XEP-0499: Pubsub Extended Discovery

Jérôme Poisson
mailto:goffi@goffi.org
xmpp:goffi@jabber.fr
https://www.goffi.org

2024-11-20
Version 0.1.0

Status	Type	Short Name
Experimental	Standards Track	pubsub-ext-disco

This specification extends the discovery requests used with the XMPP PubSub protocol by introducing mechanisms to discover linked nodes, descendants, or metadata.

Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 – 2024 by the XMPP Standards Foundation (XSF).

Permissions

Permission is hereby granted, free of charge, to any person obtaining a copy of this specification (the ”Specification”), to make use of the Specification without restriction, including without limitation the rights to implement the Specification in a software program, deploy the Specification in a network service, and copy, modify, merge, publish, translate, distribute, sublicense, or sell copies of the Specification, and to permit persons to whom the Specification is furnished to do so, subject to the condition that the foregoing copyright notice and this permission notice shall be included in all copies or substantial portions of the Specification. Unless separate permission is granted, modified works that are redistributed shall not contain misleading information regarding the authors, title, number, or publisher of the Specification, and shall not claim endorsement of the modified works by the authors, any organization or project to which the authors belong, or the XMPP Standards Foundation.

Warranty

NOTE WELL: This Specification is provided on an ”AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE.

Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall the XMPP Standards Foundation or any author of this Specification be liable for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising from, out of, or in connection with the Specification or the implementation, deployment, or other use of the Specification (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if the XMPP Standards Foundation or such author has been advised of the possibility of such damages.

Conformance

This XMPP Extension Protocol has been contributed in full conformance with the XSF’s Intellectual Property Rights Policy (a copy of which can be found at <https://xmpp.org/about/xsf/ipr-policy> or obtained by writing to XMPP Standards Foundation, P.O. Box 787, Parker, CO 80134 USA).

Contents

1 Introduction **1**

2 Requirements **1**

3 Use Cases **1**

3.1 Discovering Linked Nodes and Descendants 1

4 Business Rules **3**

5 Discovering Support **4**

6 Security Considerations **4**

7 IANA Considerations **5**

8 XMPP Registrar Considerations **5**

9 Acknowledgements **5**

1 Introduction

With the introduction of XEP-xxxx: Pubsub Node Relationships, it becomes necessary to discover child or linked nodes, enabling an entity to discover a tree-like structure and other related nodes. This specification extends the node discovery functionality in Publish-Subscribe (XEP-0060) [1] by allowing entities to discover linked nodes and descendants when performing a disco#items request. It also allows filtering to specify what the requestor is interested in, and adds metadata to identify the relationships between discovered nodes.

2 Requirements

The design goals of this XEP are:

- To ensure backward compatibility with existing implementations.
- To provide a mechanism for discovering linked nodes and descendants during a disco#items request.
- To allow control over the depth of descendant discovery.
- To provide metadata about the relationships between nodes in the discovery results.

3 Use Cases

3.1 Discovering Linked Nodes and Descendants

An entity can request to discover linked nodes and descendants by including a data form in the disco#items request. The form uses the namespace ”urn:xmpp:pubsub-ext-disco:0” and MAY include the following fields:

- **type**: A ”list-multi” field that indicates what kind of items should be returned. The values for this field MUST be either ”items” or ”nodes”, and it defaults both ”items” and ”nodes”. If ”items” is selected, pubusb node items are returned; if ”nodes” is selected, pubsub nodes are returned. Pubsub items MUST NOT be returned if ”items” is not selected, and pubsub nodes MUST NOT be returned if ”nodes” is not selected.
- **linked_nodes**: A boolean field that, when set to ”true”, indicates that linked nodes and/or their items should be included in the discovery results. The default value of this field MUST be ”false”.

[1] XEP-0060: Publish-Subscribe <https://xmpp.org/extensions/xep-0060.html>.

- **full_metadata**: A boolean field that, when set to 'true', indicates that the full node metadata form (as specified in XEP-0060 §5.4 Discover Node Metadata) MUST be included with each discovered node. If this field is set to 'false', the Pubsub/PEP service MUST return only the necessary relationship fields ('{urn:xmpp:pubsub-relationships:0}parent' and/or '{urn:xmpp:pubsub-relationships:0}link'). The default value of this field MUST be 'false'.
- **depth**: A "text-single" field that specifies the number of layers of descendants to include. The value must be castable to an integer and defaults to 0. If the "linked_nodes" field is set to "false", descendant linked nodes (i.e., descendant nodes with a set "{urn:xmpp:pubsub-relationships:0}link" field) MUST NOT be included.

If the "urn:xmpp:pubsub-ext-disco:0" data form is present in the disco#items request, the rules differ from those specified in Publish-Subscribe (XEP-0060) [2]: contrary to what is outlined in §5.5 Discover Items for a Node, the "node" attribute MUST be specified for all items as it is necessary to know to which pubsub node a pubsub item is attached.
If a disco item is actually a pubsub node, a metadata form (as specified in XEP-0060 §5.4 Discover Node Metadata) MUST be included as a child of the disco <item> element.
If the 'full_metadata' field in the request's data form is set to "true", the full node metadata form MUST be included with each discovered node. If this field is set to "false", only the necessary relationship fields ("{urn:xmpp:pubsub-relationships:0}parent" and/or "{urn:xmpp:pubsub-relationships:0}link") MUST be returned.
If the item is a child node, its "{urn:xmpp:pubsub-relationships:0}parent" field MUST be present, and if it is a linked node, its "{urn:xmpp:pubsub-relationships:0}link" field MUST be present. This data form is necessary to identify the item as a pubsub node and to see its relationships, allowing the requestor to build the hierarchy.

Listing 1: Entity Requests Discovery of Linked Nodes and Descendants

```
<iq type='get'
  from='romeo@example.net/orchard'
  to='juliet@example.org'
  id='disco1'>
<query xmlns='http://jabber.org/protocol/disco#items' node='
    urn:xmpp:microblog:0'>
  <x xmlns='jabber:x:data' type='submit'>
    <field var='FORM_TYPE' type='hidden'>
      <value>urn:xmpp:pubsub-ext-disco:0</value>
    </field>
    <field var='type'>
      <value>nodes</value>
    </field>
    <field var='linked_nodes'>
      <value>true</value>
    </field>
```

[2] XEP-0060: Publish-Subscribe <https://xmpp.org/extensions/xep-0060.html>.

```
    <field var='depth'>
      <value>1</value>
    </field>
  </x>
</query>
</iq>
```

Listing 2: PEP Service Returns Discovery Results With Linked Nodes and First Level Descendants

```
<iq type='result'
  from='juliet@example.org'
  to='romeo@example.net/orchard'
  id='disco1'>
<query xmlns='http://jabber.org/protocol/disco#items' node='
    urn:xmpp:microblog:0'>
  <item jid='juliet@example.org' node='urn:xmpp:microblog:0:comments/
      balcony-restoration-afd1'>
    <x xmlns='jabber:x:data' type='result'>
      <field var='FORM_TYPE' type='hidden'>
        <value>http://jabber.org/protocol/pubsub#meta-data</value>
      </field>
      <field var='{urn:xmpp:pubsub-relationships:0}parent'>
        <value>urn:xmpp:microblog:0</value>
      </field>
    </x>
  </item>
  <item jid='juliet@example.org' node='urn:xmpp:pubsub-attachments:1/
      xmpp:juliet@capulet.lit?;node=urn%3Axmpp%3Amicroblog%3A0;item=
      balcony-restoration-afd1'>
    <x xmlns='jabber:x:data' type='result'>
      <field var='FORM_TYPE' type='hidden'>
        <value>http://jabber.org/protocol/pubsub#meta-data</value>
      </field>
      <field var='{urn:xmpp:pubsub-relationships:0}link'>
        <value>urn:xmpp:microblog:0</value>
      </field>
    </x>
  </item>
</query>
</iq>
```

4 Business Rules

- The result described in this specification MUST NOT be used if the ”urn:xmpp:pubsub-ext-disco:0” is missing from the disco#items requests. This is to ensure backward compatibility.

- The service MUST respect the access model of each node when returning discovery results. If an entity does not have permission to access a node, it MUST NOT be included in the results.
- As the result could include a lot of disco <items>, Result Set Management (XEP-0059) [3] SHOULD be used by the PEP/Pubsub service to handle a large number of discovered items.

5 Discovering Support

If a pubsub/PEP service supports the protocol specified in this XEP, it MUST advertise it by including the "urn:xmpp:pubsub-ext-disco:0" discovery feature in response to a Service Discovery (XEP-0030) [4] information request.

Listing 3: Service Discovery Information Request

```
<iq type='get'
    from='juliet@example.org/balcony'
    to='pubsub.example.org'
    id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#info'/>
</iq>
```

Listing 4: Service Discovery Information Response

```
<iq type='result'
    from='pubsub.example.org'
    to='juliet@example.org/balcony'
    id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
    ...
    <feature var='urn:xmpp:pubsub-ext-disco'/>
    ...
  </query>
</iq>
```

6 Security Considerations

This extension does not introduce any new security considerations beyond those already present in Publish-Subscribe (XEP-0060) [5]. However, implementers should be aware that including linked nodes and descendants in discovery results may expose more information about the node structure than a basic disco#items request. Services MUST ensure that they

[3] XEP-0059: Result Set Management <https://xmpp.org/extensions/xep-0059.html>.
[4] XEP-0030: Service Discovery <https://xmpp.org/extensions/xep-0030.html>.
[5] XEP-0060: Publish-Subscribe <https://xmpp.org/extensions/xep-0060.html>.

respect the access controls of all nodes when returning discovery results.

7 IANA Considerations

This document does not require interaction with the Internet Assigned Numbers Authority (IANA) [6].

8 XMPP Registrar Considerations

TODO

9 Acknowledgements

Thanks to NLNet foundation/NGI Zero Core for funding the work on this specification.

[6] The Internet Assigned Numbers Authority (IANA) is the central coordinator for the assignment of unique parameter values for Internet protocols, such as port numbers and URI schemes. For further information, see <http://www.iana.org/>.