XEP-0489: Reporting Account Affiliations

Matthew Wild
mailto:mwild1@gmail.com

2024-03-11
Version 0.1.0

Status	Type	Short Name
Experimental	Standards Track	raa

This specification documents a way for an XMPP server to report to other entities the relationship it has with a user on its domain.

Legal

Copyright

Permissions

Warranty

Liability

Conformance

Contents

1 Introduction

This specification documents a way for an XMPP server to report to other entities the relationship it has with a user on its domain.
In practice, a server may not trust all accounts equally. For example, if a server offers anonymous access or open registration, it may have very little trust in such users. Meanwhile a user account that was provisioned by a server administrator for an employee or a family member would naturally have a higher level of trust.
Even if a server alters its own behaviour based on how much it trusts a user account (such as preventing anonymous users from performing certain actions), other entities on the network have no way of knowing what trust to place in JIDs they have not encountered before - they can only judge the server as a whole.
This lack of insight can result in the negative actions (spam, abuse, etc.) by untrusted users on a domain causing the whole domain to be sanctioned by other servers.

2 Requirements

This protocol will allow:

- Servers to communicate the affiliation of the end user that sent a stanza
- Remote servers to query a user's affiliation on demand

By providing this high-level information to other entities on the network, it allows them to make informed decisions about how to handle traffic at the account level rather than the server level.
For example, during a spam wave, a public MUC service may choose not to grant the 'participant' role by default to accounts that were very recently registered.
It aims to achieve this while preserving the privacy of the user themselves, and ensuring the user's server has full discretion over what data to provide and to which entities it is provided.

2.1 Related work

This specification has similar goals to that of XEP-0275: Entity Reputation. It differs in the following ways:

- Rather than attempting to define a semi-standardized scale, it reports qualitative actionable account status information. This makes implementation simpler, and servers don't have to guess at appropriate thresholds on a universal quantitative scale.
- This specification can be extended in the future if necessary. The scoring algorithm in XEP-0275 is coded into the document, and changing the value assignments may impact existing deployments that have defined thresholds based on the current specification.

- The 'trust' level in this specification is superficially similar to the Entity Reputation score, however with notable differences: trust levels are calculated only by a server for its own users, and we make no attempt to standardize an algorithm.

Some of this information may also be discovered through XEP-0030: Service Discovery. This specification provides more detailed information than is currently exposed via service discovery, and it is also push-based, removing the need for recipients to separately query an account's status while determining how to handle a stanza.

3 Affiliations

An affiliation is what we call the relationship that a user has with a service. Different affiliations imply different levels of trust. The affiliations defined in this specification are as follows:

- **anonymous:** the address belongs to an anonymous, temporary or guest account. The user is not known to the server.
- **registered:** the address belongs to an account that self-registered, e.g. using XEP-0077
- **member:** the address belongs to a trusted member of the server - e.g. accounts that are provisioned for known users.
- **admin:** the address belongs to a server administrator

It should be noted that these affiliations extend the account types defined in the Service Discovery Identities registry. Notably, this document adds an additional affiliation type: 'member'.

4 Protocol

4.1 Info element

An affiliation element looks like this:

Listing 1: The affiliation info element

```
<info xmlns='urn:xmpp:raa:0'
      affiliation='member'
      since='2023-06-27T00:00:00Z'
      trust='57' />
```

The 'affiliation' attribute MUST be present, and the value MUST be one of the affiliations listed in the previous section. The 'since' attribute is optional, and contains a XEP-0082 DateTime

profile string representing the date and time at which the account was approximately created. Please see the security considerations for advice on preserving privacy before exposing this information.

4.2 Query

An entity may directly query for affiliation information about a JID.

Listing 2: Service at example.com sends a query for information about a JID

```
<iq type="get" to="mr.nobody@service.example" from="example.com" id="
    123">
     <query xmlns='urn:xmpp:raa:0'/>
</iq>
```

The server SHOULD respond successfully with as much information as it permits the requesting entity to see:

Listing 3: Service responds with affiliation information

```
<iq type="result" to="example.com" from="mr.nobody@service.example" id
    ="123">
     <info xmlns='urn:xmpp:raa:0' affiliation='registered' since='
         2023-06-27T00:00:00Z' trust='47'/>
</iq>
```

Alternatively, the server MAY respond with a 'forbidden' error if it does not permit the requesting entity to view any information about account affiliations:

Listing 4: Service forbids entity from querying this information

```
<iq type="error" to="example.com" from="mr.nobody@service.example" id=
    "123">
     <error type='auth'>
         <forbidden xmlns='urn:ietf:params:xml:ns:xmpp-stanzas'/>
     </error>
</iq>
```

4.3 Embedding

Alternatively the <info/> element can be embedded into outgoing stanzas, such as presence or messages:

Listing 5: Embedding affiliation information

```
<presence to="user@example.com"
          from="mr.nobody@service.example"
          type="subscribe">
    <info xmlns='urn:xmpp:raa:0'
          affiliation='registered'
          since='2023-06-27T00:00:00Z' />
</presence>
```

The above stanza demonstrates a subscription request sent by a recently-registered user to a user on another server. When embedding, a server MUST announce which stanzas it supports embedding in. In such stanzas it MUST strip any existing <info/> elements that may have been sent by the client.
Receiving servers MUST only trust embedded <info/> elements when the origin server has announced support for them.

5 Business rules

A server may exercise discretion over when it includes affiliation information. For example, it MAY choose to only reveal this information when sending stanzas to trusted servers, or withold it when sending stanzas to untrusted servers (the definition of trusted servers is beyond this specification - it may be implementation-specific or based on a protocol such as XEP-0267: Server Buddies).
Similarly, the information does not need to be included for every type of stanza. For example, a server SHOULD only include this information for stanzas that are sent to non-contacts. For example, messages and presence to JIDs that have not granted a presence subscription to the sender yet (i.e. absent from the sender's roster, or with a subscription state of 'none' or 'from').
A server MAY also withold or reduce the information for certain affiliations - e.g. by reporting server administrators as simply 'member' if the server does not want to expose this information to the recipient.
The inclusion of the 'since' attribute is optional, but it SHOULD be included for accounts with the affiliation 'registered' accounts that were created within the past 30 days. It MAY be approximate (e.g. rounded to the nearest day) for performance or privacy reasons (the latter is discussed in the security considerations section).
The inclusion of the 'trust' attribute is optional, but it SHOULD be included for 'registered' accounts if 'since' is not included. That is, at least one of 'since' or 'trust' SHOULD be present for accounts with affiliation 'registered' to ensure a recipient has sufficient information to act on. The value of the 'trust' attribute MUST be an integer from 0 to 100 (inclusive). The value may be calculated by the server using any algorithm it deems appropriate. However, the same algorithm MUST be used for all users of the same affiliation, so that comparison between them is meaningful.

6 Discovery

If a server supports this protocol and the query functionality, it MUST advertise the 'urn:xmpp:raa:0' feature in response to service discovery queries on its domain JID.
If the server also supports embedding affiliation into stanzas, it MUST also advertise the appropriate features from this list:

urn:xmpp:raa:0#embed-presence-sub <info/> may be embedded in presence subscription requests originating from the user's bare JID.

urn:xmpp:raa:0#embed-presence-directed <info/> may be embedded in directed presence (including e.g. XEP-0045 join requests) from the user's full JID.

urn:xmpp:raa:0#embed-message <info/> may be embedded in message stanzas.

7 Security Considerations

This specification describes a protocol that can be used to help enhance the security of the XMPP network. However, some considerations do apply.

7.1 Account age privacy

If a server chooses to expose an account's creation timestamp to untrusted entities, the reported value SHOULD be approximate - e.g. rounded to the day on which the account registered - to preserve privacy. Providing a value with a high precision may allow entities to correlate the account registration with other actions performed by the user, or determine a user's likely time zone.
In particular, and in accordance with the security considerations of XEP-0082, the 'since' value MUST be reported in UTC.

7.2 Spoofing

The payloads described in this specification may be embedded by the server in stanzas originating from a user's JID. This makes it trivial for clients to send spoofed <info/> elements if the server does not remove them. To protect against such spoofing:

- Origin servers MUST advertise the correct features according to the stanza types they embed the <info/> element within.
- Origin servers MUST strip client-originating <info/> elements from any stanza types they have advertised support for.

- Receiving servers MUST ignore <info/> elements embedded within stanzas from other servers unless that server advertises support for embedding within that specific stanza type.

8 IANA Considerations

None.

9 XMPP Registrar Considerations

This document extends the 'Identity Categories and Types Registry' defined by XEP-0030 with the following addition to the 'account' category:

```
<type>
  <name>member</name>
  <desc>The user@host is a trusted member of the server - e.g. an
     account that was provisioned for known user</desc>
  <doc>XEP-xxxx</doc>
</type>
```