This specification defines the profile parameter for the SIP Accept header field. This parameter is used to negotiate support for MIME media type extensions.
Table of Contents

1. Introduction .................................................. 3
2. Terminology .................................................... 3
3. Profile Accept Header Field Parameter ......................... 4
4. User Agent Behavior ............................................ 4
5. Applicability to Other Protocols ............................... 5
6. Security Considerations ....................................... 5
7. IANA Considerations ............................................ 5
8. References ....................................................... 5
   Authors’ Addresses ............................................. 6
   Intellectual Property and Copyright Statements ............... 8
1. Introduction

The Accept header field in the Session Initiation Protocol (SIP) [11] provides a mechanism for user agents to negotiate the media types they can accept in message bodies. With this mechanism, a user agent can announce the MIME media types it supports to another user agent.

Many media types used in SIP message bodies are based on the Extensible Markup Language (XML) [13]. XML features a powerful extension mechanism that enables the use of multiple vocabularies in a single XML document. Components from different vocabularies can be uniquely identified using XML namespaces [5].

The extensibility of XML poses a problem for MIME media type based content negotiation. The negotiation only covers major media types and does not include language extensions and variations. For example, a user agent can indicate its support for the media type ‘application/pidf+xml’ (Presence Information Data Format) [12] but cannot indicate that it is also capable of handling the Location Object Extension to PIDF [10] and the location format GML 3.0 [9].

The capability to negotiate support for a language extension or variation is in particular useful in the following cases:

1. The interpretation of the content by the receiver depends on understanding the extension or language variation used.
2. Different versions of the content can be made available depending on the extensions or language variations supported by the receiver. Unsupported extensions can be omitted by the sender to avoid, for example, transmission overhead or costs associated with gathering the information.

In this specification, we propose a new parameter for the Accept header field that enables the negotiation of extensions to MIME media types.

The problem of negotiating support for language extensions and variations has been addressed for a number of specific MIME media types. For example, [1] and [6] define a profile parameter for the MIME media types XHTML and SMIL respectively. However, these solutions are limited to a specific media type and require the registration of a new MIME parameter for each media type that is extended. An abstract framework for content negotiation has been defined in [7].

2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "MAY", and "OPTIONAL".
3. Profile Accept Header Field Parameter

The profile parameter is a new parameter for the Accept header field. It MAY appear multiple times (including zero) in an Accept header field. The profile parameter contains a URI that identifies an extension or a variation of the underlying MIME media type. Currently, the profile parameter is only defined for XML-based media types, which are registered with a ‘+xml’ suffix. For these media types, the URI in the profile parameter identifies an XML namespace. This URI is identical to the URI that would be used in an XML document (e.g., in the XMLNS tag) to identify the same namespace.

The syntax of the ‘profile’ Accept header field parameter is:

```
profile-param = absoluteURI
```

This extends the existing definition of the Accept header field parameters in [11], so that its BNF now looks like:

```
accept-param   =  ("q" EQUAL qvalue) / profile-param / generic-param
```

Example:

```
Accept: application/pidf+xml;
   profile=urn:ietf:params:xml:ns:pidf:geopriv10;
```

4. User Agent Behavior

A user agent, which supports an extension or variation of a MIME media type, SHOULD list this extension or language variation in a profile parameter inserted into the Accept header field of the MIME media type. A user agent SHOULD identify all supported extensions and language variations. It SHOULD NOT list items (e.g., XML namespaces) that are by default part of the media type.

A user agent receiving an Accept header field with a profile parameter can assume that the sender supports the listed extensions and variations and MAY use them to create message bodies.

Note: The absence of an extension in the profile parameter (or the absence of the profile parameter) does not mean the sender does
5. Applicability to Other Protocols

Although this extension has been developed for the SIP Accept header field, it is applicable to the Accept header field of other protocols such as HTTP.

Note: For this broader use, the profile parameter needs to be registered for the respective protocols or in the registry defined in [8].

6. Security Considerations

Security considerations for the Accept header field also apply here.

7. IANA Considerations

This document adds a parameter to the SIP header field parameter registry [4]:

Header field in which the parameter can appear: Accept

Name of the parameter: profile

The parameter only accepts a set of predefined values: No

Reference: this document

8 References


Authors' Addresses

Volker Hilt
Bell Labs/Lucent Technologies
101 Crawfords Corner Rd
Holmdel, NJ  07733
USA

EMail: volkerh@bell-labs.com

Jonathan Rosenberg
Cisco Systems
600 Lanidex Plaza
Parsippany, NJ  07054
USA

EMail: jdrosen@cisco.com
Intellectual Property Statement

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.

Disclaimer of Validity

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 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.

Copyright Statement

Copyright (C) The Internet Society (2005). 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.

Acknowledgment

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