

SIPPING Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 24, 2008

S. Channabasappa
CableLabs
J. Littlefield
Cisco Systems, Inc.
S. Loreto
Ericsson, Inc.
February 21, 2008

Extension to the ua-profile Event Package to Support the Application
Profile Type
[draft-channabasappa-sipping-app-profile-type-02](http://www.ietf.org/internet-drafts/draft-channabasappa-sipping-app-profile-type-02)

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on August 24, 2008.

Copyright Notice

Copyright (C) The IETF Trust (2008).

Abstract

The Framework for Session Initiation Protocol User Agent Profile Delivery specifies an event package (ua-profile) that can be used by user agents to retrieve profile data. The framework also allows for optional notification of changes to the retrieved profiles. Three

profile types are specified: local-network, device, and user. This document extends that event package to support an additional profile type, application. This would enable User Agents to retrieve profile data specific to one or more applications.

Table of Contents

1. Introduction	3
2. Terminology	3
3. Motivation	3
4. Overview	4
5. Profile Type Definition	4
5.1. Parameter 'appids'	4
5.2. Summary of Event Header	5
5.3. SUBSCRIBE Bodies	6
5.4. NOTIFY Bodies	7
6. Example Usage	7
7. IANA Considerations	8
8. Security Considerations	8
9. Acknowledgements	8
10. Normative References	8
Authors' Addresses	9
Intellectual Property and Copyright Statements	11

1. Introduction

SIP User Agents require profile data to function properly. A mechanism to obtain profile data is specified by the Framework for SIP User Agent Profile Delivery [I-D.ietf-sipping-config-framework]. The framework separates profile data into three categories, termed profile types, local-network, device and user. Each profile type deals with a specific data set, e.g., the device profile type is used to obtain device-specific configuration. The framework also allows for future extensions to support additional profile types. This document specifies one such extension to support an additional profile type, application. This can be used by user agents for requesting profile data for one or more applications that they support.

2. Terminology

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].

This document also reuses the SIP terminology defined in [RFC3261] and [RFC3265], and specifies the usage of the following terms.

3. Motivation

The motivation for an independent application profile type can be demonstrated using the scenario described in Figure 1. The scenario considers a device (not shown) that supports three applications (X, Y, Z). It also considers two users (A, B). Applications X and Y are user-specific, i.e., restricted to known end-users, whereas Application Z can be used by any user (e.g., Weather Information).

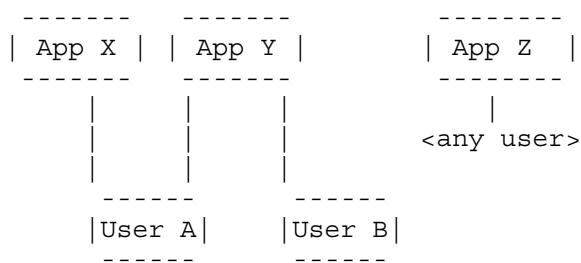


Figure 1: Motivation for application profile type

Each application needs specific profile data to function. For instance, an application such as Video on Demand (VoD) would require VoD server information, codecs for rendering, minimum bandwidth requirements etc. It may also have requirements specific to users, such as rating and cost restrictions (parental controls). Further, the presence of an application does not always mean that it is enabled. For example, a Service Provider may disable VoD for certain subscription levels.

Profile data related to such applications, especially those that are unrelated to specific users, would need to be retrieved for successful operation. This profile data may be retrieved during device boot-up if it is configured to do so, e.g., via the device profile. The profile data can also be retrieved dynamically, e.g., when the application is enabled. Such profile data does not qualify under any existing profile types specified by the SIP UA configuration framework, viz., local-network, device and user. The only exception is application profile data that is specific to users, which can be provided via the user profile type. Thus the need for an additional profile type specific to applications.

4. Overview

5. Profile Type Definition

This document specifies a new profile type for use with the SIP UA configuration framework. The name of the profile type is 'application'. This document also defines an additional event header parameter for use with the application profile type. This parameter is termed "appids".

5.1. Parameter 'appids'

The "appids" parameter describes the application profiles being requested. Its value is an identifier for the application, or a comma-separated list of such identifiers. Each application identifier is a unique value defined by the application specification, along with the profile content, and is in the form of a URI [RFC4395], preferably a URN [RFC3406]. This parameter value SHOULD be provided in the SUBSCRIBE request for the 'application' profile-type only, along with the other three parameters (vendor, model and version) specified in [I-D.ietf-sipping-config-framework]. This parameter is useful to the PDS to affect the profile provided. Behavior when the "appids" parameter is omitted is currently

undefined and treated as an error. Future standards action may specify this behavior.

In the following ABNF defining the syntax, EQUAL and DQUOTE are defined in [RFC3261]:

```
appids          = "appids=" list-of-app-ids
list-of-app-ids = DQUOTE app-id *(," app-id) DQUOTE
app-id         = 1*(subset-print-chars)
subset-print-chars= %x21 /%x23-25 / %x27-29 / %x2D-3C / %x3E-7E
                  ;All printable characters except ", =, &, *, +
                  ;comma or white-space characters.
```

The "appids" parameter appears in the Event header of the NOTIFY request to specify the actual application the NOTIFY belongs to. In the initial NOTIFY following a SUBSCRIBE, the appids parameter SHOULD list all applications obtained in the subscription, which may be a subset of the applications listed in the SUBSCRIBE. The only case in which the "appids" parameter MAY be omitted from the initial NOTIFY is when only one application was listed in the SUBSCRIBE. If the SUBSCRIBE included an "appids" parameter, the "appids" parameter of the initial NOTIFY MUST NOT list applications not present in the SUBSCRIBE. If the parameter contains a list of applications, the order in the appids parameter MUST be the same as followed in the body (see below). Subsequent NOTIFY requests on a single application subscription MAY omit the "appids", since the application context is implied by the subscription dialog.

5.2. Summary of Event Header

The following are example Event headers which may occur in SUBSCRIBE requests. The examples are not intended to show complete SUBSCRIBE requests.

```
Event: ua-profile;profile-type=application;
       vendor="vendor.example.com";model="Z100";version="1.2.3"

Event: ua-profile;profile-type=application;
       vendor="vendor.example.com";model="Z100";version="1.2.3";
       appids="myapplication"

Event: ua-profile;profile-type=application;
       vendor="vendor.example.com";model="Z100";version="1.2.3";
       appids="myapplication1,myapplication2,myapplication3"
```

The following are example Event headers which may occur in NOTIFY requests. These example headers are not intended to be complete NOTIFY requests.

Event: ua-profile;profile-type=application

Event: ua-profile;profile-type=application;appids="myapplication1"

Event: ua-profile;profile-type=application;
appids="myapplication2,myapplication3"

The table shows the use of Event header parameters in SUBSCRIBE requests for the application profile type:

profile-type application
=====
appids s
vendor o
model o
version o
effective-by

m - mandatory

s - SHOULD be provided

o - optional

The table shows the use of Event header parameters in NOTIFY requests for the application profile type:

profile-type application
=====
appids s
vendor
model
version
effective-by o

5.3. SUBSCRIBE Bodies

This draft defines an enhancement to the [I-D.ietf-sipping-config-framework] by specifying a use for the SUBSCRIBE request body. If the appids parameter contains a single application identifier, the SUBSCRIBE message body MAY contain a single body part appropriate for the application. If the appids parameter contains a list of applications, the body of the SUBSCRIBE MAY contain a "multipart/mixed" content-type, with appropriate body parts for each of the applications for which the UA is subscribing.

The body parts MUST be in the same order in which they are listed in the "appids" parameter, and if any body parts are present, all applications must have a corresponding part, even if empty.

If present in the SUBSCRIBE request, the body SHALL be used by the application-specific PDS to tailor the NOTIFY responses to the subscribing UA for each of the applications listed. The meaning and form of the SUBSCRIBE body is specified by each application.

COMMENT: An alternative to requiring all applications to have body parts if any do, and to using "empty" parts where a body part is not needed, is to employ Content-Description to name the appid to which the part corresponds.

5.4. NOTIFY Bodies

The NOTIFY message body contains a content type specific to the requested application (this type must be listed in the Accept header of the SUBSCRIBE). If the subscription is for multiple applications, the initial NOTIFY message body will contain a "multipart/mixed" content-type, and the ordering of the body-parts corresponds to the ordering of the "appids" application values.

6. Example Usage

```
SUBSCRIBE sip:urn%3auuid%3a00000000-0000-1000-0000-00FF8D82EDCB
    @example.com SIP/2.0
Event: ua-profile;profile-type=application;appids="sampleapplication"
From: sip:urn%3auuid%3a00000000-0000-1000-0000-00FF8D82EDCB
    @example.com;tag=1234
To: sip:urn%3auuid%3a00000000-0000-1000-0000-00FF8D82EDCB@example.com
Call-ID: 3573853342923422@192.0.2.44
CSeq: 2131 SUBSCRIBE
Contact: sip:urn%3auuid%3a00000000-0000-1000-0000-00FF8D82EDCB
    @example.com
    ;+sip.instance=<urn:uuid:00000000-0000-0000-123456789AB0>
Via: SIP/2.0/TCP 192.0.2.41;
    branch=z9hG4bK6d6d35b6e2a203104d97211a3d18f57a
Accept: message/external-body, application/x-z100-device-profile
Content-Length: 0
```

7. IANA Considerations

There is one consideration associated with this document. Specifically it registers a new profile type as specified in [I-D.ietf-sipping-config-framework].

Profile Type	Reference
----- application	----- [RFCXXXX]

CONTACT:

sumanth@cablelabs.com

Note to RFC editor: Please replace RFCXXXX with the RFC number assigned to this document.

8. Security Considerations

This document is an extension to the SIP Configuration Framework and as such inherits the security considerations for profile delivery as listed in [I-D.ietf-sipping-config-framework]. In addition, the presence of application ids in the SUBSCRIBE and NOTIFY bodies plays an important role in how the profile data is received by the client. A man-in-the-middle who manipulates the application ids can effectively cause a disruption in application profile data delivery. (Editor's note: Need to add more security considerations, e.g., when is the presence of an app-id a threat.)

9. Acknowledgements

The authors appreciate the feedback received on the SIPPING WG so far, specifically Sam Ganesan from Motorola, Brett Tate from Broadsoft and Paul Kyzivat from Cisco.

10. Normative References

- [I-D.ietf-sipping-config-framework]
Channabasappa, S., "A Framework for Session Initiation Protocol User Agent Profile Delivery", draft-ietf-sipping-config-framework-15 (work in progress), February 2008.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3261] 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.
- [RFC3265] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.
- [RFC3406] Daigle, L., van Gulik, D., Iannella, R., and P. Faltstrom, "Uniform Resource Names (URN) Namespace Definition Mechanisms", BCP 66, RFC 3406, October 2002.
- [RFC4395] Hansen, T., Hardie, T., and L. Masinter, "Guidelines and Registration Procedures for New URI Schemes", BCP 115, RFC 4395, February 2006.

Authors' Addresses

Sumanth Channabasappa
CableLabs
858 Coal Creek Circle
Louisville, Co 80027
USA

Email: sumanth@cablelabs.com
URI: <http://www.cablelabs.com/>

Josh Littlefield
Cisco Systems, Inc.
1414 Massachusetts Avenue
Boxborough, MA 01719
USA

Email: joshl@cisco.com
URI: <http://www.cisco.com/>

Salvatore Loreto
Ericsson, Inc.
Hirsalantie 11
Jorvas, 02420
Finland

Email: Salvatore.Loreto@ericsson.com
URI: http://www.ericsson.com/

Full Copyright Statement

Copyright (C) The IETF Trust (2008).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).

