TeraGrid Science Gateway AAAA Model: Implementation and Lessons Learned

Jim Basney
NCSA
University of Illinois
jbasney@illinois.edu

Von Welch
Independent Consultant
Champaign, IL
von@vonwelch.com

Nancy Wilkins-Diehr
San Diego Supercomputer Center
University of California, San Diego
wilkinsn@sdsc.edu

ABSTRACT
In this paper, we present our experience implementing on the TeraGrid the "Science Gateway AAAA Model" we proposed in our 2005 paper. We describe how we have modified the model based on our experiences, the details of our implementation, an update on the open issues we identified in our paper, and our lessons learned.

Categories and Subject Descriptors
K.6.5 [**Management of Computing and Information Systems**]: Security and Protection – *authentication.*

General Terms
Security

Keywords
Science Gateways, TeraGrid, SAML, PKI

1. INTRODUCTION
In 2005, we (the authors of this paper along with Jim Barlow and Doru Marcusiu) proposed a security architecture [1] to support what was then the emerging concept of Science Gateways on the TeraGrid [3]. Science Gateways have become a mechanism for increasing the impact of TeraGrid by providing high end resources to hundreds of end users through community designed and supported graphical interfaces. Support costs to TeraGrid are lowered through the use of community accounts where responsibility for security and accounting is transferred to the developers supporting the Science Gateway.

In this proposed architecture, which we refer to as the Science Gateway Authentication, Authorization, Auditing and Accounting (AAAA) Model, we replace a user's traditional remote access to a dedicated Unix account on a compute resource (Figure 1) with a model in which users indirectly access compute resources through a web portal-based Science Gateway. A Science Gateway, which typically presents a domain-specific graphical interface, accepts requests from the user and then invokes those requests on one of the dozen or so TeraGrid compute resources provided by the TeraGrid Resource Providers in an account specific to the Science Gateway, but shared by all of its users (Figure 2). This architecture means that Science Gateway users do not need to have accounts on the TeraGrid compute resources, but instead just have authorization to make requests of the Science Gateway. Our goal with this model was to allow Science Gateways to handle user enrollment and be able to do so in a manner that made the most sense for their user community, which would in turn foster broader user by larger number of users than through TeraGrid's normal allocation process.

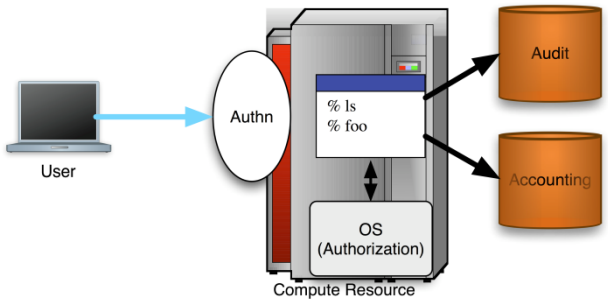

Figure 1. Traditional user access model with the user having access to a shell in a Unix account specific to the user.

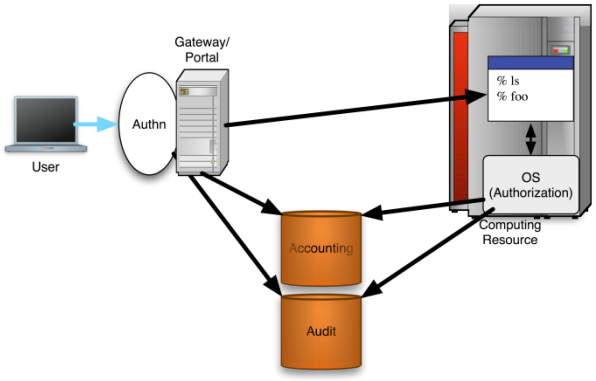

Figure 2. The proposed "AAAA Model" from our 2005 paper in which users access computational resources through a Science Gateway, which presents a limited, domain-specific interface and services user requests in a "community account" shared by all the users of the Science Gateway.

TeraGrid'10, August 2-5, 2010, Pittsburgh, PA, USA.

Since 2005, we have implemented this architecture in production on the TeraGrid with adoption by the Science Gateways and Resource Providers, and with integration into the TeraGrid accounting system. In this paper we discuss this implementation experience, including the evolution of the model, the lessons learned from real-world experience and how the open issues from the original paper were or were not resolved.

2. IMPLEMENTATION OVERVIEW

In our original paper we discussed two possible modes in which our architecture could be implemented:

- A *Transitive Mode* in which the trust from the compute resource to the user is transitive. In this mode a user authenticates to the Science Gateway and then the Science Gateway authenticates independently to the compute resource, with no information passed regarding the user from the Science Gateway to the compute resource. That is, the compute resource trusts the Science Gateway to authenticate and authorize the user without any information regarding its decision.
- A more complicated *Authorization Credentials Mode* in which the credential used by the Science Gateway to authenticate to the compute resource is decorated with some cryptographically protected proof of the user's authentication to the Science Gateway. This would provide the compute resource with greater knowledge and assurance regarding whom the Science Gateway was servicing and even allow the compute resource to authorize the request based on that information.

Figure 3 shows the implemented architecture and technology choices (which are discussed in Section 4). Our implemented architecture initially followed the Transitive Mode because this satisfied initial funding agency requirements, was simpler to deploy and could be deployed without developing any new technology. We later extended the initial architecture to fulfill the more complicated Authorization Credentials Mode with extensions to allow per-user audit and, for some job submission services, authorization on the compute resources. This later extension was driven by evolving funding agency requirements including the need to programmatically account for the number of different users submitting jobs using the Science Gateways.

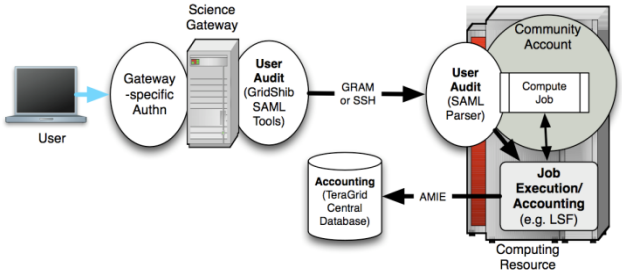

Figure 3. The implemented architecture showing updated terminology and selected technologies.

An obvious change in the implemented architecture from the original vision in Figure 2 is the lack of auditing and accounting by the Science Gateway to a central repository. Science Gateways retain the responsibility for tracking individual usage. The addition of attribute-based authentication means that TeraGrid security staff can limit access on a per-user basis rather than de-authorizing a Science Gateway and all of its users.

There also was no separate central collection of auditing information from the compute resource apart from what was collected in the accounting database. TeraGrid has not established any central audit repository. As the collection of accounting information becomes routine, this could be a service that TeraGrid offers Science Gateways, thereby potentially removing the burden for individual accounting by the Science Gateway.

The term "community account" has been adopted to describe a Unix account on a compute resource to which a Science Gateway has access and uses to invoke computational jobs on behalf of its users.

3. ORIGINAL PAPER'S OPEN ISSUES

In our original AAAA Model paper (Section 5), we had a number of open issues. In this section we describe what progress has been made on these issues. Each bullet describes an open issue from the original paper and the progress on its resolution.

- Issue: Standardization of community-resource owner agreements: In order for Resource Providers to trust the Science Gateways, we saw the need for agreements between the two parties in terms of how the Science Gateways would operate, e.g. how they would authenticate their users, what auditing they would perform, how they could be contacted in the event of an incident, etc. Ideally, instead of a series of bilateral agreements between all TeraGrid Resource Providers and Science Gateways, both communities should agree to a standard set of terms. Resolution: The two communities have agreed on a set of best practices that are documented on the TeraGrid Science Gateways web pages [2]. A Science Gateway Security Summit was held in January 2008. Science Gateway usage models were presented, as were methods Resource Providers were considering to secure community accounts. Currently efforts are focused on developing a more standardized approach for securing these accounts across the 11 Resource Provider sites.
- Issue: Policies regarding group accounts: Group accounts are not historically allowed by policy. Those policies need to be modified to allow accounts that support invocation of jobs for multiple users by Science Gateways. Resolution: Policies that previously dictated one user per account have been modified to allow for community accounts that serve all members of a Science Gateway [4].
- Issue: Restricted accounts: Given their domain-specific nature, Science Gateways are expected to only invoke a limited set of applications to service their user communities. To mitigate the impact of a compromised Science Gateway, standard mechanisms should be determined to limit applications run in community accounts. Resolution: There has been no agreement on the extent to which community accounts should be restricted and methods for doing so. Some Resource Providers handle the process manually, some use tools such as commsh [6], while others have decided the risk is minimal and have not put technical controls into place. Work toward a standardized approach across Resource Providers is scheduled for completion before the end of the TeraGrid program.

- Issue: Community Administrators: In addition to invocation of applications to service user requests, Science Gateways need to administer the contents of their accounts to install and configure applications and similar tasks. The compute Resource Providers should agree to a standard process for this administration. Resolution: Developer accounts [5] were established via the TeraGrid accounting process to allow for administration of the community accounts.
- Issue: Manageability: There need to be mechanisms for Science Gateways to manage their user communities and manage privileges with regard to access regarding the audit and accounting information generated by those users. Resolution: There has been no progress on standardization of user management by Science Gateways or accessing audit or accounting information. Once the collection of per-user Science Gateway data is complete, management tools may be developed.
- Issue: Risk Analysis: An analysis of the proposed architecture, with portals working in conjunction with computer resources, needs to be undertaken. Resolution: In 2008, under the auspices of the TeraGrid Security working group, a risk analysis was undertaken. While the results are not public, they led to many of the policies and implementation details discussed in this document.
- Issue: Standardization and broad adoption of auditing and accounting messages and interfaces: Having standard mechanisms to record audit and accounting information for both the Science Gateways and compute resources will allow for easier correlation of events and debugging of problems. Resolution: Two types of messages were standardized related to auditing and accounting: First, as described in Section 4, SAML is used to convey user identity from a Science Gateway to the compute resource. Second, AMIE [7], which was already in place to report accounting information from compute resources to the TeraGrid Central Database (TGCDB), was extended to include the user information.
- Issue: Detection of malicious activity by a Science Gateway: Given that Science Gateways should normally behave in a constrained manner invoking a limited number of applications, detection of abnormal behavior that may indicate a Science Gateway compromise should be easier to detect than with a normal TeraGrid user. Currently security issues with Science Gateways have been detected in the same way they are for command line users. Resolution: While the set of security recommendations for Science Gateways [2] includes some basic monitoring for malicious activity, there is no standardized methodology.
- Issue: Adoption: Adoption of our proposed architecture has some sociological challenges in that administrators of compute resources will effectively outsource authentication and trust they are used to managing to the Science Gateways. Resolution: Adoption of the security model has been successful as we describe in Section 5.

4. IMPLEMENTATION DETAILS

We initially deployed the "Transitive Mode" as described in our original paper. Each Science Gateway used an X.509 end entity credential specific to the Science Gateway (typically referred to as a "community credential") that it used to authenticate its requests to compute resources. Compute resources mapped the identity associated with the community credential to a community account they would create for each Science Gateway. This mode of deployment required no new software to be developed or deployed.

The two shortcomings of this model that motivated us to extend it were:

- Inability to identify individual use of the TeraGrid through community accounts. This primarily instantiated itself as a problem for reporting (to TeraGrid's funding agency, the National Science Foundation) the total number of users that were being serviced by the compute resources. This number had to be manually gathered and tabulated from the different Science Gateways, which was a tedious and error-prone process.
- A lack of authorization at the level of individual users. Compute resources could not distinguish between different users coming from a Science Gateway, so if a problem ever arose such as suspicion that a Science Gateway user's account had been compromised and was being illicitly and maliciously utilized, the only option available to the compute resource administrator would be to de-authorize the Science Gateway itself, impacting all of its users.

Because of these reasons we extended our implementation to correspond with the "Authorization Credentials Mode" described in our original paper. In this architecture the Science Gateway is responsible for generating an X.509 proxy certificate [8] for each user session from their community credential. That temporary credential is decorated with information that provides the identity of the user.

The software stack on the compute resources was then enhanced with software to parse this extension and extract the user identity. This decoration is done in a manner that did not prohibit operation by compute resources that were not augmented to parse it (i.e. it is a non-critical X.509 extension), avoiding a "flag day" when all Science Gateways and compute resources would have had to implement this at once.

The final change to support this architecture was changes to the AMIE protocol to transport the user information to the TGCDB, which records accounting information TeraGrid-wide. Resource Provider sites had to make changes specific to their accounting systems to obtain the user information and insert it into the AMIE packets they were already generating.

In the following subsections we discuss the implementation details for the Science Gateways, the compute resources and the accounting system.

4.1 Science Gateway Implementation

In our initial Transitive architecture, the only implementation requirement for Science Gateways was for them to use an X.509 credential and one of the job submission methods TeraGrid has in place that supports authentication with such credentials (GSI-enabled SSH or one of the different versions of GRAM).

According to the TeraGrid Community Account Policy [9], the Science Gateway is required to restrict the executables run in the community account to only those provided by the Science Gateway developers, since community accounts (as shared user accounts) present special security considerations for the Resource Providers. Furthermore, each Science Gateway must maintain a

user registry that contains contact information for all users accessing TeraGrid resources through the Science Gateway and collects resource usage information for each of the registered users (e.g. utilizing GRAM's audit capabilities [10]).

In the Authorization Credentials architecture, Science Gateways create a unique X.509 proxy certificate [8] for each user, generated from their original X.509 credential. This proxy certificate is decorated with information regarding the user, which is then subsequently extracted by the compute resource. This method was chosen because the decorated X.509 proxy certificate is conveyed in existing GRAM and GSI-enable SSH protocols without having to modify them.

To decorate the proxy certificate, Science Gateways install the GridShib SAML Tools software, which generates and inserts a SAML assertion containing user attributes into a proxy certificate associated with the user's session. Specifically, the SAML assertion contains a user identifier that allows the Resource Provider to uniquely associate jobs submitted using that certificate with a particular Science Gateway user. The SAML assertion may also contain the user's email address and the IP address of the user's client machine, for use in security incident investigations. Additional technical details are provided in [11].

Note that TeraGrid Resource Providers rely on the Science Gateway as a trusted entity: to properly identify users and convey that identity to the compute resource (given most users authenticate with a username and password, there is no cryptographic proof to convey), and ensure proper use of community accounts by controlling access to the community credential and limiting the executables that may run in the community account.

4.2 Resource Provider Implementation

As discussed previously, no software modifications were required by TeraGrid Resource Providers to support the Transitive architecture. TeraGrid sites already supported certificate-based access for job submission (GRAM), remote login (GSI-enabled SSH), and file transfer (GridFTP). To support Science Gateways, Resource Providers enabled community credential access to community accounts via these mechanisms.

Software modifications were required for Resource Providers to take advantage of the Authorization Credentials architecture. Since the SAML attributes provided by Science Gateways were included in a non-critical certificate extension, Resource Providers with unmodified software simply ignored the attributes. However, new "Science Gateway enabled" software capability kits were made available to Resource Providers to use these attributes.

For GRAM4, the GridShib for Globus Toolkit software [13] modifies the standard Globus certificate processing to log the SAML attributes, optionally perform attribute-based authorization decisions, and store the attributes in the GRAM Audit database [10] for later retrieval by the accounting system. The attribute-based authorization supported by the GridShib software potentially allows the resource administrator to "blacklist" a specific Science Gateway user (based on the user's SAML identity) in case of problems without denying access to other Science Gateway jobs in the community account. Otherwise, in case of problems the resource administrator may have no recourse but to (temporarily) disable the entire community account when problems occur.

With the TeraGrid's migration to GRAM5, which does not use the same underlying security implementation as GRAM4, it was necessary to separately implement SAML attribute support. At this time, GRAM5 supports storing attributes to the GRAM Audit database but not any attribute-based authorization. (We discuss in Section 6 that we have found less need for attribute-based authorization, i.e., blacklisting.)

We also learned during the deployment process that many Science Gateways prefer to submit jobs via SSH rather than through GRAM4 or GRAM5. To support these jobs, we implemented a simple script for Science Gateways to run inside the SSH session that captures the user identifier and stores it in the GRAM Audit database [14].

No enhanced implementation was done for data movement (GridFTP) since it was not deemed to be a priority as compared to job invocation.

4.3 Accounting Changes

In the initial Transitive model there were no changes needed to accounting. The allocations process however was augmented to include support for requesting community accounts. This included flagging such requests to Resource Providers so they would recognize them as such and could configure the account appropriately. This flagging was accomplished by including the word "Community" in the name of the user.

In the Authorization Credentials version of the architecture, TeraGrid Resource Providers modified their local accounting processes to include Science Gateway user identifiers in "notify project usage" (NPU) messages to the TGCDB. This process relies on the fact that the various job submission mechanisms (GRAM4, GRAM5, SSH) each store Science Gateway user identifiers in the GRAM Audit database. When constructing the NPU messages, the modified accounting process queries the GRAM Audit database to obtain the Science Gateway user identifier associated with the completed job to be included. The AMIE protocol [7] included extensibility mechanisms that allowed the additional Science Gateway attributes to be easily added.

5. STATUS OF DEPLOYMENT

As of May 2010, out of 16 active Science Gateways, 7 have performed the necessary software modifications to include attributes in the certificates used for TeraGrid job submissions, with 6 additional Science Gateways in progress. 3 additional Science Gateways use SSH-based job submissions and are waiting for the Science Gateway SSH script to be deployed at the TeraGrid Resource Providers. 4 TeraGrid Resource Providers (NCAR, NCSA, NICS, and LSU) have deployed the GRAM4 Science Gateway support and integrated it with their accounting processes. The Science Gateway support software for GRAM5 and SSH is currently being packaged by the TeraGrid software packaging team.

6. LESSONS LEARNED

In this section we describe the lessons we have learned during the process of developing and deploying the Science Gateway AAAA Model for the TeraGrid over the past 5 years.

- Avoid Technology Dependencies: When we began our project, we thought it was safe to assume the Java web services architecture of GRAM4 would become a standard mechanism for Science Gateways to submit jobs to TeraGrid.

We expended considerable effort integrating full-featured SAML support into the Java security architecture used by GRAM4. However, many TeraGrid Science Gateways continue to use GRAM2 or SSH for job submission, for performance, reliability, and other reasons. Science Gateways using GRAM2 are now beginning a migration to GRAM5. Because GRAM5 and SSH are not Java-based, it was necessary to re-implement SAML support for these other mechanisms. Looking back, it would have been better for us to implement fewer SAML features for GRAM4 and instead provide support across the different GRAM versions and SSH at an earlier stage. In addition, once the Globus team moved toward GRAM5 deployment, incorporation of changes in GRAM2 was impossible, delaying release of desired capabilities until GRAM5 was ready for release.

- Understand Participant Motivations: Typically Science Gateway developers are not funded by the TeraGrid project, so placing requirements on them (to add the GridShib software to their Science Gateway for TeraGrid) required a negotiation process with the goal of not making the bar too high to avoid discouraging their participation. Furthermore, it was difficult to motivate Science Gateways and TeraGrid Resource Providers to implement AAAA support, since the primary benefits (improved accounting for Science Gateway use of TeraGrid) accrued to the TeraGrid project rather than the Science Gateways or Resource Providers directly. Once the Science Gateways were able to successfully submit jobs to TeraGrid community accounts, it was difficult to motivate changes to the community account model (i.e., the transition from the Transitive model to the Authorization Credentials model).

- Budget Time For Software Change Management: In order to deploy new software on TeraGrid compute resource, we had to create a new release of the software and provide that release to the TeraGrid packaging team for integration into the TeraGrid software stack, which was then provided to the Resource Providers to deploy it. We found that in practice this process took months from start-to-finish due to the scheduling of effort for the various parties.

- Accepting a Good Enough Solution: The initial implementation of the Transitive architecture worked surprisingly well and pushback from security policy staff was less than we expected. All TeraGrid Resource Providers have created community accounts and serviced Science Gateway requests with them.

- Keep It Simple: We spent a lot of effort on SAML specifications, SAML metadata distribution, and attribute-based authorization, when our core requirement was to simply count Science Gateway users. In the move from GRAM4 to GRAM5 and SSH, we ended up dropping a lot of the complexity and focusing on the core requirement, to the extent that the SSH solution does not use SAML at all.

- Blacklisting Less Important: We began with a strong requirement to be able to blacklist an individual Science Gateway user, but so far we have found no need in practice for this capability. The few Science Gateway security incidents we have seen so far have impacted an entire Science Gateway rather than being specific to an individual Science Gateway user. Furthermore, the Science Gateways are generally willing to accept having their entire access shutoff during (rare) emergencies. Eliminating the blacklisting requirement allowed us to significantly simplify our software development for the GRAM5 and SSH use cases.

- Design for Incremental Deployment: Using a non-critical certificate extension allowed us to avoid a "flag day" where all Science Gateways and Resource Providers would need to upgrade at the same time. As this has been a multi-year process, it was critical that Science Gateways and Resource Providers were able to update their software independently on their own timelines.

- Web-based Science Gateways Worked Well for Restricted Interfaces: Science Gateways based on web portals are effective at providing an interface to users that only allows them to invoke a limited set of applications (and typically in a more friendly manner than a command-line interface). This was effective for ensuring Science Gateways only allowed users to run a limited set of applications as specified by the community account policy. Some Science Gateways have experimented with command-line or desktop access to community accounts, rather than portal-based access, but experience has taught us that these other access modes make it significantly more difficult to securely manage the community account.

7. CONCLUSION

We are now nearing the completion of our effort to deploy the "Science Gateway AAAA Model" on the TeraGrid. The community account model has been very successfully used by Science Gateways over the years. Along the way we moved from the *Transitive Mode* to an *Authorization Credentials Mode* to collect usage information for individual Science Gateway users. We expanded the scope of our effort from GRAM4 to also include GRAM5 and SSH access by the Science Gateways. We are relying on the GRAM Audit capability across GRAM4, GRAM5, and SSH to capture Science Gateway user attributes as part of the TeraGrid accounting process. Deploying the "Science Gateway AAAA Model" on the TeraGrid required a collaborative effort across many TeraGrid participants, particularly from Science Gateway developers and Resource Provider staff.

8. ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under grant number 0503697.

Tom Scavo and Terry Fleury both contributed significantly to our overall system design and implementation. Jon Siwek provided additional support. Stuart Martin and Joe Bester implemented the GRAM5 support. Michael Shapiro was instrumental in completing the integration with TeraGrid accounting and AMIE. Numerous Science Gateway developers and Resource Provider administrators helped by installing and testing software.

9. REFERENCES

[1] Von Welch, Jim Barlow, James Basney, Doru Marcusiu and Nancy Wilkins-Diehr. A AAAA Model to Support Science Gateways with Community Accounts. Concurrency and Computation: Practice and Experience, 2006. http://dx.doi.org/10.1002/cpe.1081

[2] Security and Accounting for TeraGrid Science Gateways. https://www.teragrid.org/web/science-gateways/security

[3] Science Gateways Home. https://www.teragrid.org/web/science-gateways/home

[4] Science Gateways for Developers. https://www.teragrid.org/web/science-gateways/developers

[5] TeraGrid Allocations and Accounts for Gateways. https://www.teragrid.org/web/science-gateways/allocations

[6] Restricted Community Accounts. http://security.ncsa.uiuc.edu/research/commaccts/

[7] Grid-Based Account Management using AMIE. http://scv.bu.edu/AMIE/

[8] Steven Tuecke, Von Welch, Doug Engert, Laura Perlman and Mary Thompson. Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile. RFC3820, 2004.

[9] David Hart and Nancy Wilkins-Diehr, TeraGrid Community Account Policy (TG-10), February 2007. http://teragridforum.org/mediawiki/index.php?title=TG-10

[10] Stuart Martin, Peter Lane, Ian Foster, and Marcus Christie, TeraGrid's GRAM Auditing & Accounting, & its Integration with the LEAD Science Gateway, TeraGrid'07.

[11] T. Scavo and V. Welch. A Grid Authorization Model for Science Gateways. International Workshop on Grid Computing Environments, 2007. http://casci.rit.edu/proceedings/gce2007

[12] Welch, V., I. Foster, T. Scavo, F. Siebenlist, C. Catlett, J. Gemmill and D. Skow. Scaling TeraGrid Access: A Testbed for Identity Management and Attribute-based Authorization. TeraGrid'07.

[13] GridShib Project Home. http://gridshib.globus.org

[14] Gateway-Submit-Attributes Script. http://www.teragridforum.org/mediawiki/index.php?title=Gateway-Submit-Attributes