

An open assertion and evidence exchange and query language – requirements and abstract syntax

Author: Giles Hogben

Affiliation: European Commission Joint Research Centre, Ispra, Italy

Introduction

Efficient, legal and user-friendly management of identities has emerged as one of the key problems to be solved in current IT infrastructure. Industry and ecommerce require high volume, well-authenticated and accurate personal data, with efficient protocols and architectures for collecting, updating and querying such data.

End users struggling with password-fatigue require solutions which automate the provision of authorization credentials. Surveys also show that, given the choice, significant numbers of end users prefer to minimize the amount of identifiable information disclosed in electronic transactions [Business Week/Harris Poll] [FORRESTER][Crano]. It has been shown, however that users are willing to invest very little time and effort in understanding identity management systems and in dealing with mechanisms for increasing their privacy [Sheehan][Palen].

Government legislation places stringent requirements for privacy of personal information, but under certain conditions, requires strong identification and re- vocability of this privacy. A key requirement coming from legislation is created by the principle of minimization of data collection, whereby data collected by services should be minimal for the purpose required.[DIRECTIVE] This is an important requirement in the design of identity management solutions. Existing solutions almost always request more data than is required. This is in part because ad- ditional data often cannot be captured later and it is difficult to determine the minimum information required for a given purpose. Furthermore, even when the data corresponding to minimum disclosure can be clearly determined, traditional data structures and even certificates often do not provide the appropriate semantics for selecting and transmitting such data.

Traditional certificates for example either show all the attributes contained in the certificate or none at all. Federated identity solutions using e.g. SAML [SAML] allow for selection of single attributes, the flexibility available within the attribute/value data structure provided does not allow for true minimization. For example, strictly speaking, for the purpose of renting out a car, a rental agency needs to know that a person is the holder of a valid driver's licence and that they have paid for the service. Any further information is required only under certain conditions, such as the occurrence of an accident. However, existing data structures are set up in such a way that the only way to prove that one is the holder of a valid driver's licence is to certify all the information held in a driver's licence. They do not provide for proving for example, that you hold a credential of a certain type (e.g. a drivers' licence), without providing the credential.

Apart from such problems for servers in expressing assertions required for access control another important problem is the expression and evaluation of evidence offered in support of assertions. The most common manifestation of this problem is the difficulty of users in evaluating the trustworthiness of security certificates. Users often do not know whether to trust individually named organizations which are described by cryptographic certificates. Such decisions are also very time-consuming and users often make poor decisions as a result. Furthermore there is no semantics available for automatic evaluation of such certificates based on delegation to preferences expressed in machine-readable rules.

Requirements for an Assertion and Evidence Language

Based on the constraints outlined in the introductory discussion, we derive the following requirements for a general Identity Management system. These also apply to both traditional IDM systems and systems based on anonymous credentials.

1. A mechanism for requesting assertions about identity principals (either by the owner or from an identity provider). This mechanism should be able to describe a set of assertions which a desired response must be a member of. In other words, the relying party must be able to communicate to the principal which assertions it needs to be assured of before allowing access to a service. This request can also be passed on to the identity provider as a request for security tokens.
2. A mechanism for communicating assertions about identity principals (either by the owner or from an identity provider). For example the system must be able to communicate an assertion that 'Bob's email address is 'bob@foo.com''. Such assertions may also be communicated by the identity provider.
3. The assertion language should be able to express assertions which minimize the increase in the server's knowledge about the user for a given interaction. For example instead of providing an exact birthdate in order to buy a 12 rated movie ($\text{Equals}(\text{valueOf}(\text{User}, \text{Age}), 12)$, it should be possible to prove the exact minimal assertion required - i.e. $\text{Greaterthan}(\text{User}, \text{age}, 12)$. When interpreted strictly, this requirement has the important implication that the assertion request and return language must allow for arbitrary arity predicates rather than restricting assertions to name-property-value triples as in state of the art identity management systems.
4. Assertion requests should, according to the knowledge available to the requester, include in the set of assertions requested, the assertion which minimizes the knowledge transferred about the user, while still satisfying the access control request. Henceforth, we shall call this the minimal-disclosure assertion. Consider, for example, a cinema ticket webstore which needs to know that the person being issued the ticket is over the age of 12. If the server requests $\text{Greaterthan}(\langle U \rangle, \text{age}, 12)$ - evidence that the user's age is greater than 12, then all assertions stating an exact birth date before 1994 would be a member of the requested set, AND the minimal disclosure assertion - that the user's age is greater than 12 is also be a member of the requested set. If instead the server requests $\text{Equals}(\text{valueOf}(\text{User}, \text{Age}), ?)$ - i.e. only the exact birth date of the user is satisfactory, then the minimal disclosure assertion, $\text{Greaterthan}(\langle U \rangle, \text{age}, 12)$ is not included in the requested set, therefore this requirement is not met. The following diagrams illustrate this requirement in set-theoretic terms.

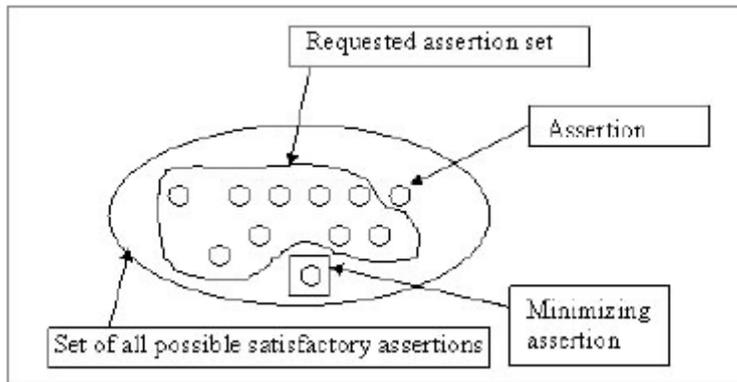


Fig 1. Legal assertion request

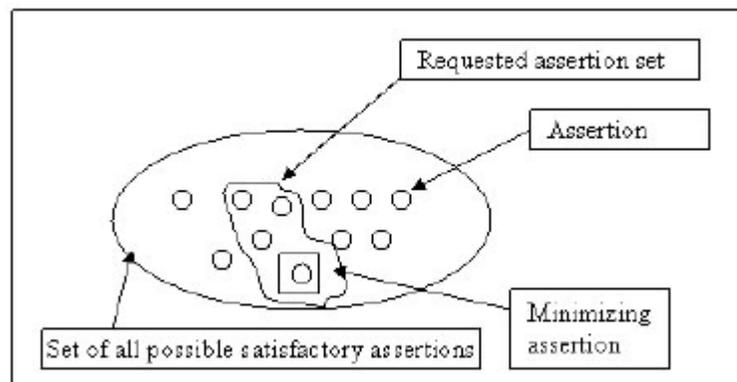


Fig 2. Illegal assertion request

5. Note that conversely to the minimization requirement, the service may deliberately choose not to specify the full set of assertions which would satisfy the request. From the point of view of the server's policy secrecy, the assertion set should be as big as possible (to release as little information on the policy as possible). The larger the set of possible assertions in the server request, the less information is revealed about the server's policy. This maximization of the assertion request set is known as policy sanitization. For example a recruitment company might ask for assertions about the user's height instead of the minimal set of assertions it actually requires which is that he/she must be over 1m60 tall.
6. A mechanism for describing, requesting and evaluating evidence for assertions (in order to make trust/risk evaluations). For example, at a minimum, the assertion `Greaterthan(User,age,12)` can be simply declared to be true. In this case, no evidence is offered other than the fact that it is being asserted by a party whose identity is known with a certain degree of trust. Or at the other extreme, it can be signed by a vouching party whose identity is assured by cryptographic means and who is trusted by the relying party. In between these 2 extremes, evidence may also be presented which is not cryptographic, but still increases the trust of the relying party to a certain degree. For example the URL of a web page which appears to have been created by the user more than 12 years ago might be presented as evidence that the user is more than 12 years old. Obviously not all evidence carries equal weight when evaluating the trustworthiness of an assertion but meta-data structures must be provided which allow different grades of evidence to be expressed unambiguously.
7. Evidence meta-data should allow for the efficient evaluation of the trustworthiness of assertions. Generally, decision making processes are based on access control policies which express conditions over assertions and the evidence offered to support them (currently usually none, or cryptographic certificates). Such policies are usually evaluated automatically by the relying party's system. So for example if a set of url's is offered as evidence, then the characteristics of these urls which support the assertions should be described by the metadata in such a way as to allow an automated trust decision on their basis.
8. If evidence is provided in the form of a testimonial from a vouching party (usually a public key certificate), then the metadata must provide a. the means to verify the identity of the vouching party (in most cases this means that the name, verification method and means to access the public key of the verifier should be accessible). b. all properties of the vouching party relevant to a trust decision on this evidence. In other words, the meta-data should allow a means to verify the identity of vouching parties and, given successful authentication, meta-data to decide the weight to be given to the vouching party's testimonial.

Abstract Semantics

We propose here a formal abstract semantics to describe the properties set out in the requirements. A full description of concrete semantics and syntax for several key cases (Public Key certificates, Idemix proofs) can be found in Hogben, Sommer [IBM report]. Detail of abstract semantics of assertion-evidence tuples.

Assertions and Evidence

A key point to note here is that there is a clean separation between assertions and evidence offered in support of those assertions. The "factoring out of trust" allows for separate, dedicated components to evaluate trustworthiness of assertions independently of what is being asserted. It also allows for new types of evidence to be used in support of assertions. For example it allows the framework to be widened out to include evidence in the form of reputation.

In [IBM Report] we describe a broad range of types of evidence which can be used to support assertions within such a framework. Concentrating on the case of evidence in terms of cryptographic certificates, we have described in detail how an OWL ontology describing abstract properties of certificates may be used to exchange and evaluate evidence for certificates.

Describing Certificates as Evidence

Traditionally, policies for certificate evaluation are restricted to lists of trusted public keys against which certificates are matched. This tends to lead to closed federations of certification providers where often only one issuer is accepted. There has not been an attempt instead to model abstract properties of certificates. We suggest that an abstract model of certification properties brings several important advantages:

- a. It allows users to describe rules over properties which they can easily understand, rather than having to understand technical aspects of certification or be familiar with individual certification authorities. For example certificates could be modelled in relation to properties of non-electronic certification (government issue, falsifiability, etc...)
- b. It facilitates the distribution of default rule sets over certificates, which can be directly written using concepts from (for example) legislation which are contained in the abstracted certification properties. For example, a default rule set distributed might specify that the Identity Provider Trust Level should be OECD government for accessing criminal record data.

Our ontology of cryptographic evidence provides abstract properties which can be used to evaluate certificates using rulesets expressed in terms of user-friendly concepts.

Assertion semantics

State of the art identity management frameworks allow only the assertion of binary predicates with the user's pseudonym as the first argument (in other words pseudonym property object). As we have seen in the requirements, this does not satisfy the minimizeability requirement of privacy protection legislation (and best practice).

Given that the minimal-disclosure assertion for a given access control decision may be expressible only by an n-ary predicate, where $n > 3$ and only one argument as the principal's pseudonym, we need to extend the semantics to include such n-ary predicates. Minimized assertions may also be boolean formulae. For example, it is often the case that proving that you have one of 2 credentials such as a passport OR a drivers' licence (but not proving which one you have), is sufficient to gain access to a system.

This has significant implications for the anonymity of end-users. For example if a user can prove that they have a passport from any EU country (but not which one), then they do not need to reveal their nationality by proving such an assertion. It is also worth noting that restriction to simple binary predicates is a severe limitation on the power of such systems to handle arbitrary personal information, especially within enterprise processes. We therefore define the abstract syntax of an assertion and evidence to support n-ary predicates, using n-ary predicates in tuples of the form:

Definition 1. $\langle A(P_i(N_j)), E(A) \rangle$

where A is a boolean propositional formula composed of N-ary predicates P_i with arguments N_j , $(P_1(N_0, N_1) \wedge P_2(N_2) _ P_3(N_3) \wedge \dots \wedge P_i(N_j, N_{j+1}))$ etc.... E is evidence offered in support of A.

The request species a set of assertions and accompanying evidence which can be used as access credentials to the service offered by the relying party. In general, such a request may be described in terms of a set of conditions on assertions and evidence which must satisfy the access control request.

Definition 2. $\forall (A_i, E_i(A_i)) \in R, C_j(\langle A_i, E_i(A_i) \rangle)$

Where R is a satisfactory response, A_i are n-ary predicates and E_i are evidence for A_i and C_j is a condition predicate on . Note that C_j is essentially a query language - a language for specifying acceptable sets of results from within a larger set. The abstract syntax does not stipulate that this language should be expressible in terms of templates for the _nal assertions set. So for example, C could state that for to be a member of the response set, it must have been stated between 8am and 6pm. The request must be capable of addressing the space covered by assertions in the response. Therefore, as the response is defined in terms of boolean formulae of n-ary predicates, such conditions should be defined in terms of a boolean formula of n-ary predicates, in combination with a query semantics for defining conditions on the assertions space.

An example of such a condition is $C(\text{Equals}(\text{User}, \text{Age}, ?x), \text{greaterThan}(?x, 21))$.

Concrete Syntax

A full description of concrete syntax and implementation of this semantics is given in [IBM report]. Here we note the following details of the implementation:

Assertion Syntax

We have shown that n-ary predicates may be efficiently expressed using RDF [RDF] syntax such as N3 [N3]. RDF assertions may then be grouped into Named Graphs [NG] which can be referenced when applying evidence.

Evidence

Evidence is also described in terms of RDF predicates based on an OWL ontology. The ontology is built around the top level concepts of:

- Identity Verification Method - how the identity of the principal of the claims in the certificate is verified according to the procedures known for the certification provider
- Algorithm - Algorithms are also described by abstract properties which can allow users or legislators to describe them without having to understand technical details.
- Identity Provider Trust - a set of categories analogous to those found for physical certificates, which can be used to categorize identity providers issuing certificates.
- Security Method - a classification of the physical security methods applied within the authentication scheme of the identity provider. For example if processes are verified by an audit certificate or protection profile of a certain level.

Security Model

A key point to note is that decisions based on concepts contained in an OWL ontology require the provenance of that ontology to be secure to avoid attacks based on poisoning of reasoning processes by substituting false ontologies. We have therefore provided a security model to allow secure reasoning over ontologies.

Assertion/Evidence Request

We have shown in [IBM Report] that Assertion Requests can be mapped directly onto SPARQL [SPARQL] queries over named graphs. With assertions and evidence expressed in terms of RDF triples, and a well-defined mapping between assertions and evidence, SPARQL queries can be used to describe assertion and evidence sets which satisfy access-control requirements. Since SPARQL is specifically designed to operate over RDF triple sets, it is very well-suited as an assertion-request language and clearly satisfies the requirement 4. above.

Conclusion

Current implementations do not fully satisfy the requirements of legislation and useability, in particular the requirement for minimizeability of the data request and the requirements for useability of certification metadata schemes. This can be achieved using RDF for assertions, SPARQL for assertion request semantics and OWL to provide an abstraction of certification properties.

References

[IBMReport] Hogben, G., Sommer, D. A meta-data and reasoning framework for open assertion and evidence exchange and query, IBM Report number RZ 3674

[Business Week/Harris Poll] http://businessweek.com/2000/00_12/b3673010.htm

[Harris] Harris, Louis and Associates and Westin, A.F. 1998. E-commerce & Privacy: What Net Users Want. Privacy & American Business, Hackensack NJ.

[FORRESTER] "Online Consumers Fearful of Privacy Violations" (Oct. 1999), <http://www.forrester.com/ER/Press/Release/0,1769,177,FF.html>.

[CRANO] Crano et al., "Beyond Concern: Understanding Net Users' Attitudes About Online Privacy," 5 (1999), available at <http://www.research.att.com/projects/privacystudy>.

[DIRECTIVE] http://ec.europa.eu/justice_home/fsj/privacy/docs/95-46-ce/dir1995-46_part1_en.pdf

[Sheehan] Sheehan, K.B. 2002. Toward a Typology of Internet Users and Online Privacy Concerns. *The Information Society*, 18, 21-32.

[PALEN] Palen, L. and Dourish, P. 2003. Unpacking "privacy" for a networked world. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Ft. Lauderdale, Florida, USA, April 05 - 10, 2003). CHI '03. ACM Press, New York, NY, 129-136. DOI=

<http://doi.acm.org/10.1145/642611.642635>

[RDF] <http://www.w3.org/RDF/>

[N3] <http://www.w3.org/DesignIssues/Notation3>

[SPARQL] <http://www.w3.org/TR/rdf-sparql-query/>

[SAML] <http://www.oasis-open.org/committees/security/>

