TIED – Trial Integration Environment Based on DETER

QPR June 30, 2009

Introduction

TIED has focused on three areas of work this quarter: the continued development of the TIED federation system and assessing its relationship to the GENI clearinghouse model; deployment of national scale layer 2 connectivity to TIED facilities, and integration of the ABAC authorization architecture. More details follow below as well as lists of participants and collaborations.

Major Accomplishments

- Web interface to the TIED clearinghouse, and external clearinghouse documentation.
- Achieved transcontinental layer 2 VLAN connectivity from Los Angeles to Arlington, VA through path ISI West > LADWP > USC > Los Nettos > CENIC > Internet 2 DCN > MAX > ISI East.
- Continued development of TIED's *fedd* software in its role as a GENI clearinghouse.
- Initial prototyping of TIED's ABAC authorization system and integration with the *fedd* implementation.

Description of Work Performed During the Quarter

1 Development of TIED Federation as GENI Clearinghouse

TIED's implementation of a GENI control framework and clearinghouse revolves around the concept of federation as a fundamental mechanism and the DETER/TIED *fedd* code base as its implementation.

1.1 World Wide Web Access to TIED Clearinghouse

Related milestones:

- Year 1, Milestone d: Operate prototype TIED clearinghouse.
- Year 1, Milestone e: Provide user access to DETER testbed using TIED building blocks.
- Year 1, Milestone f: Demonstrate and support running federated experiments by owner(s) outside the development team by the end of year 1.
- Year 1, Milestone g: Demonstrate extended clearinghouse / component functionalities key to outreach communities (eg, extended security model access)

This quarter we have rolled out initial documentation of the TIED clearinghouse user documentation (at http://groups.geni.net/geni/wiki/TIEDClearinghouse) and deployed an initial release of web-based front ends to the federation functionality at the core of TIED. Though clearinghouse functions have been provided for some time, these deployments facilitate support for new users and simplify access to TIED.

The documentation provides a reference and tutorial for experimenters who want to use TIED, as well as laying out some of the mappings between GENI functionality and the *fedd* architecture and abstractions. It is hosted on the GENI wiki and is under steady revision, though it should be complete enough for daily use. It documents both the web interface described below and the older command-line interface.

The web interface allows users to create and terminate slices as well as retrieve information anout active and inactive slices. Saved configurations can be easily recreated. Simple visualizations of the slice topologies are also available. We include a sample screenshot.

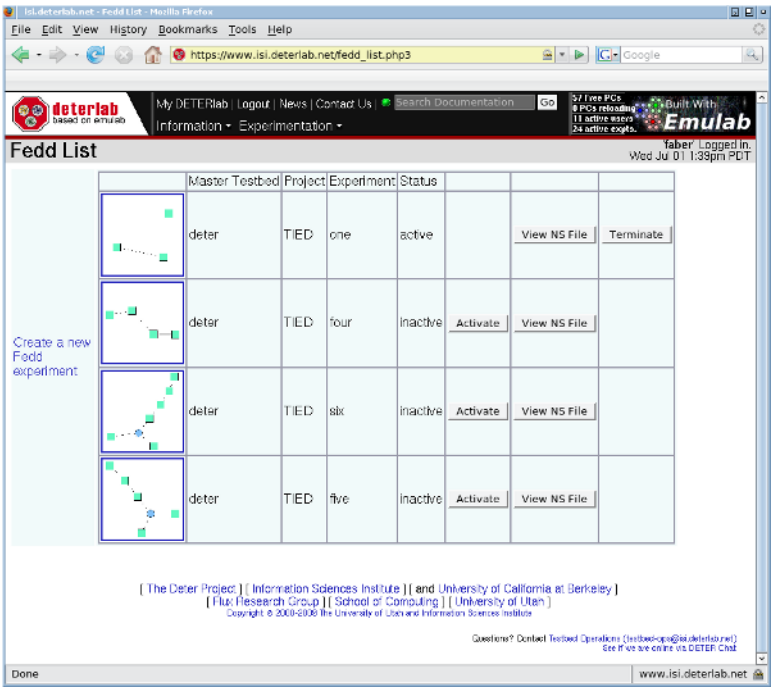

Figure 1: TIED Clearinghouse Web Interface Screenshot: Multiple Instantiated Slices

The current web interface is an initial implementation, and new features are in progress. It is intended that all user-facing operations should be completely accessible from the web interface, though currently there are some operations that can only be accomplished through the command line.

Extending the web interface will result in some new functionality being added to *fedd* as well as improvements to the web interface code itself. We are planning significant improvements in the coming quarter.

Providing a web interface opens TIED to more users and reduces the learning curve for those users, which obviously addresses Milestones d, e, and f. In addition, this simpler interface will facilitate outreach to less networking-centric experimenters, classroom use, and the like, contributing to Milestone g.

1.2 Providing the TIED Clearinghouse

In addition to the web interfaces and documentation described above, we have spent significant time this quarter improving *fedd*'s GENI clearinghouse functionality and improving the interfaces to it for users. As part of this process, we have examined the clearinghouse requirements and the implementation of *fedd*, and have found places to improve both. This section discusses how *fedd* provides clearinghouse functionality, as well as lessons we learned from running the TIED clearinghouse so far and their implications.

Related milestones:

- Year 1, Milestone d: Operate prototype TIED clearinghouse.
- Year 1, Milestone e: Provide user access to DETER testbed using TIED building blocks.
- Year 1, Milestone f: Demonstrate and support running federated experiments by owner(s) outside the development team by the end of year 1.

Fedd, an implementation of DETER/TIED's federation architecture, is the core of the TIED

clearinghouse. The TIED clearinghouse functionality has existed for some time now, but this quarter we carefully analyzed what functions were required for a GENI clearinghouse and how well *fedd* provided them. This section describes that analysis and shows how *fedd* provides the service. The next section points out the new directions and lessons we learned.

The GENI clearinghouse requirements are in some flux, but the basic concept is one of a central control point where a user can register access information and acquire resources subject to some policy. Resource providers and clearinghouses enter into some mutual trust relationship based on these policies and other agreements. Such agreements may include negotiated contracts or more loose alignment of goals and requirements.

The exact definition of a clearinghouse is somewhat difficult to nail down. The GENI Control Framework Requirements document does not directly state clearinghouse requirements, but the functional diagrams indicate that a clearinghouse is responsible for organizing principals, slices, and components as well as providing ancillary functions controlling resource allocations (tickets) and software distribution. Chip Elliot's comments at the 4th GEC outline a different set of requirements, including acting as a meeting point for resource users and providers, recording transactions, and implementing global policies.

Given this ambiguity, the TIED clearinghouse defines its functionality to include:

- Experimenter registration: Binding principals to federation access control information
- Slice management: Creating, manipulating, and terminating slices, subject to clearinghouse policy
- Top level experiment management: Managing and initializing experiment control software

In addition to these clearinghouse functions, *fedd* provides an aggregate interface, currently interfacing with Emulab-based testbeds (as described in the 1Q09 report). More importantly, our system architecture dictates that *fedd* interface with a variety of external resources (federants) using their native control protocols. *Fedd*'s plug-in architecture is designed to support such extensions simply. As described in Section 2 of this report, our first use of this capability allows TIED to dynamically control Layer 2 network paths between TIED facilities.

Experimenter Registration: Registering an experimenter is accomplished through a combination of the DETER user registration system and assigning *fedd* attributes to the experimenter. DETER user registration assigns the user a public/private keypair encoded in an X.509 certificate that can act as a *fedd* fedid, which is an implementation of a GENI GID. The combination of the new fedid and *fedd* attributes derived from information collected during registration provides the basis for *fedd* to act on the user's behalf in allocating slices. *Fedd*'s identification and authorization structure is described in [Faber08], and is currently being extended by ABAC, which we describe below.

Slice Management: Slice management consists of creating federated experiments that span multiple testbeds; in GENI terms slices are created from resources contained in several aggregates. *Fedd* manages the access control and converts the high-level description of the experiment into the configuration languages of the various aggregates.

This translation of configuration and allocation operations from the high level to a common set of low-level operations is the core of *fedd*'s aggregate implementation. The plug-in architecture is designed to simplify making TIED aggregates from new resources.

When a slice is created, *fedd* assigns it a unique global identifier, that is placed in the hands of the experimenter who created it. One can think of that identifier as picking out the owner(s) of the slice, rather than naming the slice itself. The identifier is encoded as an X.509 certificate and accompanying private key. By passing that certificate and key around multiple experimenters can manipulate the slice. This capability is also being incorporated into the developling ABAC authorization system.

Slice manipulations include acquiring several views of the slice topology and visualization information as well as terminating the slice. We are designing interfaces to support expanding slices as well.

Experiment management: The TIED slice creation system exports services into the slice including the local DETER environment of the experimenter (file systems, user information, etc.) and several support services. TIED acts as a clearinghouse (in a non-GENI sense) for those DETER services. Principal among these is the SEER experiment management software, an agent-based extensible system that manages configuration, experiment conduct, and data collection. Considerable information on SEER details is available from [Schwab07] and [SEER].

This combination of software components has been in operation since 1Q09, and has been used by experimenters outside the DETER and TIED projects as the basis for system demonstrations. A notable example this quarter is Rick McGeer (HP Labs) recently use of TIED/DETER's federation capability to explain and motivate similar capabilities proposed for DARPA's National Cyber Range. Rick is unaffiliated with the DETER or TIED projects. This directly shows our ability to meet Milestone f.

This analysis has enabled TIED to provide a more complete clearinghouse and clarified how the aggregate interface is implemented. In addition, *fedd* functionality has been tweaked in small ways that make providing the clearinghouse more straightforward. This work addresses Milestones d and e.

1.3 Clearinghouse Architecture and Fedd Refactoring

Related milestones:

- Year 1, Milestone d: Operate prototype TIED clearinghouse.
- Year 1, Milestone e: Provide user access to DETER testbed using TIED building blocks.

In deploying a clearinghouse, we were forced to consider the construct's various components and their interrelations, as well as how these components and interrelationships are instantiated by TIED's *fedd* software. We have come away from this with opinions about both the clearinghouse and about *fedd*'s architecture that will affect future TIED work.

The functions that the TIED clearinghouse provides are all essential functions for GENI, and all clearly related, but we do not believe they are so tightly connected that they must be packaged into an architectural abstraction. While assigning user attributes and privileges is related to the process of allocating resources, there is no reason they must be managed by the same entity. As GENI grows larger, requiring that resource and user information be co-administered will act as a scaling and structural bottleneck.

Using the clearinghouse as a policy enforcement point is suggested as a reason for the clearinghouse grouping, but just puts off the inherent distributed policy problem. Co-administering clearinghouse function allows more fine-grained control of policies over entities that the clearinghouse controls, but we expect that the complexity of that administration will limit the scale of clearinghouses. As other clearinghouses appear, the problem of coordinating policy reappears. Relaxing the clearinghouse's co-administration assumptions early in the prototyping will lead to better distributed policy solutions.

Similar policy problems lurk in the implicit assumption that one clearinghouse controls the aggregates registered with it to the extent needed to enforce policy. We expect that as soon as multiple clearinghouses arise, some resource providers will want to allow access to multiple clearinghouses' users, and again distributed policy issues arise.

While it is tempting to consider that clearinghouses can simplify the complex policy problems, we believe that the abstraction simply (partially) obscures those problems. In this sense the abstraction is worse than unnecessary.

In addition to encouraging prototypers to address the distributed policy problems, removing the clearinghouse abstraction prevents extraneous requirements from being placed on these functions. For example, co-location or data sharing constraints between the orthogonal capabilities is a potential requirement that we believe to be extraneous.

For these reasons, we think that the clearinghouse abstraction should be abandoned, though we continue to run one. These arguments also appear in the TIED clearinghouse documentation

described below, and will be described in a technical report during the next quarter. We believe that doing this analysis and making it public enhances our ability to meet Milestones d and e, as well as serving the larger community doing GENI design.

This analysis motivated a hard look at the *fedd* codebase, which unfortunately does not clearly reflect our architectural position. While the major functions are separated in the coding style, the functions are all provided by the same executable. The result is both more difficult to explain and more difficult to configure than a better designed representation would be. TIED implementers often describe different aspects of *fedd* as separate components when talking with each other, and *fedd* itself is frequently run at a site to provide only some of the implemented services, not all. A cleaner approach would be to have different names and executables that represent the different architectural entities, with defined interfaces for each.

A major thrust of the next quarter is to refactor the various clearinghouse functionality present in *fedd* into smaller executables that are simpler to explain and administer. By making our architectural arguments more concrete – directly reflecting them in a functioning clearinghouse – our analysis above will be supported with operational evidence, which will improve GENI design.

2 Deployment of TIED Layer 2 Network Infrastructure

Related milestones:

- Year 1, Milestone j: Provide direct external ethernet-level (VLAN) access interface to TIED resources.
- Year 2, Milestone a: Develop and deploy TIED plugin to access and control wide area network resources.

As we have previously discussed, a key aspect of the TIED control framework model is its ability to incorporate with and utilize a wide range of infrastructure resources, network types, and other facilities through TIED's extensible federation architecture. Our approach to establishing flexible layer 2 VLAN capability is to first carry out sufficient site engineering to support static VLAN configuration, and then to implement dynamic VLAN provisioning and configuration by treating the relevant external networks (in GENI parlance, aggregates that implement network connectivity) as TIED federants.

This quarter we have demonstrated end-to-end VLAN connectivity between ISI-East and the DETER cluster sited at ISI in Marina del Rey. Multiple VLANs are currently statically configured, and support 1 Gb/s aggregate bandwidth between the two sites. The path, shown in Figure 2 below, leaves ISI in Marina Del Rey over fiber provided by the Los Angeles Department of Public Works, connects to and traverses the USC campus to reach the Los Nettos regional network, briefly traverses the CENIC network and connects to Internet 2's Dynamic Circuit Network (DCN) at a large multiprovider POP located at 1 Wilshire in LA, traverses the country to the MAX regional network over Internet 2, and finally reaches ISI East on the MAX network.

As shown, this connectivity crosses several networks, but is accessible end to end only at the Los Angeles and Arlington,VA sites. We expect similar static connectivity to the UCB TIED cluster to be completed in the near future, although coordination with UCB campus networking operations is required. Further, the cross-country bandwidth is actually 10 Gb/s, but we are currently limited at the ingress to TIED. A small amount of additional hardware is required to support the full aggregate bandwidth end to end. A funding proposal (to the DoD DURIP program) has been submitted that may allow us to obtain this hardware in the near future.

The currently operating VLAN configuration is largely manually configured.[1] Our objective is that VLAN configuration be dynamically controlled through the TIED control framework. As described, this is achieved by treating external networks as federants, with the TIED managing the federation and

1 This manual configuration includes both detailed switch-level configuration at certain points and use of existing web-based “configuration tools” for potentially dynamic facilities such as I2 DCN.

controlling the external networks directly through their exported interfaces.

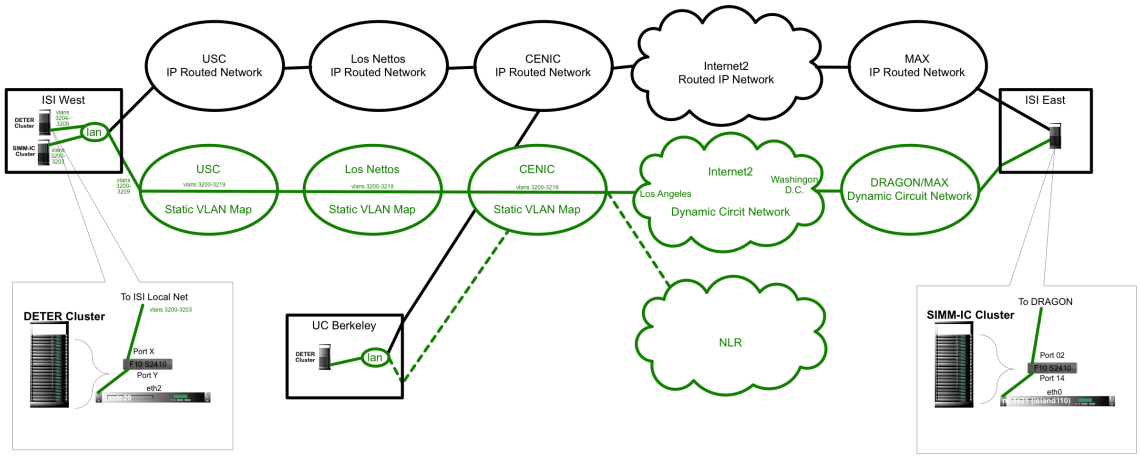

Figure 2: TIED Layer 2 Connectivity Map

In collaboration with Tom Lehman (the DRAGON project leader at ISI East), CENIC, and Los Nettos, we are working towards implementation of dynamic control capabilities within TIED for DRAGON-based networks and the Internet 2 DCN, both to meet the external VLAN milestones (milestone j) and as a motiving basis for engineering of TIED's external federation capabilities. Our ultimate goal is to provide dynamically provisionable access (while utilizing static segments where no dynamic control is available) between several TIED sites, as well as connectivity to external Internet2 DCN sites and the National Lambda Rail (NLR). This will further facilitate collaboration with other testbeds, including other GENI spiral 1 control frameworks and the Wisconsin Advanced Internet Laboratory (WAIL).

3 Prototyping and Integration of Enhanced Authorization Model (ABAC)

Related milestones:

- Year 1, Milestone a: Identify specific outreach communities for the year-1 program. Identify and document initial requirements they impose on TIED federation architecture and interfaces.
- Year 1, Milestone g: Demonstrate extended clearinghouse / component functionalities key to outreach communities (e.g., extended security model access).
- Year 1, Milestone i: Collaborate with Security team on security design for Spiral 1.

Continuing the process described in last quarter's report, we are extending and integrating a prototype ABAC implementation from NAI/SPARTA/Cobham to interface with *fedd* to provide more expressive and scalable authorization facilities. This work is intended to meet the requirements of a future, broader GENI community in general, and of our outreach communities – characterized by users and administrators with a wide range of existing equipment and skill levels -in particular.

To be successful in meeting these requirements the authorization system must scale to hundreds or thousands of researcher/users and thousands if not millions of end users, all initially authenticated by a variety of systems or in some cases operating anonymously. It must be expressive enough to control resources at a fine grain, and support the distributed policy setting and resolution that is the hallmark of federation. It must scale in the sense of reducing the administrative overhead of any single set of users. It should allow for disconnected operation rather than assuming constant communication between all components of the system. It should support clear "explanation" and auditing of its actions. Our work addresses all of these goals.

This quarter we have designed an integration of the existing *fedd* code and the ABAC authorization code that allow *fedd* (or another GENI component) to establish the authorization context and goals for an ABAC prover and to retrieve the results of an authorization negotiation. For an ABAC prover that

is protecting resources, the context contains the rules for access. For a prover requesting access the context includes the credentials that the instance can use to gain access.

The results of the negotiation are an annotated graph expressing the proof that the requester is entitled to the resource. The graph is annotated with the credentials used to support the claims. This provides two levels of function. If such a graph exists, the authorization is successful. Further, examining the graph can show *why* the authorization was successful. Our design logs graphs for auditing purposes.

In addition to integrating the proof capabilities into *fedd*, we must provide a way for clearinghouse and aggregate operators to express and monitor authorization-related policies. We are prototyping configuration and auditing tools that will allow administrators to graphically depict and manipulate authorization policies (e.g., the proof contexts of the various provers) and to browse the audit logs (that is, the proof graphs) intuitively.

We plan to demonstrate these capabilities – proofs and graphical configurations – at GEC5.

This development is being done in close collaboration with Steve Schwab and Jay Jacobs, who implemented an earlier prototype, at Cobham/SPARTA in order to roll back our prototyping experience into the broader GENI security architecture.

These authorization extensions, prompted by our study of our routreach requirements, address Milestones a and g. Our collaboration with Steve Schwab addresses Milestone i.

References

[Faber08] Ted Faber and John Wroclawski, "Access Control for Federation of Emulab-based Network Testbeds," In *Proceedings of the CyberSecurity Experimentation and Test (CSET) Workshop*, San Jose, (July 2008).

[Schwab07] Stephen Schwab, Brett Wilson, Calvin Ko, and Alefiya Hussain. "SEER: A Security Experimentation EnviRonment for DETER," In *Proceedings of the DETER Community Workshop on Cyber Security Experimentation and Test*, August 2007.

[SEER] http://seer.isi.deterlab/net

Project participants

1) Individuals directly supported by TIED award:

   John Wroclawski, PI
   Ted Faber, Research Computer Scientist

2) Individuals contributing to the project with outside support:

   Tom Lehman, Research Computer Scientist
   Jelena Mirkovic, Research Computer Scientist
   Jason Shupe, Systems Programmer
   Jay Jacobs, Systems Programmer

Publications

A Federated Experiment Environment for Emulab-based Testbeds. T. Faber and J. Wroclawski. 5th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities (TRIDENTCOM 2009) . April 6-8, 2009, Washington D.C., USA.

This paper presents an overall description of the DETER Federation Architecture that underpins our work on TIED; outlines key elements of the architecture including resource allocation, authorization and access control, and experiment control environment, and presents a brief description of the development prototype.

Initial drafts of the TIED clearinghouse documentation have been posted to the GENI wiki at

http://groups.geni.net/geni/wiki/TIEDClearinghouse.

A description of TIED's authorization subsystem ABAC has been posted to the GENI wiki at http://groups.geni.net/geni/wiki/TIEDABACModel. This document is under frequent revision.

An illustrative example of ABAC drawn from our planned demo at GEC5 has been posted to the GENI wiki at http://groups.geni.net/geni/wiki/TIEDABACDemo. This document is under frequent revision.

The DETER/TIED federation architecture and its *fedd* implementation is documented at http://fedd.isi.deterlab.net/trac. This documentation is frequently updated to reflect changed in fedd such as those described here. The documentation is linked from TIED's project page on the GENI wiki at http://groups.geni.net/geni/wiki/TIED.

Outreach activities

CSET 2009 – TIED project members are primary organizers of the 2nd Usenix Workshop on Cyber Security Experimentation and Test (CSET 2009) to be held on August 10, 2009 in conjunction with the annual Usenix Security Symposium. This workshop brings together researchers and testbed developers interested in sharing experiences and defining an agenda for the development of scientific, realistic evaluation approaches to security threats and defenses. With NSF support, CSET 2009 offers a student travel program, and makes particular effort to recruit presenters and attendees from underserved communities. TIED project member Terry Benzel serves as General Chair of CSET 2009, while contributor Jelena Mirkovic serves as co-Program Chair. Further information is available at http://www.usenix.org/event/cset09.

WISE 2009 – TIED project affiliate Jelena Mirkovic (substituting for Terry Benzel) participated and presented at the 2009 Women's Institute in Summer Enrichment (WISE 2009) hosted by the NSF-sponsored TRUST Center at UC Berkeley. WISE is a 1-week residential summer program on the University California, Berkeley campus that brings together graduate students, post-doctoral fellows, and professors from all disciplines that are interested in the technical, social, political, and economic ramifications of security technologies and security research. Leading experts from across the country teach power courses in several disciplines, including computer science, economics, law, and electrical engineering. The program structure includes rigorous classes in the mornings and opportunities to explore through hands-on experiments and team-based projects in the afternoons. Further information is available at http://www.truststc.org/wise.

Collaborations

1) Utah Emulab group (Rob Ricci and staff) – development and testing of the DETER Federation Architecture software.
2) WAIL (Paul Barford and staff) – development and testing of the DETER Federation Architecture software.
3) Cobham/SPARTA (Steve Schwab, Jay Jacobs) – Development and prototyping of attribute based security models for federation. See discussion under Activities and Findings, above.
4) Cobham/SPARTA (Steve Schwab, Brett Wilson) – Development of support for federated experiments within the SEER Experiment Control Environment.
5) HP Labs (Rick McGeer) – Early adopter of TIED/DETER federation for demonstration.
6) DRAGON project at ISI-East, CENIC, Los Nettos. VLAN interconnection and debugging. See discussion under Activities and Findings, above.

Other Contributions

Ted Faber participated in the GENI RSpec Summit in Chicago on 25 June 2009, representing the TIED project.