Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 23, 2008

X. Fu
C. Dickmann
University of Goettingen
J. Crowcroft
University of Cambridge
February 20, 2008

General Internet Signaling Transport (GIST) over SCTP
draft-ietf-nsis-ntlp-sctp-04.txt

Status of this Memo

This Internet-Draft will expire on August 23, 2008.

Copyright Notice

Abstract

The General Internet Signaling Transport (GIST) protocol currently uses TCP or TLS over TCP for connection mode operation. This document describes the usage of GIST over the Stream Control Transmission Protocol (SCTP). The use of SCTP can take advantage of features provided by SCTP, namely streaming-based transport, support of multiple streams to avoid head of line blocking, and the support

of multi-homing to provide network level fault tolerance. Additionally, the support for the Partial Reliability Extension of SCTP is discussed.

Table of Contents

1. Introduction

This document describes the usage of the General Internet Signaling Transport (GIST) protocol [1] over the Stream Control Transmission Protocol (SCTP) [2].

GIST, in its initial specification for connection mode operation, runs on top of a byte-stream oriented transport protocol providing a reliable, in-sequence delivery, i.e., using the Transmission Control Protocol (TCP) [5] for signaling message transport. However, some NSLP context information has a definite lifetime, therefore, the GIST transport protocol could benefit from flexible retransmission, so stale NSLP messages that are held up by congestion can be dropped. Together with the head-of-line blocking issue and other issues with TCP, these considerations argue that implementations of GIST should support the Stream Control Transport Protocol (SCTP)[2] as an optional transport protocol for GIST, especially if deployment over the public Internet is contemplated. Like TCP, SCTP supports reliability, congestion control and fragmentation. Unlike TCP, SCTP provides a number of functions that are desirable for signaling transport, such as multiple streams and multiple IP addresses for path failure recovery. In addition, its Partial Reliability extension (PR-SCTP) [3] supports partial retransmission based on a programmable retransmission timer.

This document defines the use of SCTP as a transport protocol for GIST Messaging Associations and discusses the implications on GIST State Maintenance and API between GIST and NSLPs. Furturemore, this document shows how GIST SHOULD be used to provide the additional features offered by SCTP to deliver the GIST C-mode messages (which can in turn carry NSIS Signaling Layer Protocol (NSLP) [6] messages as payload). More specifically:

- How to use the multiple streams feature of SCTP.
- How to use the PR-SCTP extention of SCTP.
- How to take advantage of the multi-homing support of SCTP.

The method described in this document does not require any changes of GIST or SCTP. However, SCTP implementations MUST support the optional feature of fragmentation of SCTP user messages.

2. Terminology and Abbreviations

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 [4]. Other terminologies and abbreviations used in this document are taken from related specifications (e.g., [1] and [2]) as follows:

- SCTP - Stream Control Transmission Protocol
- PR-SCTP - SCTP Partial Reliability Extension
- MRM - Message Routing Method
- MRI - Message Routing Information
- MRS - Message Routing State
- MA - A GIST Messaging Association is a single connection between two explicitly identified GIST adjacent peers on the data path. A messaging association may use a specific transport protocol and known ports. If security protection is required, it may use a specific network layer security association, or use a transport layer security association internally. A messaging association is bidirectional; signaling messages can be sent over it in either direction, and can refer to flows of either direction.
- SCTP Association - A protocol relationship between SCTP endpoints, composed of the two SCTP endpoints and protocol state information. An association can be uniquely identified by the transport addresses used by the endpoints in the association. Two SCTP endpoints MUST NOT have more than one SCTP association between them at any given time.
- Stream - A sequence of user messages that are to be delivered to the upper-layer protocol in order with respect to other messages within the same stream.

3. GIST Over SCTP

3.1. Message Association Setup

3.1.1. Overview

The basic GIST protocol specification defines two possible protocols to be used in Messaging Associations, namely Forwards-TCP and TLS. This document adds Forwards-SCTP as another possible protocol. In Forwards-SCTP, analog to Forwards-TCP, connections between peers are opened in the forwards direction, from the querying node, towards the responder.

A new MA-Protocol-ID type, "Forwards-SCTP", is defined in this document for using SCTP as GIST transport protocol. A formal definition of Forwards-SCTP is given in the following section.

3.1.2. Protocol-Definition: Forwards-SCTP

This MA-Protocol-ID denotes a basic use of SCTP between peers. Support for this protocol is OPTIONAL. If this protocol is offered, MA-protocol-options data MUST also be carried in the SCD object. The MA-protocol-options field formats are:

- in a Query: no information apart from the field header.
- in a Response: 2 byte port number at which the connection will be accepted, followed by 2 pad bytes.

The connection is opened in the forwards direction, from the querying node towards the responder. The querying node MAY use any source address and source port. The destination information MUST be derived from information in the Response: the address from the interface-address from the Network-Layer-Information object and the port from the SCD object as described above.

Associations using Forwards-SCTP can carry messages with the transfer attribute Reliable=True. If an error occurs on the SCTP connection such as a reset, as can be detected for example by a socket exception condition, GIST MUST report this to NSLPs as discussed in Section 4.1.2 of [1].

3.2. Effect on GIST State Maintenance

This document defines the use of SCTP as a transport protocol for GIST Messaging Associations. As SCTP provides additional functionality over TCP, this section dicusses the implications of using GIST over SCTP on GIST State Maintenance.

While SCTP defines uni-directional streams, for the purpose of this document, the concept of a bi-direction stream is used. Implementations MUST establish downstream and upstream (uni-directional) SCTP streams always together and use the same stream identifier in both directions. Thus, the two uni-directional streams (in opposite directions) form a bi-directional stream.

Due to the multi-streaming support of SCTP, it is possible to use different SCTP streams for different resources (e.g., different NSLP sessions), rather than maintaining all messages along the same transport connection/association in a correlated fashion as TCP (which imposes strict (re)ordering and reliability per transport level). However, there are limitations to the use of multi-streaming. All GIST messages for a particular session MUST be sent over the same SCTP stream to assure the NSLP assumption of in-order delivery. Multiple sessions MAY share the same SCTP stream based on local policy.

The GIST concept of Messaging Association re-use is not affected by this document or the use of SCTP. All rules defined in the GIST specification remain valid in the context of GIST over SCTP.

3.3. PR-SCTP Support

A variant of SCTP, PR-SCTP [3] provides a "timed reliability" service. It allows the user to specify, on a per message basis, the rules governing how persistent the transport service should be in attempting to send the message to the receiver. Because of the chunk bundling function of SCTP, reliable and partial reliable messages can be multiplexed over a single PR-SCTP association. Therefore, a GIST over SCTP implementation SHOULD attempt to establish a PR-SCTP association instead of a standard SCTP association, if available, to support more flexible transport features for potential needs of different NSLPs.

3.4. API between GIST and NSLP

GIST specification defines an abstract API between GIST and NSLPs. While this document does not change the API itself, the semantics of some parameters have slightly different interpretation in the context of SCTP. This section only lists those primitives and parameters, that need special consideration when used in the context of SCTP. The relevant primitives are repeatet from [1] to improve readability, but [1] remains authoritative.

3.4.1. SendMessage

The SendMessage primitive is used by the NSLP to initiate sending of messages.

```
SendMessage ( NSLP-Data, NSLP-Data-Size, NSLP-Message-Handle,
              NSLP-Id, Session-ID, MRI,
              SSI-Handle, Transfer-Attributes, Timeout, IP-TTL, GHC )
```

The following parameter has changed semantics:

Timeout: According to [1] this parameter represents the "length of time GIST should attempt to send this message before indicating an error". When used with SCTP, this parameter is also used as the timeout for the "timed reliability" service of PR-SCTP.

3.4.2. NetworkNotification

The NetworkNotification primitive is passed from GIST to an NSLP. It indicates that a network event of possible interest to the NSLP occurred.

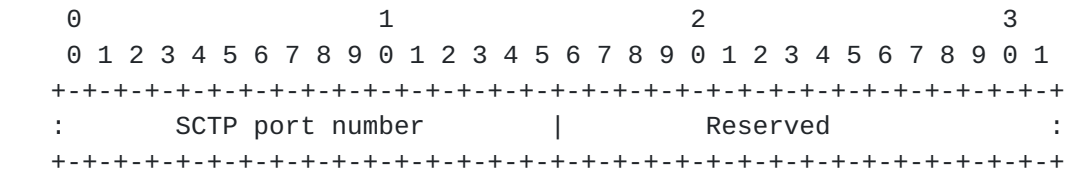
```
NetworkNotification ( MRI, Network-Notification-Type )
```

If SCTP detects a failure of the primary path, GIST SHOULD indicate this event to the NSLP by calling the NetworkNotification primitive with Network-Notification-Type "Routing Status Change". This notification should be done even if SCTP was able to remain an open connection to the peer due to its multi-homing capabilities.

4. Bit-Level Formats

4.1. MA-Protocol-Options

This section provides the bit-level format for the MA-protocol-options field that is used for SCTP protocol in the Stack-Configuration-Data object of GIST.



SCTP port number = Port number at which the responder will accept SCTP connections

The SCTP port number is only supplied if sent by the responder.

5. Application of GIST over SCTP

5.1. Multi-homing support of SCTP

In general, the multi-homing support of SCTP can be used to improve fault-tolerance in case of a path- or link-failure. Thus, GIST over SCTP would be able to deliver NSLP messages between peers even if the primary path is not working anymore. However, for the Message Routing Methods (MRMs) defined in the basic GIST specification such a feature is only of limited use. The default MRM is path-coupled, which means, that if the primary path is failing for the SCTP association, it most likely is also for the IP traffic that is signaled for. Thus, GIST would need to perform a refresh anyway to cope with the route change. Nevertheless, the use of the multi-homing support of SCTP provides GIST and the NSLP with another source to detect route changes. Furthermore, for the time between detection of the route change and recovering from it, the alternative path offered by SCTP can be used by the NSLP to make the transition more smoothly. Finally, future MRMs might have different properties and

therefore benefit from multi-homing more broadly.

5.2. Streaming support in SCTP

Streaming support in SCTP is advantageous for GIST. It allows better parallel processing, in particular by avoiding head of line blocking issue in TCP. Since a same GIST MA may be reused by multiple sessions, using TCP as transport GIST signaling messages belonging to different sessions may be blocked if another message is dropped. In the case of SCTP, this can be avoided as different sessions having different requirements can belong to different streams, thus a message loss or reordering in a stream will only affect the delivery of messages within that particular stream, and not any other streams.

6. Security Considerations

The security considerations of both [1] and [2] apply. For securing GIST over SCTP channel, it is recommended to use DTLS [7], to take the advantage of all the features provided by SCTP and its extensions. DTLS over SCTP is currently being specified in [8]. The usage of DTLS for GIST over SCTP is similar to TLS for GIST as specified in [1], where a stack-proposal containing both MA-Protocol-IDs for SCTP and DTLS during the GIST handshake phase.

7. IANA Considerations

Two new MA-Protocol-IDs (Forwards-SCTP and Fowards-DTLS) need to be assigned, with a recommended values of 3 and 4.

8. Acknowledgments

The authors would like to thank John Loughney, Robert Hancock, Andrew McDonald, Martin Stiemerling, Fang-Chun Kuo, Jan Demter for their helpful suggestions.

9. References

9.1. Normative References

[1] Schulzrinne, H. and R. Hancock, "GIST: General Internet Signalling Transport", draft-ietf-nsis-ntlp-15 (work in progress), February 2008.

[2] Stewart, R., "Stream Control Transmission Protocol", RFC 4960,

September 2007.

[3] Stewart, R., Ramalho, M., Xie, Q., Tuexen, M., and P. Conrad, "Stream Control Transmission Protocol (SCTP) Partial Reliability Extension", RFC 3758, May 2004.

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

9.2. Informative References

[5] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

[6] Hancock, R., Karagiannis, G., Loughney, J., and S. Van den Bosch, "Next Steps in Signaling (NSIS): Framework", RFC 4080, June 2005.

[7] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security", RFC 4347, April 2006.

[8] Tuexen, M. and E. Rescorla, "Datagram Transport Layer Security for Stream Control Transmission Protocol", draft-tuexen-dtls-for-sctp-02 (work in progress), November 2007.

Authors' Addresses

Xiaoming Fu
University of Goettingen
Institute of Computer Science
Lotzestr. 16-18
Goettingen 37083
Germany

Email: fu@cs.uni-goettingen.de

Christian Dickmann
University of Goettingen
Institute of Computer Science
Lotzestr. 16-18
Goettingen 37083
Germany

Email: mail@christian-dickmann.de

Jon Crowcroft
University of Cambridge
Computer Laboratory
William Gates Building
15 JJ Thomson Avenue
Cambridge CB3 0FD
UK

Email: jon.crowcroft@cl.cam.ac.uk

Full Copyright Statement

Intellectual Property

Acknowledgment

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