Duke Systems

CF AUTHN/AUTHZ
GEC10

Jeff Chase

Duke University

GENI Security Architecture

- IMHO, security is a major recurring problem in GENI Control Framework (CF) architecture.
- The problem comes when we attempt to anchor/connect GENI to the outside world.
  - Confusion about trust roots
  - Ad hoc identity silos, etc.
  - Federation → federated identity
- Solution: factor out security architecture.
  - Do it once.
  - Do it right.

Principles

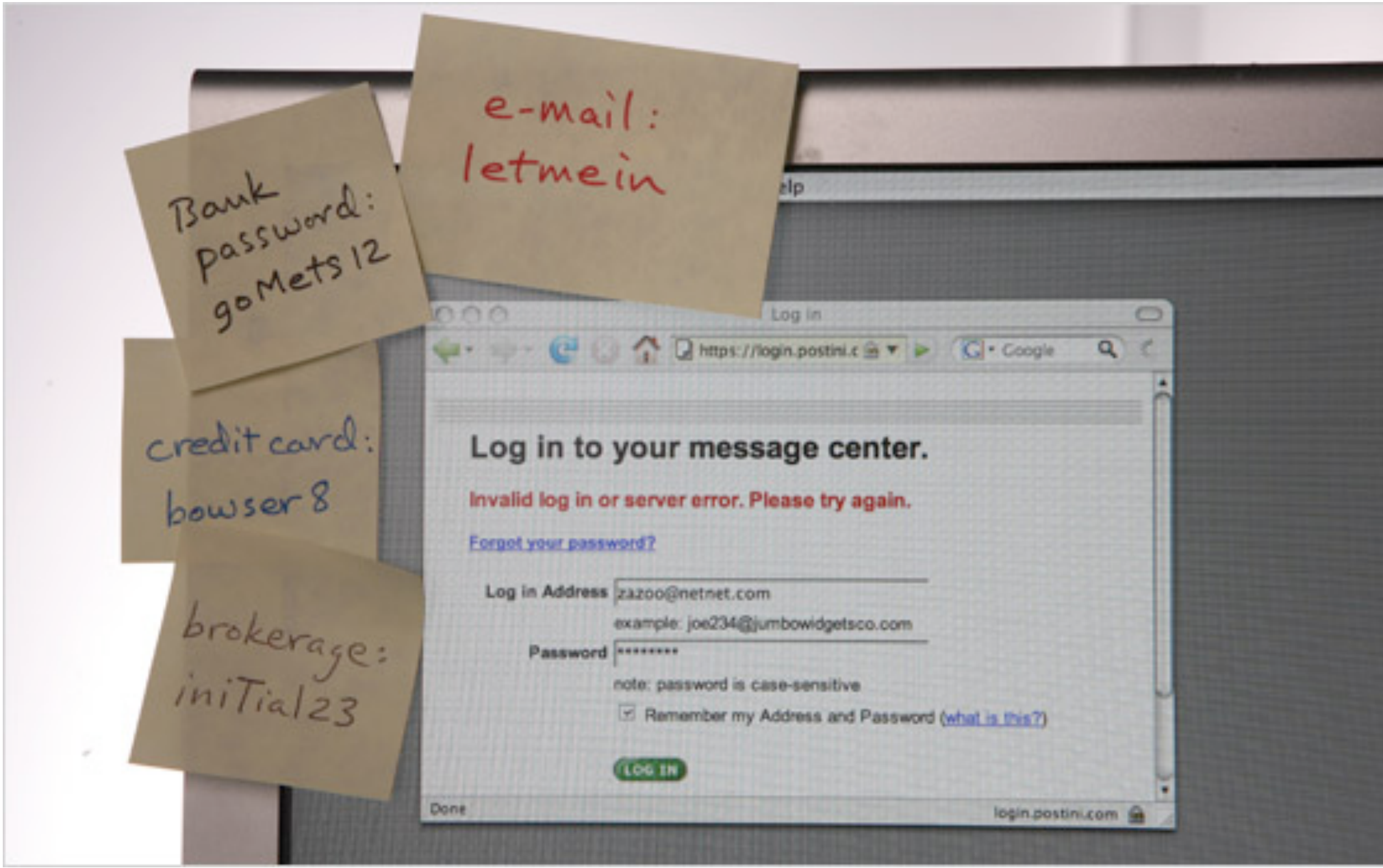
- Design in federation from the ground up.
- Separate policy from mechanism.
- Play well with others.
- Use off-the-shelf solutions when suitable.
  - External identity providers (IdPs)
    - “Web of identities”
  - Attribute-based access control (ABAC)
    - “Web of roles”
  - [See my GEC7 and GEC8 presentations.]

GENI Security Architecture

- **Agreement on underlying mechanisms:**
  - **Endorsement of identity**
  - **Assertion of attributes**
  - **Delegation of rights**
  - **Anchored in some set of trust roots**

Issued by whom? How are are the subjects named?
What are the attributes, rights, etc.? How to broker trust?
How are authorization policies specified?

[GEC7 CF-WG presentation]

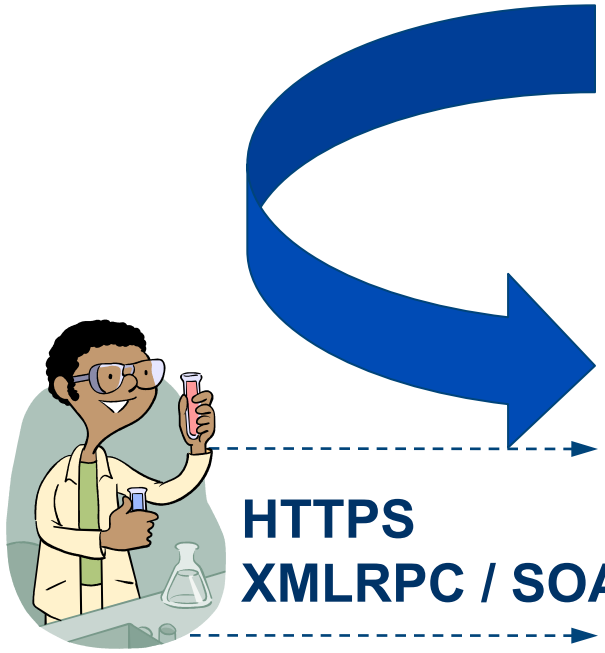

e-mail:
letmein
Bank
password:
goMets12
credit card:
bowser8
brokerage:
iniTial23
Log in
https://login.postini.c
Google
Log in to your message center.
Invalid log in or server error. Please try again.
Forgot your password?
Log in Address
zazoo@netnet.com
example: joe234@jumbowidgetsco.com
Password
note: password is case-sensitive
Remember my Address and Password (what is this?)
LOG IN
Done
login.postini.com

External Identity Providers in GENI (?)

- **GENI should enable/permit external IdPs.**
  - **Leverage powerful identity solutions developed by the large community focused on that problem.**
  - **Free GENI participants from administering identities and accounts.**
- **Which IdPs? Shibboleth and perhaps others.**
  - **Shibboleth is mature and widely deployed by universities and other institutions.**
    - **Single Sign On (SSO)**

[GEC7 CF-WG presentation]

Authenticated user identity
Attributes for authorization



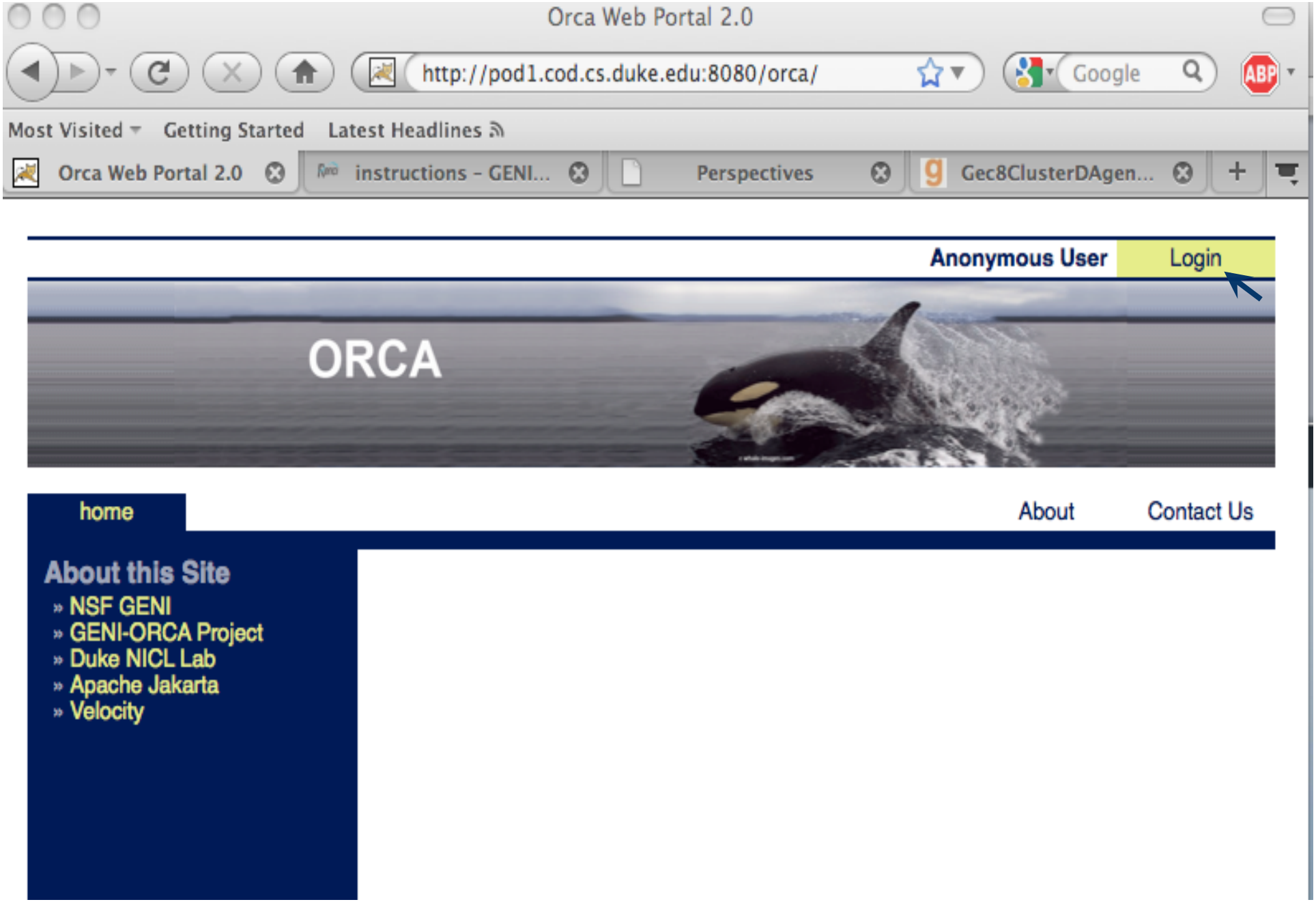
HTTPS
XMLRPC / SOAP

User/experimenters and "hands-free" tools

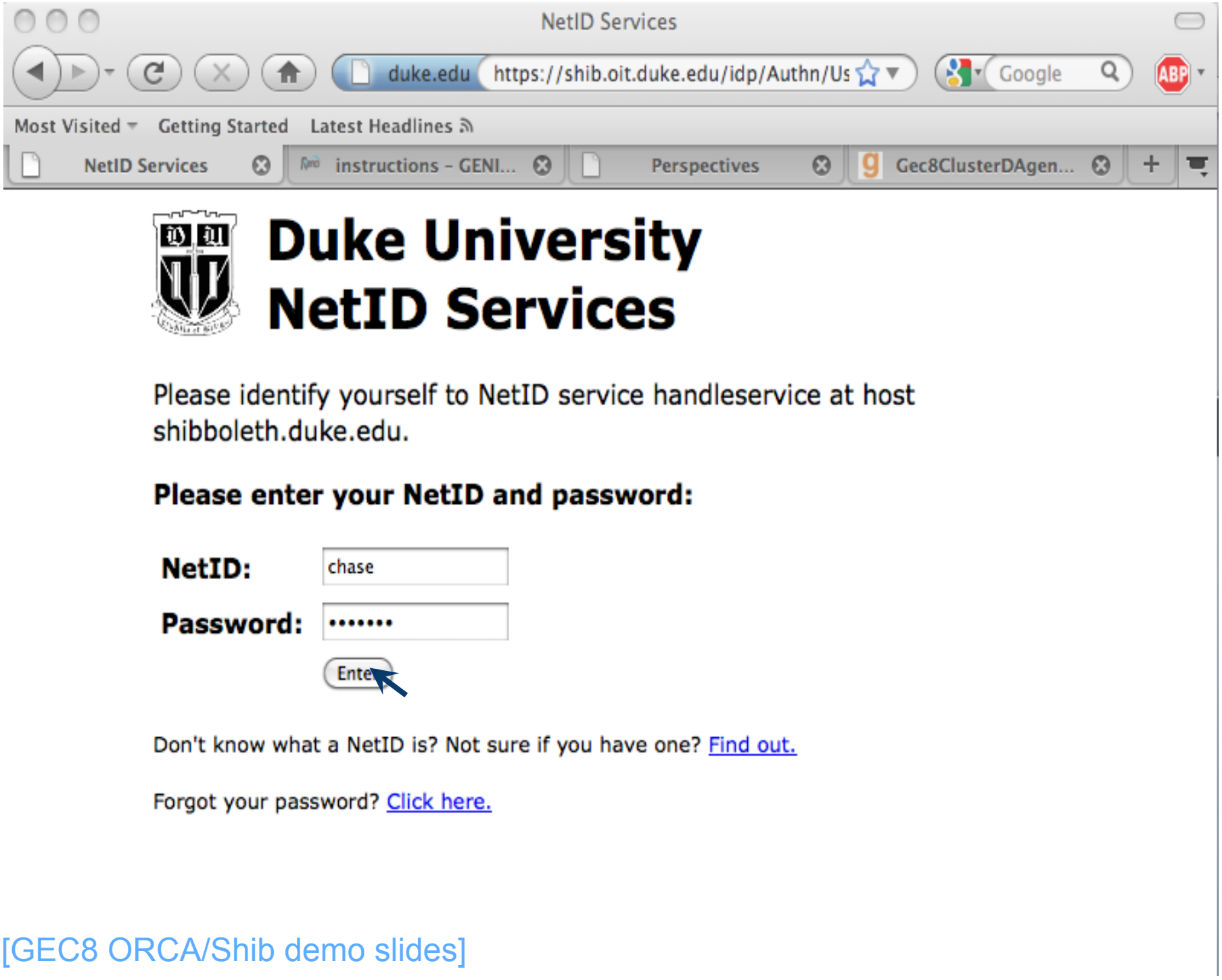
Duke Shibboleth Identity Provider (IdP)

Web Service Portal (SP)

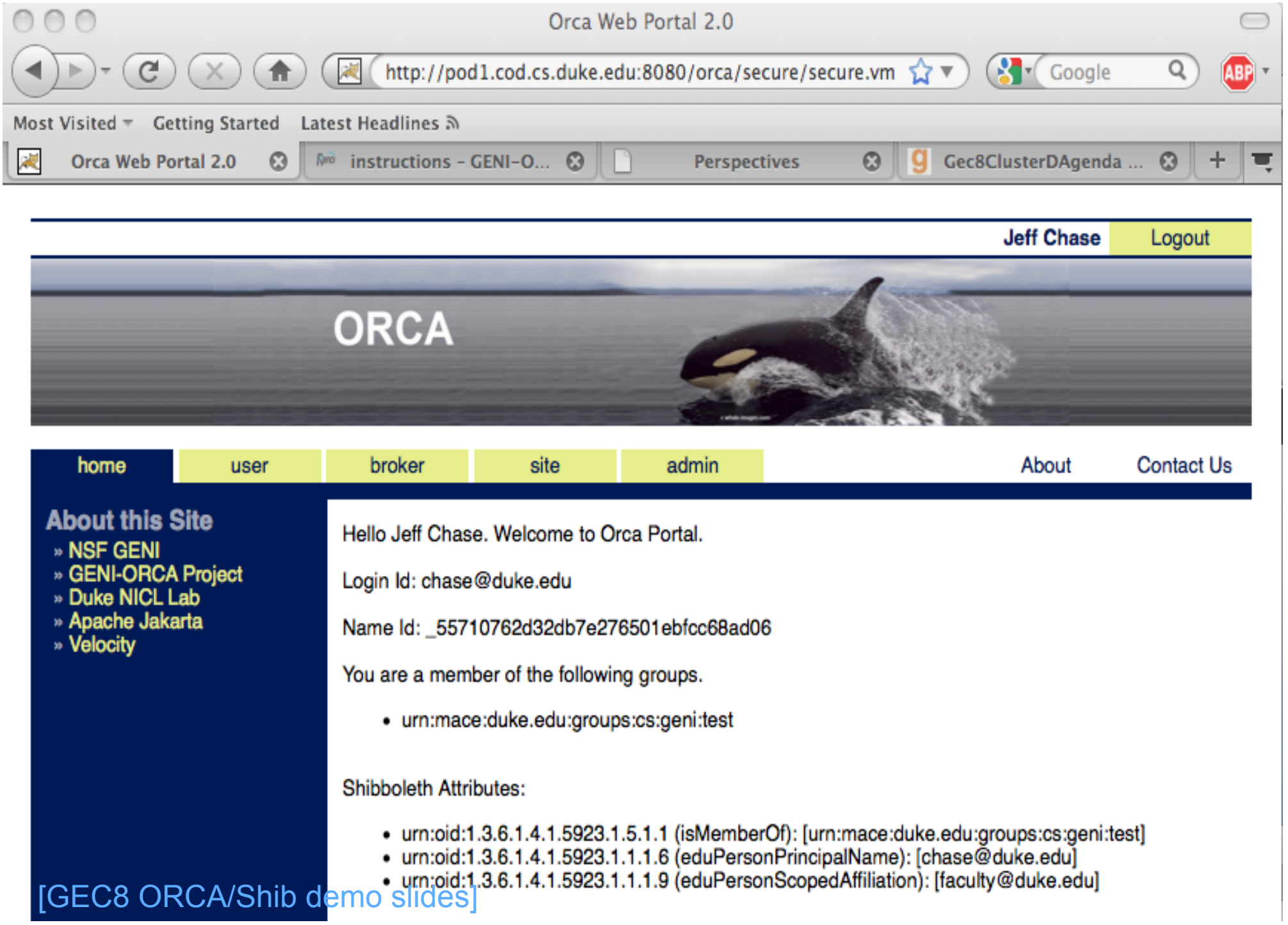
[GEC8 ORCA/Shib demo slides]



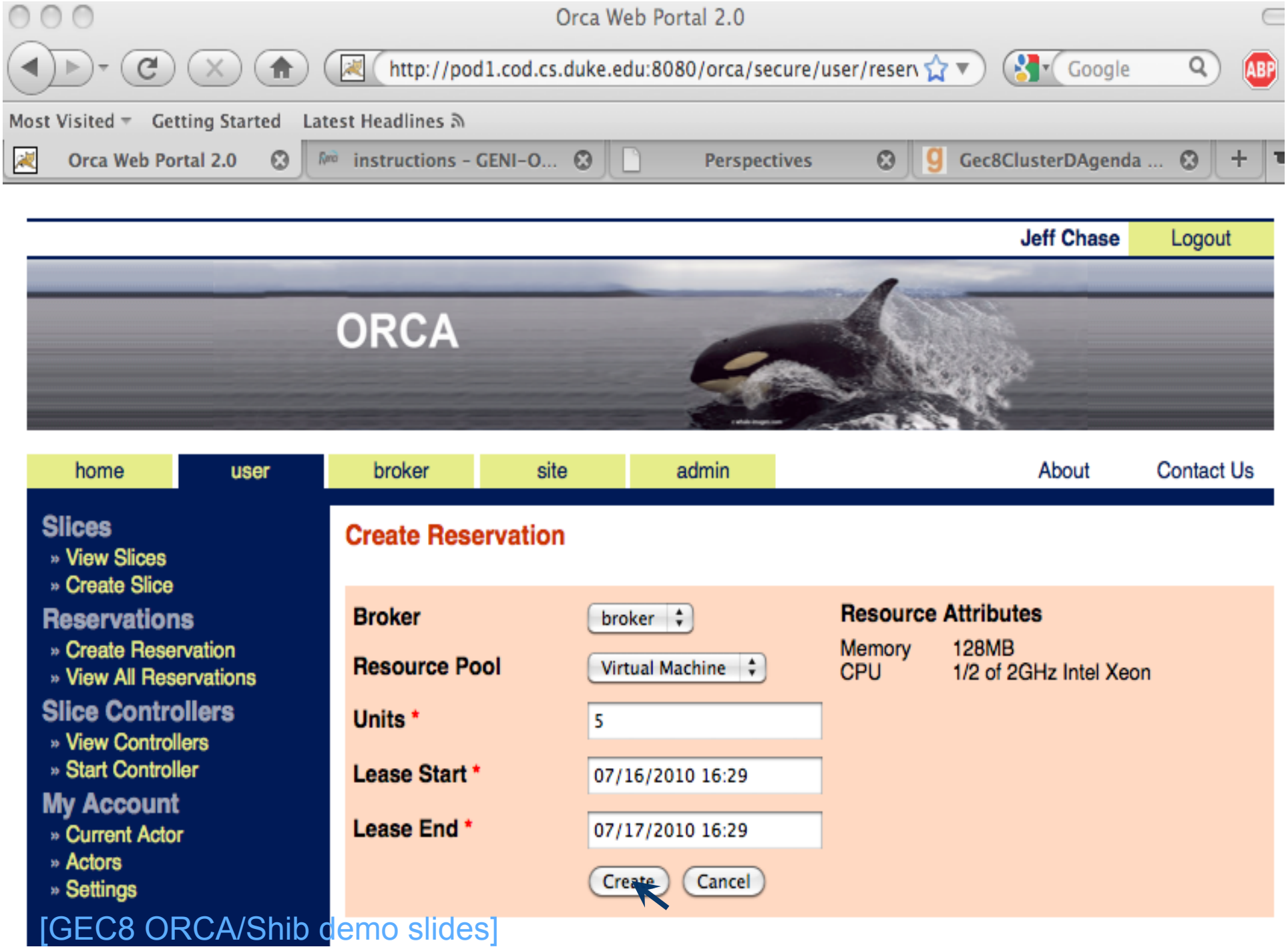
[GEC8 ORCA/Shib demo slides]



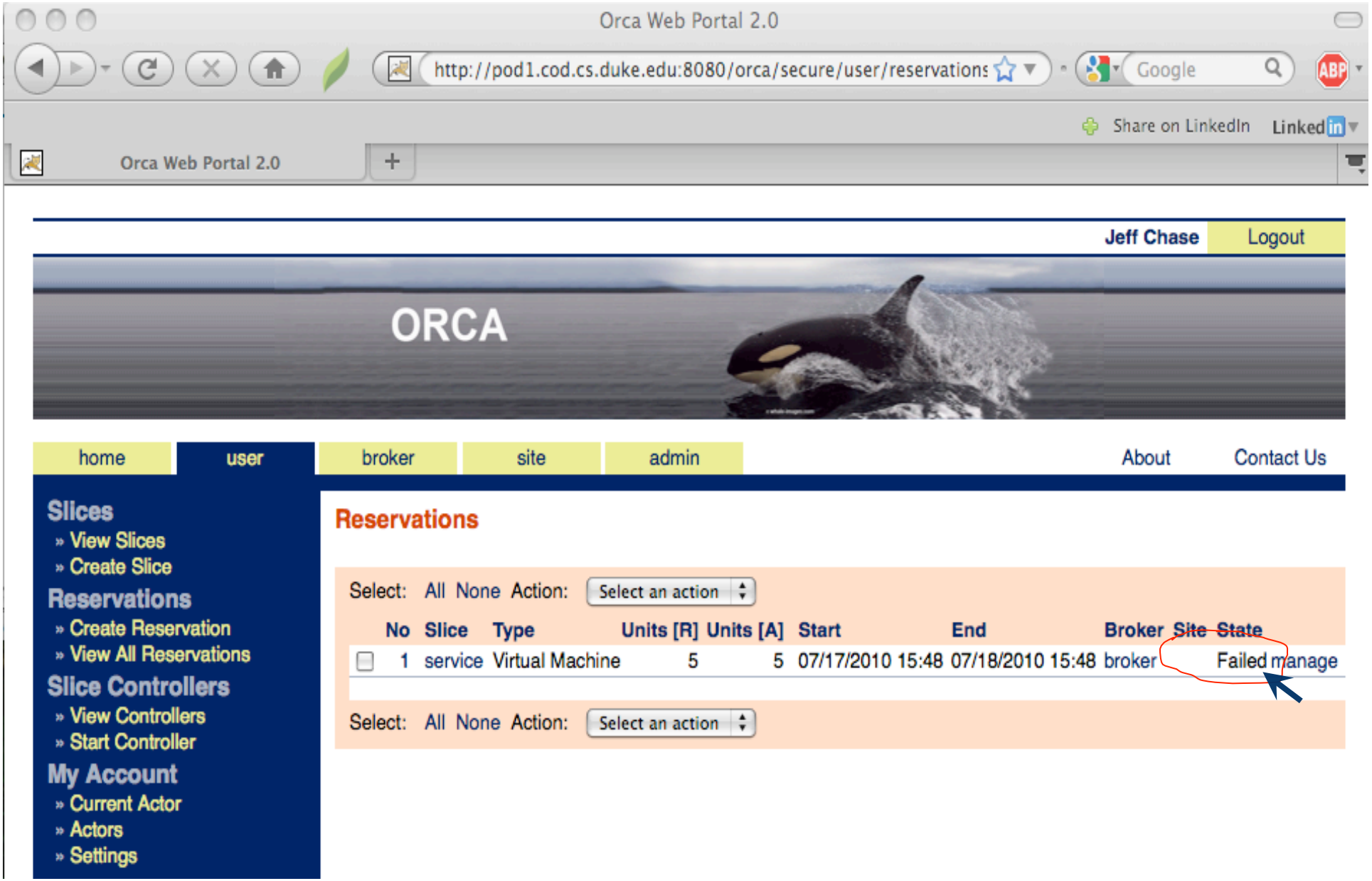
[GEC8 ORCA/Shib demo slides]



[GEC8 ORCA/Shib demo slides]



[GEC8 ORCA/Shib demo slides]



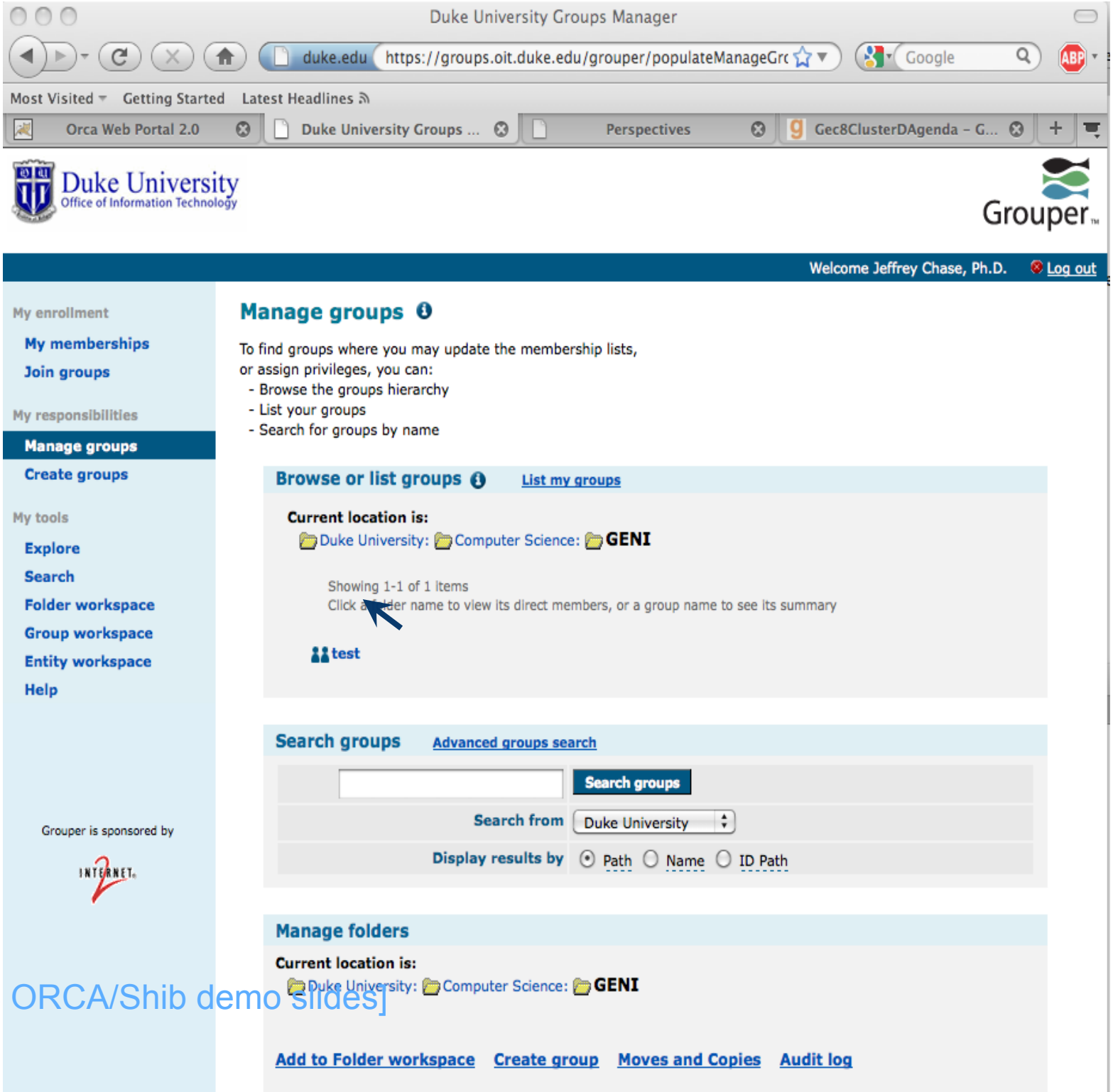
[GEC8 ORCA/Shib demo slides]

Reservation Details

Actions	Close Remove
Reservation ID	92fcd943-7675-458a-af0b-5e3480a9d6bb
Resource Type	Virtual Machine
Requested Units	5
Assigned Units	5
Leased Units	0
Lease Start	07/17/2010 15:48
Lease End	07/18/2010 15:48
Broker	broker
Site	
State	Failed
Status Message	You are authorized to reserve a maximum of 2 units.
Units	No units.

Allocation policy considers group membership attributes of requester (ABAC).

[GEC8 ORCA/Shib demo slides]



[GEC8 ORCA/Shib demo slides]

Attribute-Based Access Control (ABAC)

- This simple example illustrates Shib + ABAC.
- The attributes are asserted by a Shib IdP.
- The resource allocation policy trusts and understands attributes from this source.
- The policy uses the attributes to make a policy decision.
  - Authorization
  - Resource Control
- Shibboleth and ABAC work together.

[GEC8 ORCA/Shib demo slides]

Shibboleth in GENI, IMHO

- Easy to use to authenticate user/browser at a portal “at the edge”.
- Once authenticated, user can upload a public key for use by “hands-free” tools.
  - Standard for existing testbeds and clouds
  - Leverages external IdPs and avoids PKI
- Continue to use GENI key-based mechanisms internally.
- Continue to explore potential of delegated authentication, but do not depend on it.

[GEC7 CF-WG presentation]

GENI Portal (GIdP)

- **GENI identity portal (GIdP) is any web app that authenticates users and issues GENI certs.**
  - **Acts as a trust anchor: other GENI CF actors must trust the portal to do it right (act as a CA).**
  - **Bridges GENI to external IdPs (e.g., Shib) and/or has built-in account manager (e.g., PG&L).**
  - **Helps find "one throat to choke": if a user misbehaves, its GIdP can hold it accountable.**
  - **Interfaces to institutional IT services for users.**

Implementation: CF View

- **Factors authn and CH user registry OUT of the control framework.**
  - **So: "no implementation required."**

1. **Register trust anchors in each CF actor.**
2. **Install authz policies to consider attributes (e.g., using ABAC).**
3. **May need to pass certs through…**
4. **Allow for revocation…**

Duke Systems

CF AUTHN/AUTHZ
GEC10

Jeff Chase

Duke University

GENI Security Architecture

- IMHO, security is a major recurring problem in GENI Control Framework (CF) architecture.
- The problem comes when we attempt to anchor/connect GENI to the outside world.
  - Confusion about trust roots
  - Ad hoc identity silos, etc.
  - Federation → federated identity
- Solution: factor out security architecture.
  - Do it once.
  - Do it right.

Principles

- Design in federation from the ground up.
- Separate policy from mechanism.
- Play well with others.
- Use off-the-shelf solutions when suitable.
  - External identity providers (IdPs)
    - “Web of identities”
  - Attribute-based access control (ABAC)
    - “Web of roles”
  - [See my GEC7 and GEC8 presentations.]

GENI Security Architecture

- **Agreement on underlying mechanisms:**
  - **Endorsement of identity**
  - **Assertion of attributes**
  - **Delegation of rights**
  - **Anchored in some set of trust roots**

Issued by whom? How are are the subjects named?
What are the attributes, rights, etc.? How to broker trust?
How are authorization policies specified?

[GEC7 CF-WG presentation]

Trust Anchors

- Key question for CF architecture: what are the trust anchors/roots for the trust fabric?
  - SA, MA, CH, …etc.
- Part of the beauty of ABAC is that any entity may serve as a trust anchor for its own name space of attributes.
  - Of course…authz PDP must choose to accept those attributes.

Point 1. External IdPs and ABAC go together: IdPs serve as attribute sources for ABAC policies.

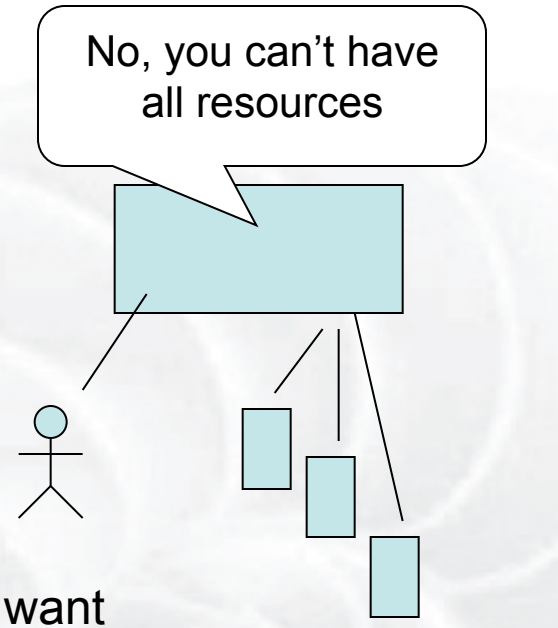
IdPs as Trust Anchors

- An IdP (e.g., Shib) is just a trust anchor maintained by an institution.
- The IdP authenticates the user agent (login).
- IdP asserts attributes of the user identity.
  - E.g., signed assertion of attributes of identity bound to an HTTPS session.
  - E.g., “Duke CS grad student”.
- Authorization policy in the server can consider these attributes (e.g., ABAC).
  - “Duke students may use this facility on Monday.”

Point 2. Resource allocation is not (just) an authorization problem.

Enforcing system-wide policy as needed

Implement system-wide policy as needed.



"The GENI system shall provide mechanisms to implement clearinghouse-wide resource allocation policies…. This will allow funding agencies or other component contributors to put overall constraints on how their components will be used."

Enable system-wide usage policies as the need arises.

Left-hand side from Chip Elliott @ GEC4

Reservation Details

Actions	Close Remove
Reservation ID	92fcd943-7675-458a-af0b-5e3480a9d6bb
Resource Type	Virtual Machine
Requested Units	5
Assigned Units	5
Leased Units	0
Lease Start	07/17/2010 15:48
Lease End	07/18/2010 15:48
Broker	broker
Site	
State	Failed
Status Message	You are authorized to reserve a maximum of 2 units.
Units	No units.

Allocation policy considers group membership attributes of requester (ABAC).

[GEC8 ORCA/Shib demo slides]

Resources: The ORCA View

- ORCA has pluggable resource allocation policy in AMs and brokering services.
- These policies may consider ABAC attributes.
- They may need other information as well:
  - Resource status
  - Allocation history of this client, group, or slice
  - Allocations and promises to other slices
  - Payment by (virtual) currency?
  - Resource delegation directives (tickets).
- ABAC is helps but is not sufficient.

Point 3. There are many other (potential) attribute roots other than IdPs.

Slice owners as attribute roots

- Would an AM trust a “random” user/ experimenter as a credential source?
- Yes, to delegate control privileges for the objects they create.
- E.g., slice owner to empower others to operate on a slice.
- ABAC delegation primitives are sufficiently powerful to do this.

SFA 2.0, Section 8

“A capability system is a special case of an ABAC framework in which all attributes directly represent specific privileges for specific objects. This restriction offers a significant simplification: since a credential represents directly the privileges that it enables, any entity may determine those privileges by inspecting that credential alone: no inference procedure is required.”

But…

- For GENI, ABAC needs a limited form of parameterized roles/attributes.
- Ownership attributes are rooted in the object creator (or the SA), not in the AMs.
- That requires some parameterization of the authz policy for objects on creation.
- We can replace the SFA registered capability authz model in a straightforward way.

Point 4. ABAC can support other features we need in the GENI trust fabric.

Example: “Stop the Experiment!”

- **Add local attributes to objects.**
  - **“A slice endorsed by a GENI-affiliated SA is a GENI slice.”**
- **Add local object attributes to the ABAC inference engine.**
  - **“If S is a GENI slice, then any entity with the GMOC role may suspend S.”**

Example: User Delegation of Authority

- SpeaksFor attribute for automated controllers.
  - “Designated driver”
- E.g., “This controller speaks for me with respect to operations on slice S.”
  - This server can act as an owner of S.
  - But I am responsible for what it does.

Other?

- Cyberphysical systems?
  - "Don't point the camera at the sun?"
- OpenFlow control of flowspace?
- Experiment opt-in?

Conclusion

- I support this proposal.
- Credential format needs some work.
- We need to standardize conventions for the basic attributes and their flow.
  - e.g., for user-created objects
- ABAC may need “just a few tweaks”.
- Eschew credential negotiation.

eom

ABAC Credential Types

1. A.r ← D
A says that D has the role A.r.

2. A.r ← B.r1
A says that any member of the role B.r1 is also a member of A.r.

3. A.r ← A.r1.r2
"If someone who A says has the attribute r1 then says somebody has the attribute r2, then A says that somebody has the attribute A.r." "This is an attribute-based delegation

Implicit type three rule: A.r ← A.r.r
(delegation of identity attribute is a special case: speaks for …not transitive)

4. intersection

A.r ← (intersection of a bunch of other roles)