SIPCORE Working Group
Internet-Draft
Intended status: Standards Track
Expires: June 15, 2012

C. Holmberg
I. Sedlacek
Ericsson
H. Kaplan
Acme Packet

December 13, 2011

Indication of features supported by proxy
draft-holmberg-sipcore-proxy-feature-04.txt

Abstract

This document defines a new SIP header field, Feature-Caps, that can be used by SIP entities to indicate support of features and capabilities, in cases where the Contact header field contains a URI that does not represent the SIP entity that wants to indicate support of its features and capabilities.

Table of Contents

1. Introduction

The Session Initiation Protocol (SIP) "Caller Preferences" extension, defined in RFC 3840 [RFC3840], provides a mechanism that allows a SIP message to convey information relating to the originator's features and capabilities, using the Contact header field.

This document defines a new SIP header field, Feature-Caps, that can be used by SIP entities to indicate support of features and capabilities, in cases where the Contact header field contains a URI that does not represent the SIP entity that wants to indicate support of its features and capabilities. Such cases are:

- - The SIP entity acts as a SIP proxy.
- - The SIP entity acts as a SIP registrar.
- - The SIP entity acts as a B2BUA, where the Contact header field URI represents another SIP entity.

2. Conventions

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 BCP 14, RFC 2119 [RFC2119].

3. Definitions

Downstream SIP entity: SIP entity in the direction towards which a SIP request is sent.

Upstream SIP entity: SIP entity in the direction from which a SIP request is received.

4. User Agent (UA) Behavior

4.1. General

If the URI in a Contact header field of a request or response represents a UA, the UA MUST NOT indicate supported features and capabilities using a Feature-Caps header field within that request or response.

When a UA receives a SIP request, or response, that contains one or more Feature-Caps header fields, the Feature Tags in the header field inform the UA is about the features supported by the entities that

inserted the header fields. Procedures how features are invoked are outside the scope of this specification, and MUST be described by individual Feature Tag specifications.

When the UA receives the SIP request or the response, the feature tags in the topmost Feature-Caps header field will represent the supported features "closest" to the UA.

4.2. B2BUA Behavior

The procedures in this section applies to UAs that are part of B2BUAs, but where the URI in the Contact header field does not represent the UA, because the B2BUA is also acting as a proxy and inserts its URI e.g. in a Record-Route header field.

When a UA sends a SIP request, if the UA wants to indicate support of features towards its downstream SIP entities, it adds a Feature-Caps header field to the request, together with one or more Feature Tags associated with the supported features, before it forwards the request.

If the SIP request is triggered by another SIP request that the B2BUA has received, the UA MAY forward those Feature-Caps header field by copying them to the outgoing SIP request, similar to a SIP proxy, before it adds its own Feature-Caps header field to the SIP request.

When a UA receives a SIP response, if the UA wants to indicate support of features towards its upstream SIP entities, it adds a Feature-Caps header field to the response, together with one or more Feature Tags associated with the supported features, before it forwards the response.

If the SIP response is triggered by another SIP response that the B2BUA has received, the UA MAY forward those Feature-Caps header field by copying them to the outgoing SIP response, similar to a SIP proxy, before it adds its own Feature-Caps header field to the SIP response.

4.3. Registrar Behavior

If a SIP registrar wants to indicate support of features towards its upstream SIP entities, it can insert a Feature-Caps header field, together with Feature Tags associated with the supported features, in a REGISTER response.

5. Proxy behavior

When a proxy receives a SIP request, if the proxy wants to indicate support of features towards its downstream SIP entities, it adds a Feature-Caps header field to the request, together with one or more Feature Tags associated with the supported features, before it forwards the request.

When a proxy adds a Feature-Caps header field to a SIP request, it MUST place the header field before any existing Feature-Caps header field in the request.

When a proxy receives a SIP response, if the proxy wants to indicate support of features towards its upstream SIP entities, it adds a Feature-Caps header field to the response, together with one or more Feature Tags associated with the supported features, before it forwards the response.

When a proxy adds a Feature-Caps header field to a SIP response, it MUST place the header field before any existing Feature-Caps header field in the response.

6. Feature-Caps Header Field Definition

6.1. General

This section describes how the Feature-Caps header field is used, and the associated semantics, with different SIP methods and response codes.

NOTE: Future specification can define usage semantics of the Feature-Caps header fields for SIP methods, response codes and request types not specified in this specification.

6.2. SIP Dialog

The Feature-Caps header field can be used within an initial SIP request for a dialog, within a target refresh SIP request, and within any 18x or 2xx response associated with such requests.

If a Feature Tag is inserted in a Feature-Caps header field of such request or such response, the feature associated with the Feature Tag MUST be supported for the dialog, until the next time the dialog target is refreshed, or the dialog is terminated.

For a given dialog the entity MUST use the same Feature-Caps header field value (if included) in all 18x and 2xx responses for the same

transaction.

6.3. SIP Registration (REGISTER)

The Feature-Caps header field can be used within a SIP REGISTER request, and within the 200 (OK) response of such request.

If a Feature Tag is inserted in a Feature-Caps header field of a SIP REGISTER request or response, the feature associated with the Feature Tag MUST be supported for the registration, and all SIP transactions associated with the registration, until the registration is refreshed or terminated.

6.4. SIP Stand-Alone Transactions

The Feature-Caps header field can be used within an request for standalone SIP transaction, and within any 18x or 2xx response associated with such request.

If a Feature Tag is inserted in a Feature-Caps header field of an request or response for a standalone transaction, the feature associated with the Feature Tag MUST be supported for the duration of the standalone transaction.

6.5. SIP Capability Query (OPTIONS)

7. Syntax

7.1. General

Each value of a Feature-Caps header field MUST contain a "*" value, followed by one or more Feature Tags, associated with the supported features, separated by semicolon (";").

NOTE: A "*" value means that no information regarding which proxy, or domain, that support the features associated with the Feature Tags, is provided.

NOTE: When used in a Contact header field, a "*" value has an "any URI" meaning. When used in a Feature-Caps header field, it simply means that no URI information is provided.

7.2. ABNF

The ABNF for the Feature-Caps header fields is:

```
Feature-Caps = ("Feature-Caps" / "fc") HCOLON fc-value
               *(COMMA fc-value)
fc-value     = "*" *(SEMI feature-param)
               ;;feature-param from RFC 3840
```

Figure 1: ABNF

8. Feature Tag Usage With Feature-Caps

Feature tags inserted in a Feature-Caps header field indicate that the SIP entity that inserted the header field supports the associated features.

In order to use a Feature Tag in a Feature-Caps header field, the Feature Tag specification MUST specify the semantics of the feature tag when inserted in the Feature-Caps header field. Unless the feature specification defines such semantics, a the Feature Tag MUST NOT be used in a Feature-Caps header field.

Within a given Feature-Caps header field, Feature Tags are listed in a non-priority order, and for a given header field any order of listed Feature Tags have the same meaning. For example, "foo;bar" and "bar;foo" have the same meaning (i.e. that the entity that inserted the Feature Tags supports the features associated with the "foo" and "bar" Feature Tags.

9. Feature Tag Requirements

9.1. General

This section provides guidance on how to define an Feature Tag, and what information needs to exist in an Feature Tag specification.

It is bad practice for Feature Tag specifications to repeat procedures defined in this document, unless needed for clarification or emphasis purpose.

Feature Tag specifications MUST NOT weaken any behavior designated with "SHOULD" or "MUST" in this specification. However, Feature Tags specifications MAY strengthen "SHOULD", "MAY", or "RECOMMENDED" requirements to "MUST" strength if features associated with the Feature Tag require it.

Feature Tag specifications MUST address the issues defined in the following subsections, or document why an issue is not applicable for

the specific Feature Tag.

9.2. Overall Description

The Feature Tag specification MUST contain an overall description of the Feature Tag: a description of the feature associated with the Feature Tag, and a description what information is carried in associated Feature Tag values (if any).

9.3. Applicability

The Feature Tag specification MUST describe why the support of the feature can not be indicated in a SIP Contact header field [RFC3261], using the mechanism defined in RFC 3840. The reason is either that the entity inserting the Feature Tag is acting as a SIP proxy, or SIP registrar, or a B2BUA but is not represented by the URI in the Contact header field.

9.4. Feature Tag Name

The Feature Tag specification MUST define an Feature Tag name, which entities use as a header field value to identify the Feature Tag in the Feature-Caps header field. The Feature Tag name MUST conform to the ABNF defined in Section 7.2.

9.5. Feature Tag Values

The Feature Tag specification MAY define Feature Tag values, associated with the Feature Tag. A Feature Tag value MUST conform to the ABNF defined in Section 7.2.

The Feature Tag specification MUST define the syntax and semantics of the values associated with the Feature Tag. In addition, the specification MUST define whether there are restrictions regarding the usage of specific values.

Feature Tags can share value defined for other Feature Tags, but the value is Feature Tag specific, and the value semantics MUST be defined for each Feature Tag tag uses the value.

9.6. SIP Option Tags

The Feature Tag specification MAY define SIP option tags, which can be used as describe in RFC 3261.

The registration requirements for option tags are defined in RFC 5727 [RFC5727].

9.7. Feature Tag Usage Restrictions

If there are restrictions on how entities can insert a Feature Tag, the Feature Tag specification MUST document such restrictions.

There can be restrictions related to whether entities are allowed to insert Feature Tags in registration related messages, standalone transaction messages, or dialog related messages, whether entities are allowed to insert Feature Tags in requests or responses, whether entities also need to support other features in order to insert a Feature Tag, and whether entities are allowed to insert Feature Tags togheter with other Feature Tags.

9.8. Feature Tag Security Considerations

If the information carried in a Feature Tag requires a certain level of security, the Feature Tag specification MUST describe the mechanisms that entities need to use in order to provide the required security.

If the Feature Tag specification does not require any additional security, other than what the underlying SIP protocol provides, this MUST be stated in the Feature Tag specification.

9.9. Implementation Details

It is strongly RECOMMENDED that the Feature Tag specification defines the procedure regarding how implementors shall implement and use the Feature Tag, or refer to other locations where implementors can find that information.

NOTE: Sometimes an Feature Tag designer might choose to not reveal the details of an Feature Tag. However, in order to allow multiple implementations to support the Feature Tag, Feature Tag designers are strongly encouraged to provide the implementation details.

9.10. Examples

It is RECOMMENDED that the Feature Tag specification provide demonstrative message flow diagrams, paired with complete messages and message descriptions.

Note that example flows are by definition informative, and do not replace normative text.

10. IANA Considerations

10.1. Registration of the Feature-Caps header field

This specification registers a new SIP header field, Feature-Caps, according to the process of RFC 3261 [RFC3261].

The following is the registration for the Feature-Caps header field:

RFC Number: RFC XXX

Header Field Name: Feature-Caps

Compact Form: fc

11. Security Considerations

Feature tags can provide sensitive information about a SIP entity. RFC 3840 cautions against providing sensitive information to another party. Once this information is given out, any use may be made of it.

12. Acknowledgements

13. Change Log

[RFC EDITOR NOTE: Please remove this section when publishing]

Changes from draft-holmberg-sipcore-proxy-feature-03

- Hadriel Kaplan added as co-author.
- Terminology change: instead of talking of proxies, talk about entities which are not represented by the URI in a Contact header field (http://www.ietf.org/mail-archive/web/sipcore/current/msg04449.html).
- Clarification regarding the usage of the header field in 18x/2xx responses (http://www.ietf.org/mail-archive/web/sipcore/current/msg04449.html).
- Specifying that feature support can also be indicated in target refresh requests (http://www.ietf.org/mail-archive/web/sipcore/current/msg04454.html).
- Feature Tag specification registration information added.

Changes from draft-holmberg-sipcore-proxy-feature-02

- Definition, and usage of, a new header field, instead of Path, Record-Route, Route and Service-Route.

Changes from draft-holmberg-sipcore-proxy-feature-01
- Requirement section added
- Use-cases and examples updated based on work in 3GPP

Changes from draft-holmberg-sipcore-proxy-feature-00
- Additional use-cases added
- Direction section added

14. References

14.1. Normative References

[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.

[RFC3840] Rosenberg, J., Schulzrinne, H., and P. Kyzivat, "Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)", RFC 3840, August 2004.

[RFC5727] Peterson, J., Jennings, C., and R. Sparks, "Change Process for the Session Initiation Protocol (SIP) and the Real-time Applications and Infrastructure Area", BCP 67, RFC 5727, March 2010.

14.2. Informative References

[3GPP.23.237] 3GPP, "IP Multimedia Subsystem (IMS) Service Continuity; Stage 2", 3GPP TS 23.237 10.7.0, September 2011.

[3GPP.24.837] 3GPP, "IP Multimedia (IM) Core Network (CN) subsystem inter-UE transfer enhancements; Stage 3", 3GPP TR 24.837 10.0.0, April 2011.

Authors' Addresses

Christer Holmberg
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: christer.holmberg@ericsson.com

Ivo Sedlacek
Ericsson
Scheelevaegen 19C
Lund 22363
Sweden

Email: ivo.sedlacek@ericsson.com

Hadriel Kaplan
Acme Packet
71 Third Ave.
Burlington, MA 01803
USA

Email: hkaplan@acmepacket.com