank* Wuerthwein[1,***]

[1]University of California San Diego, 9500 Gilman Dr, La Jolla, CA 92093
[2]Morgridge Institute, 330 N Orchard St, Madison, WI 53715

Abstract. LHC data is constantly being moved between computing and storage sites to support analysis, processing, and simulation; this is done at a scale that is currently unique within the science community. For example, the CMS experiment on the LHC manages approximately 200PB of storage across ~ 100 sites and, on a daily basis, moves 1PB between sites via GridFTP as primary protocol. This paper describes the performance results we have achieved by exploring alternatives to the GridFTP protocol for these data movements. In particular the HTTPS third party copy over Xrootd data servers as a possible replacement of GridFTP for LHC big data movements.

1 Introduction

The current model for wide area network access of a site's compute storage on the Worldwide LHC Computing Grid (WLCG)[1] is the *storage element* (SE). The Storage Element is a WAN (Wide Area Network) facing service that usually aggregates some fault tolerant array of services. Once each site advertises its Storage Element, tools like Rucio [2] or PhEDEx [3] can transfer datasets among sites by scheduling file transfers using File Transfer Service (FTS) [4].

FTS uses two tuples (source and destination) to schedule a transfer between sites. Each tuple consist of a Storage Element (and type Xrootd, HTTP, GridFTP, etc) and a file name. In the case of the Open Science Grid (OSG) [5] until 2016 BestMan [6] used to be the storage element of choice with several Grid FTP servers behind it. Since then it has been phased out to a pure GridFTP and Linux Virtual Server (LVS) solution [7]. Now in preparations for the High Luminosity Large Hadron Collider (HL-LHC) a joint taskforce between OSG and Iris-HEP was setup to explore the possibility of having all the transfer done using less proprietary protocols (GridFTP) in favor of more widely used protocols: HTTP, while also leveraging other existing pieces of the grid software stack: XRootD [8].

The proposed replacement for the LVS with GridFTP configuration consists of using XRootD servers behind a redirector using HTTP as transfer protocol. In this document we explore the scalability of this solution to be considered a replacement for GridFTP transfers for the WLCG community. We begin in Section 2 by mentioning the current architecture for file transfers infrastructure. Then we show our proposed solution and its scalability compared

*e-mail: emfajard@ucsd.edu
**e-mail: bbockelman@morgridge.org
***e-mail: fkw@ucsd.edu

to the currently deployed solution in Section 3, and finish with a demonstration on how the scheme performs in production for transfers among US CMS Tier 2 sites.

2 Current Grid FTP LVS Architecture

OSG sites are currently setup to source and sink data via a set of GridFTP Servers behind a Linux Virtual Server (LVS). In this setup LVS acts as a network load balancer in the sense that it works by redirecting TCP packets (in a round robin fashion) to the different GridFTP servers and then spoofing the ARP packets for the rest of the connection as shown in Figure 1. A more in depth description of this setup can be found in [7]. However as successfully as this architecture is at doing transfers and reducing the software overhead it comes with its own drawbacks. It is based on the Globus Protocol, adding a layer of software to maintain that is no longer supported by its developers; the authentication is limited to GSI (No token based authentication); and LVS is not aware of GridFTP. The latter is due to the fact that LVS is a general purpose load balancer which instead of working at the application level works at the network level. This results in operational problems due to LVS not knowing if one server has failed a transfer several times nor how loaded a server is. LVS incorporates heart-beats, but this only guarantees that the server has an active network connection, and says nothing about the performance or reliability of that connection.

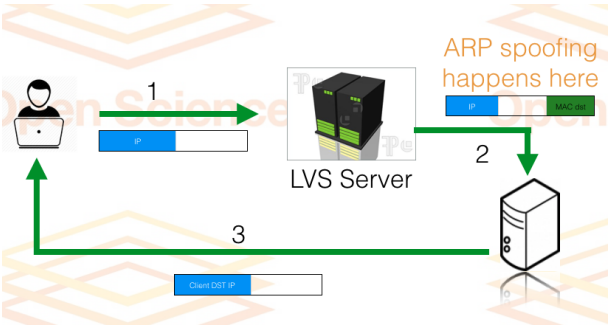

Figure 1. Diagram of LVS spoofing for gridftp

3 XrootD Third Party Copy

In this work we propose a solution that addresses the drawbacks mentioned in Section 2. The solution is based on XRootD Servers, Third Party Copy and HTTP as the protocol of choice for transfers. In the proposed solution several XRootD servers (six for a typical Tier 2 Site) are placed behind a redirector with both (servers and redirector) listening on HTTP. Then at this step FTS or any HTTP client (for example curl) can initiate a transfer between two XRootD HTTP servers as seen in Figure: 2. The client sends a request to the redirector which then picks a server to start the TPC based on several metrics (IOWait, NetworkPerformance and so on). The initiater of the TPC can be either the source or destination, i.e. both "push" and pull" are supported. Authentication can occur via standard X509 proxies, Macaroons [9] or Scitokens [10], thus allowing phase-out of X509 over time. The present paper documents the performance characteristics of this solution.

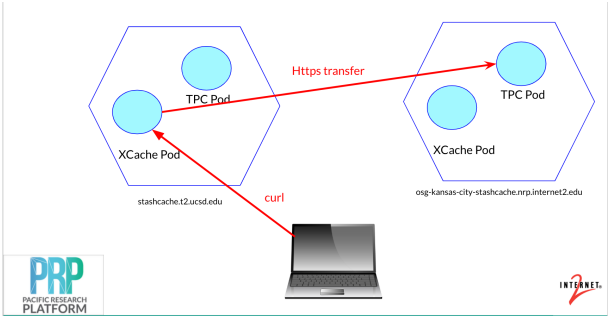

Figure 2. Diagram of TPC transfer tests

Table 1. Kansas City (left) and UCSD(right) server specifications

Memory	128GB
Network Card	100Gbps
Core Count	40
Disk Setup	NVME 20TB

Memory	128GB
Network Card	100Gbps
Core Count	40
Disk Setup	6NVME 10TB

4 Stress Tests and Current Deployment

In order for us to benchmark the performance of file transfers over HTTP using XRootD Third Party Copy mechanisms we set up two $100 Gbit/sec$ network capable servers (the specs of the machines are in Table 1.) and placed them such that transfers cross several regional networks. To easily try different applications, be fully reproducible in the deployment, and use production hardware for the short duration benchmarking, we performed all tests by deploying the XRootD Third Party Copy Servers via Kubernetes.

For the testing we set up two containers [11] and create basic scripts based on curl to initiate up to one hundred simultaneous transfers with files roughly 3GB in size. We picked 3GB because it reflects the average file size for CMS and ATLAS. The number of simultaneous transfers was chosen to be able to have sufficient data in-flight such that connection start-up effects don't prevent us from reaching maximum possible bandwidth. After several rounds of tests we were able to achieve close to $3 GB/sec$ as can bee seen in Figure: 3 in point to point transfers, i.e. using one server each for source and destination. For comparison, Figure 4 shows the production transfer bandwidth achieved at the UCSD Tier 2 using six GridFTP servers. A peak of roughly $25 Gbit/sec$ is reached there.

5 Production Deployments of HTTP via XRootD at US CMS Tier 2 Centers

As a next step, we started deploying HTTP via XRootD at four US CMS Tier 2 centers for production transfers to gain long term operational experience. Production transfers via this infrastructure are being tracked using GRACC [12]. Initial experience with this production deployment is shown in Figure 5. Although HTTP transfers are still a small percentage of transfers this is expected to grow over the years as more sites are enabled in production to transfer over HTTP.

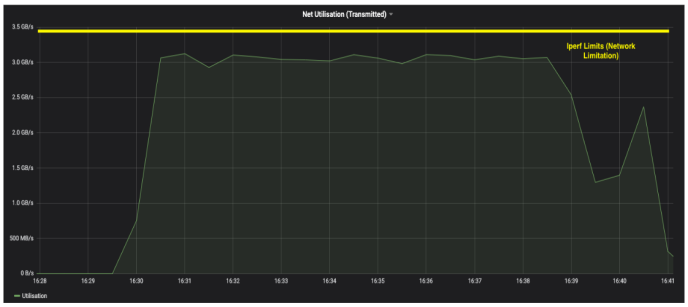

Figure 3. Best achieved network performance on XRootD HTTP TPC tests. The yellow line represents the achieved limit running Iperf3 between same pods.

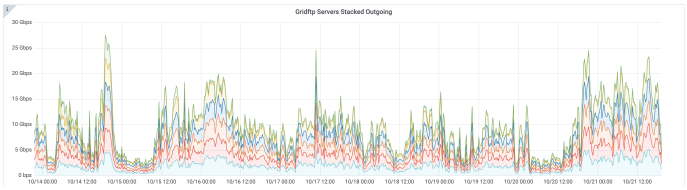

Figure 4. Network usage of production of six GridFTP servers at UCSD

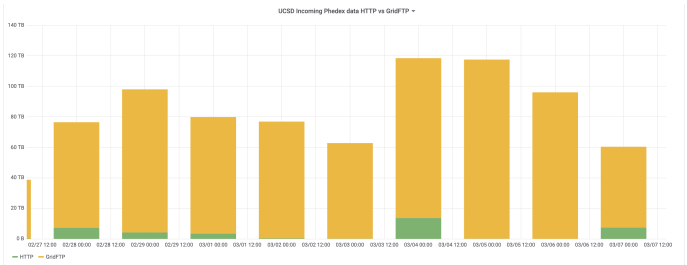

Figure 5. Current production transfers discriminated between GSIFTP (Yellow) and HTTP (Green)

6 Conclusions

Present, based on GridFTP servers aggregated via LVS, and proposed future, based on HTTP implemented via XRootD servers, architecture for WLCG file transfers are discussed. Initial benchmarking indicates that $40 Gbit/sec$ can be achieved using HTTP Third Party Copy as implemented in XRootD servers, even for source-destination pairs of single servers each. It was shown that typical high capacity production deployments using GridFTP achieve $30 Gbit/sec$ today by aggregating six servers via LVS. We thus conclude that the existing XRootD implementation of Third Party Transfer via HTTP is sufficiently scalable to serve as a replacement to the existing combination of GridFTP and LVS.

Initial production deployments have begun, and we showed the first usage plots from those deployments. More experience needs to be gained with these deployments before we can unambiguously conclude that HTTP as implemented in XRootD is ready to replace GridFTP via LVS at US LHC Tier 2 centers.

Acknowledgement

The authors would like to thank the different funding agencies for this work, in particular the National Science Foundation through the following grants: OAC-1836650, MPS-1148698 and OAC-1541349.

References

[1] I. Bird, *Computing for the Large Hadron Collider* (Annual Reviews, 2011), Vol. 61, pp. 99–118, http://www.annualreviews.org/doi/abs/10.1146/annurev-nucl-102010-130059

[2] M. Barisits, T. Beermann, F. Berghaus, B. Bockelman, J. Bogado, D. Cameron, D. Christidis, D. Ciangottini, G. Dimitrov, M. Elsing et al., Computing and Software for Big Science **3**, 11 (2019)

[3] A. Sanchez-Hernandez, R. Egeland, C.H. Huang, N. Ratnikova, N. Magini, T. Wildish, Journal of Physics: Conference Series **396**, 032118 (2012)

[4] E. Karavakis, A. Manzi, M.A. Rios, O. Keeble, C.G. Cabot, M. Simon, M. Patrascoiu, A. Angelogiannopoulos (2020)

[5] R. Pordes, D. Petravick, B. Kramer, D. Olson, M. Livny, A. Roy, P. Avery, K. Blackburn, T. Wenaus, F. Würthwein et al., Journal of Physics: Conference Series **78**, 012057 (2007)

[6] *Berkeley storage manager (bestman)*, https://sdm.lbl.gov/bestman/

[7] E. Fajardo, C. Pottberg, B. Bockelman, G. Attebury, T. Martin, F. W^{o}rthwein, Journal of Physics: Conference Series **1085**, 032004 (2018)

[8] A. Dorigo, P. Elmer, F. Furano, A. Hanushevsky, WSEAS Transactions on Computers **1** (2005)

[9] A. Birgisson, J. Politz, Erlingsson, A. Taly, M. Vrable, M. Lentczner, *Macaroons: Cookies with Contextual Caveats for Decentralized Authorization in the Cloud* (2014), ISBN 1-891562-35-5

[10] A. Withers, Z. Miller, B. Bockelman, D. Weitzel, D. Brown, J. Patton, J. Gaynor, J. Basney, T. Tannenbaum, Y. Gao, *SciTokens: Demonstrating Capability-Based Access to Remote Scientific Data using HTCondor* (2019), pp. 1–4, ISBN 978-1-4503-7227-5

[11] E. Fajardo, *efajardo/Xrootd-TPC: Fist release of the TPC container for tests* (2020), https://doi.org/10.5281/zenodo.3626200

[12] K. Retzke, D. Weitzel, S. Bhat, T. Levshina, B. Bockelman, B. Jayatilaka, C. Sehgal, R. Quick, F. Wuerthwein, Journal of Physics: Conference Series **898**, 092044 (2017)