Freshness and Affiliation

1. Background

When organisational affiliation is used for authorisation (or any similar purpose), it is necessary for a RP to know whether they can trust the attribute.

For a community proxy relaying information about the user (see AARC-G056), the affiliation may have been verified in the past by the user registering with their then home organisation, but the user has continued to log in with a linked lower assurance account.

2. Freshness of Authorisation Attributes

Attribute metadata is least poorly developed for identity attributes.

For email address, we have an attribute that says whether it has been verified, but not when it was verified (the implicit assumption is close to the account being created), nor who has verified it (the assumption here is that it is the service that manages the user account).

For other identity attributes, we have RAF which provides a language for communicating assurance about (a) uniqueness of the principal, (b) identity proofing, and (c) freshness, in the limited sense of providing a time window for changing publication of attributes when the user's relevant circumstances change, the choice for the latter being a one day or one month time window.

For the most part people look at the assurance only for identity attributes, e.g. OpenID's acr.

2.1. The problem(s) statement

For the sake of exposition, we assume the attribute source is a community proxy (using AARC-G056).

2.1.1. Problem 1 - "can I trust this assertion"

In a federated environment or in an environment with a community proxy, SPs may get user attributes from several parties channeled through the SP's nearest proxy. There are several difficulties with this approach as the SP does not know when the attributes were asserted or by whom and how recently they have been reasserted.

This is particularly complicated when the user's account has linked multiple IdPs to it, each of which is a source for different attributes.

Authorisation decisions are almost always performed using non-identity attributes, such as organisational affiliation. When the SP takes an authorisation decision, they sometimes need to know who asserted it and when the assertion was made to the community proxy.

2.1.2. Problem 2 - authority

When communicating organisational affiliation in particular, it was noted that the community proxy is not authoritative for organisational affiliation, which led to voPerson attributes such as voPersonExternalAffiliation (AARC-G059). This could be a simplification, as communicating organisational affiliation is already complicated - with attributes like organization and schacHomeOrganization etc., and with short names and long names for organisations. Which leads to

2.1.3. Problem 3 - organisation name

Although not the main focus of this document, organisation is difficult to communicate. There is not only the choice of language, choosing abbreviation vs full name, legal name etc. For example, eduGAIN metadata requires (short) name, display name, and URL, and where relevant an English language version is provided with optionally a native version. If an SP is to match the name as a string against attributes provided, how should they do this?

As an example, STFC's short name is (unsurprisingly) "STFC", but the (legal) organisation name is (since 2018) "UKRI" as short name and "UK Research and Innovation" as long name. For historical and practical reasons, the different parts of UKRI (of which STFC is one) have their own IdPs, so a single IdP cannot assert "UKRI" as it would not be authoritative for all of UKRI. In practice, therefore, the short name "STFC" or "UKRI-STFC" is asserted. Many other organisations have similar complexities, so string matching organisations based on attributes becomes difficult.

2.1.4. Problem 4 - clever people

Being clever people, we naturally want to solve the problem in the cleanest and most reusable way. It is tempting to overdesign a solution which either does not get fully implemented - or at all - or does not get used to its full extent. As an example, the VOMS membership groups and roles and capabilities were rarely used to their full potential.

2.1.5. Problem 5 - hacks

The opposite part of Problem 4 is to hack a short term solution which will solve the narrow problem but not be generalisable. The classic example is email _verified from OpenID Connect which is worth quoting in full:
True if the End-User's e-mail address has been verified; otherwise false. When this Claim Value is true, this means that the OP took affirmative steps to ensure that this e-mail address was controlled by the End-User at the time the verification was performed. The means by which an e-mail address is verified is context specific, and dependent upon the trust framework or contractual agreements within which the parties are operating.

While simple and easy to use, the problems with this approach is that

1. The time the verification is not available, though it could be communicated elsewhere.
2. The OP took "affirmative steps" but no information is available about these steps (though they may be available out of band). For email, it is natural to assume that it tried to mail the user with a code and asked the user to return the code, but a community proxy could also take a verified email address from an organisational IdP.
3. There is no stipulation of re-verification. If the user's access to the mailbox changes, we only have RAF and then only if it's an organisational mail address. Conversely a community may choose to keep a persistent email address which does not change when the user changes their organisational affiliation.
4. The approach is not orthogonal - it is specific to email and does not generalise to other attributes.
5. The approach does not allow for multiple email addresses. If the user had multiple email addresses (see AARC-G056), the "verified" assertion can be made only for one.

### 2.1.6. Problem 6 - Inertia

A final problem is that national educational and research federations are notoriously slow to make any changes to their attributes. Therefore, any change will need to be implemented in the community proxy or at the federation level (using SAML assertions or OIDC-Fed trust marks or similar). Even then, changes take a long time to get implemented in proxy implementations and even more to roll out into production environments. Obviously this is not a problem with freshness in particular but applies to all changes required by communities or required for interoperation. Having communities with requirements for the change can help.

In other words, the stakeholders for this proposal are the communities that rely on the attribute (and their AAI/proxy operators), the relying parties, and the infrastructure operators, but not the IdPs or federation operators.

### 3. Prior Work

Recognising the problem in AARC1, we conducted a brief study into the existing proposed solutions to the problem. The results were not published as a deliverable (it was not planned as a deliverable), but the results were as follows:

1. The most mature work in the field was NIST IR-8112
2. Also Internet2 had looked into the problem
3. (And there was a third group - who?)

At the same time, ELIXIR looked into the matter, proposing to

- verify using a login from the user’s home organisation (this would be done regularly by the community proxy itself), communicating a freshAffiliation attribute (in addition to voPersonExternalAffiliation) to assert which affiliations are used in the immediate login; or,
- providing a hash table (a “dict” in pythonese) mapping the organisation (short) domain name to a timestamp of the most recent verification.