Building Bridges from the Campus to XSEDE

Lee Liming
University of Chicago
5735 South Ellis Avenue
Chicago, IL 60637
+1 (505) 899-4098
lliming@uchicago.edu

Ian Foster
University of Chicago
5735 South Ellis Avenue
Chicago, IL 60637
+1 (630) 252-4619
foster@uchicago.edu

Steven Tuecke
University of Chicago
5735 South Ellis Avenue
Chicago, IL 60637
tuecke@uchicago.edu

ABSTRACT

XSEDE is the integration framework for national-scale, public HPC resources in the United States. XSEDE is used by thousands of researchers at hundreds of college and university campuses throughout the country, as well as many international collaborators. Over the past several program years, XSEDE has redefined its identity management, security, and service interfaces to bridge the gap between national-scale HPC resources and campus-based computing resources. These changes make it easier for research performed on campus to access our national computing resources and make them a part of the everyday research process. We report here on XSEDE's new identity management system and how it provides a smooth bridge between campus and national identity systems. We also describe how this federated security system supports two additional bridges between campuses and national HPC services, one involving data movement and another involving scientific workflows.

1. ENVISIONING BRIDGES TO XSEDE

In 2012, the XSEDE project [1] undertook a focused effort to document the needs of its user community in a wide set of areas. One of these areas was campus bridging. The resulting description of campus bridging use cases [2] listed the following as the most useful "bridges" between campus and XSEDE systems.

1. XSEDE systems and services should allow use of campus identity credentials by supporting federated identity and authorization mechanisms.
2. XSEDE should help campus system administrators create XSEDE-like environments on campus systems to smooth their users' transitions between campus and national systems.
3. XSEDE services should provide "remote desktop" access so that users on campuses can remotely view graphical displays generated by XSEDE systems.
4. XSEDE should enable users on campuses to conduct integrated data analysis that includes data on both campus and XSEDE systems.
5. XSEDE should enable users on campuses to initiate automated workflows (scripted series of computation and file management tasks) that involve both campus and XSEDE systems and services.

The second use case (enabling campuses to make their systems "look more like XSEDE systems") is speculative: it is unclear whether campus IT administrators want to do this, and it is uncertain that XSEDE's environment would be a good fit for campus systems. Nevertheless, XSEDE makes heavy use of open source and free-license software components, and the processes that XSEDE service providers use to create commonality across their systems are open and freely available to the public. We believe that existing documentation for XSEDE systems and services is sufficient to allow anyone to replicate the most common elements of the XSEDE system environment, should they so choose.

The third use case (remote desktop access) is highly specific to the type of resource being accessed. While important in many cases, it is not universally appropriate for all services. Therefore, each service provider must design and implement the mechanism in a manner that best serves the intended purpose of their service.

The remaining three use cases—the first, fourth, and fifth—are amenable to, and in some cases require, solutions that are XSEDE-wide in nature. The rest of this paper focuses on these three use cases.

2. SURVEYING THE SITE

The first step in any construction project (for bridges or anything else) is to survey the existing site and structures and identify the pieces that must change. In 2013 and 2014, XSEDE's Architecture and Design (A&D) team was charged with documenting the existing XSEDE system as the basis for expansion plans. The authors of this paper, along with Andrew Grimshaw (University of Virginia) and Morris Riedel (Juelich Supercomputing Centre), performed this work. The output of this activity is contained in two technical reports: a brief high-level architectural overview of the XSEDE system [3] and a lengthy inventory of the system's detailed architecture and implementation [4].

The XSEDE system has one basic purpose: it allows researchers and scientists to form a relationship with the collection of national services that XSEDE provides and to maintain that relationship over time as their data and computing needs change and as the services themselves change. Six core functions support this basic purpose: identity management, interactive login, accessing remote files, submitting and managing computations, transferring datasets, and discovering and providing resource information. These six core functions are shown as the vertical columns in Figure 1.

Because XSEDE serves many different kinds of researchers and scientists (individuals, small teams, national or international collaborations) and supports several different modes of interaction (single project use, program-wide use, open-ended partnerships), the architecture offers three distinct integration approaches: the access layer (abstract user and developer interfaces for XSEDE's core functions), the service layer (abstract protocol-based interfaces for similar classes of XSEDE

services), and the resource layer (direct access to individual, specialized XSEDE systems and services). These three approaches to integration are shown as the horizontal layers in Figure 1.

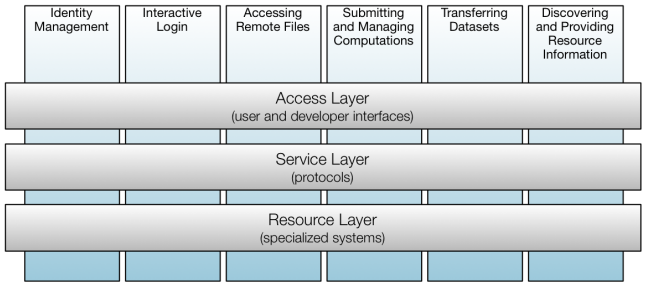

Figure 1: XSEDE core functions and architecture layers

As Figure 1 shows, each of the six core system functions can be accessed via any of the three integration layers, depending on the purpose of the integration and the nature of the relationship between the integrators and XSEDE. The access layer is intended for end users and application developers and provides software APIs (libraries and classes), Web interfaces (REST APIs), and command-line interfaces (CLIs). The service layer is for application developers and long-term partners and their system builders (including campus IT systems), and its interfaces are in the form of standardized network protocols. The resource layer is for short-term, specialized uses, and its interfaces are highly specific to each XSEDE service. Because the primary XSEDE services provide high-performance computing, the interfaces in the resource layer tend to be Unix-based command-line interfaces.

In this section, we provided a brief overview of the initial state of the system prior to our bridge-building projects. The next three sections describe the plans for each of the three bridges identified in §1.

3. BRIDGING CAMPUS AND NATIONAL IDENTITIES

The first bridge needed between campuses and XSEDE is a way for campus IT administrators to allow their users to use campus credentials to access XSEDE services. This bridge corresponds directly to the first XSEDE core function: identity management.

XSEDE's baseline system (inherited from the previous program, TeraGrid [5]) provided identity management via two key mechanisms: an XSEDE-specific userid and password, and an XSEDE-specific X.509 certificate system. The former allowed users to register as XSEDE users and access XSEDE's Web user portal. The second allowed users to obtain and use short-term X.509 digital certificates for single sign-on access to XSEDE services.

While reasonably robust, this baseline system had numerous deficiencies. The tools provided were specific to XSEDE and costly to develop and maintain, and the support across XSEDE services was uneven. Furthermore, X.509 has not achieved widespread adoption by campuses, so it could not provide a smooth bridge between campuses and XSEDE. It was clear that in order to fully support the first campus bridging need, we would need to redefine XSEDE's identity management interfaces. This task became one of XSEDE's primary development activities in 2014 and 2015. (The XSEDE project is intentionally not focused on development, but rather on operation of the existing system. Resources for development are strictly controlled by the project management. Development resources are provided for only the most critical needs.)

The route that this bridge must take had been mapped out by several earlier efforts: notably, the activities of the NSF Advisory Committee for Cyberinfrastructure Task Force on Campus Bridging [6], the formation of the InCommon community [7], and a prototyping activity in TeraGrid [8]. From these efforts it was clear that a key element would be support for OAuth (now OAuth2), the mechanism for federated authorization used widely on the Internet in both academia (InCommon) and public enterprise (e.g., Facebook, Google) [9]. Supporting OAuth2 would create a smoother bridge between XSEDE and campus systems.

OAuth2, however, is an abstract framework for authorization. It does not specify many of the details necessary to build a complete solution for XSEDE and campuses. Ultimately, XSEDE was able to form a consensus around a design that built on OAuth2 to provide the following elements.

Every XSEDE user initially registers with the XSEDE User Portal (XUP), a website accessible via any standard Web browser [10]. Users may identify themselves to XUP via any of a wide set of identity providers, including InCommon campuses and other OAuth2-based systems. (Users without access to such systems—or who choose not to use them—can register without a pre-existing identity.) Registration consists of creating an XSEDE user ID (unique for each XSEDE user) and an XSEDE password. Once registered, the user may link his/her XSEDE identity to others, such as a campus identity. This linking mechanism is based on OAuth2. Once identities are linked, the user may use any linked identity to login to XUP and gain access to XSEDE system-wide functions.

Authorization to use specific XSEDE services (e.g., a supercomputer) is granted by the operators of individual XSEDE resources. (Merely having an XSEDE identity does not allow the user to do much.) Before a user is authorized to use a national HPC system, XSEDE account management staff independently verify the user's identity. XUP provides the user interface for requesting authorization to use XSEDE resources and for managing access to the resulting "allocations."

XSEDE provides several interfaces for translating the user's OAuth2 token (obtained at XUP login) into a different kind of credential understood by a specific service interface. For example, the MyProxy interface [11] allows creation of a short-term X.509 certificate used by some legacy TeraGrid interfaces; the WS-Trust STS interface [12] allows creation of a signed SAML chain used by XSEDE's Web services-style interfaces.

Application and service developers are working on campus bridging, science gateways, connecting instruments, etc., may require CLI, GUI, or API access to XSEDE's identity services. XSEDE provides all of these mechanisms via its access layer.

Finally, for application developers and system integrators, XSEDE offers a range of APIs for identity and group management, including:

OAuth2: A widely used interface for sharing user identity and authorization information between systems.

OpenID Connect: Another widely used interface for sharing user profile information (name, email address, etc.) based on OAuth2 authorizations [13].

Globus Auth API: The interfaces used for XSEDE's user management functions, most especially for managing links between identities.

WS-Trust STS: A standard Web services interface for translating one set of credentials into another set based on access rules.

MyProxy: A legacy interface used to obtain an X.509 credential that can be used to access some XSEDE system-wide services.

The public rollout of these new interfaces in support of the "campus identity management" bridge is scheduled to be completed by the end of 2015.

4. THE DATA ANALYSIS BRIDGE

The fourth type of bridge identified in §1 concerns integrated analysis of both data on campus systems and data on XSEDE systems. The challenges that arise when trying to do this work include:

- Navigating XSEDE security mechanisms
- Moving the data from campus systems to XSEDE (or vice versa)
- Moving the results from XSEDE to campus (or vice versa)
- Keeping track of which portions of the data have and have not been moved

The first challenge is related closely to the campus identity management bridge discussed in the preceding section, and in fact, we believe that the identity management solution described there is also the solution to this part of the data analysis challenge.

The second and third challenges involve moving data between campus systems and XSEDE systems. This task can be straightforward if the data is small and in only a few files, but is often enormously difficult because the data is large (>100GB) and/or in many files. This campus bridging challenge was the topic of an earlier paper from 2012 [14]. In that paper, we reported on the use of Globus software-as-a-service to simplify the campus-to-XSEDE data analysis bridge. We will not repeat that material here, but note that the addition of the new identity management bridge simplifies the previous one even further, because now all XSEDE users automatically have the credentials needed to authenticate to Globus. There is no longer any need to maintain a separate Globus account or to login separately to Globus and XSEDE to use the methods described in that paper.

The fourth challenge is an important twist on this problem: even if it is easy to move data from campus to XSEDE and back, how can one keep track of what data is where? Without assistance from the system, a researcher or research assistant could spend hours, even days, checking the status of hundreds or thousands of files in a large dataset. Rather than repeating the entire description of how Globus solves the data movement problem, we simply note that Globus services include a synchronization feature that allows users to specify that a complete copy of all files from a given source should be created on the destination, with the understanding that only files that are not already at the destination will be copied. This capability makes it easy for researchers to assure themselves that all of the data that they expect to be on campus (or on XSEDE systems) really is there, without painstakingly inspecting hundreds or thousands of files.

5. THE WORKFLOW BRIDGE

The fifth type of bridge identified in §1 concerns automated workflows (i.e., a scripted series of computation and file management tasks) that involve both campus and XSEDE systems and services. The challenges that arise when trying to do this work include:

- Navigating XSEDE security mechanisms
- Moving data as part of file management tasks
- Allowing computation tasks to access remote data when moving the data is more costly than accessing it remotely
- Controlling the execution of computation tasks on campus and XSEDE systems

As in the preceding section, the first challenge is a restatement of the campus identity management bridging issue, and we refer to §3 for the solution. The second challenge is a restatement of the data analysis bridging issue, and we refer to §4 for the solution.

The third challenge arises when the scientific software used in the workflow expects the input or output data to be locally available (on a storage system connected to the system running the software) and it is time-consuming to make a local copy available. For example, the file may be huge (>100GB) and only part of it is needed. Or, there may not be sufficient local storage to make a local copy. In such cases, it is preferable to create a virtual file system interface [15] to access the data. To the scientific application, it appears that the data is on the local system, when in fact it is on the other end of a network connection.

To overcome this challenge, part of XSEDE's "campus workflow bridge" is a user-initiated virtual filesystem. XSEDE provides users with a thick-client interface (downloadable software) called Genesis II [16], which includes a virtual filesystem, GFFS [4 §3.4.1]. The user installs the software where the data resides (on campus or on XSEDE systems) and exports the data into the virtual filesystem, which is accessible only to themselves and anyone they choose to authorize. The science code used in the workflow sees this filesystem—and the data exported to it—as a local filesystem.

The fourth challenge listed above presents the need for a consistent interface for controlling computational tasks on campus and XSEDE systems: an interface that can be used by software rather than by humans. Specifically, the user's workflow system needs a way to remotely control tasks on campus and XSEDE computation services.

A second part of XSEDE's campus workflow bridge is therefor a remote computation interface. Like the GFFS, this interface is provided as a thick client and is also part of Genesis II. This interface accesses the UNICORE service [17] provided by each XSEDE computation service (via protocols specified in XSEDE's service layer) to initiate, queue, execute, interrupt (if necessary), and obtain the results from computational tasks on the compute service. Many workflow systems in use today have

UNICORE drivers that allow the system to access UNICORE task management services.

In summary, XSEDE's campus workflow bridge includes support for campus credentials when authenticating to XSEDE, Globus services for moving data between campus and XSEDE, Genesis II's GFFS for providing a virtual filesystem interface when necessary, and Genesis II's UNICORE client interface for remote task management on XSEDE compute services.

6. CONCLUSIONS

During its five-year program cycle, XSEDE has constructed several new and improved bridges between campus IT systems and XSEDE's portfolio of national services. The lynchpin of this work was restructuring XSEDE's identity management function to support modern federated identity and authorization mechanisms. In particular, basing XSEDE's identity management function on OAuth2—with additional features from OpenID Connect, Globus Auth, WS-Trust STS, and MyProxy—has made it possible for campus-based researchers to access national HPC services using their campus credentials. To enable this access, campus ID administrators can join the InCommon federation or provide their own OAuth2-based campus identity and authorization system. Once enabled, researchers on campus have easier access to integrated campus and national services for data analysis and scientific workflow.

7. ACKNOWLEDGMENTS

The Extreme Science and Engineering Discovery Environment (XSEDE) is supported by National Science Foundation grant number ACI-1053575.

The development of Globus services has been supported by research grants from the US Department of Energy, National Science Foundation, and National Institutes of Health; by the University of Chicago; and by a grant of computer time from Amazon Web Services. Increasingly, operation of Globus services is financially supported by its subscribers.

8. REFERENCES

[1] J Towns, T Cockerill, M Dahan, I Foster, K Gaither, A Grimshaw, V Hazlewood, S Lathrop, D Lifka, GD Peterson, R Roskies, JR Scott, N Wilkins-Diehr, "XSEDE: Accelerating Scientific Discovery", Computing in Science & Engineering, vol.16, no. 5, pp. 62-74, Sept.-Oct. 2014, doi:10.1109/MCSE.2014.80

[2] Campus Bridging Use Cases. Technical Report, April 25, 2013. (http://hdl.handle.net/2142/43882)

[3] XSEDE Architecture Overview. F Bachmann, I Foster, A Grimshaw, D Lifka, L Liming, M Riedel, S Tuecke, Technical Report, September 1, 2014. (http://hdl.handle.net/2142/50274)

[4] XSEDE Architecture Level 3 Decomposition. Technical Report, July 14, 2013. (http://hdl.handle.net/2142/45117)

[5] C Catlett, et al, "TeraGrid: Analysis of Organization, System Architecture, and Middleware Enabling New Types of Applications." In High Performance Computing and Grids in Action, L Grandinetti (ed), IOS Press, 2008.

[6] NSF Advisory Committee for Cyberinfrastructure Task Force on Campus Bridging. Final Report. March 2011. (http://www.nsf.gov/od/oci/taskforces/TaskForceReport_CampusBridging.pdf) Also available as print-on-demand book from: https://www.createspace.com/3597300.

[7] Barnett, W., V. Welch, A. Walsh and C.A. Stewart. "A Roadmap for Using NSF Cyberinfrastructure with InCommon." 2011. (http://hdl.handle.net/2022/13024) Also available as print-on-demand book from https://www.createspace.com/3630011.

[8] J Basney, T Fleury, and V Welch, "Federated Login to TeraGrid," 9th Symposium on Identity and Trust on the Internet (IDtrust 2010), Gaithersburg, MD, April 2010. (http://dx.doi.org/10.1145/1750389.1750391)

[9] Hardt, D., Ed., "The OAuth 2.0 Authorization Framework," IETF RFC 6749, October 2012. (http://tools.ietf.org/html/rfc6749)

[10] https://portal.xsede.org/.

[11] Basney, J., Humphrey, M., and Welch, V., "The MyProxy online credential repository." Softw. Pract. Exper. 35, 9 (July 2005), 801-816. (DOI=10.1002/spe.v35:9; http://web-test.ncsa.illinois.edu/~jbasney/myproxy-spe.pdf)

[12] "WS-Trust 1.4." 25 April 2012. OASIS standard incorporating approved errata. (http://docs.oasis-open.org/ws-sx/ws-trust/v1.4/errata01/os/ws-trust-1.4-errata01-os-complete.html)

[13] "OpenID Connect Core 1.0." 8 November 2014. OpenID standard incorporating approved errata. (http://openid.net/specs/openid-connect-core-1_0.html)

[14] I Foster, R Kettimuthu, S Martin, S Tuecke, D Milroy, B Palen, T Hauser, and J Braden. "Campus bridging made easy via Globus services." In Proceedings of the 1st Conference of the Extreme Science and Engineering Discovery Environment: Bridging from the eXtreme to the campus and beyond (XSEDE '12). ACM, New York, NY, USA, 2012, Article 50, 8 pages. (DOI=10.1145/2335755.2335847; http://doi.acm.org/10.1145/2335755.2335847)

[15] "Virtual file system." Wikipedia, retrieved on June 19, 2015. (https://en.wikipedia.org/wiki/Virtual_file_system)

[16] Morgan, M.M; Grimshaw, A.S., "Genesis II - Standards Based Grid Computing," Cluster Computing and the Grid, 2007. CCGRID 2007. Seventh IEEE International Symposium on , vol., no., pp.611,618, 14-17 May 2007. (DOI=10.1109/CCGRID.2007.53)

[17] D Snelling. "UNICORE and the Open Grid Services Architecture." In F. Berman, G. Fox, and T. Hey, editor, Grid Computing: Making The Global Infrastructure a Reality, pages 701–712. John Wiley & Sons, 2003.