Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: February 6, 2015

L. Johansson, Ed.
NORDUnet
H. Flanagan
Spherical Cow Consulting

August 5, 2014

Requirements on an Attribute Registry
draft-johansson-areg-reqs-02

Abstract

This document establishes requirements for a registry of attribute-type definitions.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on February 6, 2015.

Copyright Notice

Table of Contents

1. Introduction and Motivation

An attribute is a representation of a single datum of information associated with an entity. The type of the attribute (the 'attribute type') is defined by semantics and syntax that allow it to be used in a variety of protocols and representations.

This document lists requrements for a registry of such attribute-type definitions. For a long time, protocols that rely on the transfer of

attributes (like OpenID Connect, OAUTH, WS-Federation or SAML) often rely on, at least in the case of attributes associated with accounts and persons, attribute-type definitions that are borrowed from LDAP or X.509 schema even though those particular protocols no longer represent the common method to transfer and consume attributes.

Claims-based protocols (for instance SAML or OpenID Connect) are widely used on the Internet today. A common use-case for such protocols is to establish identity federations that rely on the transfer of attribute-values as a means to communicate subject information. Identity federations are often purposed to specific communities. Increasingly such communities need to engage in transactions across federation boundaries (e.g., when sharing services with other communities). This practice is called inter-federation. Inter-federation raises the need for a way to discover information about the attributes used in the protocols employed inside and between federations.

This document attempts to address these problems by establishing a set of requirements for an Internet-wide registry of attribute-type definitions. This document does not attempt to establish the registry, that will be the work of future specifications.

2. Core Concerns

In order to set the stage for and properly frame the registry requirements, the following section lists a set of core concerns that MUST be address by the registry requirements proper:

2.1. Naming

It is implied that attribute types have names that uniquely identify them. This requirement will be spelled out in detail below. A core concern implied by the existence of names is one of name management. A common way to implement name management is to structure the names in such a way as to establish name-spaces - parts of the name that can be allocated, delegated and used to stablish global uniqueness.

There are examples of attribute-type definitions that are in common use today that employ a variety of name spaces including both OIDs, http-based URIs and URNs.

Another aspect of naming is name agility. Depending on the protocol use to represent the name it is sometimes necessary to have to create an alias for a name within another namespace. Name agility has implication for the structure and contents of an attribute registry.

Attribute names sometime need human-readable (aka "friendly") labels. This leads to questions of internationalization and possibly security considerations in analogy to how IDNs can result in new attack-vectors when used in URIs.

2.2. Use

The core usage-question is this: will the attribute registry be used in conjunction with individual transactions or as a tool for configuration, discovery and information related to the task of setting up federations and other relationships using claims-based protocols. The former use-case requires a global service available 24x7 while the latter requires the availability typically found in a website providing documentation.

This document is skewed towards the former use-case. The authors believe that the operational issues involved in the latter type of registry would be daunting to say the least; it is only presented here for completeness.

2.3. Data Locality

There are two fundamental models for registries (as for any data store): centralized and distributed. In a central registry all the information is kept and maintained in one place, whereas a distributed registry shares information in the registry over multiple cooperative instances that together make up the full registry. It is possible to concieve of hybrid models where for instance a central index is used to store referrals to a set of distributed nodes.

The distributed model is most often used when the expected use of the registry would imply a very high load on a single registry instance. An example of a system with this property is the DNS. A distributed registry model has implications for requirements on lookup (cf below). Specifically the registry may need a central or well-known entry-point unless there is a mechanism for performing lookups.

The central model by contrast is simpler in that no protocol needs to be specified for communicating between registry instances and that lookup can be handled to a single well-know instance. This model may be preferred if the total amount of data in the registry is relatively small (at least compared to the DNS or systems of similar scale). The fact that the registry is operated in a single instance does not necessarily imply lowered requirements on availability and security. An example of this type of registry is the Time Zone Database [RFC6557].

One possible basis for a distributed registry is the Dynamic Delegation Discovery System (DDDS) as described in [RFC3401], [RFC3402], [RFC3403]and [RFC3404].

2.4. Schema

As was stated in the introduction, an LDAP and X.509 attribute schema is commonly used to describe attribute-types for claims-based protocols. Recently however there is a trend towards defining "raw attributes", i.e, attribute-types that are not supported by a corresponding directory schema. Thus there may be a need to define a "directory-neutral" attribute-type schema langue. In either case there will probably be a need to support multiple schema in the registry.

Note that LDAP and X.509 schema have a property that is not currently used in claims-based protocols: objectClass definitions. These are schema elements that often list a set of mandatory and/or optional associated attributes.

Depending on the intended use of the registry, a native attribute schema may need to exist for the registry that may or may not need to represent the complete set of properties of each attribute type. For instance, if the intended use of the registry is to support configuration and setup of federation, rather than in-transaction discovery of attribute properties, the registry native schema may not have to include all information of each attribute. Instead it would be possible to maintain a minimal set of core properties in the registry and provide references to external information sources that could be chaised for additional information.

2.5. Lookup and Search

Lookup and Search may appear to be very similar operations but they are in fact quite dissimilar in that they place very different requirements on the representation and schema of the data to be searched. To draw an example from the DNS again: The DNS supports lookup but not search. In other words it is possible to, given a domain name, lookup the corresponding records in the DNS but it is not in generally possible to search for records given knowledge of only part of a domain name.

3. Requrements

The following terminology is used in this section:

registry An instance of an attribute registry fulfilling these requirements.

consumer A user, device, process, or other entity that consumes information from the registry.

attribute type An element of the registry.

attribute name Synonymous with attribute-type name

3.1. Use

- A consumer MUST NOT directly use the registry for in-transaction lookup.

The registry is primarily intended for use as a tool to help discover attribute-type information related to setup and configuration of federations. While services that directly tie in to authentication events (for instance, in order to provide for the internationalization of attribute-friendly names) may be needed, such services can always be developed as commercial spin-offs from the basic registry.

3.2. Data Locality

- The registry SHOULD be established as a central, non-distributed registry.

Since the primary use of the registry is not for in-transaction lookups, the registry does not need to be distributed. This reduces the complexity of the registry.

3.3. Naming

- The registry MUST support multiple name spaces for naming attribute types.

- The registry MUST support attribute-type name aliases.

- The registry MAY support aliases that are namespace-free short names.

- The registry SHOULD (if such names are supported) impose restrictions on registering short names.

3.4. Schema

- The registry SHOULD support a native attribute type schema.

- The native attribute-type schema MUST map cleanly (in)to X.520/ LDAP schema for attribute types

- The native attribute-type schema MAY only represent a subset of the features of X.520/LDAP schema
- The native attribute-type schema SHOULD support multiple serializations (XML,JSON,etc)

3.5. Lookup and Search

- The registry MUST support lookup based on attribute-type name.
- The registry MUST support lookup based on attribute-type aliases if they are provided.
- The registry MAY support search but registry consumers MUST NOT assume support for search.

4. Acknowledgements

This work was inspired by discussions at the ISOC identity ecosystem workshops held in Amsterdam and Gathersburgh MD in 2011 and 2012.

5. Contributors

Main contributors for this work has been

- Heather Flanagan (ISOC/Internet2)
- James Bryce Clark (OASIS)

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

Attributes are often used to carry sensitive information as part of claims-based protocols. It is common for claims to contain attribute values that are used to allow or deny access to a protected resource. Some attributes carry identifiers as values. A discussion of the security implications of handling identifiers can be found in "Issues in Identifier Comparison for Security Purposes" [RFC6943].

8. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3401] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part One: The Comprehensive DDDS", RFC 3401, October 2002.

[RFC3402] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Two: The Algorithm", RFC 3402, October 2002.

[RFC3403] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database", RFC 3403, October 2002.

[RFC3404] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Four: The Uniform Resource Identifiers (URI)", RFC 3404, October 2002.

[RFC6557] Lear, E. and P. Eggert, "Procedures for Maintaining the Time Zone Database", BCP 175, RFC 6557, February 2012.

[RFC6943] Thaler, D., "Issues in Identifier Comparison for Security Purposes", RFC 6943, May 2013.

Authors' Addresses

Leif Johansson (editor)
NORDUnet

Email: leifj@nordu.net

Heather Flanagan
Spherical Cow Consulting

Email: hlf@sphericalcowconsulting.com