Catching Up With XRootD

FTS/XRootD Workshop

September 9-13, 2023

Andrew Hanushevsky, SLAC
http://xrootd.org

SLAC

SLAC
NATIONAL ACCELERATOR LABORATORY

XRootD Release Notation

- X.0.0
  - Major release that can break ABI
    - Every 2 to 4 years
- X.Y.0
  - Feature release that can't break ABI
    - 1 to 2 each year
- X.Y.Z
  - Patch release to fix bugs & minor enhancements
    - As often as needed

What's Happened Since 2023

- 27-March-23 **XRootD** Workshop @ JSI
  - 08-May-23 Patch Release 5.5.5
  - **30-June-23** **Feature Release 5.6.0**
  - 11-July-23 Patch Release 5.6.1
  - 15-September-23 Patch Release 5.6.2
  - 27-October-23 Patch Release 5.6.3
  - 11-December-23 Patch Release 5.6.4
  - 22-January-24 Patch Release 5.6.5
  - 25-January-24 Patch Release 5.6.6 (Oops)
  - 06-February-24 Patch Release 5.6.7
  - 23-February-24 Patch Release 5.6.8
  - 08-March-24 Patch Release 5.6.9
  - **01-July-24** **Feature Release 5.7.0**
  - **04-July-24** **Patch Release 5.7.1**
- 09-September-24 **XRootD** Workshop @ RAL

The largest number of patch releases for a feature release

Performance Features 5.6.0 @ 06-30-23

- Allow **Xcache** file evictions
  - Can be done using the **xrdprep** command
    - xrdprep **–E** *host*[:*port*] *path* […]
  - Can be done using **xrdfs** command
    - xrdfs *host*[:*port*] {**evict** | **fevict**} *path*
      - evict means evict only if file is not in use
      - fevict means forced eviction (i.e. evict no matter what)
        - All clients using the file get an I/O error
- Allow **Xcache** origin to be a locally mounted filesystem
  - pss.origin *path*
    - This allows fronting file systems with a smarter cache
      - E.g. Lustre when workload uses small reads.

Performance Features 5.6.0 @ 06-30-23

- Increase default parallel event loops to 10
  - This improves **Xcache** and Proxy server performance
  - This is a client option settable via envar XRD_PARALLELEVTLOOP
    - The client default was 1 but reset to 3 for proxy servers

Security Features 5.6.0 @ 06-30-23

- **gsi** option to display DN when it differs from entity name
  - **-showdn:{true|false}**
    - false is the default
- Allow **Xcache** origin to be a locally mounted file system
  - pss.origin *path*
    - This allows fronting file systems with a smarter cache
      - E.g. Lustre when workload uses small reads.
- Implement ability to have the token username as a separate claim
  - This allows for identity tokens
- Use SHA-256 for signatures, and message digest algorithm
  - This is now the minimum required for any new OS

Security Features 5.6.0 @ 06-30-23

- Allow ztn authentication protocol to handle other tokens
  - Originally, ztn was geared for Sci/WLCG tokens
  - Now ztn can pass through any type of token
    - This was mostly to satisfy EOS requirements
- Allow a client to point to a token file using a URL CGI element
  - xrd.ztn=*tokenfile*
    - Allows use of tokens in a multi-threaded environment

Operational Features 5.6.0 @ 06-30-23

- Allow generic prepare plug-in to handle large responses.
  - Response was limited to 64K now this is a default
    - You specify the maximum size in configuration
      - Driven by Harvard's tape integration
- Allow specification of min and max creation mode.
  - Separate specifications for directories and files.
    - There are many options for backward compatibility
      - https://xrootd.slac.stanford.edu/doc/dev56/ofs_config.htm#_Toc136617291
      - Driven by sites wanting to deterministic way of specifying permissions
- Make **maxfd** configurable with a default of 256K
  - Sets maximum number of file descriptors allowed
    - https://xrootd.slac.stanford.edu/doc/dev56/xrd_config.htm#_Toc152596839
    - Driven by sites specifying unlimited which is impossible to implement

Monitoring Features 5.6.0 @ 06-30-23

- Include token information in the monitoring stream
  - Uses new monitor map message ID – '**T**'
  - **T** *userid***n***tokeninfo**
    - **&Uc=***udid***&s=***subj***&n=**[*un*]**&o=**[*on*]**&r=**[*rn*]**&g=**[*gn*]
      - *https://xrootd.slac.stanford.edu/doc/dev56/xrd_monitoring.htm#_Toc138968517*

Client Features 5.6.0 @ 06-30-23

- Increase number of parallel copy jobs from 4 to 128
  - This allows more parallelism in xrdcp

Other Features 5.6.0 @ 06-30-23

- Move to CMake 3.16
- Support musl libc
  - Rich Felker library released under MIT license
    - Used by Alpine Linux, Dragora 3, and optionally Gentoo Linux
- General modernization of build system
  - More from our ReleaseManager later
- Better support for creating python binary wheels
  - More from our ReleaseManager later
- Improved HTTP protocol conformance
  - Supply caching object information
  - Accept-Ranges in HEAD response
  - Enabling trailer information for better error handling

Performance Features 5.7.0 @ 07-01-24

- **cmsd** load balancing algorithm with randomized affinity
  - This avoids creating hot spots in DFS type environments
    - Learn about it in Jyothish Thomas' talk later in the week
- Implement the **kXR_seqio** open option for sequential I/O
  - FS plug-in can use fadvise() to tell kernel to optimize for xrdcp
    - Actual implementation awaiting code contribution
- Avoid some repeated calls of EVP_PKEY_check
  - Avoid gsi performance degradation in OpenSSL 3.0
- Option to force the destination IP address for HTTP-TPC
  - **tpc.fixed_route** {**true**|**false**}
    - True uses same interface that requested the TPC
      - Used by SENSE project for controlling net throughput

Security Features 5.7.0 @ 07-01-24

- Update min/default RSA bits to 2048
  - This is the currently accepted minimum
    - It will likely increase as time goes on
- Option to allow for http tpc unrestricted redirection
  - **http.auth tpc fcreds**
    - Thus allows curl to forward credentials upon redirect
      - https://xrootd.slac.stanford.edu/doc/dev57/xrd_config.htm#_Toc171719991
- Enable ability to have token groups as a separate claim
  - Feature used by certain token issuers
    - This is configured via the token issuer profile
- HTTP external handlers can now be loaded without TLS
  - **http.exthandler** *name* [**+notls**] *path* [*token*]
    - https://xrootd.slac.stanford.edu/doc/dev57/xrd_config.htm#_Toc171720007

Security Features 5.7.0 @ 07-01-24

- New option to configure authorization strategy for tokens
  - Adds new option to the scitokens.cfg file
    - authorization_strategy = *type*
      - *type* may be **capability** in addition to **group** and **mapping**
    - Refer to the SciTokens README file

Monitoring Features 5.7.0 @ 07-01-24

- Allow G-Stream reporting for the throttle plug-in
  - **xrootd.mongstream throttle use …**
    - https://xrootd.slac.stanford.edu/doc/dev57/xrd_config.htm#_Toc171719980

Other Features 5.7.0 @ 07-01-24

- New only-if-cached cache control option using
  - Allows Xcache to use cache only if file is already in the cache
    - Allows better control of where files are cached
- Add support for pelican:// protocol
  - You will hear more about pelican in subsequent talks
- Move baseline C++ requirement to C++17

Enhancements 5.7.1 @ 09-04-24

- Obtain full server information
  - Motivation: CMSSW blacklisting proxies
    - The real culprit is the origin server
  - Additional information is available
    - Whether or not this is a caching server
      - Yet to be exposed via client API
    - For proxies and caching servers
      - xrdfs query config proxy
        - The fixed origin, if any
        - Whether or not the origin is specifiable in the url

Enhancements 5.7.1 @ 09-04-24

- Accommodate API endpoints
  - New directive: **pss.hostarena** ***text***
    - Inserts ***text*** between origin URL and client path
  - Assume **pss.origin** **xroot://myhost/**
  - Assume URL: **xrootd://proxysrv//a/b/c**
  - Result: **xroot://myhost/*text*/a/b/c**
    - Used to access the requested data
      - Thus limiting the scope of client URLs
    - The ***text*** is exported via envar XRDXROOTD_PROXYARENA
      - Composite origin exported as XRDXROOTD_PROXYURL

Enhancements 5.7.1 @ 09-04-24

- Implement Read/Only redirector option
  - Motivation: Simplify CMS AAA redirection when R/W sites join a redirector
  - Standard solution requires AAA sites to use the **globalro** export option
    - Not easily enforced and is an XRootD-only option
    - New redirector directive centralizes enforcement
      - **cms.mode** {**r/o** | **readonly** | **r/w** | **readwrite**}
        - Works for any XRootD protocol storage provider
          - Will be documented in R6 reference

Enhancements 5.7.1 @ 09-04-24

- The cconfig command
  - New –o option to write out *config* to a file
    - Helpful in unraveling continuations
      - Or creating a single configuration file

Conclusion

- **XRootD** remains relevant with good support
  - Framework widely used as a core component
    - The tagline – "It's **XRootD** Inside!" applies
- Our core partners
  - CERN SLAC UCSD
- Community & funding partners *(not a complete list)*
  - OSG WLCG ATLAS EXPERIMENT ALICE CMS LSST dCache GSI NSF

Funding from US Department of Energy contract DE-AC02-76SF00515 with Stanford University