SIPPING Working Group
Internet-Draft
Expires: December 27, 2006

S. Loreto
G. Camarillo
Ericsson
June 25, 2006

The Session Initiation Protocol (SIP) Dialog Correlation draft-loreto-sipping-dialog-correlation-01.txt

Status of this Memo

Copyright Notice

Abstract

This document defines a new header field for use with SIP. The Same-Session header field is used to logically correlate an existing SIP dialog with a new SIP dialog when the media sessions established by both dialogs can be considered a single logical session. This mechanism can be used to share the user interface and other resources between all the media streams from both sessions.

Table of Contents

1. 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 [3].

2. Overview

This document defines a new SIP [5] header field: Same-Session. The Same-Session header field is used to logically correlate an existing SIP dialog with a new SIP dialog when the media sessions established by both dialogs can be considered a single logical session. This is especially useful in peer-to-peer call control environments.

While is possible insert a new participant into a multimedia conversation with the Join header field [6], the Join operation is normally used to create or join a conference. It adds a dialog to the conversation space associated with the matched dialog and performs a media mixing or media combining.

Instead, the Same-Session operation inserts a new dialog into a multimedia conversation. It enables a dialog to share all the resources and the user interface with the matched dialog.

Obviously it is also possible to achive the Same-Session operetion effect using Third Party Call Controll (3pcc) [18] and the SIP Session Mobility [20]. However there are various disadvantages in the use of 3pcc.

Appendix A provides some concrete examples regarding the different complexity level using the 3pcc or the Same-Session header.

3. Requirements

This specification was created in order to meet the following requirement:

It should be possible for a user agent to correlate two dialogs so that all the media streams associated to them are treated as a single media session.

4. Use case

Alice establishes a voice session with Bob. Alice wants add video to the session using her SIP-enabled camera. Alice sends a REFER to her

camera, which has SIP user agent on it, so that her camera sends an INVITE request to Bob in order to establish a video stream. Alice wants Bob to treat the video stream from her camera and the voice stream from her voice-only user agent as part of the same media session. That is, Alice wants Bob to treat both streams as if both had been established using a single SIP dialog.

5. Same-Session Header Field Syntax

The following is the augmented Backus-Naur Form (BNF) syntax [2] of the Same-Session header field:

```
        Same-Session         = "Same-Session" HCOLON callid * (SEMI same-
session-param)
        same-session-param  = to-tag / form-tag / strictly-flag / generic-
param
        to-tag              = "to-tag" EQUAL token
        from-tag            = "from-tag" EQUAL token
```

Examples:

```
        Same-Session: 98732@sip.example.com
                     ;from-tag=r33th4x0r
                     ;to-tag=ff87ff

        Same-Session: 12adf2f34456gs5;to-tag=12345;from-tag=54321;strictly

        Same-Session: 87134@171.161.34.23;to-tag=24796;from-tag=0
```

6. User Agent Server Behavior

The Same-Session header field contains information used to match an existing SIP dialog (Call-ID, to-tag, and from-tag). Upon receiving an INVITE with a Same-Session header field, the UA (User Agent) attempts to match this information with a confirmed or early dialog. The to-tag and from-tag parameters are matched as if they were tags present in an incoming request. In other words the to-tag parameter is compared to the local tag, and the from-tag parameter is compared to the remote tag.

If more than one Same-Session header field is present in an INVITE, or if a Same-Session header field is present in a request other than INVITE, the UAS (User Agent Server) MUST reject the request with a 400 (Bad Request) response.

The Same-Session header has specific call control semantics. If both

a Same-Session header field and another header field with contradictory semantics (for example a Replaces [7] header field) are present in a request, the request MUST be rejected with a 400 (Bad Request) response.

If the Same-Session header field matches more than one dialog, the UA MUST act as if no match is found.

If no match is found, the UAS rejects the INVITE and returns a 481 (Call/Transaction Does Not Exist) response. Likewise, if the Same-Session header field matches a dialog which was not created with an INVITE, the UAS MUST reject the request with a 481 (Call/Transaction Does Not Exist) response.

If the Same-Session header field matches a dialog which has already terminated, the UA SHOULD decline the request with a 603 (Decline) response.

If the Same-Session header field matches an active dialog, the UA MUST verifies that the initiator of the new INVITE is authorized to be part of the session previously established by the matched dialog. If the initiator of the new INVITE has authenticated successfully as equivalent to the user who established the matched dialog, then the merging of both session is authorized. For example, if the user who established the initial dialog and the initiator of the new INVITE request share the same credentials for Digest authentication [8], or they sign the correlation request with S/MIME [11] with the same private key and present the (same) corresponding certificate used in the original dialog, then the merging of the session is authorized.

Alternatively, the Referred-By mechanism [9] defines a mechanism that the UAS can use to verify that an INVITE request with a Same-Session header field was sent on behalf of the other participant in the matched dialog (in this case, triggered by a REFER request). If the INVITE request contains a Referred-By header which corresponds to the user that established the matched dialog, the UA SHOULD authorize the merging of the sessions. The Referred-By header field MUST reference a corresponding, valid Referred-By Authenticated Identity Body [10]. The UA MAY apply other local policy to authorize the remainder of the request. In other words, the UAS may apply different policy to the new dialog than was applied to the matched dialog.

If authorization is successful, the UA attempts to accept the new INVITE and treats the session newly-established and the previously established session as if they were one. It SHOULD return in the response the Contact header filled in the same way as it returned during the original dialog establishment phase; in this way, subsequent users joining the session will be able to use the same

URL.

If the authorization is successful, but the UA cannot accept the new INVITE (for example: it cannot establish required QoS or keying, or it has incompatible media), the UA MUST return an appropriate error response and MUST leave the matched dialog unchanged.

If the UAS is incapable of satisfying the Same-Session request, it MUST return a 488 (Not Acceptable Here) response.

7. User Agent Client Behavior

A User Agent that wishes to add a new dialog of its own to a single existing early or confirmed dialog sends the target User Agent an INVITE request containing a Same-Session header field. The UAC (User Agent Client) places the Call-ID, to-tag, and from-tag information for the target dialog in a single Same-Session header field and sends the new INVITE to the target.

If the User Agent receives a 300-class response, and acts on this response by sending an INVITE to a Contact in the response, this redirected INVITE MUST contain the same Same-Session header which was present in the original request. Although this is unusual, this allows INVITE requests with a Same-Session header to be redirected before reaching the target UAS.

Note that use of the Same-Session mechanism does not provide a way to match multiple dialogs, nor does it provide a way to match an entire call, an entire transaction, or to follow a chain of proxy forking logic.

8. New Same-Session Option Tag

This specification defines a new Require/Supported header option tag "Same-Session". UAs which support the Same-Session header field MUST include the "Same-Session" option tag in a Supported header field. UAs that want explicit failure notification if Same-Session is not supported MAY include the "Same-Session" option in a Require header field.

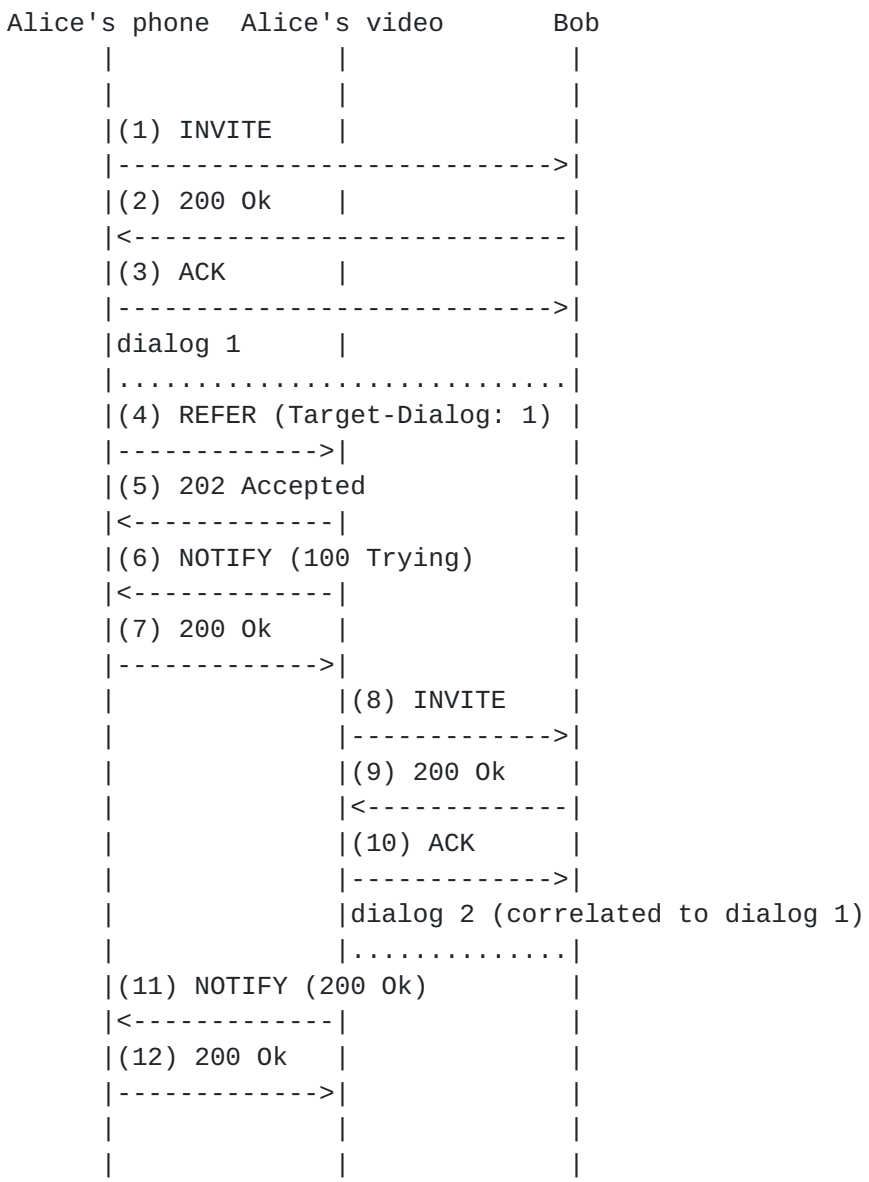
The following is an example of a Require header field with the "Same-Session" option tag:

Require: Same-Session

9. Usage Example

The following non-normative examples are not intended to enumerate all the possibilities for the usage of this extension, but rather to provide examples or ideas only.

9.1. Correlate a Dialog



In this example, Alice starts a phone call with Bob (messages 1,2,3). At a later point, Alice wants to add video to the session using a different user agent that supports video. Alice wants Bob to treat media stream (i.e., audio and video)as if they had been established using a single INVITE-initiated dialog. Consequently, Alice's user agent generates the following REFER request.

```
REFER sip:aliceVideo@b.example.org SIP/2.0
To: <sip:aliceVideo@example.org>
From: <sip:alicePhone@example.org>;tag=iii
Call-Id: 7@a.example.org
CSeq: 1 REFER
Contact: <sip:alicePhone@example.org>
Refer-to: <sip:Bob@example.com?Same-Session=98732@example.com
        %3Bfrom-tag=r33th4x0r%3Bto-tag=ff87ff>
Referred-By: < sip:alicePhone@example.org>
```

When Alice video-enabled user agent receives the REFER request, it establish a new dialog (message 8,9,10) with Bob using the information received in the REFER request.

```
INVITE sip:bob@b.example.org SIP/2.0
To: <sip:bob@example.org>
From: <sip:aliceVideo@example.org>;tag=iii
Call-Id: 777@a.example.org
CSeq: 1 INVITE
Contact: <sip:aliceVideo@example.org>
Same-Session: 425928@phone.example.org;to-tag=xyz;from-tag=pdq
```

```
SIP/2.0 200 OK
To: <sip:bob@example.org>
From: <sip:aliceVideo@example.org>;tag=iii
Call-Id: 777@a.example.org
CSeq 1 INVITE
Contact: <sip:bob@b.example.org>
```

10. Security Considerations

The extension specified in this document significantly changes the relative security of SIP devices. It has the same problems of both "Join" and "Replace" header fields.

This extension can be used to insert a new dialog in a multimedia conversation in order to monitor potentially sensitive content. As such, invitations with the Same-Session header field MUST only be accepted if the peer requesting a Same-Session has been properly authenticated as a user already involved in the call.

11. IANA Considerations

11.1. Registration of Same-Session SIP header field

Name of Header:	Same-Session
Short form:	none
Normative description:	RFC xxxx

11.2. Registration of Same-Session SIP Option-tag

Name of option:	Same-Session
Description:	Support for the SIP Correlation header
SIP headers defined:	Same-Session
Normative description:	RFC xxxx

12. Acknowledges

Goeran Ericsson provided valuable ideas for this document.

13. References

13.1. Normative References

[1] Handley, M., "SDP: Session Description Protocol", draft-ietf-mmusic-sdp-new-26 (work in progress), January 2006.

[2] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997.

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

[4] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

13.2. Informational References

[5] 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.

[6] Mahy, R. and D. Petrie, "The Session Initiation Protocol (SIP) "Join" Header", RFC 3911, October 2004.

[7] Mahy, R., Biggs, B., and R. Dean, "The Session Initiation Protocol (SIP) "Replaces" Header", RFC 3891, September 2004.

[8] Franks, J., Hallam-Baker, P., Hostetler, J., Lawrence, S., Leach, P., Luotonen, A., and L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, June 1999.

[9] Sparks, R., "The Session Initiation Protocol (SIP) Referred-By Mechanism", RFC 3892, September 2004.

[10] Peterson, J., "Session Initiation Protocol (SIP) Authenticated Identity Body (AIB) Format", RFC 3893, September 2004.

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

[12] Campbell, B., Rosenberg, J., Schulzrinne, H., Huitema, C., and D. Gurle, "Session Initiation Protocol (SIP) Extension for Instant Messaging", RFC 3428, December 2002.

[13] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.

[14] Sparks, R., "The Session Initiation Protocol (SIP) Refer Method", RFC 3515, April 2003.

[15] Donovan, S., "The SIP INFO Method", RFC 2976, October 2000.

[16] Rosenberg, J., "The Session Initiation Protocol (SIP) UPDATE Method", RFC 3311, October 2002.

[17] Rosenberg, J. and H. Schulzrinne, "Reliability of Provisional Responses in Session Initiation Protocol (SIP)", RFC 3262, June 2002.

[18] Rosenberg, J., Peterson, J., Schulzrinne, H., and G. Camarillo, "Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP)", BCP 85, RFC 3725, April 2004.

[19] Camarillo, G., Marshall, W., and J. Rosenberg, "Integration of Resource Management and Session Initiation Protocol (SIP)",

RFC 3312, October 2002.

[20] Shacham, R., "Session Initiation Protocol (SIP) Session Mobility", draft-shacham-sipping-session-mobility-02 (work in progress), March 2006.

Appendix A. Same Session header AND Third Party Call Controll

It is possible to achive the Same-Session operation effect using Third Party Call Controll (3pcc) [18] and the SIP Session Mobility [20]. However there are various cons in the use of 3pcc:

- complexity: some use cases that are quite complex implemented using the Third Party Call Controll (3pcc) become more simpler using the Same-Session Header.
- implementation: not many terminals are going to implement what is needed to be a 3pcc controller. However, any terminal will implement REFER.
- support of an extension at the remote end: the controller needs to understand all the SIP extensions applied to both dialogs.

Moreover Same-Session Header solves the SIP lack, underlined in the SIP Session Mobility [20], of a standard way to associate multiple sessions as part of a single call in SIP.

The following examples show the different complexity, in term of amount of messages, using the Same-Session header or the 3pcc, in the scenario where the user A has an on-going session with the user agent B and then A wants to add a new media to the session using a different user agent C.

Appendix A.1. Example: preconditions using the Same-Session Header

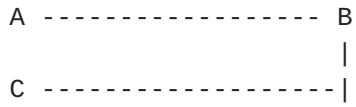


Figure 1: Same-Session architecture

Fig.1 shows the architecture achived using the Same-Sessione header peer to peer call control model.

Using the same-session header, as showed in fig.2, A issues a REFER transaction to C, then C send an INVITE to B following the basic session establishment call flow showed in Figure 1 of [19]. The flow if fig.2 has 17 messages.

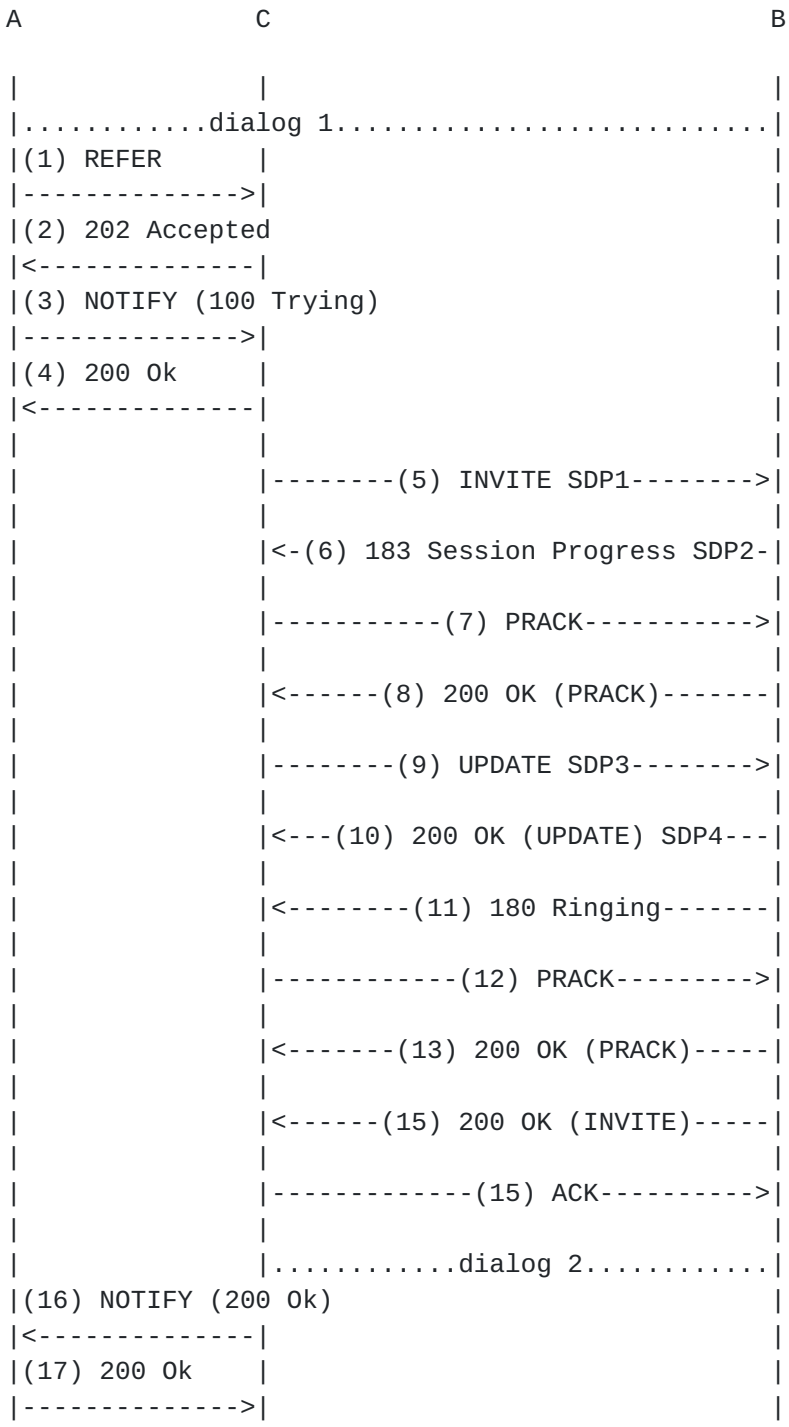


Figure 2: Basic session establishment using Same-Session and preconditions

Appendix A.2. Example: preconditions using the 3pcc

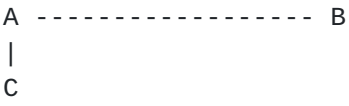
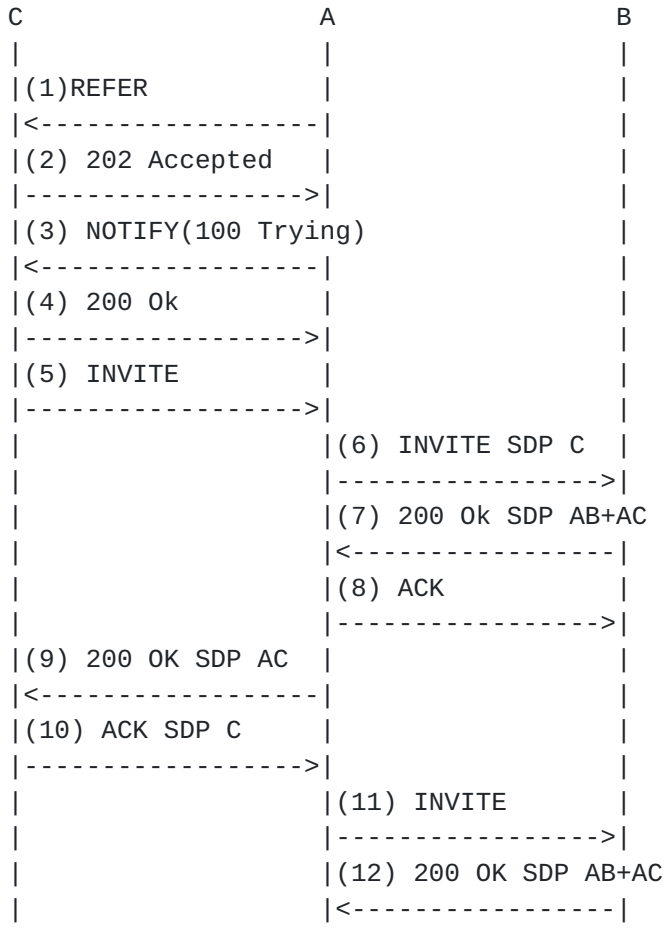


Figure 3: 3pcc architecture

Fig.3 shows the architecture achived using the Third Party Call Control (3pcc) model.

Using 3pcc, A behaves as the controller in Figure 11 of [18]. In this scenario the flow contains 26 messages. We don't insert the figure for the sake of space.

Alternatively, it is possible using 3pcc in a different way. A issues a REFER to C and C send the INVITE towards A. The flow, as showed in fig.4, without precondition already has 16 messages.



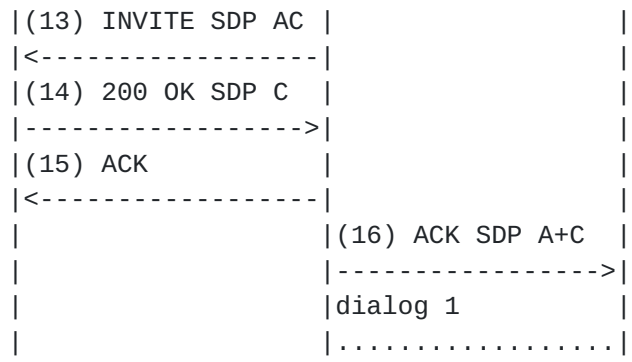


Figure 4

Authors' Addresses

Salvatore Loreto
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: Salvatore.Loreto@ericsson.com

Gonzalo Camarillo
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: Gonzalo.Camarillo@ericsson.com

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

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