SIP Load Management
draft-sparks-sipping-load-00

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 16, 2004.

Copyright Notice

Copyright (C) The Internet Society (2003). All Rights Reserved.

Abstract

As deployments of services based on the Session Initiation Protocol increase in scope, managing load across the systems providing their services becomes increasingly important. This document looks at generic load management requirements and the tools to address them currently available in the protocol. Potential new tools to better meet the requirements are proposed.
Table of Contents

1. Problem statement ................................. 3
   1.1 Load contributors ............................ 3
1.2 Clustering ........................................ 3
2. Currently available SIP mechanisms ..................... 4
   2.1 Redirection ..................................... 4
   2.2 503 unavailability ............................ 4
2.3 REFER ............................................ 5
3. Proposals for new tools ............................... 5
   3.1 Throttling ...................................... 5
   3.2 Modifying route-sets ........................... 6
4. Recommendations ....................................... 6
   References ......................................... 6
   Author's Address .................................. 7
   Intellectual Property and Copyright Statements ....... 8
1. Problem statement

A SIP element, either an endpoint or an intermediary, has an upper bound on the volume of SIP traffic it can (or is willing to) process. When ambient demand exceeds that volume, the element and its SIP peers must behave such that

- The application constructed using SIP can continue to exhibit predictable behavior.
- The element continues to receive volume near, but not above, its limit.

When an element's capacity changes, either increasing or decreasing, the element and its SIP peers must behave such that existing volume toward the element is modified to be near, but not above, the new capacity. This behavior must encompass the special case of changing to a capacity of zero (allowing a controlled shutdown for example).

These behaviors might be achievable using mechanisms entirely outside SIP, but providing support from within the protocol may allow them to be realized with less effort. Some in-protocol support already exists.

This document discusses realizing those behaviors using proxies and PSTN gateways as concrete examples of intermediaries and endpoints to take advantage of the widespread familiarity with those elements. The discussion is applicable, however, to arbitrary SIP elements. The document also uses the terms "traffic volume" and "load" interchangeably.

1.1 Load contributors

Load can arrive at an element from one or multiple sources. These sources may vary the size of their contribution over time. The contribution can range from being evenly distributed across all sources to one source providing nearly the total load. A proxy could be processing traffic for thousands of directly connected endpoints, or could be isolated such that it only ever communicates with one or two neighboring proxies. Likewise, a gateway could have thousands of endpoints utilizing it directly, or could be isolated such that all SIP traffic comes through one proxy. The behaviors above must be effective given any configuration of contributors.

1.2 Clustering

Systems frequently provide higher capacity than is achievable with a single element by distributing load across a cluster of elements.
These systems must be able to distribute load across these elements in such a way that each element can operate near, but not above, its limit.

2. Currently available SIP mechanisms

2.1 Redirection

An element can redirect some types of request to a peer using a 3xx response. This can be used to redirect load. For instance, a gateway can redirect initial INVITEs to a peer, thus avoiding the load associated with the remainder of the messages in the dialog it might create. Only load directed toward the Request-URI of this request will be redirected, so this tool is most valuable when requests to this URI represent a significant fraction of the total load.

The duration of this redirection can be controlled with a 1 second granularity by using a 302 response with either an Expires header field, or an expires parameter on the Contact header field value.

Care needs to be taken in such systems to avoid redirect-loops. If A redirects to B and B redirects to A, a requester will consider the last redirection an unrecoverable failure.

This mechanism is only applicable to out-of-dialog requests. Behavior upon receiving a 300-class response in-dialog is not explicitly specified in [1], nor is there a prohibition against issuing a 300-class response in-dialog. The immutability of a dialog's route-set adds difficulty to making in-dialog redirection meaningful, particularly at a proxy.

2.2 503 unavailability

An element can squelch all traffic from a source by returning a 503 to any request from that source. Per [2] the peer will treat this response as a transport error and quit sending traffic over this transport, failing over to another through SRV. The duration of the squelch can be controlled with a 1 second granularity by providing a Retry-After header field in the response. This can be useful for controlling load when there are multiple contributors. Its all-or-nothing nature reduces its utility when the environment contains only one or two peers. It is particularly unhelpful for a gateway behind a single protecting proxy or a proxy in a pipeline with exactly one peer to either side.

Behavior on receipt of a 503 without a Retry-After header is underspecified. It is possible that emitting a 503 without a Retry-After could squelch all future traffic from a peer (until the

2.3 REFER

Load at an endpoint within an existing dialog can increase (or capacity may decrease) driving total load beyond the endpoint's capacity. Shedding non-dialog-related load may not be sufficient to bring total load below capacity. An endpoint can shed dialog related load by terminating the dialog (BYE to INVITE, unSUBSCRIBE or NOTIFY/Terminated for SUBSCRIBE or REFER related dialogs). If a system can preserve the value of the dialog to the SIP application by transferring the state-association related to the dialog to a peer endpoint, the endpoint can use REFER to do so before terminating the dialog locally. For instance, a gateway might transfer all calls currently using one of its trunks to a peer to allow that trunk to be taken out of service.

3. Proposals for new tools

3.1 Throttling

The all-or-nothing nature of 503 is too coarse. A tool that allowed traffic to continue to arrive from a peer, but at a reduced rate, would be useful. As a strawman, consider a new method "THROTTLE" that contains a "Gap:" header field with an integer value representing milliseconds. A peer accepting a THROTTLE agrees to send messages no closer than that number of milliseconds apart. This will allow an element with, say, one peer to continue to receive load near its capacity, whereas the 503 mechanism introduces bursts of no-load.

A request is proposed over a response for the following reasons:

o It is less error-prone to control the propagation distance of a request (using both the Request-URI and Max-Forwards). It also provides a feedback channel, in the form of the SIP response, that can carry information from the peer about its support for throttling.

o An element can be pro-active in managing load using this request before it reaches saturation.

Using an error response would require rejecting a request that was otherwise acceptable. It will be difficult to extend success responses to carry a throttle indicator and limit the effect of that indicator to the first hop (a proxy that didn't understand the indicator would forward it, resulting in throttling load to itself unintentionally).
3.2 Modifying route-sets

It is necessary to be able to take a proxy out of service without destroying the future of any existing dialogs that happen to have that proxy in their route-set. Currently, the only protocol-supported way to do this is to provide an alternate instantiation of that proxy via DNS SRV records. It would be useful for a proxy to indicate that it wants out of the path. Consider, for example, a proxy on the path to a presence server carrying thousands of active subscriptions. If this proxy had to be removed from service and another instance couldn't be provided through DNS, the system would have to endure thousands error detections and generation of new subscriptions. If notifications on the original dialogs are infrequent, this may mean the presence server will carry double its subscription load until the original dialogs expire.

It would be better if the proxy could either modify the route set inside this dialog or indicate to both ends that it wants this dialog torn down. The later could be achieved with something like a

Proxy-Desires: terminate-session

added to both the request and response of any transaction within the dialog.

4. Recommendations

If this brief analysis survives review, then we should

- Refine the 503 caching mechanics in RFC 3263
- Begin work on a throttling tool in the SIP working group
- Begin work on allowing a proxy to modify route sets or request dialog termination in the SIP working group
- Capture the results in this document and expand it into an informational document recommending load management practices that involve using SIP mechanisms.

References


draft-sparks-sip-noninvite-01 (work in progress), October 2003.

Author's Address

Robert J. Sparks
dynamicsoft
5100 Tennyson Parkway
Suite 1200
Plano, TX 75024

EMail: rsparks@dynamicsoft.com
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any intellectual property 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; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication 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 implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

Full Copyright Statement

Copyright (C) The Internet Society (2003). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assignees.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION
Acknowledgment

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