SIP -- Session Initiation Protocol
Working Group
Internet-Draft
Expires: April 12, 2004

D. Willis
B. Campbell
dynamicsoft Inc.

October 13, 2003

Session Initiation Protocol Extension to Assure Congestion Safety
draft-ietf-sip-congestsafe-02

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

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/1id-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 April 12, 2004.

Copyright Notice

Abstract

The Session Initiation Protocol allows the use of UDP for transport of SIP messages. The use of UDP inherently risks network congestion problems, as UDP itself does not define congestion prevention, avoidance, detection, or correction mechanisms. This problem is aggravated by large SIP messages which fragment at the UDP level. Transport protocols in SIP are also negotiated on a per-hop basis, at the SIP level, so SIP proxies may convert from TCP to UDP and so forth. This document defines by which a SIP User Agent may require that its requests are not sent over UDP or other transports having congestion-related characteristics similar to those of UDP.

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 [RFC2119].

2. Background

The Session Initiation Protocol [4] provides application support over multiple transport protocols, including UDP and TCP. Extensions to support SCTP are under consideration, and other transport protocols may be proposed for future use. Transport negotiation is not "end to end" with SIP. Instead, each SIP hop individually determines which transport to use towards the next hop. For example, a User Agent Client (UAC) may use TCP to talk to a proxy, that proxy my use UDP to talk to another proxy, and that second proxy may use SCTP to talk to a destination User Agent Server (UAS).

UDP has inherent issues with congestion management or reliability. The protocol has no explicit mechanisms for avoiding, detecting, or adapting to network congestion. SIP attempts to deal with this in two ways:

1. Retransmission timers with exponential back offs.
2. Attempting to limit the size of transmissions over UDP to reduce the effects of fragmentation.

The fundamental problem with UDP is that it provides no feedback mechanism to allow a sender to pace its transmissions against the real performance of the network. While this tends to have no significant effect on extremely low-volume sender-receiver pairs, the impact of high-volume relationships on the network can be severe. Consider the following scenario, wherein the traffic between multiple UAs is funnelled through a single proxy-proxy relationship.

Example of large-fan out/fan-in likely to encounter congestion:

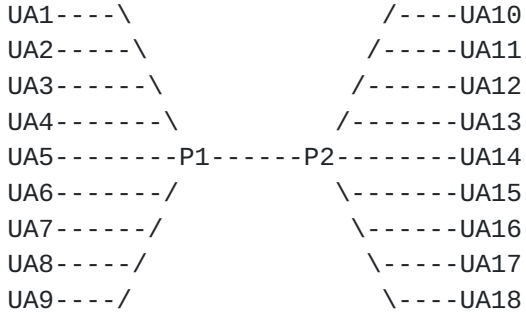

Figure 1

In this scenario, any requests from UA(1..9) to UA(10..18) traverse the proxy-proxy link P1<-->P2. Assuming current SIP practices, if this link is UDP and every UA emits a request simultaneously, each proxy will insert nine (one for each UA) requests, resulting in eighteen simultaneous requests on the P1<-->P2 link. Each request may require retransmissions, and large requests may require fragmentation to fit the link MTU -- at the worst case, producing more than one hundred packets per request, or approximately 2,000 simultaneously expressed packets in this scenario. If the capacity of link P1<-->P2 is inadequate to deliver these messages within the SIP retransmission window, the originating UAs (or the proxies, if acting in transaction-stateful mode) generate retransmissions, further compounding the problem into a "retransmission storm". Real-world scenarios may scale far more seriously. It is not unreasonable to assume that there may be tens of thousands of UAs on each side of the network.

It should be noted that the fundamental problem not just between UAs and proxies, but whenever there is a high fan-out or fan-in ratio. If in the above example, each UA were behind a "residential proxy", the problem would occur in similar fashion.

3. Scope of Work

One solution might be to deprecate UDP entirely for SIP. However, there is a large installed base using UDP, and there are legitimately places where UDP appears to be quite useful such as tiny mobile phones and in extremely high-volume proxies connecting over dedicated networks.

As an alternative, this draft defines mechanisms whereby:

1. a UAC may require that any proxy processing its requests transmit those requests over a transport protocol providing congestion management.
2. a UAC may inform a UAS receiving its requests that those requests were transmitted over a route supporting congestion management, and require that that UAS respond in similar fashion.
3. A proxy may reject requests that require congestion-managed transport when that proxy finds that the only route it has to the next hop is over transport that does not support congestion management.
4. A proxy may reject requests that would be fragmented, even for requests that do not indicate a requirement for congestion-managed transport.
5. A UAS may reject requests that would result in responses that require congestion-managed transport if the originating request

did not require congestion-managed transport.

Note that SIP has no fundamental mechanism whereby a proxy may reject a response. This precludes requiring congestion management for responses being processed by a proxy except as provided by the original request. If, due to an issue of network topology change or similar event between the processing of the request and the processing of the response by a proxy the only path available to the proxy is not congestion managed, the proxy has no choice but to send the response over that path. It's not perfect, but seems to be all we can do at this time.

4. Assuring Transitive Congestion-Managed Transport with Require and Proxy-Require

SIP provides mechanisms whereby a user agent making a request can be assured that any proxy servicing or UAS responding to that request support a specific extension or set of behavior.

To be assured that a proxy servicing the request meets the requirements, the UAC includes a "Proxy-Require" header field with a value indicating a tag for the specific extension or behavior required. As per [4], proxies not recognizing a specific tag or unwilling to support the associated behavior reject a request referencing that tag with a 420 response, which has the semantic "Bad Extension".

To be assured that a UAS responding to a request meets the requirements, the UAC includes a "Require" header field with a value indicating a tag for the specific extension or behavior required. As per [4], UASs not recognizing a specific tag or unwilling to support the associated behavior reject a request referencing that tag with a 420 response, which has the semantic "Bad Extension".

We herein define a an option-tag value of "congestion-managed". There is an IANA registration process for these tags defined in [4], and the "IANA Considerations" of this document fulfills the requirements of the IANA registration process.

5. New Behaviors at SIP Nodes

5.1 Behavior at the UAC

5.1.1 Sending a Request

A UAC exercising this extension adds a Require header field and a Proxy-Require header field value including the option tag "congestion-managed" to each request.

For any request that exercises this extension (i.e., contains the "congestion-managed" option tags), the UAC MUST transmit the request using a protocol that supports congestion maangement.

Any UA supporting this extension SHOULD exercise this extension on all initial requests.

5.1.2 Receiving a 514 Response to a Request

A 514 response (semantic "No available route with congestion management) indicates that an intermediate proxy found that its only vailable routes toward the required next hop did not support congestion management. A UA receiving a 514 response has the options of giving up, trying the request without the "Proxy-Require: congestion-management" (which will likely return a 516) or trying a different set of proxies, presumably through using a different pre-loaded Route header field.

5.1.3 Receiving a 515 Response to a Request

A 515 response (semantic "Response requires congestion management") indicates that the response generated by the UAS responding to the request is larger than the UAS' understanding of path MTU and that the UAS does not know that the route indicated by the VIA headers is over congestion-managed transport. A UAC receiving a 515 to a request may either retry the request in a congestion-managed manner (adding the "congestion-managed" option tag to Require and Proxy-Require)) or abandon the request.

5.1.4 Receiving a 516 Response to a Request

A 516 response (semantic "Proxying of request would induce fragmentation") indicates that a proxy forwarding the request detected that the request was larger than the next hop link MTU from that proxy and that the transport protocol toward that next hop does not support congestion management. A UAS receiving a 516 response may retry the request with a "Proxy-Require: congestion-management" added (which will probably return a 514), retry the request using an alternate route, or abandon the request.

5.2 Behavior at the Proxy

A proxy forwarding a request containing a Proxy-Require with this tag value MUST trasmit that request using a transport protocol (such as TCP) supporting congestion-management. All proxies SHOULD attempt to reduce fragmentation following the procedure described below.

5.2.1 Proxy Rejects Request Requiring Congestion Management When Route with Congestion Management Not Available

When a SIP proxy processing a request marked with a Proxy-Require header field containing the value "congestion-managed" determines that the next hop is reachable only via a transport proocol not supporting congestion management (such as UDP) the proxy MUST reject that request with a 514 response.

5.2.2 Proxy Rejects Request Not Requiring Congestion Management When Forwarding That Request Would Induce Fragmentation

When a SIP proxy supporting this extension and processing a request not marked with a Proxy-Require header field containing the value "congestion-managed" determines that the next hop is reachable only via a transport protocol not supporting congestion management (such as UDP) and the size of the request is larger than the MTU of the interface towards that next hop, the proxy MUST reject that request with a 516 response.

5.2.3 Forwarding of Responses

When any proxy supporting this extension forwards a request or response and there is a choice of transport protocols toward the next hop, the proxy SHOULD choose a transport protocol supporting congestion management if one is available.

When a proxy supporting this extension forwards a response containing a Proxy-Require header field with the option-tag "congestion-managed" as a value and the relevant Via header field value allows for a choice of transport protocols, the proxy MUST select a transport supporting congestion management if such a transport is available.

SIP provides no mechanism whereby a proxy may reject a response. Consequently, proxies may receive responses that require fragmentation over a transport not supporting congestion management. One example of a situation where this might be expected to occur is as follows: A UAC not supporting this extension makes a request via UDP. This request transits the proxy in question without inducing fragmentation. The responding UAS generates a response that is larger than the request. When the proxy prepares to send the request, it finds that the increase in size now requires fragmentation. Discarding the response would result in a timeout and retransmission of the request and response, thereby doing more harm than good. There seems to be nothing that the proxy can do to correct the situation, so it MUST forward the response as specified in [4].

5.3 Behavior at the UAS

A user agent server server (UAS) receiving a SIP request generates a response to that request. Delivery of this response may raise issues of congestion management. Because SIP requires that responses traverse exactly the reverse of the route taken by the request (recorded in the Via: header field values), the server has no options about routing the response. If the request was delivered in a congestion-managed manner, it is likely that the response will also be returned in a congestion-managed manner, as it must traverse exactly this recorded route. However, if the request was NOT received in a congestion-managed manner, the server cannot negotiate a congestion-managed path for the response, as the response must follow the path of the request.

When a UAS supporting this extension responds to a request over a route supporting congestion management (as indicated by the presence of the congestion-managed option tag in the request), the UAS MUST include the congestion-managed option tag in a "Proxy-Require" header field in the response. Furthermore, it MUST transmit that response using a protocol supporting congestion management. If it is unable to transmit the response using a protocol supporting congestion management, it MUST reject the request and return an error response using response code 515, which has the semantic of "Response requires congestion management."

When a UAS supporting this extension generates a response to a request that is larger than the UAS' understanding of path MTU and that request was not received over a congestion-managed route (as indicated by the presence of a "Require: congestion-managed"), it cannot be assumed that the response can be safely transmitted. As the UAS cannot respond safely, it SHOULD reject the request and return an error response using response code 515, which has the semantic of "Response requires congestion management". Note that this does not absolutely preclude fragmentation of the response, as the request may be fragmented by intervening routers. However, this sort of fragmentation is outside of the UAS' capacity to detect or control.

6. IANA Considerations

This document defines the SIP option tag "congestion-managed" which IANA will add to the registry of SIP option tags defined in [4].

This document defines the SIP response code 514, with the semantic "No congestion-managed route available" which IANA will add to the registry of SIP response codes defined in [4] in the section for 5xx clase response codes.

This document defines the SIP response code 515, with the semantic "Response requires congestion management" which IANA will add to the registry of SIP response codes defined in [4] in the section for 5xx clase response codes.

This document defines the SIP response code 516, with the semantic "Proxying of request would induce fragmentation" which IANA will add to the registry of SIP response codes defined in [4] in the section for 5xx clase response codes.

The following is the registration for the congestion-managed option tag:

RFC Number: RFCXXXX [Note to IANA: Fill in with the RFC number of this specification.]

Option Tag: congestion-managed

The following is the registration for the SIP response code 514:

RFC Number: RFCXXXX [Note to IANA: Fill in with the RFC number of this specification.]

Response Code: 514 No available route with congestion management

The following is the registration for the SIP response code 515:

RFC Number: RFCXXXX [Note to IANA: Fill in with the RFC number of this specification.]

Response Code: 515 Response requires congestion management

The following is the registration for the SIP response code 516:

RFC Number: RFCXXXX [Note to IANA: Fill in with the RFC number of this specification.]

Response Code: 516 Proxying of request would induce fragmentation

7. Acknowledgements

This document is a product of the SIP Working Group and contains input from many contributors in that group. The named authors of this document claim no personal contribution to the content excecpt as provided in their capacity as participants in the working group. Rather, they have attempted to act only in an editorial fashion, documenting the consensus of the working group as it emerged. Somebody had to do the typing.

Normative References

[1] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

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

[3] Postel, J. and J. Reynolds, "Instructions to RFC Authors", RFC 2223, October 1997.

[4] 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.

[5] Mankin, A., Bradner, S., Mahy, R., Willis, D., Ott, J. and B. Rosen, "Change Process for the Session Initiation Protocol (SIP)", BCP 67, RFC 3427, December 2002.

Authors' Addresses

Dean Willis
dynamicsoft Inc.
5100 Tennyson Parkway
Suite 1200
Plano, TX 75028
US

Phone: +1 972 473 5455
EMail: dean.willis@softarmor.com
URI: http://www.dynamicsoft.com/

Ben Campbell
dynamicsoft Inc.
5100 Tennyson Parkway
Suite 1200
Plano, TX 75028
US

Phone: +1 972 473 5452
EMail: bcampbell@dynamicsoft.com
URI: http://www.dynamicsoft.com/

Intellectual Property Statement

Full Copyright Statement

Acknowledgement

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