Essential Correction to the Session Initiation Protocol (SIP) 503 (Service Unavailable) Response
draft-hilt-sip-correction-503-00

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

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 November 5, 2007.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

Overload occurs in the Session Initiation Protocol (SIP) when SIP servers have insufficient resources to process all SIP messages they receive. The SIP protocol specified in RFC 3261 provides the 503 (Service Unavailable) response code as a remedy for servers under overload. However, the current definition of 503 (Service Unavailable) has problems and can in fact amplify an overload condition. This document proposes an essential correction to RFC 3261 that avoids these problems and helps SIP servers to better cope
with overload.

Table of Contents

1. Introduction .................................................... 3
2. Terminology ...................................................... 3
3. Reason for Change ............................................... 4
4. Summary of Change ............................................... 4
5. Consequences if not approved ................................. 6
6. The Change ....................................................... 6
   6.1. 503 Service Unavailable ................................. 6
7. Security Considerations ........................................ 7
8. IANA Considerations ............................................. 7
Appendix A. Acknowledgements ................................... 7
9. References ....................................................... 7
   9.1. Normative References ..................................... 7
   9.2. Informative References .................................... 7
Author’s Address ................................................... 8
Intellectual Property and Copyright Statements .................. 9

Hilt                    Expires November 5, 2007                [Page 2]
1. Introduction

As with any network element, a Session Initiation Protocol (SIP) server can suffer from overload when the number of SIP messages it receives exceeds the number of SIP messages it can process. Generally, a SIP server is overloaded when it does not have sufficient resources to process all incoming SIP messages.

RFC3261 defines the 503 (Service Unavailable) response code to enable servers to handle temporary overload as follows:

The server is temporarily unable to process the request due to a temporary overloading or maintenance of the server. The server MAY indicate when the client should retry the request in a Retry-After header field. If no Retry-After is given, the client MUST act as if it had received a 500 (Server Internal Error) response.

A client (proxy or UAC) receiving a 503 (Service Unavailable) SHOULD attempt to forward the request to an alternate server. It SHOULD NOT forward any other requests to that server for the duration specified in the Retry-After header field, if present.

Servers MAY refuse the connection or drop the request instead of responding with 503 (Service Unavailable).

Unfortunately, this mechanism has proven to be problematic in actual deployments. Problems observed include load amplification, server underutilization, off/on semantics and ambiguous usages, which can eventually lead to a congestion collapse [todo: cite design team simulation work].

This specification proposes an essential correction to RFC3261 following the process defined in [4]. The specification does not attempt to provide a complete solution for SIP overload control. Such a solution is left for further study.

Section 3 describes the specific problems identified in RFC3261. Section 4 and Section 6 introduce the proposed change and Section 5 discusses the consequences if this change is not approved.

2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in BCP 14, RFC 2119 and indicate requirement levels for compliant implementations.
3. Reason for Change

The current specification of 503 (Service Unavailable) responses has proven to be problematic [5] and is ineffective in helping a SIP server to handle overload in many cases [todo: cite overload design team simulation work]. The use of 503 (Service Unavailable) responses can in fact worsen an overload condition and may lead to congestion collapse; a condition in which the message throughput of a server drops to a small fraction of its capacity.

The following specific mechanisms defined for 503 (Service Unavailable) responses contribute to overall problems:

Retry-After: A server can insert a Retry-After header into a 503 (Service Unavailable) response to tell its client to stop sending traffic for the given period of time. After this time is over, the client may start sending again. This mechanism causes problems when used by a server that receives requests from a small number of clients (e.g. a SIP proxy that receives requests from a few other SIP proxies). It leads to traffic oscillation and shifts load between alternate servers [5]. However, this mechanism does work well for servers that have large client population.

Alternate server forwarding: After receiving a 503 (Service Unavailable) response, a client can send the request to an alternate server, if available. This mechanism amplifies load since clients send each request to all alternative servers before a request is eventually rejected [5].

Dropping requests: A server is allowed to drop requests or refuse connections instead of sending a 503 (Service Unavailable) response. Requests that do not receive a response will eventually be retransmitted by the client, which again amplifies load during periods of overload.

Blocking hostnames: A client that has received a 503 (Service Unavailable) response with a Retry-After header may decide to stop forwarding traffic to this server based on the servers hostname. However, if this hostname represents a cluster of servers (e.g. via a DNS mapping), the client would block traffic to all servers. The other servers would then be underutilized [5].

4. Summary of Change

The following changes are proposed:
1. Deprecate the use of Retry-After headers in 503 (Service Unavailable) responses except for servers with a large (> 20) client population (e.g. edge proxies). Proxies that create a 500 (Server Internal Error) response after receiving a 503 (Service Unavailable) MAY include a Retry-After header in the 500 (Server Internal Error) response to prevent the UAC from instantly retrying the request.

2. Deprecate forwarding a request to alternate servers after receiving a 503 (Service Unavailable) response.

3. Change dropping requests or refusing the connection as a replacement for sending a 503 (Service Unavailable) response from MAY to SHOULD NOT.

4. Recommend the use of IP addresses for blocking traffic after receiving a 503 (Service Unavailable) with Retry-After and not the hostname.

The following proposals are an alternative to 1. and 4. above:

5. Limit scope of 503 (Service Unavailable) with Retry-After header to one request only. The Retry-After header only affects a single request and clients keep forwarding other requests after receiving a 503 (Service Unavailable) with Retry-After header to a server.

6. Deprecate conversion of 503 (Service Unavailable) to 500 (Server Internal Error) by proxies. It is no longer needed since 503 (Service Unavailable) only affects one request.

OPEN ISSUE 1: is proposal 1./4. or 5./6. preferable? Proposal 5./6. seems cleaner and simpler than 1./4. It is independent of the number of clients a server has. It also enables an overloaded proxy to set a Retry-After header for each request that is passed all the way to the UAC. However, proposal 5./6. requires a change in the semantics of 503 (Service Unavailable) responses. It is therefore not fully backwards compatible without an indicator for its support.

OPEN ISSUE 2: is 20 a good delineation for the use of 503 (Service Unavailable) in proposal 1.?

OPEN ISSUE 3: RFC 3261 [2] and RFC 3263 [3] define that transport failures (generally, due to fatal ICMP errors in UDP or connection failures in TCP) should be treated as a 503 (Service Unavailable) response. Also, 503 (Service Unavailable) is currently recommended for servers that are offline for maintenance. This mixes overload control and other failure cases. Should all non-overload failures be changed to 500 (Server Internal Error)? It would, e.g., enable a proxy to try alternate servers for all non-overload failure cases.
5. Consequences if not approved

Without these changes, networks of SIP servers are vulnerable to overload and, in the worst case, congestion collapse. A network of SIP servers can be significantly impacted by overload due to the problems described above.

While the proposed changes do not provide a full solution for overload control and cannot always prevent a congestion collapse, they avoid the problems described above and improve SIP server performance under overload.

6. The Change

The following sentence is added to the end of the paragraph starting with "A proxy which receives..." on top of page 110 (Section 16.7 step 6.) in RFC3261 [2]:

It MAY indicate when the client should retry the request in a Retry-After header field added to the 500 (Server Internal Error) response.

The following section replaces Section 21.5.4 in RFC3261 [2].

6.1. 503 Service Unavailable

The server is temporarily unable to process the request due to a temporary overloading of the server.

A server that is temporarily overloaded SHOULD reject those requests that exceed its processing capacity with 503 (Service Unavailable) responses.

Servers with a large population of clients (proxies or UACs) MAY indicate when the client should retry the request in a Retry-After header field. Servers that fall into this category typically receive traffic from 20 or more (often much more) clients. An example for such a server is an edge proxy. All other servers SHOULD NOT include a Retry-After header in a 503 (Service Unavailable) response. If no Retry-After is given, a client MUST act as if it had received a 500 (Server Internal Error) response.

A client (proxy or UAC) receiving a 503 (Service Unavailable) SHOULD NOT attempt to forward the request to an alternate server. Forwarding the request to alternate servers would increase the load on all servers and thereby amplify an overload condition.
If the Retry-After header field is present in a 503 (Service Unavailable) response, the client SHOULD NOT forward any other requests to that server for the duration specified in the Retry-After header field. The client SHOULD block traffic to a server based on the servers IP address and not the hostname since hostnames can represent multiple servers.

Servers SHOULD NOT refuse the connection or drop the request as a replacement for responding with 503 (Service Unavailable).

7. Security Considerations

The procedures introduced in this document have no security implications beyond what is already specified in RFC3261 [2].

8. IANA Considerations

None.

Appendix A. Acknowledgements

Many thanks to Jonathan Rosenberg and Keith Drage for their suggestions. A big thanks to Indra Widjaja, Eric Noel, Carolyn Johnson, Ping Wu, Tadeusz Drwiega and the overload control design team for simulation results.

9. References

9.1. Normative References


9.2. Informative References


draft-drage-sip-essential-correction-01 (work in progress),
March 2007.

[5] Rosenberg, J., "Requirements for Management of Overload in the
Session Initiation Protocol",
draft-rosenberg-sipping-overload-reqs-02 (work in progress),
October 2006.

Author’s Address

Volker Hilt
Bell Labs/Alcatel-Lucent
101 Crawfords Corner Rd
Holmdel, NJ  07733
USA

Email: volkerh@bell-labs.com