

StoRM

The StoRM disk-based SRMv2 server.

StoRM

Contents

- What is StoRM?
 - ◆ Planned Features
 - ◆ Who's who
- Sample Use cases
 - ◆ Use Case 1. POSIX I/O usage
 - ◆ Use Case 2: moving large dataset
 - ◆ Use Case 3: computing centre wishes to join Grid
- Release roadmap
 - ◆ June2006 release features
- Further information
 - ◆ Papers and presentations
 - ◆ Bugs
 - ◆ Source code repository
 - ◆ Mailing lists

What is StoRM?

StoRM is a disk-based SRM v2.1.1 server, which aims at allowing direct POSIX filesystem access to files (`file:` transport protocol) in addition to standard protocols already in use within the Grid community (GridFTP, RFIO, HTTPG, etc.).

StoRM has a security subsystem that lets you configure access to the files based on different policies, and will let you securely grant restricted access to files even with a pool accounts setup.

StoRM runs on top of any POSIX filesystem on Linux, but can take advantage of the special features of IBM's GPFS cluster filesystem or SGI's XFS advanced filesystem.

Planned Features

- *support for `file:` protocol: direct filesystem access to files:*
 - native filesystem performance
 - compatible with legacy applications
- support for all (ACL-capable) POSIX filesystems on Linux, but takes advantage of GPFS and XFS special capabilities for advance space reservation.
- *guaranteed* advance space reservation with GPFS or XFS; "best effort" advance space reservation with other POSIX filesystems.
- flexible authorization system, lets you grant or deny access to files depending on a the user's certificate subject DN and VOMS attributes.
- "promiscuous" access to files: Grid and local cluster files can co-exist within the same directory tree — easy migration of existing clusters to Grid storage.
- full VOMS support.
- support for both "permanent" and "volatile" file types.
- split front-end/back-end server setup, allows replication of front end server for load-balancing and high availability.

Who's who

StoRM is a collaboration effort between the EGRID project and INFN / GRID.IT .

The source code is presently being developed by Ezio Corso, Flavia Donno, Luca Magnoni, Riccardo Murri, Heinz Stockinger and Riccardo Zappi.

Sample Use cases

(See [EGEE4] for more details and context.)

Use Case 1. POSIX I/O usage

- StoRM presides over files on a SE: GPFS Filesystem spread over all WN. Access to data is granted simply on VO membership basis.
- Grid user submits job; job reaches WN; job first executes SRM client for getting the file directly.
- StoRM verifies grid user has right permissions; StoRM returns a TURL with file handle; if StoRM is using JiT: it sets up an ACL for local user to which grid credential is being mapped.
- Job processes the file. If JiT: StoRM removes ACL when job finishes.

Use Case 2: moving large dataset

- SRM client issues srmReserveSpace on StoRM; destination StoRM checks requesting user has permissions; destination StoRM returns SpaceToken.
- SRM client issues srmCopy to source StoRM for pushing data set to destination StoRM, given space token.
- Source StoRM checks permissions + negotiates with destination server a GSIFTP transfer.

Use Case 3: computing centre wishes to join Grid

A computing centre wishes to join its local cluster to an existing grid infrastructure; the move needs to have no/little impact on users' organisation of files: users should continue to organise files on the local cluster as they have always done.

- CC's user submits job to the grid; output file Centre's gets saved on CC's StoRM SE.
- Another centre's user wishes to perform local computation on newly created file: user has no need to be aware of special arrangements for data produced from the grid.

Release roadmap

Next release is due in June, 2006. It will be part of EGRID release 3 and will be made available to the general public.

Release notes and installation instructions are available at <http://www.egrid.it/sw/storm/install>

June2006 release features

- srmPrepareToGet
- srmPrepareToPut
- srmCopy: both "push" and "pull" mode
- srmReserveSpace
- srm***RequestStatus

- `srmLs`: recursive, but with a limited number of returned entries.

Further information

Papers and presentations

[EGEE4] Ezio Corso, StoRM presentation at the 4th EGEE Conference in Pisa,
<http://indico.cern.ch/materialDisplay.py?contribId=129&sessionId=22&materialId=slides&materialType=slides>

[CHEP2004] F. Donno, A. Ghiselli, L. Magnoni, R. Zappi, "StoRM: Grid Middleware for Disk Resource Management", http://infnforge.cnaf.infn.it/docman/view.php/11/64/StoRM_chep2004.pdf

Please find other documents at http://infnforge.cnaf.infn.it/docman/?group_id=11

Bugs

Please report bugs through the [EGRID issue tracking system](http://www.egrid.it/cgi-bin/bugzilla/editcomponents.cgi?product=StoRM) at <http://www.egrid.it/cgi-bin/bugzilla/editcomponents.cgi?product=StoRM>

Source code repository

The source code is hosted in the [EGRID CVS and Subversion repository](http://www.egrid.it/cvs/storm?root=cvs) ; although public anonymous access is not yet available, you can browse it at the URL <http://www.egrid.it/cvs/storm?root=cvs>

Mailing lists

The storm-users@infn.it mailing list can be used to contact the developers; its archives can be browsed at <http://server11.infn.it/archive-storm-users>

The storm@infn.it mailing list is a closed mailing list for developers only.