Globus Data Sharing: Security Assessment

November 14, 2014

For Public Distribution

Randy Heiland, Scott Koranda, Von Welch

About CTSC

The mission of the Center for Trustworthy Scientific Cyberinfrastructure (CTSC, trustedci.org) is to improve the cybersecurity of NSF science and engineering projects, while allowing those projects to focus on their science endeavors. This mission is accomplished through one-on-one engagements with projects to solve their specific problems, broad education, outreach and training to raise the practice-of-security across the community, and looking for opportunities for improvement to bring in research to raise the state-of-practice.

Acknowledgments

CTSC's engagements are inherently collaborative; the authors would like to thank the Globus team for the collaborative effort that made this document possible.

This document is a product of the Center for Trustworthy Scientific Cyberinfrastructure (CTSC). CTSC is supported by the National Science Foundation under Grant Number OCI-1234408. For more information about the Center for Trustworthy Scientific Cyberinfrastructure please visit: http://trustedci.org/. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Using & Citing this Work

Cite this work using the following information:
R. Heiland, S. Koranda, and V. Welch, "Globus Data Sharing: Security Assessment," Center for Trustworthy Scientific Cyberinfrastructure, trustedci.org, November 2014. Available: http://hdl.handle.net/2022/19165

This work is available on the web at the following URL:
http://trustedci.org/globus

Table of Contents

Executive Summary

The Globus service for fast and reliable file transfer includes sharing functionality that allows a Globus user to securely share access to files with another user without the need for the second user to have a local account on the storage system on which the files are stored. This report is the result of a Globus-CTSC engagement during which CTSC reviewed the design, architecture, and high-level implementation for the sharing functionality.

CTSC did not find any high security risks with the Globus data sharing feature. There were, however, several recommendations made to address low-to-medium risks and those are detailed in this report. The most significant type of risk discovered in this assessment involves inadequate or confusing documentation for system administrators who are responsible for creating and maintaining a Globus endpoint, especially with regards to the sharing functionality. The second most significant type of risk involves an inability for resource providers (RPs) to monitor which files and directories are being shared.

Overall, the engagement has been very positive.

Response from Globus Team

This section represents the response authored by the Globus Team to this assessment.

The Globus team has been responsive to CTSC's questions and has indicated a willingness to address the recommendations. Globus will report on their actions that address the recommendations at https://www.globus.org/technology/security/globus-security-reviews.

Overview

Globus (formerly *Globus Online*) is a set of services around transfer and management of data available at https://www.globus.org. The Globus service recently added Data Sharing (https://www.globus.org/data-sharing) functionality to allow a Globus user to securely share access to files with another user without the need for the second user to have a local account on the storage system on which the files are stored.

This review covers the design, architecture, and high-level details of the implementation of the Globus sharing functionality and assesses any potential security vulnerabilities. The review assumes the reader is familiar with Globus and related technologies (e.g., GSI).

Scope of Our Review

This review is scoped in significant ways:

- It focuses only on the data-sharing aspects of Globus. Other aspects of Globus are assumed to be generally sound and are out of scope of this review.
    - For a prior review of Globus more generally, please see [1].
- We did not review the source code directly but rather the perceived and documented behavior of the sharing functionality.
- The review does not cover the command-line interface (CLI) or the Transfer (REST) API to Globus; rather, it focuses on the Web interface.

Materials Reviewed

The following materials were reviewed as part of this report:

1. Sharing Data Using Globus webpage: https://support.globus.org/entries/23602336-Sharing-Files-and-Folders-Using-Globus-Online
2. Globus Security Deep Dive. Presentation by Steve Tuecke at GlobusWorld 2014. http://www.globusworld.org/files/2014/20-globus-security-deep-dive-tuecke.pdf
3. Globus Sharing Security. Unattributed and undated document reviewing Globus Sharing from San Diego Supercomputer Center (SDSC).
4. GridFTP Sharing Extensions, unattributed document with 2013 copyright.
5. SDIACT-027 ADR Feedback Summary: Unattributed and undated document representing questions asked as part of the XSEDE process of reviewing Globus Data Sharing.
6. SDIACT-027 Design Document: Globus Sharing. Lee Liming. Version 1.0, May 15, 2014. "This document describes the design specification for the Globus Sharing system as part of the XSEDE SD&I activity SDIACT--027."
7. Big Data Mgt for Science, from ESNet and Globus Online (GO) Webinar. August 8, 2013.
8. Installation and configuration of the Globus Connect Server (GCS), version 5.2.5, that was based on the Globus Toolkit version 5.2.5.

Review Principles

In the experience of the reviewers, it is worth expressing a set of principles as an expectation for a service we are evaluating. This gives us a firm foundation by which we evaluate the service and gauge its shortcomings (if any).

The following principles were used for this assessment and are based on those put forth by Saltzer and Schroeder [2], and Saltzer and Kaashoek [3], with strong influence from Smith [4]. The principles are additionally shaped by the experience of the reviewers and CTSC staff.

1. Open Design: The system should be such that secrecy of its design is not a requirement for its security. The system and its usage should be well documented.
2. Minimize Secrets: Similar to principle 1, the number of secrets in the system should be minimized and designed to provide as much security as possible (typically by maximizing entropy).

3. Defined trust and authority model: The system should have a defined system for identification of actors and well defined policy regarding their authorities and how they are trusted to behave.
4. Transparency: Do actors have appropriate ability to determine policy and be aware of actions that impact them?
5. Fail-safe Defaults: The defaults for the system should be reasonable from the perspective of cybersecurity.
6. Principle of Least Astonishment: The system, in particular the interface to its configuration and policy, should reflect the mental model of a typical user and minimize the likelihood of unintended consequences (particularly those contrary to cybersecurity).
7. Least Privilege: The system should support the notion of minimizing the privileges of each actor. In practice this principle is often under great tension with the need to provide usability.
8. Lifecycle Management: Does the system provide appropriate measures such that its security configuration is coherent with external events (e.g. users departing).
9. Coherency: In distributed systems, coherency is difficult to enforce. Does the system fail gracefully from a cybersecurity perspective when coherency is not achieved?

Recommendations

We have the following recommendations for the Globus Team and organize them by the principles given previously. We note that the mapping of recommendations to principles is not precise and exists mainly to provide context as to why a recommendation was made.

Each recommendation is annotated with a priority level - Low, Medium, or High - to guide the Globus team in addressing the recommendations. These levels are based on the subjective opinion of the authors using the criteria of how likely we expect the identified issue to create a risk, and the resulting impact if the risk were to come to fruition. The authors assumed that Globus was being used to share data that has equal and moderate risk tolerances with regards to threats to confidentiality, integrity and availability. If Globus were used for data that varied from this risk profile, our relative prioritizations would change.

Open Design

1. [LOW] Globus should provide *primary* documents, as opposed to having information scattered throughout the Globus forums (e.g. https://support.globus.org/entries/23857088-Installing-Globus-Connect-Server, https://support.globus.org/entries/23602336)
2. [MED] Globus should provide a “best security practices” document for configuring a GCS with sharing enabled. For example, when using the “globus-gridftp-server” command, document how a Resource Provider (RP) should use the:

- "-c", "-C", and "-config-base-path" arguments for specifying directories related to configuration files. We recommend using these arguments versus the default behavior. Furthermore, once these directories are created, instructions should be provided for restricting their permissions to prevent files from being overwritten.
- "-sharing-dn", "-sharing-state-dir", "-sharing-control", and "-sharing-rp" arguments.[1] Proactively restrict sharing of inappropriate files and directories. Specifically restrict or caution against sharing of all 'dot' files for users since they are less likely needed to be shared.

3. [MED] The code responsible for receiving and then processing the USER :globus-sharing: , SITE SHARING, and SITE RESTRICT commands should be independently audited since it implements new functionality not previously used in Globus GridFTP deployments.

Minimize Secrets

- No issues.

Defined Trust and Authority Model

4. [LOW] Globus Sharing is basically allowing for users to create an administrative overlay on existing administrative RPs. Globus should document a set of principles as to how these administrative domains interact, for example:
   - Does the RP administrator always have veto rights on what a user can share and with whom? *[Our proposed answer: Yes as to what can be shared, but restricting with whom seems overly complicated.]*
   - Does the RP administrator have the right to know with whom the user is sharing and who is actually accessing shared files from their service? *[Our proposed answer: Yes]*
   - Does the user have the right to privacy either of the data or sharing configuration from RP administrators in their overlay? *[Our proposed answer: No, unless the RP promises privacy as a matter of policy above and beyond a normal deployment scenario.]*
   - What is each of the 3-4 parties (sharing user, RP admin, Globus, shared-to user) responsible for and trusted to do?
5. [MED] Globus should provide a relatively simple explanatory document for users and RP administrators describing (in both non-technical and technical language) the trust model laid out in the previous recommendation.

[1] At this time, these options are only partially described at http://toolkit.globus.org/toolkit/docs/5.2/5.2.5/gridftp/admin/#globus-gridftp-server .

Transparency

6. [LOW] Globus should notify users when "they" create a share (in case it's not really them).
7. [MED] Globus should allow RP administrators to be notified when shares are created on their resources.
8. [MED] Additional logging capabilities should be added to the GridFTP server to allow configuration by an administrator to log the full details of all credentials (including all certificates in a chain both for sharing and non-sharing access) used when a client accesses the server.
9. [MED] Globus should document approaches to monitoring the sharing status of endpoints, including an endpoint's current configuration and summary/statistics of usage.

Fail-safe Defaults

- No issues.

Principle of Least Astonishment

10. [MED] Globus should pursue a tighter integration of the GCS-layer tooling for managing configuration files and the servers, particularly the GridFTP server, so that both new and experienced GCS/GridFTP administrators have a single place where the GridFTP server configuration is managed. The risk is that an experienced administrator may build on her existing experience and directly configure the GridFTP server only to later find that configuration overwritten by the globus-connect-server-setup (and other) commands.
11. [LOW] Provide a best security practices document for users that use the data sharing functionality. For example, users might append a "_share" suffix to the name of a directory being shared.
12. [LOW] The GridFTP server functionality for parsing all files in a directory to determine the server configuration (e.g."-C") should be modified so that only whitelisted files or files with a particular extension will be parsed, in order to prevent accidental misconfiguration by the presence of an additional file (e.g. copying `globus-connect-server-sharing` to `globus-connect-server-sharing.backup`).
13. [MED] As pointed out by SDSC, Globus should sanity check share-<uuid> file to ensure it properly owned and only readable by the owning user.
14. [MED] As pointed out by SDSC, shares for disabled account should not be active.

Least Privilege

15. [LOW] It should not be possible to create a share such that the configuration for sharing could be modified via the share.

16. [Ed: *This risk was removed after discussion with the Globus team, but the item number was retained to maintain consistent numbering of the issues.*]
17. [MED] It should not be possible to create shares for privileged paths (e.g. /root under UNIX variants), even if file permissions will later prevent any files from actually being shared.
18. [LOW] Deployments with lower risk tolerances should create a "sharing dock" location and configure the "-sharing-rp" configuration option for the GridFTP server so that only files and directories proactively copied or moved into the sharing dock location may be shared. Do not enable sharing of the normal user home directory.
19. [LOW] Deployments with lower risk tolerances should configure the GridFTP server so that the directory used to create sharing configuration files per user (SharingStateDir configuration option for GCS or "-sharing-state-dir" for direct GridFTP configuration) is not the default $HOME/.globus/sharing but instead an administrator-managed location. in order to control which users may or may not leverage the sharing capabilities.

Lifecycle Management

20. [MED] Shares should have a lifespan should they need to be periodically (annually seems like a reasonable default) verified by the creating user as valid.
    - The review from SDSC suggested linking lifespan to the lifespan of the proxy credential, but this seemed too short to be reasonable.
21. [LOW] Document that RPs should periodically audit the GridFTP configuration for the "-sharing-dn" option to ensure that the only value for that option is the official Globus application DN "/C=US/O=Globus Consortium/OU=Globus Online/OU=Transfer User/CN=__transfer__".

Coherency

22. [LOW] We understand the central Globus server and endpoints have their own views of what shares exist. In theory, these should always be coherent, but is there a mechanism to ensure coherency, or verify coherency and recover if coherency is lost? We imagine some corner-case failures where shares may continue to exist even if the central server believes them to be gone.

Comments on XSEDE Review

In general we find no fault with the XSEDE review and credit their documents as a basis of our own review. We do differ on one point:

- In "Globus Sharing Security" the issue of a compromised user credential being used to create a share is discussed and the proposal is made to limit share lifetime to the lifetime of a credential. We believe that would be too unwieldy and instead make

recommendations #6 (notify user when a share is created) and #20 (require periodic validation of shares).

Conflict of Interest Disclosure

During the course of this review, the spouse of one of the CTSC authors (Welch) of this document was hired by the Globus project. Integrity of the review was maintained by the inclusion of the other two CTSC authors.

References

1. Von Welch. Globus Online Security Review. Indiana University ScholarWorks. February 3, 2012. http://hdl.handle.net/2022/14147
2. Saltzer and Schroeder. The Protection of Information in Computer Systems, 1975. http://web.mit.edu/saltzer/www/publications/protection/
3. Saltzer and Kaashoek. Principles of Computer Design, 2009. http://books.google.com/books?id=I-NOcVMGWSUC
4. R.E. Smith, A Contemporary Look at Saltzer and Schroeder's 1975 Design Principles, 2012. http://cryptosmith.com/node/365

Appendices

Terminology and Acronyms

Terms and acronyms used in this document. More can be found in the online globus documentation[2].

- **Globus**: A Platform-as-a-Service (PaaS) to provide secure data transfer between endpoints. It uses the cloud (currently Amazon EC2) as its primary hosting infrastructure. (Previously known as "Globus Online")
- **Endpoint**: One of the two file transfer locations, either the source or the destination, between which files can move.
- **Shared Endpoint**: An endpoint whose files can be shared with another endpoint. For example, endpoint *mary#macbook* ("sharing") can share files with endpoint *uchicago#ITServices* ("sharee"). The sharing functionality requires a "Globus Plus" subscription, but only for the sharing endpoint, not the sharee.
- **Globus Connect Personal (GCP)**: An application that creates a Globus endpoint on your laptop or other personal computer and allows you to transfer and share files (available for Mac OS X, Windows, and Linux). In the example above, the *mary#macbook* is probably on a GCP.
- **Globus Connect Server (GCS)**: A software package that creates a Globus endpoint on multi-user systems such as a lab servers, campus research computing clusters, and other high-performance computing or storage resources. In the example above, the *uchicago#ITServices* is probably on a GCS. The primary sub-packages of GCS include Globus GridFTP, Globus MyProxy, and MyProxy OAuth.
- **ACL**: Access Control List. A list of user/process permissions for an object (e.g. file). For Globus, it consists of a Globus username, a path in a shared endpoint, and the assigned privileges (read or write) stored and managed by the Globus application.
- **CLI**: Command Line Interface. An alternative to using the web interface for Globus. (Ability to share/set ACLs still somewhat unknown from CLI)
- **Transfer API**: A REST-style interface to the Globus reliable file transfer service. The API is still in beta and is subject to change (as of this writing: https://transfer.api.globusonline.org/v0.10/doc/index.html).
- **GSI**: GSI stands for Grid Security Infrastructure and is used to describe the original infrastructure of GT security, which is comprised of SSL, PKI and proxy certificates.
- **credentials:** A combination of a certificate and the matching private key.
- **MyProxy**: A secure identity provider for GCS. MyProxy manages X.509 credentials (certificates and private keys). MyProxy combines an online credential repository with an online certificate authority to allow users to securely obtain credentials.

[2] http://toolkit.globus.org/toolkit/docs/5.2/5.2.5/

- **RPs**: Resource Providers

Anatomy of a Share

Below we detail the steps, operations, and effects of activating an endpoint, creating a shared endpoint and sharing it, and then having the share accessed by the sharee.

Activating the Endpoint

To share a directory or file a user must first authenticate to the Globus application and activate an endpoint that includes the directory or file the user wishes to share. Activating the endpoint gives or delegates to Globus a RFC 3820 proxy certificate and associated private key that the Globus application can use to authenticate to the GridFTP server exposing the directory or files. The Globus application offers a number of mechanisms for users to obtain and delegate the proxy to Globus and a full exploration of those mechanisms is not in scope here. Typically the proxy certificate has a lifetime measured in hours. Note that activation does not involve directly interacting with the GridFTP server. In order for an activated endpoint to be accessed, however, by the Globus application on behalf of the user the GridFTP server must be configured to accept and validate the proxy certificate using the well known and understood GSI protocol[3] and the the server must be capable of mapping the subject of the delegated proxy certificate to a local user account on the server. Activating an endpoint is required for sharing but is not specifically part of sharing and it is the same activation requires to use the basic Globus file transfer capabilities.

Creating the Share

After activating the endpoint a user with the Globus application sharing privilege clicks on the "Sharing" tab for the activated endpoint and then "Add Shared Endpoint" to expose a form for adding a share. The user must specify the following form inputs:

- Source Endpoint: the endpoint through which the sharing is enabled. This is the same endpoint that the user activated.
- Source Path: the path to the file or directory to be shared.
- New Endpoint Name: a new name for the shared endpoint that can be given to those with whom the user wants to share the file or directory.
- Description: A text description that will assist other users when attempting to identify the shared file or directory.

Figure 1 shows the form the user must complete.

[3] http://toolkit.globus.org/toolkit/docs/latest-stable/gsic/key/#gsicKey

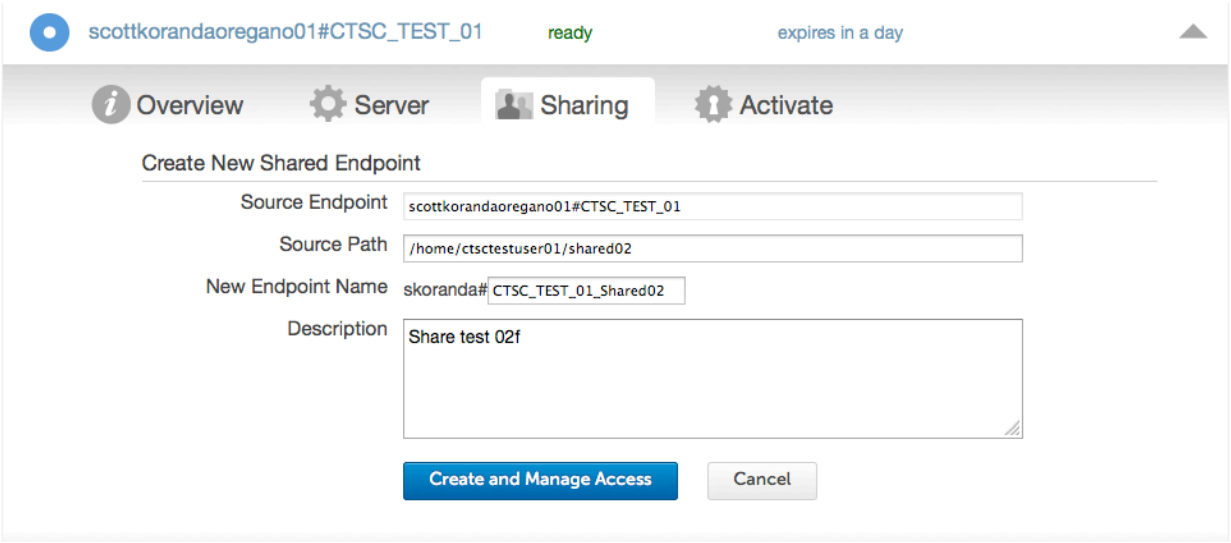

Figure 1: Globus web client: creating a shared endpoint.

When the user clicks "Create and Manage Access" the following actions occur:

1. Globus uses the user's delegated proxy credential to authenticate to the GridFTP server and is mapped to the user's local account.
2. The Globus application issues the extended FTP command SITE SHARING TESTPATH. The value passed is the value for "Source Path" that the user entered into the form to create the share. The GridFTP log file (if configured appropriately) contains an entry like this: `[11065] Wed Apr 23 15:10:34 2014 :: cli.globusonline.org: 50646: [CLIENT]: SITE SHARING TESTPATH /home/ctsctestuser01/shared02/`
3. The server consults its configuration to determine if the file or directory is allowed to be shared. If the GridFTP server is configured to allow sharing of the file or directory the server returns "250 OK.". The GridFTP server if configured appropriately will show in the log file a line of the form `[10645] Wed Apr 23 14:44:56 2014 :: cli.globusonline.org:37009: [SERVER]: 250 OK`.If the GridFTP server is configured to not allow sharing of the file or directory (using the sharing_rp configuration option) the server returns a "500 Command failed". The GridFTP server if configured appropriately will show in the log file a line of the form `[11065] Wed Apr 23 15:10:34 2014 :: cli.globusonline.org:50646: [SERVER]: 500 Command failed : Requested path can not be accessed via sharing`. Note that restrictive UNIX file permissions on the file or directory to be shared such that the user who is attempting to share the files *does not* prevent the share from being created.
4. If the server configuration allows for the share to be created and the server returns "250 OK" to the Globus client then the client opens a new connection to the server, again authenticates using the user's delegated credential, and is authorized as the user. The

Globus client then issues the extended FTP command SITE SHARING CREATE. The value passed contains two elements: (i) a uuid created by Globus and (ii) the path to the file or directory being shared. If so configured, the GridFTP log file will show a line like: `[11087] Wed Apr 23 15:15:29 2014 :: cli.globusonline.org: 53353: [CLIENT]: SITE SHARING CREATE id=04607a10-cb24-11e3-b483-22000a971261;path=/home/ctsctestuser01/shared03/;`

5. The server uses the uuid passed to it from the client to create a sharing configuration file. The location where the sharing configuration file is created is configurable and does not need to be writable by the user, allowing the administrator to control creating of sharing configuration files per user. The default location for the sharing configuration file, however, is in the directory $HOME/.globus/sharing/. The GridFTP server will create the sharing file regardless of the UNIX permissions on the directory used to hold the sharing configuration files. The name of the sharing file has the form `share-<uuid>` where the uuid is that passed by the Globus client. The sharing file is created with UNIX permissions 0400. An example sharing file is`# This file is required in order to enable GridFTP file sharing.# If you remove this file, file sharing will no longer workshare_path "/home/ctsctestuser01/shared03"`
6. The GridFTP server returns “250 OK” for a successful share creation.
7. The Globus application closes the connection with the server and creates the share object in its internal representation. The share object includes:
    a. The uuid used in the SITE SHARING CREATE call.
    b. All certificates (the full chain) that make up the delegated credential used by the Globus client to access the GridFTP server. Note that the associated private key for the proxy credential is not included (nor any other private keys).
8. Other metadata about the share including the path.

Note that at this point no other user or sharee has been given access to the shared file or directory, though any credential otherwise able to access the shared file or directory through the normal GridFTP operations has access to the file or directory.

To actually share a file or directory the user uses the Globus web application to select a Globus user as the sharee, the path of the file or directory to share (relative to the shared endpoint, with files and subdirectories below a directory inheriting), and the privileges (read or write) to give to the sharee. Figure 2 shows the view the user sees. The path(s), user(s), and privilege(s) (collectively known as the ACL) selected by the user are saved in the Globus application--there is no record about which files and directories are shared with which users sent to the server.

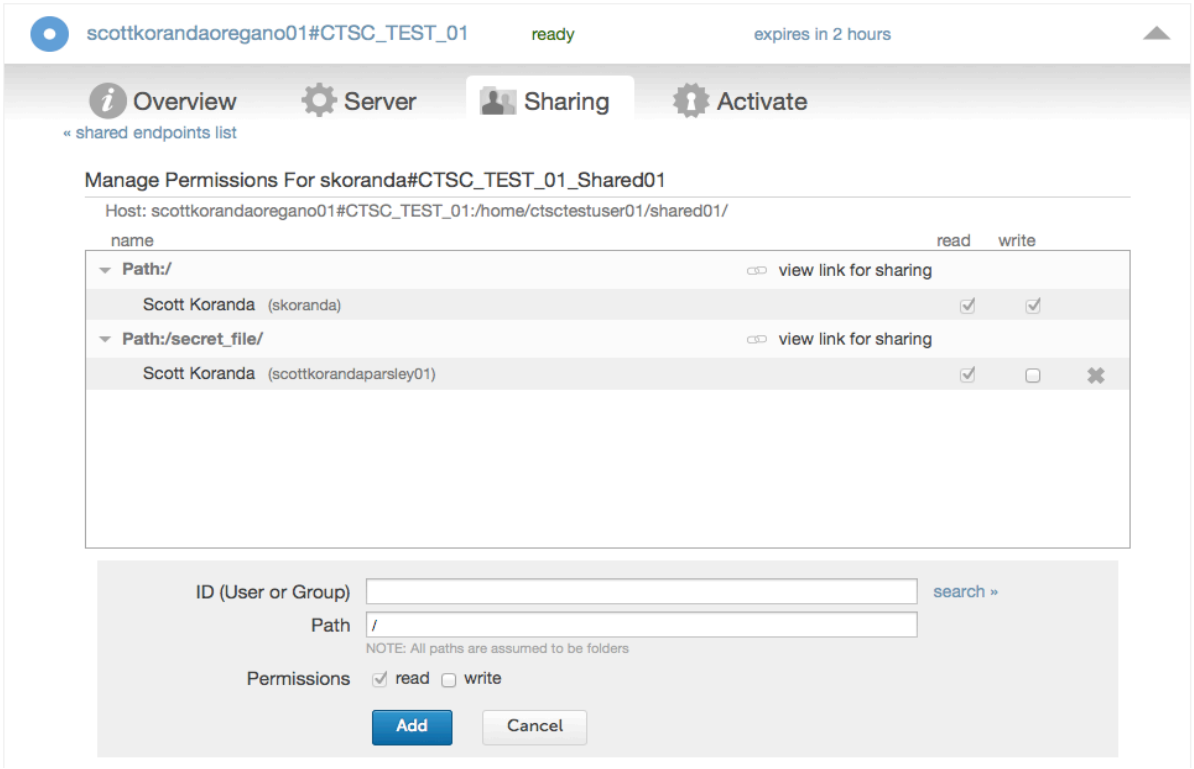

Figure 2: Globus web client: managing sharing permissions.