SIP Event Package for Message Waiting Indication

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 document is an individual submission to the IETF. Comments should be directed to the authors.

1. Abstract

This draft proposes an event package for [SIP], which uses the SUBSCRIBE and NOTIFY methods [SUB/NTFY] to carry message waiting status and message summaries from a messaging system to an interested User Agent.

2. Conventions used in this document

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. Background and Overview

Messaging Waiting Indication is a common feature of telephone networks. It typically involves playing a special dial tone (called message-waiting dial tone), lighting a light or indicator on the phone, or both. Message-waiting dial tone is similar but distinct from stutter dial tone. Both are defined in [GR506].
The methods in SIP 2.0, as defined in RFC2543, were only designed to solve the problem of session initiation for multimedia sessions, and rendezvous. Since Message Waiting Indication is really status information orthogonal to a session, it was not clear how an IP telephone acting as a SIP User Agent would implement comparable functionality. Members of the telephony community viewed this as a shortcoming of SIP.

Users want the useful parts of the functionality they had using traditional analog and PBX telephones. It is also desirable to provide comparable functionality in a flexible way that allows for more customization and new features.

Using the SUBSCRIBE and NOTIFY methods, a Subscriber User Agent (typically an IP phone or SIP soft-phone) subscribes to the status of their messages. A SIP User Agent acting on behalf of the user’s messaging system then notifies the Subscriber whenever the account’s messages have changed. The Notifier sends this message summary information in the body of the NOTIFY, encoded in a new MIME type defined later in this draft. A User Agent can also explicitly fetch the current status.

A SIP User Agent MAY subscribe to multiple accounts (distinguished by the URL in the To header). Multiple SIP User Agents MAY subscribe to the same account.

Before any subscriptions or notifications are sent, each interested User Agents must be made aware of its messaging server(s). This MAY be manually configured on the User Agents, manually configured in the proxies, or dynamically discovered using caller preferences [Caller-Prefs]. (For more information on usage with caller preferences, see section 4.2)

6. Simple Message Summary Format

A simple text-based format is proposed to prevent an undue burden on low-end user agents, for example, inexpensive IP phones with no display. Although this format is text-based, it is intended for machine consumption only.

The format specified in this proposal attempt to separate orthogonal attributes of messages as much as possible. Messages are separated by media type (audio, text, image, and video); by message status (new and old); and by urgent and non-urgent type.

The text format begins with a simple status line, and optionally a summary line per media type. Valid media types are Voicemail (audio), Email (text), Fax (image), and Video (video). For each media type the total number of new and old messages is reported in the new and old fields.
message-summary = status-line {status-line} [mheaders]
status-line = "Messages-Waiting: " status CRLF
summary-line = media-type ": " new "/" old [urgent] CRLF
urgent = "(" new-urgent "/" old-urgent ")"
status = "yes" | "no"

In some cases, detailed message summaries are not available. The status line allows messaging systems or messaging gateways to provide the traditional boolean message waiting notification.

   Messages-Waiting: yes

In the example that follows, more functionality is available to User Agent. There are one new and three old voice messages.

   Voicemail: 1/3

After the summary, the format can optionally list a summary count of urgent messages. Of the one new and three old voice messages, none of the new messages are urgent, but one of the old messages is.

   Voicemail: 1/3 (0/1)

Optionally, after the summary counts, the messaging systems MAY append message headers, which further describe newly added messages. A messaging system which includes message headers in a NOTIFY, MUST provide an administrator configurable mechanism for selecting which headers are sent. Likely headers for inclusion include To, From, Date, Subject, Message-ID, and Priority. Note that the syntax for these headers is more restrictive than for [SMTP].


For SMTP encoded messages, messaging systems SHOULD follow the recommendations of the VPIM Working Group using either MIME type information of [Message-Context] to map messages to a specific media type. Systems using other encodings MAY define their own mappings.

4. Examples of Usage

4.1 Example Message Flow

In the example call flow below, Rohan’s IP phone subscribes to the status of Rohan’s messages. Via headers are omitted for clarity.

    Subscriber       | Notifier
                       | Mahy, Slain
A1: SUBSCRIBE (new)  
----------------------> 
A2: 200 OK 
<---------------------- 
A3: NOTIFY (sync) 
<---------------------- 
A4: 200 OK 
----------------------> 
A5: NOTIFY (change) 
<---------------------- 
A6: 200 OK 
----------------------> 
A7: (re)SUBSCRIBE 
----------------------> 
A8: 200 OK 
<---------------------- 
A9: NOTIFY (sync) 
<---------------------- 
A10: 200 OK 
----------------------> 
A11: (un)SUBSCRIBE 
----------------------> 
A12: 200 OK 
<---------------------- 
A13: NOTIFY (sync) 
<---------------------- 
A14: 200 OK 
----------------------> 

A1: Subscriber (Rohan’s phone) -> Notifier (Rohan’s voicemail gateway)  
Subscribe to Rohan’s message summary status for 1 day.  

SUBSCRIBE sip:rohan@vmail.cisco.com SIP/2.0  
To: <sip:rohan@cisco.com>  
From: <sip:rohan@cisco.com>;tag=78923  
Date: Mon, 10 Jul 2000 03:55:06 GMT  
Call-Id: 1349882@rmahy-phone.cisco.com  
CSeq: 4 SUBSCRIBE  
Contact: <sip:rohan@rmahy-phone.cisco.com>  
Event: simple-message-summary  
Expires: 86400
Accept: application/simple-message-summary  
Content-Length: 0

A2: Notifier -> Subscriber

SIP/2.0 200 OK  
To: <sip:rohan@cisco.com>  
From: <sip:rohan@cisco.com>;tag=78923  
Date: Mon, 10 Jul 2000 03:55:07 GMT  
Call-Id: 1349882@rmahy-phone.cisco.com  
CSeq: 4 SUBSCRIBE  
Event: simple-message-summary  
Expires: 86400  
Content-Length: 0

A3: Notifier -> Subscriber (immediate synchronization of current state: 2 new and 8 old [2 urgent] messages)

NOTIFY sip:rohan@rmahy-phone.cisco.com SIP/2.0  
To: <sip:rohan@cisco.com>  
From: <sip:rohan@cisco.com>;tag=78923  
Date: Mon, 10 Jul 2000 03:55:07 GMT  
Call-Id: 1349882@rmahy-phone.cisco.com  
CSeq: 20 NOTIFY  
Contact: <sip:rohan@vmail.cisco.com>  
Event: simple-message-summary  
Content-Type: application/simple-message-summary  
Content-Length: 45

Messages-Waiting: yes  
Voicemail: 2/8 (0/2)

A4: Subscriber -> Notifier

SIP/2.0 200 OK  
To: <sip:rohan@cisco.com>  
From: <sip:rohan@cisco.com>;tag=78923  
Date: Mon, 10 Jul 2000 03:55:08 GMT  
Call-Id: 1349882@rmahy-phone.cisco.com  
CSeq: 20 NOTIFY  
Event: simple-message-summary

A5: Notifier -> Subscriber

This is a notification of new messages. Some headers from the new messages are appended.

NOTIFY sip:rohan@rmahy-phone.cisco.com SIP/2.0  
To: <sip:rohan@cisco.com>  
From: <sip:rohan@cisco.com>;tag=78923  
Date: Mon, 10 Jul 2000 04:28:53 GMT  
Contact: <sip:rohan@vmail.cisco.com>  
Call-ID: 1349882@rmahy-phone.cisco.com
CSeq: 31 NOTIFY
Event: simple-message-summary
Content-Type: application/simple-message-summary
Content-Length: 420

Messages-Waiting: yes
Voicemail: 4/8 (1/2)

To: <rohan@work.com>
From: <friend@home.com>
Subject: carpool tomorrow?
Date: Sun, 09 Jul 2000 21:23:01 -0700
Priority: normal
Message-ID: 13784434989@vmail.cisco.com

To: <rohan@work.com>
From: <the-boss@work.com>
Subject: HELP! at home ill, present for me please
Date: Sun, 09 Jul 2000 21:25:12 -0700
Priority: urgent
Message-ID: 13684434990@vmail.cisco.com

A6: Subscriber -> Notifier

SIP/2.0 200 OK
To: <sip:rohan@cisco.com>
From: <sip:rohan@cisco.com>;tag=78923
Date: Mon, 10 Jul 2000 04:28:53 GMT
Call-ID: 1349882@rmahy-phone.cisco.com
CSeq: 31 NOTIFY
Event: simple-message-summary
Content-Length: 0

A7: Subscriber -> Notifier
Refresh subscription.

SUBSCRIBE sip:rohan@vmail.cisco.com SIP/2.0
To: <sip:rohan@cisco.com>
From: <sip:rohan@cisco.com>;tag=78923
Date: Mon, 10 Jul 2000 15:55:06 GMT
Call-ID: 1349882@rmahy-phone.cisco.com
CSeq: 8 SUBSCRIBE
Contact: <sip:rohan@rmahy-phone.cisco.com>
Event: simple-message-summary
Expires: 86400
Accept: application/simple-message-summary
Content-Length: 0

A8: Notifier -> Subscriber

SIP/2.0 200 OK
To: <sip:rohan@cisco.com>
From: <sip:rohan@cisco.com>;tag=78923
Date: Mon, 10 Jul 2000 15:55:07 GMT
Call-Id: 1349882@rmahy-phone.cisco.com
CSeq: 8 SUBSCRIBE
Contact: <sip:rohan@rmahy-phone.cisco.com>
Event: simple-message-summary
Expires: 86400
Content-Length: 0

A9: Notifier -> Subscriber (immediate synchronization of current state)

NOTIFY sip:rohan@rmahy-phone.cisco.com SIP/2.0
To: <sip:rohan@cisco.com>
From: <sip:rohan@cisco.com>;tag=78923
Date: Mon, 10 Jul 2000 15:55:07 GMT
Call-Id: 1349882@rmahy-phone.cisco.com
CSeq: 47 NOTIFY
Contact: <sip:rohan@vmail.cisco.com>
Event: simple-message-summary
Content-Type: application/simple-message-summary
Content-Length: 45

Messages-Waiting: yes
Voicemail: 4/8 (1/2)

A10: Subscriber -> Notifier

SIP/2.0 200 OK
To: <sip:rohan@cisco.com>
From: <sip:rohan@cisco.com>;tag=78923
Date: Mon, 10 Jul 2000 15:55:08 GMT
Call-Id: 1349882@rmahy-phone.cisco.com
CSeq: 47 NOTIFY
Contact: <sip:rohan@vmail.cisco.com>
Event: simple-message-summary

A11: Subscriber -> Notifier
Un-subscribe after "rohan" logs out.

SUBSCRIBE sip:rohan@vmail.cisco.com SIP/2.0
To: <sip:rohan@cisco.com>
From: <sip:rohan@cisco.com>;tag=78923
Date: Mon, 10 Jul 2000 19:35:06 GMT
Call-Id: 1349882@rmahy-phone.cisco.com
CSeq: 17 SUBSCRIBE
Contact: <sip:rohan@rmahy-phone.cisco.com>
Event: simple-message-summary
Expires: 0
Accept: application/simple-message-summary
Content-Length: 0
A12: Notifier -> Subscriber

SIP/2.0 200 OK
To: <sip:rohan@cisco.com>
From: <sip:rohan@cisco.com>;tag=78923
Date: Mon, 10 Jul 2000 19:35:07 GMT
Call-Id: 1349882@rmahy-phone.cisco.com
CSeq: 17 SUBSCRIBE
Contact: <sip:rohan@rmahy-phone.cisco.com>
Event: simple-message-summary
Expires: 0
Content-Length: 0

A13: Notifier -> Subscriber (immediate synchronization of current state, which the subscriber can now ignore)

NOTIFY sip:rohan@rmahy-phone.cisco.com SIP/2.0
To: <sip:rohan@cisco.com>
From: <sip:rohan@cisco.com>;tag=78923
Date: Mon, 10 Jul 2000 19:35:07 GMT
Call-Id: 1349882@rmahy-phone.cisco.com
CSeq: 56 NOTIFY
Contact: <sip:rohan@vmail.cisco.com>
Event: simple-message-summary
Content-Type: application/simple-message-summary
Content-Length: 45

Messages-Waiting: yes
Voicemail: 4/8 (1/2)

A10: Subscriber -> Notifier

SIP/2.0 200 OK
To: <sip:rohan@cisco.com>
From: <sip:rohan@cisco.com>;tag=78923
Date: Mon, 10 Jul 2000 19:35:08 GMT
Call-Id: 1349882@rmahy-phone.cisco.com
CSeq: 56 NOTIFY
Event: simple-message-summary
Content-Length: 0

4.2 Use with Caller Preferences

The use of caller preferences is purely optional but its use is encouraged. If caller preferences is used, a messaging server MAY REGISTER a Contact with a "SUBSCRIBE" methods tag. If SUBSCRIBE is used by other services, the messaging server MAY also REGISTER as a Contact with the feature="voice-mail" tag. An example of this kind of registration follows below.
The following SUBSCRIBE message would find the Contact which registered in the example above.

```
SUBSCRIBE sip:rohan@cisco.com SIP/2.0
...
Accept: application/simple-message-summary
Request-Contact: *;feature="voice-mail";methods="SUBSCRIBE"
```

5. Specific Use of SUBSCRIBE and NOTIFY

5.1 Behavior of Subscriber (Ordinary User Agent)

Subscriber User Agents will typically SUBSCRIBE to message summary information for a period of hours or days, and automatically attempt to re-SUBSCRIBE when the subscription is half-expired. If re-subscription fails, the Subscriber SHOULD periodically retry again (for example, after the subscription is 66%, 75%, 80%, 85%, 90%, 95%, and 99% expired) until a subscription is successful. The example retry sequence is similar to that used by [DHCP] Clients trying to renew a lease. If a subscription has expired, the Subscriber MAY attempt to renew the expired subscription in intervals of about 10% of the expires period of the original or requested subscription. These new subscriptions MUST use a new Call-ID.

The Subscriber SHOULD SUBSCRIBE to that user’s message summaries whenever a new user becomes associated with the device (a new login). The Subscriber MAY also explicitly fetch the current status at any time. The subscriber SHOULD renew its subscription immediately after a reboot, or when the subscriber’s network connectivity has just been re-established.

The Subscriber MUST be prepared to receive and process a NOTIFY with new state immediately after sending a new SUBSCRIBE, a SUBSCRIBE renewal, an unSUBSCRIBE or a fetch; or at any time during the subscription.

When a user de-registers from a device (logoff, power down of a mobile device, etc.), subscribers SHOULD unsubscribe by sending a SUBSCRIBE message with an Expires header of zero.
A Subscriber MAY use an "alias" or "group" in the To header and/or Request-URI if that name is significant to the messaging system. User Agents SHOULD tolerate non-SIP URI schemes in the To and From fields.

A Subscriber MUST be prepared to receive NOTIFYs from different Contacts corresponding to the same SUBSCRIBE. (the SUBSCRIBE may have been forked).

Implementers MAY create a service which consolidates and summarizes NOTIFYs from many Contacts.

5.3 Behavior of the Notifier (Messaging System)

When a SIP Messaging System receives SUBSCRIBE messages with the message-summary event-type, it obeys the following rules.

Immediately after a subscription is accepted, the Notifier MUST send a NOTIFY with the current message summary information. This allows the Subscriber to resynchronize its state. This initial synchronization NOTIFY MUST NOT include the optional message headers.

When the status of the messages changes sufficiently for a messaging account to change the number of new or old messages, the Notifier SHOULD send a NOTIFY message to all active subscribers to that account. NOTIFY messages sent to subscribers of a group or alias, MUST place the account name in the user part of the Contact header URL.

A Notifier MAY choose to buffer NOTIFY responses for a short administrator-defined period (seconds or minutes) when the message status is changing rapidly. Note that timely notification of a newly added message is probably more significant to the end user than notifications of newly deleted messages.

5.4 Behavior of a Proxy Server

There are no additional requirements on the Proxy, other than to transparently forward the SUBSCRIBE and NOTIFY methods. However, Proxies SHOULD allow non-SIP URLs. Proxies and Redirect servers SHOULD be able to direct the SUBSCRIBE request to an appropriate messaging server User Agent. Proxies SHOULD support routing to Contacts based on the existence of methods="SUBSCRIBE" and feature="voice-mail" parameters in a Request-Contact header (as specified in the caller preferences specification).
6. Formal Syntax

The following syntax specifications use the augmented Backus-Naur Form [BNF] as described in RFC-2234.

6.1 New event-type definition

The syntax of the Event header is defined in SUBSCRIBE/NOTIFY as shown below.

```
Event = "Event": SP 1#event-type
event-type = token
```

This document amends that definition as shown below.

```
event-type = message-event | token
message-event = "simple-message-summary"
```

6.2 Body Format Syntax

The formal syntax for application/simple-message-summary is below:

```
messsage-summary = status-line *(summary-line) [ mheaders ]
status-line = "Messages-Waiting: " status CRLF
summary-line = media-type "": SP new "/" old [ urgent ] CRLF
status = "yes" | "no"
media-type = "Voicemail" | "Email" | "Fax" | "Video"

mheaders = *(CRLF message-headers)
message-headers = hname "": SP hvalue CRLF
hname = #( alphanum | "-" | "_" )
hvalue = <Any [UTF-8] except CTL>
alphanum = lowalpha | upalpha | DIGIT

new = WHOLENUMBER
old = WHOLENUMBER
new-urgent = WHOLENUMBER
old-urgent = WHOLENUMBER
WHOLENUMBER = *DIGIT
DIGIT = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
upalpha = "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" | "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" |
         "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"
lowalpha = "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" | "j" |
           "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" | "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z"

CTL = %0x00-1f | %0x7f ; (0-31 or 127)
CRLF = %d13 %d10
SP = %d32
```
7. IANA Considerations

7.1 MIME Registration for application/simple-message-summary

MIME media type name: application
MIME subtype name: simple-message-summary
Required parameters: none.
Optional parameters: none.
Encoding considerations: This type is only defined for transfer via [SIP].
Security considerations: See the "Security Considerations" (Section 8) section in this document.
Interoperability considerations: none
Published specification: This document.
Applications which use this media: The simple-message-summary application subtype supports the exchange of message waiting and message summary information in SIP networks.

Additional information:
1. Magic number(s): N/A
2. File extension(s): N/A
3. Macintosh file type code: N/A

8. Security Considerations

At a minimum, SUBSCRIPTION requests for message waiting, and message waiting NOTIFY messages SHOULD be authenticated. Because of the potential privacy implications, encrypting NOTIFY messages is encouraged to keep potentially sensitive information private. Additionally, SUBSCRIPTION requests MAY be encrypted. Hop-by-hop encryption could discourage unauthorized disclosure of the address or location of the user’s account.

9. Changes from -00

This draft greatly simplifies and shortens the -00 version.

1) The generic behavior of SUBSCRIBE/NOTIFY is now greatly clarified in [SUB/NTFY] and made consistent with PINT and SIP for presence. This message waiting draft is now consistent with [SUB/NTFY].
2) The XML format has been removed due to lack of immediate interest. At a future date, similar functionality may be added as another event package with an appropriate MIME type.

3) An IANA Considerations section was added to register the new "application/simple-message-summary" MIME type and the "simple-message-summary" SIP event package.

4) The "flag-list" was removed due to lack of interest and to encourage simplicity.

5) Due to synchronization issues, and the recommendation of the VPIM Working Group, support for message count "deltas" was removed.

6) The Messages-Waiting line in the body is now mandatory.

7) This version of the draft clarifies the role of caller preferences as optional but encouraged.

8) A set of SMTP-like headers from the triggering messages may now optionally follow the message summaries, provided that the resulting NOTIFY on UDP fits in a single datagram.

10. References


[GR506] "GR-506: Signaling for Analog Interfaces, Issue 1, Revision 1", Telecordia, November 1996.


Acknowledgments

Thanks to Dave Oran, Bill Foster, Steve Levy, Denise Caballero-McCann, Jeff Michel, Priti Patil, Satyender Khatte, Bich Nguyen, Manoj Bhatia, David Williams, and Bryan Byerly of Cisco; Jonathan Rosenberg of Dynamicsoft; and Adam Roach of Ericsson.

Author’s Addresses

Rohan Mahy
Cisco Systems
170 W Tasman Dr, MS: SJC-I/2/3
San Jose, CA 95134 USA
Phone: +1 408 526 4000
Email: rohan@cisco.com

Ilya Slain
Cisco Systems
170 W Tasman Dr, MS: SJC09/2/2
San Jose, CA 95134 USA
Phone: +1 408 527 9446
Email: islain@cisco.com
Full Copyright Statement

"Copyright (C) The Internet Society (date). 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 assigns. 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 HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE."