SIMPLE
Internet-Draft
Expires: January 6, 2006

J. Ala-Kurikka
M. Ylianttila
E. Harjula
Univ. of Oulu
July 5, 2005

Mapping non-URIs to contact element in Presence Information Data Format

draft-ala-kurikka-simple-map2pidf-00

Status of this Memo

This Internet-Draft will expire on January 6, 2006.

Copyright Notice

Abstract

The Presence Information Data Format (PIDF) defines a basic XML format for presence information. The contact element in PIDF is understood as a URI. However, some existing Internet services do not identify users (presentities) natively with a URI. Mapping non-URIs to contact element in Presence Information Data Format (Map2PIDF) specifies the mapping of addresses to a URI in the case of such currently existing services.

1. Introduction

The Presence Information Data Format (PIDF) [1] defines a basic XML format for presenting presence information of a presentity. Presence tuples [2] have an optional <contact> element. The contents of the <contact> element are understood to refer to a URI [3].

There are, however, services and applications in the Internet that do not natively use URIs to identify users, and/or the mapping of users to URIs is more or less cumbersome. Examples include Instant Messaging, VoIP and peer-to-peer for file sharing. While PIDF seems flexible enough to handle also such communication addresses, it would be helpful if users' capability of communicating with such services that do not share a common naming convention could also be announced with PIDF. This document specifies how to map user names from a such services to a presentity's <contact> element.

One goal of this specification is to be backwards compatible with PIDF. This would ensure that existing non-Map2PIDF watchers and gateways will continue to work properly, without access to the functionality specified here. Old User Agents (later UAs) would not know how to act with URIs with these new schemes. However, if the contents of <contact> elements were to be shown in plain text form to a user acting as a watcher, he/she could possibly understand the form and follow the link manually (e.g. by opening a file sharing client program and connecting to a hub that the presentity is connected to). At this phase, no action has been taken to filter URIs with formerly undefined schemes defined in this document from presence information for watchers or gateways that do not conform to this specification. It is assumed that URIs with these new schemes in the presence information do not hinder UAs or gateways from reading the tuples that have well-known schemes according to "Handling of Unrecognized Element Names" in PIDF.

1.1 Motivation

The PIDF specification does not rule out the set of supported communication means. The only requirement set is that the communication address has to be expressed with a URI. However, some application level protocols - of which some are proprietary - use naming conventions other than URIs to define individual presentities. While these are normally ruled out so that their addresses would never be added to a presentity tuple, this is an excellent opportunity for adding connectivity awareness for PIDF. Providing listing of the ways that a presentity can communicate with would be a major advantage for SIP [4] leveraging SIP's status as a necessity for all interconnected IP devices, being the unifying factor.

Of course, details on supported application level protocols could also be queried from remote UAs after establishing a session with them. Having that information without a session initiation with the remote UA could, however, prove an important feature. Queries such as "who of my friends are using Skype" would probably be more efficient with a centrally managed presence information service than with having to establish sessions to all friends first.

Mapping the "non-URIs" to a URI without a standard way could also be used. But this would limit the usefulness of the presence tuple to what the user can do, and even this only if the mapped URI would be shown to the user in plain text form. There are many useful scenarios where an automata would benefit from being able to understand all the different ways of interaction (i.e. communication addresses) of a presentity, say, a user's friend. The UA could then provide e.g. a list of different possible operations with the friend to the user. Or it could suggest using less costly Skype URI to establish a telephone call instead of using a more expensive tel URI, if the friend had one in one of his presence tuples.

Such feature would probably prove especially useful in mobile devices because of their limited resources, including processing capabilities, memory and screen. Also user friendliness could be substantially better with one UI to control actions in several protocols instead of multiple separate applications, one for each protocol. The need for such multiprotocol communication will be even bigger in the future, as there seems to be no consensus on which protocol to use for e.g. file sharing. This leads often to loss of interoperability. A certain type of interoperability would be made possible, though, with the proposed specification.

1.2 Terminology and Conventions

This memo makes use of the vocabulary defined in the IMPP Model document [2]. Terms such as COMMUNICATION ADDRESS, COMMUNICATION MEANS, CONTACT ADDRESS, and PRESENTITY in the memo are used in the same meaning as defined therein.

The key words MUST, MUST NOT, REQUIRED, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL in this document are to be interpreted as described in BCP 14, RFC 2119 [5].

2. Mapping

Every protocol being mapped to a URI MUST have a well-defined, individual URI scheme. Schemes do not exist for the protocols specified in this document. Schemes for non-existing protocols MUST be formulated through the following process (conforms with [5]):

The official name of the protocol is converted to lowercase, removed of any special characters (including white space) other than digits, plus ("+"), period (".") or hyphen ("-").

If the name's first character is a digit, replace the digit with the first character of the written English form in lowercase (e.g. 1->one->o)

3. Example

Using a default XML namespace:

```
<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf"
                entity="pres:Joe@example.com">
        <tuple id="sg89ae">
                <status>
                        <basic>open</basic>
                </status>
                <contact priority="0.8">skype:joe</contact>
        </tuple>
</presence>
```

This would indicate to a WATCHER that the PRESENTITY Joe@example.com is registered to Skype with a user name of "joe" and that his Skype client is online, so Joe is able to receive VoIP calls.

4. Implementing Map2PIDF

Our specification would benefit from some way of exchanging presence information between SIP UAs [4] providing PRESENCE INFORMATION and other applications (possibly UAs) communicating with other protocols. One way of doing this would be to enable the SIP UA itself to be able to communicate with various other protocols as well as SIP. Another possibility would be to have interfaces between applications (or UAs) through which presence information for different protocols could be exchanged.

If no such intercommunication of applications can be achieved, presence information provided to other protocols might be harder to obtain by the SIP UA. This would, on the other hand, restrict functionality. The SIP UA would not know if the device was online with other protocols, therefore suggestions of using other protocols might lead to useless attempts. Operating system level application search and start capabilities could still be used to e.g. invoke an external application to establish a VoIP session.

5. Security Considerations

Exposing PIDF information to others requires security mechanisms itself. However, they are already specified in the PIDF specification. Map2PIDF reveals some more information, such as 3rd party applications installed and/or running on a PRESENTITY's device. Care has to be taken, and the developer should pay special attention to all notes in the PIDF security considerations. The 3rd party applications might have security vulnerabilities themselves, however, addressing them is out of scope of this document.

If a password to a certain server (or similar) was to be included in a PRESENTITY's PRESENCE INFORMATION to enable certain (perhaps selected) set of WATCHERs to gain access to the server, S/MIME [6] MUST be used for staging security at least for that PRESENCE TUPLE. Otherwise the password is exposed almost certainly allowing eavesdroppers to also gain access to the server using the WATCHER's privileges.

6. References

[1] Sugano, H., Fujimoto, S., Klyne, G., Bateman, A., Carr, W., and J. Peterson, "Presence Information Data Format (PIDF)", RFC 3863, August 2004.

[2] Day, M., Rosenberg, J., and H. Sugano, "A Model for Presence and Instant Messaging", RFC 2778, February 2000.

[3] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998.

[4] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

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

[6] Ramsdell, B., "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Message Specification", RFC 3851, July 2004.

Authors' Addresses

Jussi Ala-Kurikka
MediaTeam / University of Oulu
Department of Electrical and Information Engineering
Information Processing Laboratory
P.O.BOX 4500
University of Oulu 90014
FI

Phone: +358 44 304 0098
Email: jussi.ala-kurikka@oulu.fi
URI: http://www.mediateam.oulu.fi

Mika Ylianttila
MediaTeam / University of Oulu
Department of Electrical and Information Engineering
Information Processing Laboratory
P.O.BOX 4500
University of Oulu 90014
FI

Phone: +358 40 535 0505
Email: mika.ylianttila@oulu.fi
URI: http://www.ee.oulu.fi/~over

Erkki Harjula
MediaTeam / University of Oulu
Department of Electrical and Information Engineering
Information Processing Laboratory
P.O.BOX 4500
University of Oulu 90014
FI

Phone: +358 50 303 0478
Email: erkki.harjula@oulu.fi
URI: http://www.mediateam.oulu.fi

Appendix A. Acknowledgments

The authors would like to thank the Application Supernetworking (All-IP) project partners: the National Technology Agency (TEKES), Nokia, TeliaSonera Finland, Elektrobit Group, IBM and Serv-It for their financial support and Henning Schulzrinne for his comments during the early phases of sketching the idea.

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.