



Michal Simon

XRootD5: what's in it for you



Outline

- Short introduction to XRootD5
- Secure root/xroot, why and how?
- Other goodies
- Summary



XRootD5 in few words

- **Major release**, with the most important new feature being **encryption**
- Protocol and API level backwards compatibility, it is not ABI compatible – plugins will require recompilation
- Released in July 2020, followed by 3 bugfix releases and one feature release
 - Released in OSG repo and EPEL
 - By the end of 2020 pushed to Debian distribution

Secure root/xroot protocol

- roots/xroots is the old good **root/xroot protocol plus TLS**
- Based on OpenSSL
 - Version 1.0.0 and above
 - Custom hostname verification added to cover the older versions
- Encrypted and unencrypted version of root/xroot protocol run on the same port (by default 1094)

Why do we need encryption?

- Allows for authorization token handling (e.g. SciToken)
 - Prerequisite for **replacing X509 with access tokens** in WLCG
- Encrypt confidential data
 - Encrypt 'in transit' data for the **CERNBox** use case
- Encrypt possibly destructive metadata operations (could replace in the future request signing)
- Improves data integrity and allows for further evolution of Third-Party-Copy

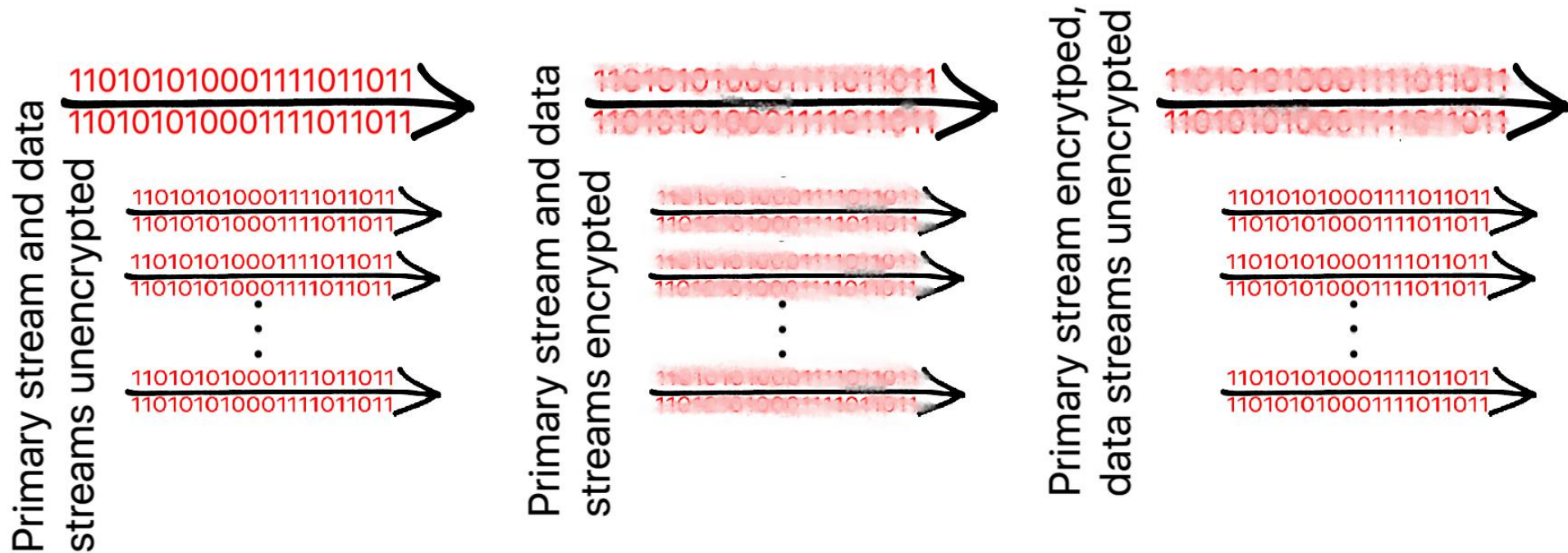
What triggers encryption?

- On the client side the **roots/xroots** protocol;
 - **--notlsok** options allows to proceed without encryption if the server is too old to support it
 - **--tlsmetalink** option allows to apply encryption to all URLs in a metalink file
- On the server side the **xrootd.tls** configuration directive, with few compatibility options:
 - by default it is **off**
 - enforce encryption only for clients that support it (**capable**)
 - do encryption only at client discretion (**none**)

How flexible is it?

- It is pretty flexible ;-) – not everything needs encryption and (at the beginning) not everyone will support encryption
- One can configure the server to encrypt:
 - only the **third-party-copy** orchestration
 - **control channel** after login (handy for GSI auth)
 - control channel before login
 - **data streams**
 - everything
- On the client side:
 - **--tlsnodata** allows to apply roots/xroots only to the control stream

How flexible is it?



What lies beneath the flexibility?

- Handshake negotiation
 - All connections are **initially non-encrypted**
 - The connection is being **upgraded to TLS on client or server request**
- If only control channel should be encrypted we open a second (or multiple) physical connection for the raw data
- **Encrypted and unencrypted traffic uses the same port** number (not like http vs https) to ease operators lives

Is roots/xroots widely available?

- **GFAL2 has been ported to XRootD5** (in EPEL)
- **DPM has been ported to XRootD5** (available in EPEL)
- **EOS has been ported to XRootD5** (successful encrypted transfer executed in PPS)
- **dCache** devel team (with our help) implemented roots/xroots support (in Java!!!)

Certificates, certificates, ...

- XRootD server needs a host certificate in order to enable encryption
 - configurable with **xrd.tls** directive
- If roots/xroots is being used client will **enforce host verification**
 - the hostname must match the one in the host certificate (or one of the SAN extensions)

Certificates, certificates, ...

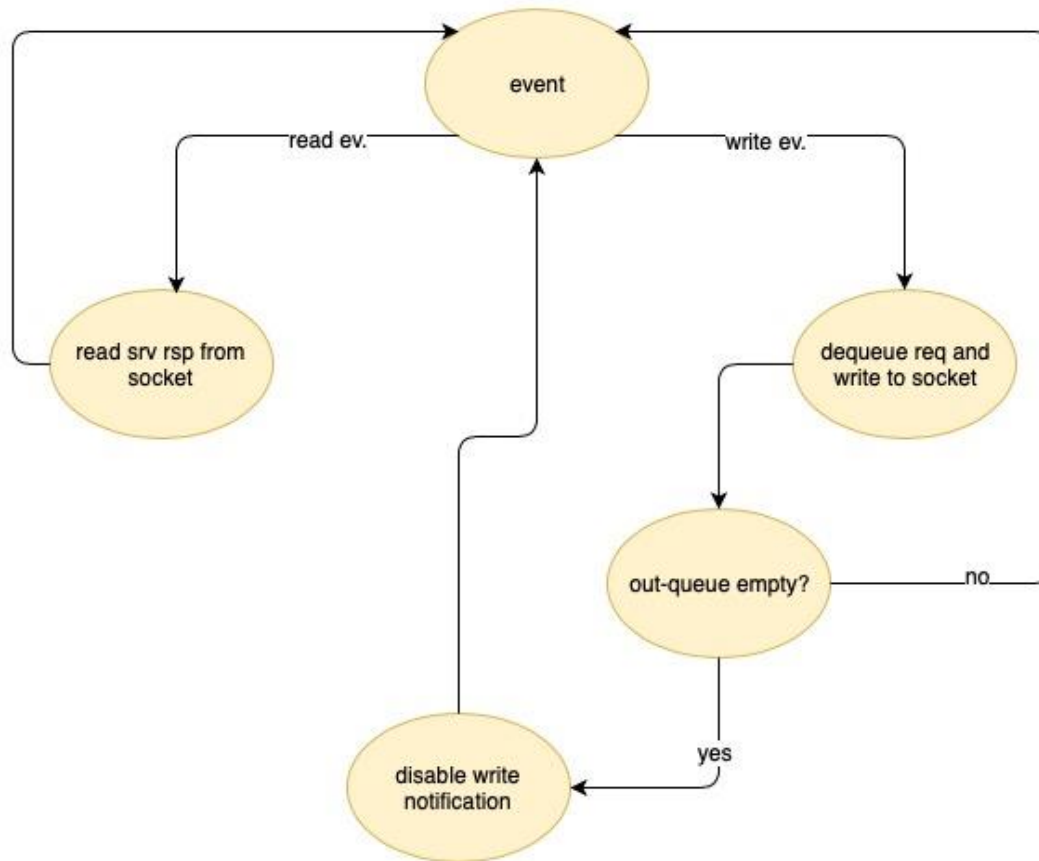
- The client does not need to have a certificate
 - the user **may use his proxy certificate** in order to establish a TLS connection
 - server can be configured to enforce client certificate verification with: **xrd.tlsca**
- Allowing the client to establish the TLS connection based on user X509 proxy certificate opens door to a new **more concise implementation of gsi authentication** in the future

Implementation

- roots/xroots implementations is based on OpenSSL
 - for better performance, **asynchronous APIs and socket BIOs** were used
- **All TLS actions are logged** (e.g. when connection is upgraded to TLS, what version of TLS is being used)
- We are aiming at **isolating OpenSSL in the XrdTls** component
 - should facilitate migration from OpenSSL in the future (e.g. to NSS)

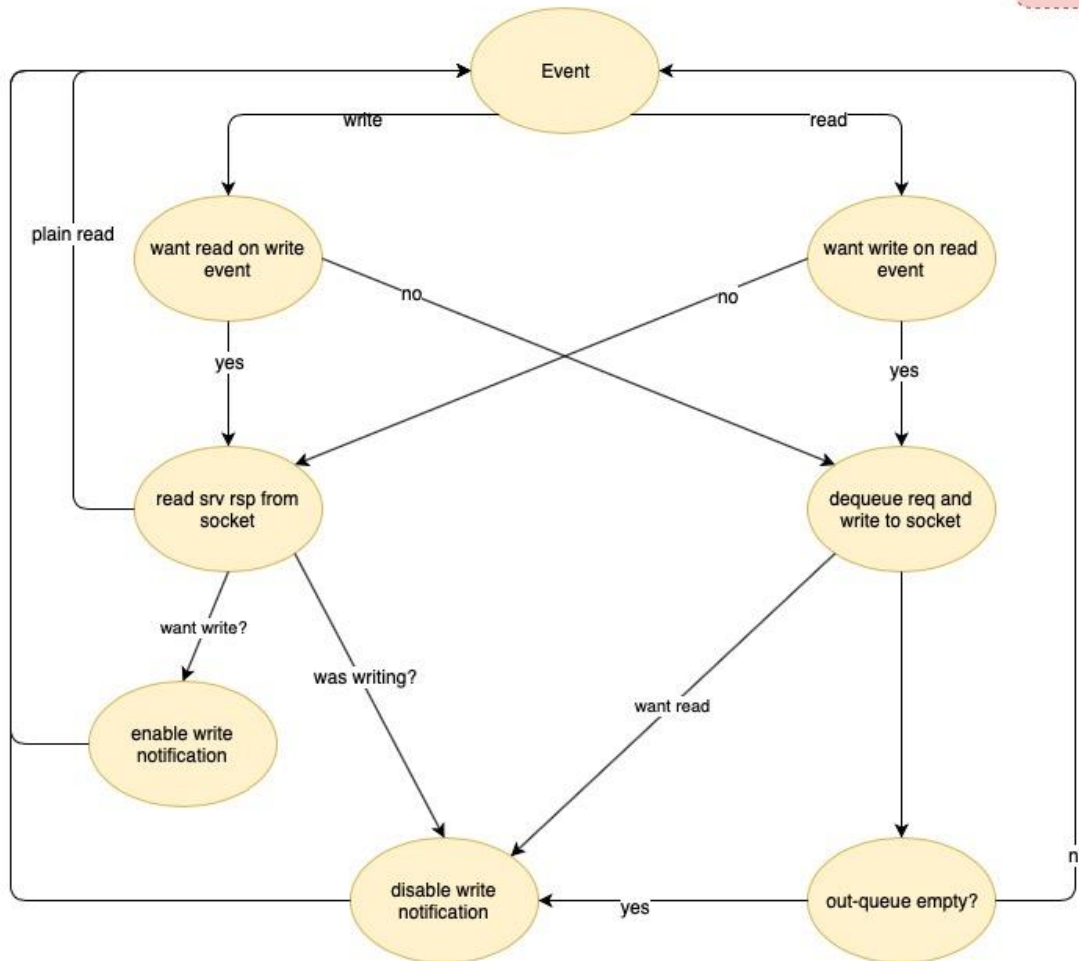
Implementation: event loop (no TLS)

* enable write notification when user enqueues new request



Implementation: event loop (with TLS support)

*enable write notification when user enqueues new request



One last word on TLS ...

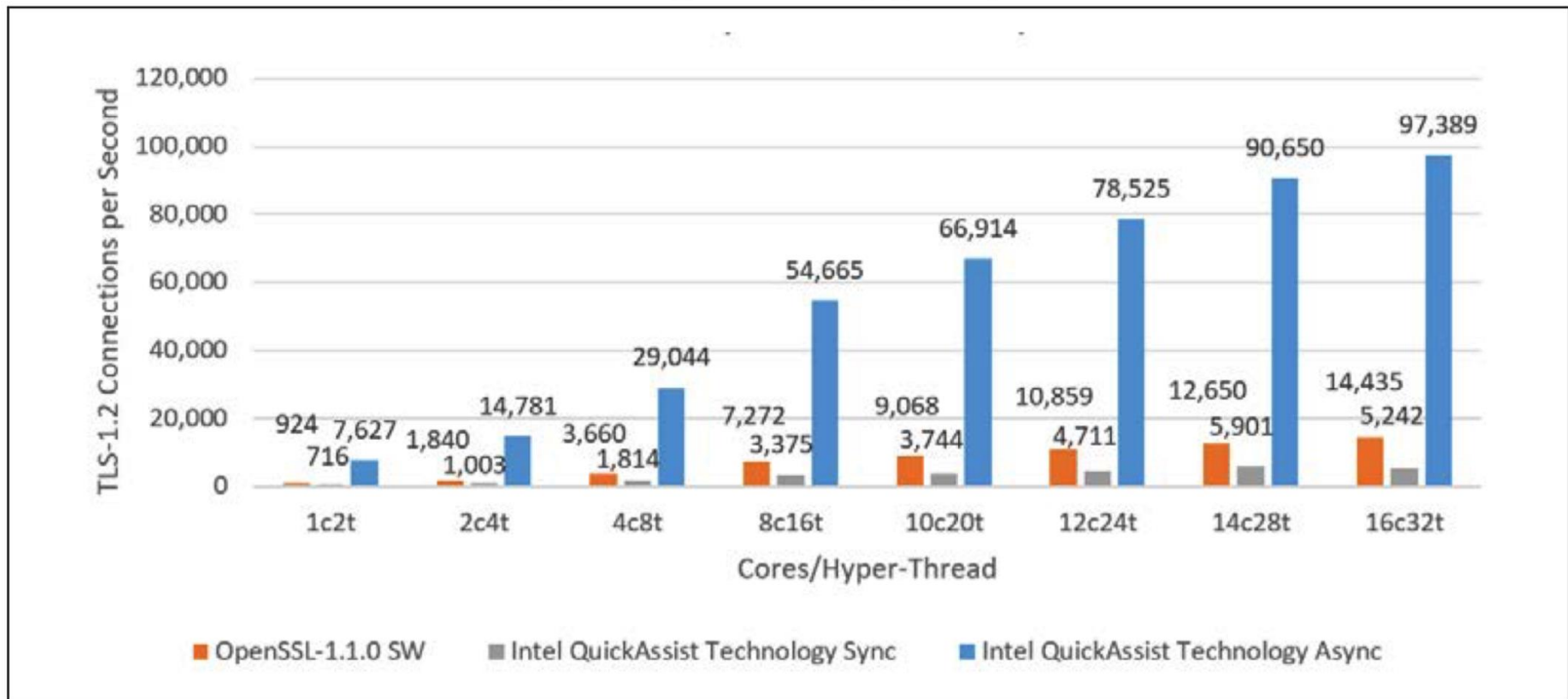
Design choices (OpenSSL)

- Memory BIO vs **socket BIO**
 - avoid **copying the data in memory**
 - saves time, memory and CPU cycles
- Synchronous vs **asynchronous** interface
 - reducing **context management overhead** (single-threaded way to handle multiple TLS connections)
 - cryptographic transformations can be more easily **processed on dedicated h/w** (Intel QuickAssist)
 - allows for more optimization
 - parallel processing at crypto-level

Async vs Sync

Intel NGINX test (2018)

- async + QuickAssist is **significantly faster** than sync
- <https://01.org/sites/default/files/downloads/intelr-quickassist-technology/337003-001-intelquickassisttechnologyandopenssl-110.pdf>



Upgrade to TLS overhead

- CC7, openssl 1.02k-21
- Overhead of upgrading the connection to TLS
 - ~2.3 msec to carry out the plain XRootD HS
 - ~7.6 msec to carry out the XRootD HS including upgrade to TLS (host & client cert verification)
 - ~37.6 msec to carry out the XRootD HS including GSI authentication

Other goodies

SecEntity re-mastered:

- X509 capabilities, key-value attributes
- **Credential forwarding**, Multi-VO credentials
- Easily **extensible without breaking API**

Universal (both **root/xroot** and **http**) **VOMS attribute extractor** plugin

- xrdvoms plugin was the starting point
- Shipped as a sub-package of XRootD in **EPEL**
- Obsoletes several packages (vomsxrd, xrootd-voms-plugin and xrdhttpvoms)

Other goodies

General purpose new features

- **Extended file attributes**
- Extended stat (sets stages for proper **uid/gid tracking**)
- Hardware assisted **CRC32C**
- gstream (monitoring stream optimized to deliver periodic medium-level info)
- Server side **plug-in stacking** with `++` directive
 - User plugin gets a pointer to the level-up plugin so it can call it's implementation
- **SciTokens** plug-in
- Client **declarative API**

Other goodies

Intel ISAL based erasure coding library

- For the AliceO2 use case

Ensure data integrity in XCache; significantly reduce transfer failures due to checksum errors

- **Paged read:** read request with **CRC32C** (hardware assisted) **per 4KB block**

Other goodies

New features for EOS

- **Write recovery at MGM** (allows to **recover 99% of I/O errors** for *xrdcp* transfers to EOS)
- **Collapse redirect from passive to active MGM** in xrootd client
 - Facilitate FUSE interaction with passive-active MGM deployment
- **Simplify buffer management & avoid copying data between kernel and user space**
 - Using **splice/vmsplice** syscalls
 - Speed up data transfer: for **slow medium ~3-5%**, for **fast (like ramdisk) ~40%**; reduces CPU usage **by a factor of 3-4**

XRootD4 status

- We had 5 bugfix releases in 4.12.x series after releasing XRootD5
 - Mostly to backport **important fixes for the third-party-copy with HTTP**
- **End of full support scheduled/proposed for April 1st, 2021**
 - XRootD5 will move to **C++14** hence it won't be possible to back-port some of the new developments/bugfixes to XRootD4
 - If desired we can offer a additional period of limited support for XRootD4

2021 plans

- Follow up and support **XRootD5 deployment**

High priority new developments

- **Finalize client EC plugin for Alice O2** (Hook it up to EOS)
- **ZIP append** (initial work done by a summer student, will use checkpoint support on server side)
- **Data at rest integrity option**

Other possible developments

- **uid/gid tracking; connect control and data streams on different interface**; recursive delete (driven by webdav semantics)
- Get/put file (new TPC); channel level plug-ins; RDMA support; Extending testing infrastructure (mock event-loop)

Summary

- We have a working and fully functional **secure roots/xroots protocol**
- Many backwards compatibility ‘features’ to **facilitate forward migration path**
- **Plethora of new features and enhancements** both for site admins and developers

Questions?

