The Three S’s Of Distributed Authorization: Safe, Simple, Scalable

Eve Maler, chair of @UMAWG
tinyurl.com/umawg | tinyurl.com/umafaq

June 14th, 2013

Kantara Initiative
The “data price” for online service is too high: typing…

- Provisioning by hand
- Provisioning by value
- Oversharing
- Lying!
The “data price” for online service is too high: connecting…

- Meaningless consent to unfavorable terms
- Painful, inconsistent, and messy access management
- Oblivious oversharining
The “data price” for online service is too high: private URLs...

- Handy but insecure
- Unsuitable for really sensitive data
Most data “sharing” today is back-channel and unconsented.
Privacy is about context, control, choice and respect – so UMA enables a “digital footprint control console”

- Web 2.0 access control is inconsistent and unsophisticated
- To share with others, you have to list them literally
- You have to keep rebuilding your “circles” in new apps
- You can’t advertise content without giving it away
- You can’t get a global view of who accessed what

- You can unify access control under a single app
- Your access policies can test for claims like “over 18”
- You can reuse the same policies with multiple sites
- You can control access to stuff with public URLs
- You can manage and revoke access from one place
UMA turns online sharing into a privacy-by-design solution

Historical
Municipal
Financial
Vocational
Artistic
Social
Geolocation
Computational
Genealogical
Biological
Legal
...
UMA turns online sharing into a privacy-by-design solution

I want to **share** this stuff selectively
- Among my own apps
- With family and friends
- With organizations

I want to **protect** this stuff from being seen by everyone in the world
UMA turns online sharing into a privacy-by-design solution

I want to **control** access proactively, not just feel forced to consent over and over
UMA is a profile of OAuth, with bits added for interop and scale.
UMA solves for
1) individual choice and
2) fully modular cloud services
UMA solves for
1) individual choice and
2) fully modular cloud services
UMA solves for
1) individual choice and
2) fully modular cloud services
Key use cases

http://kantarainitiative.org/confluence/display/uma/Case+Studies

• Subscribing to a friend’s personal cloud
• Sharing accessibility attributes ("GPII")
• E-transcript sharing ("HEAR")
• Patient-centric health data access
• Enterprise "access management 2.0"
Key implementations

http://kantarainitiative.org/confluence/display/uma/UMA+Implementations

- SMARTAM.net (running authorization service from Cloud Identity UK)
- Puma (Python libraries for RS- and client-enabling web apps) from ditto
- Fraunhofer AISEC open-source implementation in Java
- Gluu OX open-source implementation for Access Management 2.0 use cases
Steve Yegge’s rant crystallized a key challenge for data sharing

[Jeff Bezos] issued a mandate that was so out there, so huge and eye-bulgingly ponderous, that it made all of his other mandates look like unsolicited peer bonuses… ‘1) All teams will henceforth expose their data and functionality through service interfaces.’

Like anything else big and important in life, accessibility has an evil twin who, jilted by the unbalanced affection displayed by their parents in their youth, has grown into an equally powerful arch-nemesis (yes, there’s more than one nemesis to accessibility) named security. And, boy howdy, are the two ever at odds.

But I’ll argue that accessibility is actually more important than security because dialing accessibility to zero means you have no product at all, whereas dialing security to zero can still get you a reasonably successful product such as the Playstation Network.
We’re finally getting around to loosely coupled identity in steps

...but we’re often not deeply protected when we do it

Fed authn tech
First for B2E/B2B (web SSO, SAML), then for B2C (social sign-in, ultimately OpenID Connect)

Fed authn biz
A burgeoning number of trust models and best practices, but little public law and few test cases; privacy is just now making a meaningful entrance

Fed authz tech
Mostly for B2E/B2B so far (“web access management”, XACML); new OAuth and UMA use cases stretch the domain boundaries

Fed authz biz
Effectively nonexistent
A technical innovation: machine-readable scope descriptions

(now modularized so OAuth and OpenID Connect can potentially use this feature too)

• AS presents “protection API”
• RS makes calls to it to register resources for protection, along with their scopes
• Scope IDs point to descriptions
• Dazza G’s innovation: include formal terms of authz in them
A business innovation: enabling “binding obligations” between parties

Diagram showing relationships between
- Authorizing Party
- Requesting Party
- Authz Server Operator
- Resource Server Operator
- Client

Connections and actions:
- Manage
- Consent
- Control
- Protect
- Trust
- Authorize
- Access
- Manage

Legal zone sign with a pedestrian crossing.
Obligations are tied to auditable changes of protocol state

• Phase 1: protect resources
  • Obligations revolve around the introduction of the AS and RS
  • The state change: issuance of a “protection API token” for OAuth-mediated access to that API

• Phases 2 and 3: get authorization and access resource
  • Obligations run the gamut of types and state changes
  • The two key ones:
    • Requesting Party-Authorizing Party: Adhere-to-Terms
    • Authorizing Party-Requesting Party: Adhere-to-Terms
    • Scope terms of authz can be surfaced up into this agreement if the AS requests a claim that confirms consent
Next steps

• We’re working on optimization opportunities when UMA, OpenID Connect, XDI, etc. are used together
• We will issue an “Implementor’s Draft” by ~end of summer
• We have liaison relationships with projects in the “trusted identities in cyberspace” ecosystem
• We are profiling and working to pilot UMA for higher ed, accessibility attribute sharing, and healthcare use cases
• We welcome your involvement and contributions
  – Become an UMAitarian!
  – Follow @UMAWG on Twitter and UserManagedAccess on Facebook
Questions?
Thank you

@UMAWG
tinyurl.com/umawg | tinyurl.com/umafaq
IIW 16, May 2013
Phase 1: protect a resource
Phases 2 and 3: get authorization and access resource

Section references are from http://docs.kantarainitiative.org/uma/draft-uma-core.html dated 6 Jan 2013

Token terminology:
* PAT = protection API token
* AAT = authorization API token
* RPT = requesting party token

Binding obligations terminology, as shown in notes over entities representing obligated parties (see http://docs.kantarainitiative.org/uma/draft-uma-trust.html):
* Subject = Individual or Non-Person Entity
* Authorizing Party = Subject acting as resource owner
* AS Operator = Subject operating authorization server endpoint
* RS Operator = Subject operating resource server endpoint
* Requesting Party = Subject acting as requesting party

Flow scenario:
* Client starts out with no AAT or RPT but is ultimately able to qualify for the required authorization

---

Client presents no RPT (Sec 3.1.1)

Learn protected resource location and scopes out of band (Sec 3.1)

Attempt access with no RPT

401 with AS location

Look up AS config data (Sec 1.5)

AS config data (Sec 1.5)

Dynamic client registration if necessary (Sec 3.4, draft-ietf-oauth-dyn-reg)
Phases 2 and 3: 
get authorization and access resource 
2 of 3
Phases 2 and 3: get authorization and access resource

1 of 3
Spec call tree for the UMA profile of OAuth

UMA core

- OAuth 2
- OpenID Connect
- Token introspection
- OAuth resource set registration
- UMA binding obligations
- Dynamic client registration
- hostmeta

UMA native spec

Required external component

Optional external component

Individual IETF I-D