Trust as the Foundation of Resource Exchange in GENI

Marshall Brinn
Raytheon BBN Technologies
Cambridge, MA
mbrinn@bbn.com

Nicholas Bastin
University of Houston
Houston, TX
nbastin@uh.edu

Andrew Bavier
Princeton University
Princeton, NJ
acb@cs.princeton.edu

Mark Berman
Raytheon BBN Technologies
Cambridge, MA
mberman@bbn.com

Jeffrey Chase
Duke University
Durham NC
chase@cs.duke.edu

Robert Ricci
University of Utah
Salt Lake City, UT
ricci@cs.utah.edu

ABSTRACT
Researchers and educators in computer science and other domains are increasingly turning to distributed test beds that offer access to a variety of resources, including networking, computation, storage, sensing, and actuation. The provisioning of resources from their owners to interested experimenters requires establishing sufficient mutual trust between these parties. Building such trust directly between researchers and resource owners will not scale as the number of experimenters and resource owners grows. The NSF GENI (Global Environment for Network Innovation) project has focused on establishing scalable mechanisms for maintaining such trust based on common approaches for authentication, authorization and accountability. Such trust reflects the actual trust relationships and agreements among humans or real-world organizations. We describe here GENI's approaches for federated trust based on mutually trusted authorities, and implemented via cryptographically signed credentials and shared policies.

Categories and Subject Descriptors
C.2.1 [**Computer Systems Organization**]: Network Architecture and Design – *Network communications, network topology, distributed networks*

*C.2.3 [***Computer Systems Organization**]: Network Operations – *Network monitoring*.

D.4.6 [**Software**] Security and Protection - *Authentication*

K.6.5 [**Management of Computing and Information Systems**]: Security and Protection - *Authentication*.

TRIDENTCOM 2015, June, 2015, Vancouver, BC, Canada.

General Terms
Design, Reliability, Experimentation, Human Factors, Management

Keywords
Federation, Trust, Network, Testbeds, Cloud, Authentication, Authorization, Policy

1. THE CHALLENGE OF RELIABLE RESOURCE EXCHANGE
Network and computer science research typically requires the use of large sets of resources, configured in particular topologies often spanning wide distances. Most researchers do not have ready access to such resources within their campuses or through their professional connections. Such resources are available, however, across large regions and varied administrative domains such as state, national or international institutions.

The potential exists, then, for a market through which the exchange of resources can take place. There is a healthy demand for resources from researchers, experimenters and educators; on the other side, there is a healthy supply of resources that could be made available to such potential users. However, both sides of the transaction have reason for hesitation.

The researcher needs assurances that the resources are *reliable* along several dimensions:

- *Maintenance*: the resources are well maintained, likely to have acceptable availability and provide support if there are issues or questions
- *Performance*: The resources are likely to provide reasonable performance and more important, observed performance consistent with the provider's claims.
- *Security*: Including resources from this provider will not introduce vulnerabilities into the broader topologies (e.g. viruses, worms, attacks).

The resource provider needs assurances to the experimenter's integrity:

- *Identification*: The experimenter must be known and known to be reliable to the resource owner will do no harm to his resources or to any of the other users of his resources:

TRIDENTCOM 2015, June 24-25, Vancouver, Canada

DOI 10.4108/icst.tridentcom.2015.259683

- *Limitation*: The resource owner can establish and enforce policies on who can get resources and in what quantities and circumstances
- *Protection*: The resource provider can determine when there is some misbehavior on some resources, isolate or shut down those resources and hold, as necessary, the users of these resources responsible.

What is needed is a mechanism to establish mutual trust between prospective consumers and providers of resources. A particular researcher and resource owner may know and trust one another. However, requiring such pair-wise trust relationships to build large experimental topologies will not scale (i.e. M*N for M experimenters and N resource providers).

Establishing trusted exchange of resources between owners and experimenters requires these elements:

- *Authentication*: There are mechanisms to ensure that the requesting party and provider own or have access to particular resources and accounts.
- *Identity*: There are mechanisms assuring that all parties are who they claim to be
- *Authorization*: There are mechanisms to limit access to particular resources or actions on those resources by policy and based on the identity and attributes of the requestor.
- *Accountability*: If something goes wrong with either a resource or an experiment, there are mechanisms to identify such issues, contain the damage and identify the responsible party or parties.
- *Reliability*: Both the experimenter and resource owner need some trust in the basic integrity and trustworthiness of the other.

Establishing such notions of identity and reputation among parties who do not know each other directly requires a trusted third party.

2. GENI APPROACHES TO FEDERATED TRUST

The NSF GENI (Global Environment for Network Innovation) project has worked to establish such a context for the trusted exchange of resources for the purpose of networking research. GENI provides a framework to:

- Vouch for the experimenter's and the resource provider's identity and reputation
- Provide information about the experimenter on which to make authorization decisions
- Monitor experiments, provide alert, shutdown and forensics services, revoke privileges to resources as needed.

GENI provides an architecture for building federations among experimenters and resource owners. Such federation is a human activity of parties that join together to commit to common conventions, interfaces and procedures to support the trusted exchange of resources. The NSF GENI Federation is a *particular instance* of that architecture, comprising a broad and growing range of experimenters and resources, including resources owned and managed by GENI and resources shared from the Emulab [3], ExoGENI [13] and PlanetLab [5] federations.

2.1 GENI Principals

The allocation of resources is a transaction between people and organizations that supply and demand access to computing, storage and network resources. GENI represents these entities with the following principal concepts.

- *Aggregate*: An infrastructure hosting (IaaS) provider, provisioning resources on request based on its own policies, priorities and constraints.
- *Member*: GENI users or tenants who may allocate and control virtual infrastructure elements spanning multiple aggregates, and link them together to form end-to-end topologies for experiments or networked applications.
- *Sliver*: A virtual resource unit that is provisioned from a single aggregate and is named and managed independently of other slivers. Each sliver is bound to exactly one slice at the time that the sliver is created.
- *Slice*: A logical container for a set of virtual infrastructure resources. The slice abstraction is useful to name, control, and contain groups of virtual resources that span multiple provider sites and are allocated and used for a common purpose. A slice belongs to exactly one project at the time the slice is created.
- *Project*: A logical grouping of slices, often under the management of a common lead or PI representing the work of a single lab or organization. By GENI policy, the project lead is accountable for activities taken on slivers within the project's slices.

As noted below, each principal is uniquely identified to allow reliable authentication and provide granular authorization and accountability.

2.2 GENI Services

GENI defines several classes of ``authority'' services that coordinate identity management and authorization. These services are decentralized: each authority service may have multiple instances, and the set of instances may change over time.

- *Clearinghouse Services*: Establish federation-level authentication and authorization for experimenter use of federation resources. Most notably, these services include:
    - Member Authority (MA): manages federation member identity, attributes and associated credentials. In this case, each experimenter is a member at some MA.
    - Slice Authority (SA): manages creation of slices and projects, and manages memberships and roles of members within slices and projects.
    - Service Registry (SR): provides a directory of all federation services associated information for connecting to them (e.g. URL)
- *Monitoring Services*: Process and tools monitoring activity on GENI resources for health, performance and adherence to policies.

The experimenter and resource owner are also represented by software entities within GENI. The resource owners are represented by *Aggregate Managers*, which speak the Aggregate Manager (AM) API to negotiate allocation and management of resources at each infrastructure site (an *aggregate*). The experimenter is represented by a range of *tools* which speak the Federation API to Clearinghouse authorities and the AM API to Aggregate Managers. The relationship among GENI principals is represented in Figure 1.

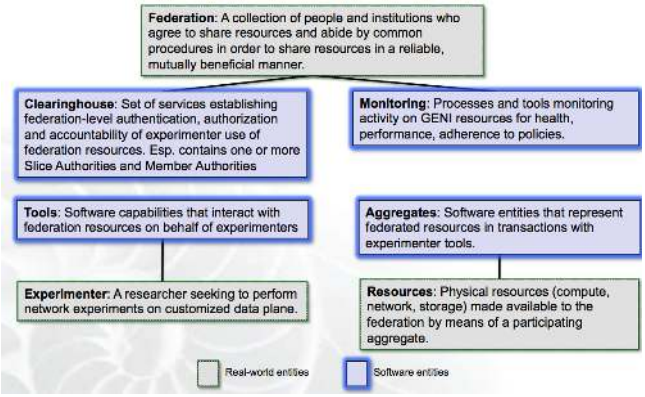

Figure 1. Principals of GENI and the software entities that represent their interests.

A given Federation is represented by a Clearinghouse. The Clearinghouse has a single Service Registry but may contain multiple Slice or Member authorities. Authorities of the same type (e.g. two Slice Authorities or two Member Authorities) are independent of one another, managing distinct sets of users and slices. An aggregate may belong to multiple federations or accept credentials from all or some of the authorities of a given federation.

2.3 GENI Identifiers

GENI defines a structured symbolic name space (URNs) for GENI objects and principals. Each entity in GENI is associated with a single domain: the entity's URN has a prefix that is the name of its containing domain. The domain name is presumed to be globally unique and stable: in practice a DNS name is used. The URN for an object or principal incorporates a user-assigned common name that is unique within its containing domain. Some of the semantic content of a GENI credential is encoded in the URNs. In particular, object URNs encode the type of the object and essential relationships among objects. For example, the URN of a slice encodes the name of its containing project, and the URN of a sliver encodes the name of its containing slice. Moreover, it is possible for authorities to subdivide their own namespace, issuing CA certs valid only for that namespace. For example, one might implement projects or other groups by having a high-level SA delegate to sub-SA's representing these groups, which issue their own slices.

GENI uses an ordinary X.509 PKI hierarchy to bind public keys to principal URNs. A root Certifying Authority (ch.geni.net) ssues X.509 identity certificates endorsing public keys for the domains; the domains then issue certificates endorsing public keys for their object authorities and other principals. The distinguishing name in a GENI identity certificate is a GENI URN.

A certificate for a URN is accepted only if its speaker is the parent in the URN name space, i.e., the certificate is signed under a public key of the parent. Given the URN of a principal one can derive syntactically the URN of its parent in the identity hierarchy. Given the URN of an object, one can derive the URN of its controlling authority.

One problem with symbolic names (e.g., URNs) is that they are not guaranteed to be unique through time: users may assign the same common name to different objects at different times. The AM API 3.0 standard resolves this problem by adding a unique machine-generated identifier (a 128-bit UUID/GUID constructed according to IETF RFC 4122) for each object; certain request APIs now use the URN and the UUID together to assure uniqueness.

GENI uses a separate credential format for speaks-for assertions. GENI also defines an object credential format containing the object's URN and various essential attributes of the object—or the rights that some principal has for the object—depending on the type of the object and certificate. A related GENI standard defines a certificate format to delegate named rights over objects, based on the SFA capability standard.

2.4 Trust Relationships in GENI

Having introduced the different players and interactions between these players in GENI, we can discuss the different types of trust relationships in GENI and how we rely on them to support a trusted framework for resource exchange.

First we note that 'trust' means different things in difficult contexts and are represented differently in the GENI architecture:

- *Credibility*: "If you claim it, I believe it". That is, I accept your assertions as true. This is typically manifested by installation of your trust roots in my bundle and the incorporation of trust statements in policy.
- *Endorsement*: "I vouch for you to others". This is typically represented by the appearance of an entry in a directory service or membership in the federation or granting of particular privileged credentials.
- *Reliance*: "I believe you can do something as I would want it done". This is typically the domain of Speaks-For credentials and Delegation credentials and the trust between people and software or people and other people.

In GENI we support the following types of trust relationships among entities, summarized in the following figure, each to be described below. [Figure 3] The decision to enter into a trust relationship happens first in a real-world, human context, where an appropriate due diligence process results in a formal or informal agreement. It is then represented in a software setting, in one of these ways.

Clearinghouse (CH) trusts User [*Endorsement*]. The members of the Federation vet prospective members to validate their credentials and identity. If validated, a Federation Member authority mints an SSL certificate for that user, attesting that the bearer of the corresponding private key has these attributes (URN, UUID, email) but also that the person passed the Federation's vetting process. This certificate allows access to GENI services and represents a statement of trust of the Federation in this person. In GENI, this trust is based on a verified identity and confirmation that the person in question is qualified to be a project lead (typically a university faculty member), or under the supervision of a qualified project lead. If that trust is broken, (e.g., upon leaving a research team or for misbehavior) the certs can allowed to expire or, by out-of-band action, revoked.

Aggregate (AM) trusts Clearinghouse [*Credibility*]. An aggregate joins a federation by including the root certificate of that federation in its trusted set. Connections to the aggregate are validated against the trusted set and only connections that can be resolved to one of the trusted roots will be accepted. Slice

Credentials and User Credential are validated against the same trust set: "If the MA trusts this user or the SA validates the user's rights at a slice, I will too". Aggregates can be members of multiple federations by including each federation's root certificate in its trusted set.

Clearinghouse trusts Aggregate [*Endorsement*]. Members of a federation trust AMs in that they vet them for proper operations and capable management and thus vouch for them. In GENI, this trust is established when the aggregate provider executes an aggregate provider agreement, which defines the provider's responsibilities towards the federation and users. These aggregates are included in the Federation's Service Registry. Users do not, themselves, need to trust the aggregates: they can rely on the endorsement from the SR. In this way we solve the M*N problem noted above: For M users and N aggregates we need only M+N trust relationships.

User trusts Tool [*Reliance*]. GENI tools enable users to invoke resource allocation and management API's while hiding, for some users, details such as key management, resource specifications and API specifics. In GENI, we distinguish two classes of tools:

- *Desktop*: These tools 'speak as' the user using the user's cert/key and run under the user's control. These run on the user's own computer: the private key never leaves the machine.
- *Hosted*: These tools 'speak as' themselves using their own cert/key and 'speak for' the user using a Speaks-For credential. A Speaks-For Credential is a statement signed by the user that they authorize the tool to speak on the user's behalf, thus maintaining the protection of the user's private.

User trusts Clearinghouse [*Reliance*]. The trust a user has in the Clearinghouse services is manifested in two ways:

- Hosted tools will validate the CH service's cert much as the CH validates the user's cert. This gives the user's assurance of correct HTTPS authentication.
- By directing tools to interact with the CH services (Slice Authority, Member Authority, Service Registry, etc.) the user implicitly trusts their correct function.

These trust relationships are summarized in Figure 2.

Trusting entity \ Trusted entity	USER	TOOL	CH	AM
USER		Reliance	Reliance	
TOOL		Reliance		
CH	Endorsement			Endorsement
AM			Credibility	

Figure 2 Trust relationships in GENI: who trusts whom.

2.5 Credentials and Trust Roots

GENI uses a number of signed statements or credentials in supporting authentication and authorization services. GENI authentication is based on standard private key infrastructure methods and every participant in GENI has a distinct key pair.

Certificates are statements saying, in effect, "The bearer of the private key associated with this public key has these attributes: UUID, URN, email, …". In GENI, these are represented in X.509 [9] format and are signed by the federation Member Authority.

These credentials are the foundation of authentication in GENI. All API calls (to aggregates through the AM API or to the Clearinghouse authorities through the Federation API) are authenticated (e.g. using SSL) using the caller's certificate and private key. [9]

The federation contains a set of trust roots, which are X.509 certificates of the federation authorities (including the federation CA or certificate authority) whose corresponding keys signed the federation credentials. The act of an aggregate joining a federation (an aggregate may belong to more than one federation) includes adding the federation's trusted root certificates to its own bundle of trusted root certificates. In so doing, any SSL connection made by members using certificates signed by a trusted authority are accepted.

GENI creates and acknowledges credentials in one of two formats:

- SFA Credential: These are credentials granting privileges or roles to a given user in a given context (role-based access control or RBAC) and conform to the SFA format. [7]
- ABAC Credential: These are credentials asserting attributes about a given user (e.g. "X is a member of the faculty at Y"). This supports attribute-based access control, hence ABAC and conforms with the standard ABAC format [1].

In these credential formats, several other types of credentials are common in GENI, which will be relevant for the upcoming discussion of authorization.

- User Credential: Statement from the MA regarding roles and rights of a user independent of slice or resource context. This is in SFA format.
- Slice Credential: Statement from the SA regarding roles and rights of a user in a particular slice or resource context. This is in SFA format.
- Attribute Credential. Statement from the MA or SA representing an assertion about a user, a superset of what can be expressed in User Credentials. These are in ABAC format.

Finally, GENI supports two different forms of credentials that support transferring or rights or privileges from one entity to another:

- Speaks-for Credential: A statement that "I grant this tool or user to speak on my behalf, in the specified context". The signer of the credential is accountable for any actions taken as a result of this grant. This credential allows the bearer to act on the signer's behalf without exposing the signer's private keying material.
- Delegation Credential: A statement that "I grant a particular right or privilege of mine to this other user. The bearer of the credential is accountable for any actions taken as a result of this grant.

2.6 GENI Authorization Pipeline

Authorization in GENI requires the combination of policies and credentials and an engine that can make authorization

determinations based on these. Credentials, as we've seen, are statements about the attributes or privileges of a given user. When the user seeks to invoke the Federation API [4] or the Aggregate Manager API [2], the presence of the user's certificate signer in the server's trust root is sufficient for authentication. For authorization, however, we want to make sure the entity is not merely recognized but is entitled to perform the requested action.

We therefore gather attributes from the user based on his certificate, federation-supplied credentials and user-supplied credentials (signed by federation authorities but provided out-of-band). A SliceCredential, for example, is a signed statement about the rights/privileges of a given member in the context of a slice. These credentials are matches against policies, which are statements about which attributes are required in which context to allow a given action. Such policies are software encodings of agreements made between the real parties (people and organizations) involved.

For example, we may have a policy at an SA that says, "Only members of a slice may obtain a SliceCredential". In such a case, that policy would allow creation of a SliceCredential for a given slice only in the presence of a credential that asserts that the requestor is a member of the given slice. Alternatively, we may have a policy at an AM that says, "Only members of federation X may allocate Y bandwidth on a given circuit". In this case, the policy would allow the satisfaction of a request for bandwidth only with a credential that asserts membership in a given federation. This pipeline for GENI Authorization is illustrated in Figure 3.

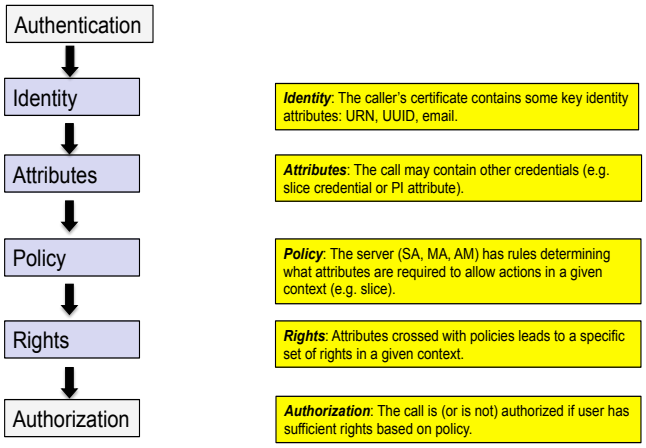

Figure 3. The GENI authorization pipeline. All decision logic are based on attributes and policies.

2.7 Trust Credentials at Work: An Example.

We describe here the interaction among a user tool, a Clearinghouse service and aggregate manager to perform mutually trusted operations between the experimenter and resource owner.

Rather than invoking the AM API call initially to the aggregate, the user contacts the Slice Authority for a slice credential. If the user is a member of good standing of the federation and the slice in question, the slice credential will be provided. Then the user will include this credential in the AM API call (in this case, a query for the set of resources on an aggregate for a given slice). The aggregate will authenticate the user by validating the caller's cert against its trust roots, but will authorize the call based on the presence of the Slice Credential signed by a trusted Slice Authority. If both authentication and authorization tests are satisfied, the AM will perform the requested operation and return the requested information. This flow is illustrated in Figure 4.

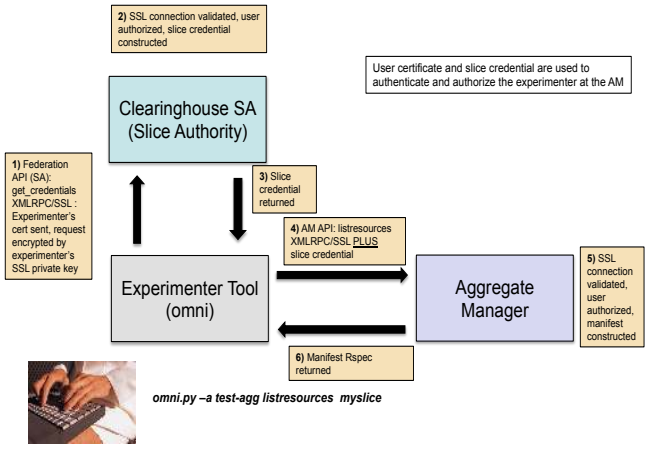

Figure 4. Trust Credentials at Work: Getting a slice credential in a desktop tool context.

A slightly more complex example involves a hosted tool making the same request on behalf of a user. In this case, the user creates a Speaks-For credential for the tool, and the tool uses it to speak to the Slice Authority and the Aggregate Manager on behalf of the user. The Slice Authority and Aggregate Manager are willing to treat the caller as if they were the user (for purposes of Authorization) because of the presence of a Credential signed by the user authorizing them to do so. This flow is illustrated in Figure 5.

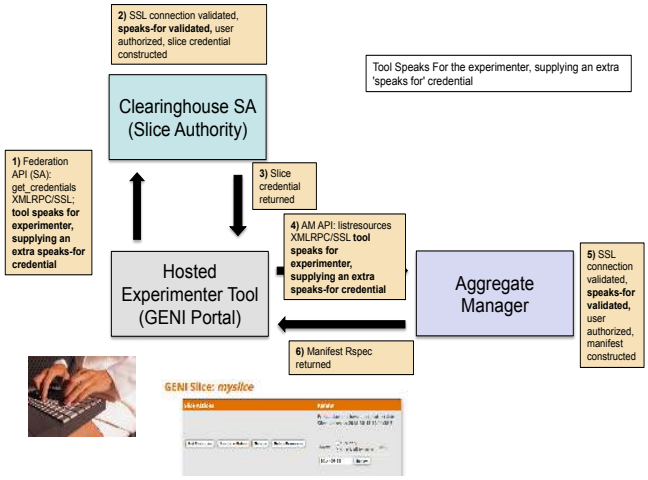

Figure 5. Trust Credentials at Work: Getting a slice credential in a hosted tool context.

In an additional use case, a researcher may authenticate to an MA with a password under a Web identity (SSO) protocol such as OAUTH or Shibboleth[16]. The MA issues credentials based on authenticated attributes bound to the user identity. For example, GENI runs a portal service that harvests attributes about each academic user from a Shibboleth identity provider at the user's institution. Once logged into the portal with an institutional identity, the user may use Web forms to supply additional information to the portal and to accept required conditions. If the user provides a public key in the authenticated session, the portal may issue endorsements to approve the user as a GENI user

and/or principal investigator (PI), based on attributes supplied by the institution via Shibboleth/ SAML (e.g., user is a faculty member). This example demonstrates one way that GENI bootstraps trust from services outside of GENI.

2.8 Accountability Foundations

GENI accountability seeks to provide assurances that misbehavior (intentional or unintentional) within resources allocated to an experimenter can be detected and the damage minimized and that future such incidents can be made less likely. GENI has a variety of processes, policies and procedures that ensure that experimenters can, if necessary, be held accountable for actions taken on federation resources.

To that end, GENI accountability rests on the following pillars:

- **Monitoring**: Gather data from Aggregates and Clearinghouse services on current system state including: current relationships among users, slices, slivers (allocated resources) and aggregates and time-series of real-time network, compute and storage resource metrics.
- **Alerting**: Determine potentially problematic behaviors or metric patterns on or across aggregate resources.
- **Forensics**: Determine what happened and who is responsible for these resources (experimenter, slice owner, project lead).
- **Response**: Depending on the severity and time-criticality, there are a number of options, including: sliver isolation, account disabling, certificate non-renewal, certificate revocation (and thus membership revocation).

The following simple use case may help illustrate how these system functions coordinate to provide accountability within GENI:

- A researcher creates a project for running a series of experiments.
- A student in the experimenter's lab requests a GENI slice within the project containing a set of resources on which to perform a particular experiment.
- The creation of this topology is registered in the GENI monitoring topology database, so that we know which 'slivers' (allocated resources) have been created for which users, including both the compute nodes and network links and paths.
- The aggregates report ongoing performance data to the GENI monitoring metrics database as well, so that network, CPU and memory data on a given virtual or physical resource can be measured and catalogued.
- The student deploys some service on the topology that starts a network storm.
- GENI monitoring automated processes to determine exceptional loads and has alert thresholds to bring humans into the loop when such loads are encountered.
- The GENI monitoring team use the topology database to determine which experimenter created the slivers on which the abnormal activity is taken. The monitoring team can contact the student directly, as they have his email address form the time the student created his/her GENI account. Alternatively, the monitoring team may contact the researcher directly. In either case, the team will attempt to determine the cause of this load and whether this is acceptable or expected behavior or not, and the intent.
- The team may then take corrective action depending on the severity and intent of the impact of the incident, including:
    - Leaving the service up until completion (if it is not interfering with other experiments)
    - Asking the student to bring down the service
    - Shutting down the student's slice (freezing all resources pending future investigation
    - Revoking the student's credentials and membership in GENI on a temporary or permanent basis
    - Revoking the researcher's credentials and membership in GENI on a temporary or permanent basis.

2.9 GENI Design Motivations

There are many approaches to handling identity, policy and accountability, and many have been used successfully in a range of similar applications. This section seeks to describe particular GENI requirements that have driven some of GENI's design choices in these areas.

GENI has users representing a broad spectrum of experience and sophistication and comfort with materials such as SSL certificates or SSH keys. Some understand these and want to manage their materials themselves; others don't want to know about these things and want GENI to take are of these materials for them. GENI has thus supported users who want to upload a CSR (certificate signing request) to get a certificate, as well as creating a new cert/key pair for those users who do not. Likewise, GENI allows for uploading public SSH keys while also supporting users who want GENI to create an SSH key pair for them. GENI automatically loads the SSH public keys onto any allocated resources so that allocating users can readily logged into their resources.

"Key hygiene" is a matter of great concern to sophisticated users: particularly not wanting private keys to leave their local machine. However, novice users would prefer convenience to security in this regard. GENI's approach to 'speaks for' credentials allows users to let tools represent them without having to divulge any private key materials, and in a way that doesn't require novice users to understand too much detail about PKI infrastructure.

The majority of GENI users are grad students and undergraduates. They have made it clear that they want to remember and manage as small a number of passwords as possible. This has driven our SSO design of the GENI portal by which people can log into their home institution's IdP, and if that IdP is a member of the appropriate class in the InCommon federation, GENI will accept the IdP's authentication to provide entry into GENI.

GENI racks reside a range of sites including regional and national backbone networks. Most GENI racks, however, reside on academic campuses and thus campus CIO's and IT staff are particularly concerned about letting people outside the campus have access to local resources. This concern is particularly strong when we talk about connections to non-GENI campus resources or federations with similar testbeds outside the US. Our quota-based policy approach is intended to allow campus managers to specify the access they want to provide local and external users.

Similarly, the requirements of campus CIO's and IT drive our approach to accountability so that they can be assured that damage due to misbehaving experiments on or connected to their resources can be limited.

2.10 System Validation and Evaluation

GENI is a program still being developed and growing rapidly. As such, it has been challenging to take a snapshot of GENI and assess its performance or validate its approaches regarding trust. That said, GENI undergoes ongoing regression tests for function, scaling, longevity, performance and security.

- We perform a set of acceptance tests on all racks and resource aggregates, pushing all resources on the path towards 'production'.
- We have functional validation on production GENI racks to ensure that experimenter and monitoring features delivered required functionality.
- Robustness tests have been run to repeatedly and aggressively handle resource requests for up to 48 hours.
- Virtual Machine (VM) allocation limits are tested within each rack, and we have successfully allocated up to 130 VM's on all GENI racks.
- VLAN allocation limits across AL2S (the Advanced Layer 2 Service of Internet2) are tested by continually creating and breaking connections and exchanging traffic.
- Longevity tests have been run on particular topologies that have been continuously running and exchanging traffic for two months and counting.
- The GENI user community has grown tremendously over the last several years, as indicated in Figure 6, demonstrating the scaling of the management of authentication processes and materials.

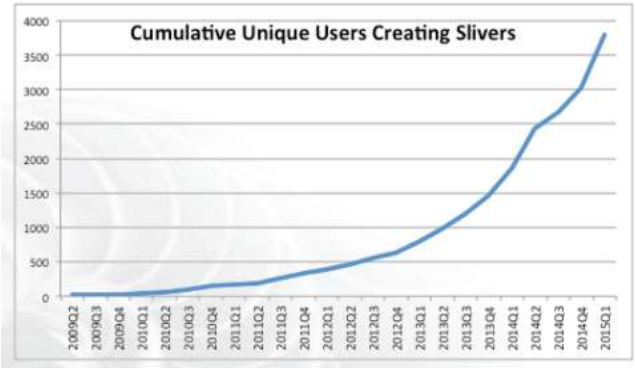

Figure 6. Growth of GENI users and corresponding identification and authentication/authorization mechanisms since 2009.

3. FUTURE DIRECTIONS

The GENI project continues to work on enhancing the infrastructure for enabling trustworthy exchanges of experimenter resources. Three current focus areas may be of particular note:

Aggregate Policy. Currently most GENI Aggregates rely strictly on SFA credentials to support their authorization decisions. We intend to support and advocate for using ABAC credentials and readable policies to guide these decisions. We expect such policies to reflect an amalgam of both federation and aggregate-local concerns.

Certificate Revocation. GENI plans to implement the full SSL certificate revocation functionality to support much more immediate isolation of habitually or egregiously misbehaving actors. We are working to maintain and publish a CRL (Certificate Revocation List) to aggregates so that even users holding a valid and unexpired certificate will be rejected immediately, without needing to wait for certificate expiration. This issue has not been a major concern in this stage of GENI's development, but we expect that it will need to be addressed as the number of users and resources continue to scale.

Credential Discovery. One shortcoming of a credential-based authorization system is the difficulty, in general, of finding all credentials that may be relevant to a particular authorization system. Recent research in shared credential repositories [14] provides hopes that such credential discovery might be available to GENI services and minimize the requirement on the part of tools or services to anticipate up-front all relevant credentials to an authorization decision.

4. RELATED WORK: GRID COMPUTING SYSTEMS

Grids emphasize federated environments similar to what we propose: they combine multiple resource providers in a unified service for a community of users that may span multiple identity domains.

The evolution of grid security architecture reflects many of the same choices in our approach for community clouds and federated clouds. For example, grid systems make frequent use of signed PKI certificates and delegation. Grids use PKI for similar reasons: PKI is convenient to authenticate messages from hands-free user tools and from programs running inside the grid.

External identity and attributes. Early grid systems used simple mappings of external user identities (distinguished names) to local user identities (user IDs) with specific rights and powers by means of a "grid map file". Over time it was recognized that user privileges flow from memberships and roles in communities, and their relationships of sharing and trust. This motivated a more dynamic and fluid trust decisions, based on identity attributes and third-party endorsements of identities that are not known to the provider.

For example, grid designers also initiated early efforts to bridge web sign-on (SSO) systems to PKI-based grid security infrastructure [15]. Many deployed grids today bridge web SSOs to their PKI systems using variants of the simple identity broker concept outlined above. Examples include recent versions of MyProxy [12], the Short-Lived Credential Service portal (SLCS), and several others.

Virtual Organizations. The concept of a Virtual Organization (VO) serves as the unit of grouping for users and providers in many deployed grid systems [10]. Most importantly, VOs are groupings of users spanning multiple identity domains (although they may also involve groupings of grid resource providers). A VO corresponds loosely to a *project* in the GENI architecture. There may be differences in granularity, but both approaches support multiple group coordinators and nested subgroups with delegated administration.

Many grid systems employ a service called Virtual Organization Management Service [11] to manage user membership in

V**O**s. Each VOMS server is recognized by other entities as authoritative for one or more V**O**s or for a specific set of groups. The VOMS issues credentials containing statements about user membership and roles in V**O**s. For example, a VOMS instance issues credentials as X.509 attribute certificates signed under its own key pair and binding a user**'**s public key to one or more roles scoped to a named VO. Resource providers that trust the VOMS to make such statements may consider them in deciding whether to grant access. This structure is similar to the GENI authorities that issue project credentials.

5. SUMMARY

GENI seeks to build a trusted environment in which experimenters and resource owners can participate in resource allocation transactions. The trust relationships among software entities reflect the trust within the corresponding human/inter-organizational relationships, nothing more. Authorization, authentication, and accountability are the pillars of constructing that trust; credentials and policies are the critical enablers of these pillars. The GENI program will continue working to make these exchanges more efficient and trustworthy and is eager to share experiences with other programs or institutions with similar problems or solutions.

6. ACKNOWLEDGMENTS

GENI is funded by the US National Science Foundation (NSF) under cooperative agreement CNS-0737890. Any opinions, findings, conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect the views of the NSF.

.

7. REFERENCES

[1] ABAC Development Team. ABAC. [Online]. http://abac.deterlab.net/

[2] (2012) GENI Aggregate Manager API Version 3. [Online]. http://groups.geni.net/geni/wiki/GAPI_AM_API_V3

[3] Brian White, Jay Lepreau, Leigh Stoller, Robert Ricci, Shashi Guruprasad, Mac Newbold, Mike Hibler, Chad Barb, and Abhijeet Joglekar, "An integrated experimental environment for distributed systems and networks," *SIGOPS Operating Systems Review*, vol. 36, no. SI, pp. 255-270, December 2002.

[4] Marshall Brinn, Jonathon Duerig, Aaron Helsinger, Tom Mitchell, Robert Ricci, Tom Rother, Leigh Stoller, Wim Van de Meerssche, Brecht Vermeulen, and Gary Wong. (2013, November) Common Federation API, Version 2.. [Online]. http://groups.geni.net/geni/wiki/CommonFederationAPIv2

[5] Larry Peterson, Tom Anderson, David Culler, and Timothy Roscoe, "A blueprint for introducing disruptive technology into the Internet," *ACM SIGCOMM Computer Communication Review*, vol. 33, no. 1, pp. 59-64, January 2003.

[6] Mark Berman, Jeffrey S Chase, Lawrence Landweber, Akihiro Nakao, Max Ott, Dipankar Raychaudhuri, Robert Ricci, and Ivan Seskar, "GENI: A Federated Testbed for Innovative Network Experiments," *Computer Networks*, no. 61, pp. 5-23, March 2014.

[7] Larry Peterson, Soner Sevinc, Jay Lepreau, Robert Ricci, John Wroclawski, Ted Faber, Stephen Schwab, and Scott Baker. (2009, April) Slice-Based Facility Architecture. [Online]. http://svn.planet-lab.org/attachment/wiki/WikiStart/sfa.pdf

[8] A. Freier, P. Karlton, and P. Kocher. (2011, August) RFC 6101: The Secure Sockets Layer (SSL) Protocol Version 3.0. [Online]. http://tools.ietf.org/html/rfc6101

[9] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk. (2008, May) RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. [Online]. http://tools.ietf.org/html/rfc5280

[10] Ian Foster, Carl Kesselman and Steven Tuecke. 2001. "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", International Journal of High Performance Computing Applications, Volume 15, 2001.

[11] Alfieri, R. and Cecchini, R. and Ciaschini, V. and dell'Agnello, L. and Frohner, Á. and Gianoli, A. and Lõrentey, K. and Spataro, F. 2004. "VOMS, an Authorization System for Virtual Organizations", Grid Computing, Lecture Notes in Computer Science, Springer Berlinen / Heidlberg, Volume 2970, 2004.

[12] Basney, Jim and Humphrey, Marty and Welch, Von. 2005. "The MyProxy online credential repository", Software Practice and Experience, Volume 35, Number 9, July 2005. RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. [Online]. http://tools.ietf.org/html/rfc5280

[13] Baldine I, Xin, Y., Mandal, A., Ruth, P., Yumerefendi, A. and Chase, J. 2012. "ExoGENI: A multi-domain infrastructure-as-a-service testbed". TridentCom: International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities, June, 2012.

[14] Jeff Chase and Vamsi Thummala. 2014. "A Guided Tour of SAFE GENI", Duke University Department of Computer Science Tech Report, CS-2014-002, June 2014

[15] Barton, Tom and Basney, Jim and Freeman, Tim and Scavo, Tom and Siebenlist, Frank and Welch, Von and Ananthakrishnan, Rachana and Baker, Bill and Goode, Monte and Keahey, Kate. 2006. "Identity Federation and Attribute-based Authorization through the Globus Toolkit, Shibboleth, Gridshib and MyProxy.". 5th Annual PKI R&D Workshop, April, 2006.

[16] R. L. Morgan and Scott Cantor and Steven Carmody and Walter Hoehn and Ken Klingenstein. 2004. "Federated Security: The Shibboleth Approach". EDUCAUSE Quarterly, Volume 27, Issue 4, 2004.