Extending the Power of Consent with User-Managed Access

A Standard Architecture for Asynchronous, Centralizable, Internet-Scalable Consent

Eve Maler
CTO Office
ForgeRock
San Francisco, USA
eve.maler@forgerock.com

***Abstract*— The inherent weaknesses of existing notice-and-consent paradigms of data privacy are becoming clear, not just to privacy practitioners but to ordinary online users as well. The corporate privacy function is a maturing discipline, but greater maturity often equates just to greater regulatory compliance. At a time when many users are disturbed by the status quo, new trends in web security and data sharing are demonstrating useful new consent paradigms. Benefiting from these trends, the emerging standard User-Managed Access (UMA) allows apps to extend the power of consent. UMA corrects a power imbalance that favors companies over individuals, enabling privacy solutions that move beyond compliance.**

Keywords—privacy; consent; authorization; permission; access control; security; personal data; digital identity; Internet of Things

I. Introduction

The go-to user consent mechanisms used in typical online interactions, such as opt-in checkboxes, buttons, and forms – or, worse, opt-outs –satisfy compliance rather than a customer or user need for managing the exposure of personal data.

These mechanisms have been suffering in effectiveness under advances in technology and digital commerce such as Big Data, biometric authentication, and Internet of Things (IoT) devices, from smart cars to smart buildings. It is becoming impractical to limit information collection and keep users fully informed about what information is being collected in the general case, and it is certainly impractical to achieve this goal moment-by-moment at collection time. If clicking an "I Agree" checkbox following the display of screens of legalese were ever a good solution to online privacy challenges, it no longer is.

According to the most recent International Association of Privacy Professionals salary survey, the top driver for corporate privacy funding was meeting compliance obligations, and more than half of privacy groups in corporations and governments reported into either a legal or a compliance department [1]. This suggests that most mature companies are prioritizing their own policy needs rather than customer desires.

Signs are appearing of a growing appetite for privacy as a business differentiator. Pew Research recently reported that 91% of Americans agree or strongly agree that consumers have lost control over how personal data is collected and used, and 80% who use social networking sites are concerned about third parties accessing their shared data [2]. Webbmedia Group, writing in the Harvard Business Review, has identified data privacy as one of the top ten technology trends of 2015 [3].

However, without tools and mechanisms that deliver post-compliance consent features, organizations will not be able to deliver on goals for privacy that go beyond current capabilities. This paper introduces an emerging web standard called User-Managed Access (UMA) [4], whose architecture enables conforming applications to offer stronger consent experiences and consent management abilities. UMA is analyzed along with other existing and emerging consent mechanisms against an optimistic set of requirements for consent.

II. Optimistic Requirements for Consent

The ordinary word *consent* has several senses: the capture of fully considered and empowered permission (with a synonym of *authorization*); or of harmonious approval (*agreement*); or of passive assent (*acquiescence*).

The Article 29 Data Protection Working Party opinion on the definition of consent [5], and Fair Information Practice Principles (FIPPs) for privacy the world over, prescribe a lofty mix of transparency, control, and consent goals. The Privacy by Design (PbD) framework [6] developed by Dr. Ann Cavoukian goes further, defining principles that encourage moving beyond privacy compliance – for example, taking a proactive rather than a remedial approach, and embedding privacy into design. In practice, however, current consent mechanisms tend towards the *acquiescence* end of the continuum.

It is useful to define a set of optimistic consent requirements that take into account new technology trends:

- **Choice:** Maximize opportunities for individual *authorization* for, and mutual *agreement* to, personal data sharing; minimize *acquiescence* to sharing and *unconsented* sharing. Rationale: Decisional autonomy is the bedrock for data privacy, and – barring the security benefits from good data protection practices – "privacy" is meaningless without individual choice.

DOI 10.1109/SPW.2015.34

- **Relevance:** Capture consent at a time and in a manner most relevant to and convenient for the individual. Rationale: The "right" thing to do must be the easiest, as observed by Donald Norman in *The Design of Everyday Things*: "Everyday activities must usually be done relatively quickly, often simultaneously with other activities. .... Subconscious thought is biased toward regularity and structure, and it is limited in formal power. It may not be capable of symbolic manipulation, of careful reasoning through a sequence of steps." [7] It is unreasonable to expect individuals to make good choices at inconvenient, unnatural, or distracted moments. This is especially true given the constrained interfaces of IoT devices.
- **Granularity:** Enable differentiation of the parameters of consent, including data sources, data items, receiving parties, and modification of consent parameters over time, including revocation, again in a manner most relevant to and convenient for the individual. Rationale: Blanket consent decisions that cover "too much ground" take away choice.
- **Scalability:** Enable consent interactions, processes, and systems to scale to accommodate the numbers of data sources, data items, and consent functions that individuals will realistically experience. Rationale: Web and API interactions are challenging enough in scale, with each user having a dozen or more websites to deal with. In an IoT world, where each kitchen appliance, door lock, and even item of clothing may be a source of personal data, and where any individual can become a "maker" of devices, scale issues will become critical.
- **Automation:** Enable machine processing and recording of consent functions. Rationale: Automation improves speed of handling, accuracy of fulfillment, and auditability. Human handling can lead not only to error but potentially to extra exposure of sensitive data; it is for this reason that some enterprise architecture platforms inject API keys into compiled software code only after developers have completed their work.
- **Reciprocity:** Capture the consent of the data-receiving party in dealing with the individual, along with capturing the consent of the individual in sharing data. Rationale: This is in the spirit of *agreement*. The receiving party may in fact be a human being as well, to whom we must grant the same privacy privileges.

III. Analysis of Typical Consent Mechanisms in Use

With these requirements defined, we can turn to an analysis of typical consent mechanisms in use and the language used to understand and describe them.

A. Consent Mechanism Classification System

Consent mechanisms are generally classified as follows, from strongest to weakest:

- **Opt-in:** The individual positively acts to consent. This is ideally **informed**, ensuring the individual appreciates relevant facts about collection, use, and consequences. One subclass is **express**, where the action is explicit and direct – for example, checking an approval box, or clicking an "I Agree" button. The other subclass is **implied**, where the action is inferred to mean consent – for example, providing a phone number in an email signature block, indicating that calling and storing the number is acceptable.
- **Opt-out:** The individual "gives consent" passively by *not refusing* to consent. If the individual takes express action to refuse consent (essentially revoking a system-default consent), then he or she has opted out.
- **Unconsented:** This category covers cases where collecting consent is impractical, such as when an individual is unconscious in an emergency room and medical personnel need access to records, or unnecessary, such as when laws allow public access to arrest records regardless of an arrestee's wishes.

B. Analysis of Existing Consent Mechanisms

The classification system works well to encompass three consent mechanisms that people frequently encounter: 1) digital opt-in interactions when accepting terms of service (ToS) (for example, when installing mobile apps); 2) opt-in/out interactions for the use of browser cookies; and 3) medical "consent directive" forms on paper that record their wishes for sharing health information with caregivers and family members.

However, the system is less helpful in the face of two modern consent-related trends: 4) OAuth-based [8] authorize/deny interactions for achieving "social" login into and connection between applications (for example, using one's Facebook account to log in to comment on a web news article); and 5) person-to-person "Share" features in web apps.

OAuth-based "opt-in" consent for an application to access a web API on the user's behalf results the issuance of a "scoped" – constrained-use – access token for the application to use. While the language of consent usually revolves around personal data, such as attribute data fields uploaded as part of online forms, APIs might involve access to user-created digital content or other digital assets that reveal personally identifiable information. APIs might even involve access for the purpose of adding or changing content – not only retrieving it. This could be classified as express consent, but not for sharing of personal data as usually conceived.

Data sharing with other people, for example with web-based word processing applications such as Google Docs, involves using the "Share" button and similar interfaces, and it also often involves "scoping down" the recipient's extent of access (say, enabling viewing vs. editing). This act – very unlike traditional opt-in consent – works like express consent.

Table I grades these mechanisms against the requirements in Section II as strong (+1), neutral (0), or weak (-1).

TABLE I. EXISTING CONSENT MECHANISMS AGAINST REQUIREMENTS

Requirements	Existing Consent Mechanisms				
	ToS opt-in	*Cookie opt-in/out*	*OAuth*	*"Share"*	*Consent directive*
Choice	-1	0	0	+1	+1
Relevance	-1	0	+1	+1	0
Granularity	-1	0	0/+1	+1	0
Scalability	-1	+1	0	-1	-1
Automation	-1	0	+1	+1	-1
Reciprocity	-1	-1	-1	0	-1

ToS interactions, which infamously force users to acquiesce (Choice), perform badly on other requirements too.

Consent directives, despite their proactive (Choice) nature, suffer because they are paper-based. Patients must fill them out at inconvenient times (Relevance), and the forms are difficult to access at information sharing time (Automation).

The European Union legislation on cookies [9] is largely responsible for preserving such strengths as this opt-in/out mechanism has, but it is still relatively weak.

The OAuth mechanism has promise. OAuth's notion of "scope" allows constraints to be imposed on the extent of access (Granularity). It earns a neutral Choice grade and an ambiguous Granularity grade because while users can revoke consent at will, and can sometimes uncheck specific scopes of access when granting consent at an app's discretion, generally they are pressured into acquiescence in order to receive service. It earns a neutral Scalability grade because many third-party applications can connect to one service, but the relationship with each service is pairwise.

The "Share" mechanism has great promise because it is proactive, despite it counterintuitive role in consent. It earns a positive Automation grade because it manages sharing automatically. It earns a negative Scalability grade because it must be implemented anew for each ecosystem.

IV. INTRODUCING USER-MANAGED ACCESS

Even if traditional styles of consent interaction comply with regulations and FIPPs and have been deployed with a robust application of privacy discipline, we can observe that they do not serve individuals particularly well, while more modern consent interactions that "draw outside the lines" of privacy conversations show important hints of improvement. The emerging web standard UMA, composed of two Version 1.0 Draft Recommendations produced by the UMA Work Group of the Kantara Initiative, takes advantage of these modern technologies and interaction models.[1]

A. UMA Capabilities, Roles, and Flows

The UMA protocol [10] is a profile of OAuth. It was designed to give an individual a unified control point for authorizing who and what can get access to his or her online personal data (such as identity attributes), content (such as photos), and services (such as creating status updates), no matter where those resources live online. Further, UMA allows the individual to configure the control point to test the requesting side's suitability for authorization, including identity (such as "Do you control the email address bob@gmail.com?") and promises (such as "Do you agree to these nondisclosure terms?"). This is known as claims-gathering [11] and it has a role to play in data usage control.

The roles of the actors in an UMA flow are:

- **Resource owner:** An individual (or organization) with primary control over access to resources.
- **Authorization server:** A unified control point the resource owner uses to manage resource access.
- **Resource server:** One of potentially many hosts of protected resources (such as personal data).
- **Requesting party:** An individual (or organization) seeking access to a protected resource; sometimes the resource owner is in the role of a requesting party.
- **Client:** An application used by a requesting party.

The UMA architecture is shown in Fig. 1. The protection and authorization APIs are UMA-standardized RESTful web APIs that coordinate protection over some application-specific interface exposed by the resource server. These standardized APIs are themselves secured with embedded OAuth flows. These embedded flows enable: 1) the resource owner to consent to having the resource server outsource protection of its resources to the authorization server (represented by an OAuth "protection API token"); and 2) the requesting party to consent to having the client send whatever personal data is required to the authorization server, satisfying the claims-gathering process, in order to seek authorized access (represented by an OAuth "authorization API token").

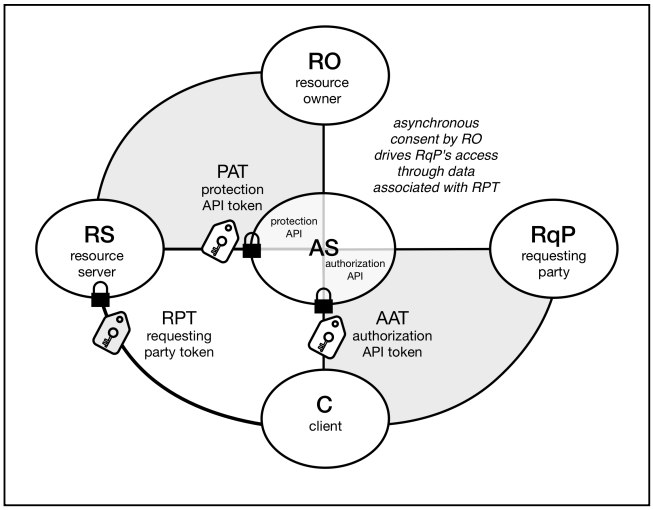

Fig. 1. UMA Architecture

[1] The author is the founder and chair of the UMA standards effort.

B. UMA Use Cases

While UMA has been under development for several years, an analysis of its impact on consent is apropos given that the specifications have now stabilized, have seen multiple implementations, and are motivated by a widening variety of use cases. Following are two currently under active discussion:

- **Financial data for tax returns:** Taxpayer Alice wants to share, for a limited length of time, access to the data about how much income she made last year with her chartered accountant Bob. Alice is the resource owner and Bob is the requesting party. Her paycheck application is a resource server, exposing an API and scopes for accessing her income data. A central data-sharing hub application (authorization server) helps her manage her data exposure to Bob and others. Bob uses a tax return preparation client app.
- **Individual-centric health data sharing:** Nurse and veteran Alice has a cardiac defibrillator installed and also uses a fitness wearable to manage her health. She lives in the United States but travels to developing countries to apply her nursing expertise there, so she uses several doctors' medical portals as resource servers. She uses a generic data sharing manager (authorization server) offered by her former university to manage health data flow. She and her doctors are requesting parties, using a variety of clients. Some of her devices both generate (resource server) and consume (client) data.

C. UMA Consent Experiences and Implications

Whereas ordinary OAuth is reactive in nature, presenting a user with an Authorize/Deny interface ("opt-in"), UMA gives the resource owner control. The requesting party can attempt access without requiring the resource owner's presence, and the resource owner can choose when and whether to consent. This makes consent *asynchronous*. The resource owner might experience this type of consent as follows:

- **Before access attempts:** The resource owner proactively sets access policies, potentially involving "Share" interface paradigms and also "Register resource" paradigms for onboarding new resources, such as light bulbs in a home automation scenario. Implications: The resource owner expressly consents and sets the conditions for access (to the limits of the authorization server's ability) rather than having to accept offers from others – a powerful position from which to negotiate, and a powerful moment to capture a person's intent: the moment he or she desires to share a digital asset. Setting up sharing parameters ahead of time is also especially convenient for IoT devices that have unfriendly interfaces for real-time consent.
- **After access attempts:** The resource owner handles access approval requests and consent modifications and revocations. If the requesting party requires a timely response from the owner in real time, the owner could be given the opportunity to respond, for example through a push notification sent to a smart mobile device. Implications: The resource owner chooses when and whether to opt in, retaining control of the data-sharing relationship and creating incentives for requesting parties to keep relationships on an even keel.
- **During access attempts:** When the requesting party is also the resource owner, the attempt coincides with a live online "consent" session. Implications: Consenting to "one's own access" can be managed consistently.

D. Analysis of UMA Against the Requirements

Table II grades UMA against the optimistic requirements.

TABLE II. UMA AGAINST REQUIREMENTS

Requirements	Consent Mechanism
	User-Managed Access
Choice	0/+1
Relevance	+1
Granularity	+1
Scalability	0/+1
Automation	0/+1
Reciprocity	+1

UMA's reliance on OAuth gives it strength, and its asynchronous nature strengthens it further. Individuals can direct sharing proactively and handle requests reactively (Choice and Relevance), and leverage scopes in policy setting (Granularity). Its handling of requesting-party consent earns it a good Reciprocity grade. It earns ambiguous Choice and Scalability grades only because businesses have incentives to reserve consent-handling powers to themselves, and the jury is still out on wide-scale adoption. It earns an ambiguous Automation grade because it does not standardize a policy expression format, nor yet an audit mechanism.

V. FUTURE WORK

After Version 1.0 Recommendation completion, the UMA Work Group will consider issues for potential future development, such as formal auditability. The healthcare IT community has also launched a standards group called Health Relationship Trust (HEART) that includes an UMA profiling component for patient-centric health data sharing use cases.

One area requiring special attention is claims-gathering, as each application ecosystem's process will vary. For example, in health data sharing, the resource owner may want to ensure that the requesting party is a family member in control of a known email address, an accredited doctor, or a hospital employee. Ecosystem members will need to join agreements – known as access federation trust frameworks – to ensure expectations are enforceable. A draft UMA specification [12] assists with potential contract clauses that might appear in such frameworks. External work on chain-link confidentiality

[13] is also relevant here, to ensure that sharing of a resource owner's data that takes place "downstream" from the initial requesting party is controllable to some level even in the absence of purely technical means of control, such as encryption or digital rights management (DRM).

VI. Conclusion

While the practice of privacy is an increasingly mature discipline, traditional consent tools are unable to live up to customer and business demand for new data privacy options. Further, consent language is not keeping up with newer options for engaging with online data-sharing flows involving web API access authorization and the "Share" paradigm.

The UMA protocol offers features that expand the power of consent, both materially through centralizing, standardizing, and increasing the "grain" of consent, and rhetorically through the notion of asynchronous, centralizable consent. As demands to share personal data increase, and new reasons arise for people to wish to share data on a selective basis, UMA gives new opportunities for reclaiming positive senses of "privacy" and "consent".

VII. Related Work

Two additional standards efforts are of particular interest. The first is the Extensible Access Control Markup Language (XACML) [14], which provides standard declarative formats for authorization policy. These formats could be used in conjunction with UMA to increase the latter's Automation grade. The second is the Consent Receipt effort at Kantara [15], which could improve the grades of existing consent mechanisms against the requirements. It could also eventually be used in concert with UMA mechanisms, for example in reciprocal person-to-person sharing scenarios to encourage virtuous circles of online selective sharing.

Acknowledgment

The author thanks Heidi Shey and Richard Mardling for their kind review of this paper, and various "UMAnitarians" for their review and ongoing support: Maciej Machulak; Domenico Catalano for graphical help; Robert Lapes for discussions about his work curating and building consent taxonomies for the GUIDE Project; and Mark Lizar for the notion of "mere assent".

References

[1] International Association of Privacy Professionals. (2012). *2012 Privacy Professionals Role, Function and Salary Survey* [Online]. Available: https://www.privacyassociation.org/media/pdf/knowledge_center/IAPP_Salary_Survey_2012.pdf

[2] M. Madden. (2014, November 12). *Public Perceptions of Privacy and Security in the Post-Snowden Era* [Online]. Available: http://www.pewinternet.org/2014/11/12/public-privacy-perceptions/

[3] A. Webb. (2015, January 5). *The Tech Trends You Can't Ignore in 2015* [Online]. Available: https://hbr.org/2015/01/the-tech-trends-you-cant-ignore-in-2015

[4] M. Machulak, E. Maler, D. Catalano, and A. van Moorsel, "User-managed access to web resources" in Proceedings of the 6th ACM workshop on Digital identity management. ACM: New York, NY, 2010, pp. 35-44.

[5] EU Article 29 Working Party. (2011, July 13). *Opinion 15/2011 on the definition of consent* [Online]. Available: http://ec.europa.eu/justice/policies/privacy/docs/wpdocs/2011/wp187_en.pdf

[6] A. Cavoukian (2011, January). *Privacy by Design: The 7 Foundational Principles* [Online]. Available: http://www.privacybydesign.ca/content/uploads/2009/08/7foundationalprinciples.pdf

[7] D. Norman, The Design of Everyday Things. Basic Book, New York, NY: Basic Book, 2002, p. 125.

[8] D. Hardt, Ed. (October 2012). *The OAuth 2.0 Authorization Framework* [Online]. Available: http://tools.ietf.org/html/rfc6749

[9] European Commission (2015, March 2). *The EU Internet Handbook: Information Provider's Guide: Cookies* [Online]. Available: http://ec.europa.eu/ipg/basics/legal/cookies/index_en.htm

[10] T. Hardjono, *et al.* (2015, February 23). *User-Managed Access (UMA) Profile of OAuth 2.0* [Online]. Available: https://docs.kantarainitiative.org/uma/draft-uma-core.html

[11] E. Maler. (2010, October). *Controlling Data Usage with User-Managed Access (UMA)* [Online]. Available: http://www.w3.org/2010/policy-ws/papers/18-Maler-Paypal.pdf

[12] E. Maler and T. Hardjono. (2013, January 25). *Binding Obligations on User-Managed Access (UMA) Participants* [Online]. Available: http://docs.kantarainitiative.org/uma/draft-uma-trust.html

[13] W. Hartzog, "Chain-Link Confidentiality" in Georgia Law Review, Vol. 46, 2012, p. 657.

[14] E. Rissanen. (2014, January 22). *eXtensible Access Control Markup Language Version 3.0* [Online]. Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf

[15] M. Lizar and J. Wunderlich. (2014, October 1). *Minimum Viable Consent Receipt (MVCR) Specification v.05* [Online]. Available: https://kantarainitiative.org/confluence/display/infosharing/Minimum+Viable+Consent+Receipt+(MVCR)+Specification+v.05