Federated Shibboleth, OpenID, oAuth, and Multifactor

Russell Beall
Senior Programmer/Analyst
University of Southern California
beall@usc.edu
University of Southern California

- Private research university, founded 1880
- Budget:
  - $2.9 billion annually
  - $560.9 million sponsored research
- Locations:
  - two major LA campuses
  - six additional US locations
  - four international offices
- http://about.usc.edu/facts
University of Southern California

- 298,000+ affiliated individuals
  - 17,500 undergraduate students
  - 20,500 graduate and professional students
  - 3,400 full-time faculty
  - 11,800 staff
  - 240,000 alumni
  - 5200 sponsored affiliates with active services
  - 157 self-registered guest accounts (so far)
Problems solved

- Faculty/Staff/Student/Guest access
  - systems of record
  - automated account creation
  - sponsorship and vetting
  - mature access control infrastructure
Problems solved

- Federated access
  - self-registration
  - no USC account to maintain
  - limited sponsorship/vetting
  - works within mature access control infrastructure
New Interesting Problems

- OAuth
- OpenID
- Multifactor
OAuth and OpenID

- Alternative to Shibboleth Federated login
  - useful for:
    - non-participants
    - strict access IdPs
    - replacement of ProtectNetwork
OAuth and OpenID

- **OAuth** – secure
  - Data retrieved on backend
    - server-to-server communication
    - trust token exchange
    - secret key/token signing of requests
  - Not subject to spoofing

- **OpenID** – insecure (untrustworthy)
  - Data returned in HTTP GET parameter
  - Easily spoofed using proxy server
  - Haven’t run a spoof test, so I may be proven wrong…
Multifactor

- several quality levels:
  - lightweight version
    - local credential plus OAuth
    - local credential plus federated Shibboleth
  - full-fledged options
    - tiqr
    - Duo
    - others
Multifactor

- Two types possible with Shib:
  - decided by application
    - app chooses other factor(s) and requests as needed
    - authentication contexts coordinated by SP config
  - IdP-based
    - IdP uses multiple factors within a single context
Trust

- What is the trust model?
- How do we trust:
  - a hardware token?
  - an OAuth authentication event?
  - a Federation member authentication event?
Trust

- Trust is established with:
  - vetting
  - token management practices
- Applies equally to hard or soft tokens
- Either must be registered
Registration

- Multifactor
  - requires hardware token linking
    - phone
    - keyfob

- OAuth/Federated authentication
  - good = vetted registration under supervision
  - sufficient(?) = telephone/email communication of identifier to be trusted

- If nobody controls registration, neither can be trusted
Registration

USCnet Login

Duo Authentication

Add phone  ▶  Install Duo Mobile  ▶  Activate Duo Mobile

Add phone
1. Phone details
   United States
   +1
   ex. (201) 204-5678

Phone type  Choose...

2. Verify phone
We can call or text you with a verification code.
   Call me  or  Text me

Verification code (6 digits):  Verify

Continue
Registration

Duo Authentication

1. Get the application
   - Text me the installation link
   - Can't get text messages?

2. Install the application
   - Finish installing the application on your phone and then click Continue.
Registration

Duo Authentication

Activate Duo Mobile
1. Activate the application
   Click to get the activation link by text message:
   - Activation link sent
   - Can't get text messages?

2. Verify activation
   After activating, generate a passcode and type it below.
   - Passcode: [input field] Verify

Continue to login
Skip this step You'll still be able to log in using phone callback or SMS pass.
Registration

USCnet Login

Duo Authentication

Enrollment successful! This is the Duo login prompt that you'll normally see when logging in.

Log in using: Apple iOS (XXX-XXX-1828)

- Duo Push
- Phone call
- Passcode
  - Send SMS passcodes

Log in
Registration
Registration
Enrichment

- Registration of OAuth or Federation guest creates simple LDAP account
- No password
- Allows for enrichment of additional data including access entitlements
- Targeted enrichment creates a layer of trust
  - Layer is thin but workable
Enrichment
Demo
For Future Pondering

- With sufficient vetting, could a software token as a second factor authentication be acceptable?
  - Plus points:
    - low budget
    - hacking one account is easy, hacking two and knowing the linkage is not
  - Negatives:
    - duplicated passwords
    - depends on free APIs with no service contract