

Network Working Group
Request for Comments: 4411
Category: Standards Track

J. Polk
Cisco Systems
February 2006

Extending the Session Initiation Protocol (SIP)
Reason Header for Preemption Events

Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2006).

Abstract

This document proposes an IANA Registration extension to the Session Initiation Protocol (SIP) Reason Header to be included in a BYE Method Request as a result of a session preemption event, either at a user agent (UA), or somewhere in the network involving a reservation-based protocol such as the Resource ReSerVation Protocol (RSVP) or Next Steps in Signaling (NSIS). This document does not attempt to address routers failing in the packet path; instead, it addresses a deliberate tear down of a flow between UAs, and informs the terminated UA(s) with an indication of what occurred.

Table of Contents

1. Introduction	2
1.1. Conventions Used in This Document	4
2. Access Preemption Events	4
2.1. Effects of Preemption at the User Agent	6
2.2. Reason Header Requirements for Access Preemption Events	6
3. Network Preemption Events	7
3.1. Reason Header Requirements for Network Preemption Events ..	10
4. Including a Hybrid Infrastructure	10
4.1. Hybrid Infrastructure Requirements	11
5. Preemption Reason Header Cause Codes and Semantics	11
5.1. Access Preemption Event Reason Code	12
5.1.1. Access Preemption Event Call Flow	12
5.2. Network Preemption Events Reason Code	14
5.2.1. Network Preemption Event Call Flow	15
5.3. Generic Preemption Event Reason Code	16
5.4. Non-IP Preemption Event Reason Code	16
5.4.1. Non-IP Preemption Event Call Flow	17
6. Security Considerations	17
7. IANA Considerations	17
7.1. "Preemption" Namespace Registry	18
7.2. Default Reason-Text IANA Registry for the SIP Reason Header	20
8. Contributions	20
9. Acknowledgements	20
10. References	21
10.1. Normative References	21
10.2. Informative References	21

1. Introduction

With the introduction of the SIP Resource-Priority (R-P) header [4], there became the possibility of sessions being torn down for (scarce) resource reasons, meaning there weren't enough resources for a particular session to continue. Certain domains will implement this mechanism where resources may become constrained either at the user agent (UA) or at congested router interfaces where more important sessions are to be completed at the expense of less important sessions. Which sessions are more or less important than others will not be discussed here. What is proposed here is a SIP [2] extension to synchronize SIP elements as to why a preemption event occurred and which type of preemption event occurred, as viewed by the element that performed the preemption of a session.

The SIP Reason Header is an application layer feedback mechanism to synchronize SIP elements of events; the particular event explained here deals with preemption of a session. Q.850 [5] provides an

indication for preemption (cause=8) and for preemption "circuit reserved for reuse" (cause=9). Q.850 Cause=9 does not apply to IP, as IP has no concept of circuits. Some domains wish to differentiate appropriate IP reasons for preemption of sessions and to indicate topologically where the preemption event occurred. No other means exists today to give feedback as to why a session was torn down on preemption grounds.

In the event that a session is terminated for a specific reason that can (or should) be shared with SIP Servers and UAs sharing dialog, the Reason Header [1] was created to be included in the BYE Request. This was not the only Method for this new Header; [1] also discusses the CANCEL Method usage.

This document will define two use cases in which new preemption Reason values are necessary:

Access Preemption Event - This is when a UA receives a new SIP session request message with a valid R-P value that is higher than the one associated with the currently active session at that UA. The UA must discontinue the existing session in order to accept the new one (according to local policy of some domains).

Network Preemption Event - This is when a network element - such as a router - reaches capacity on a particular interface and has the ability to statefully choose which session(s) will remain active when a new session/reservation is signaled for under the parameters outlined in SIP Preconditions per [3] that would otherwise overload that interface (perhaps adversely affecting all sessions). In this case, the router must terminate one or more reservations of lower priority in order to allow this higher priority reservation access to the requested amount of bandwidth (according to local policy of some domains).

This document will cover the semantics for these two cases and request IANA registration of the new protocol value "Preemption" for the Reason Header field, with 4 cause values for the above preemption conditions. Additionally, this document will create a new IANA Registry for reason-text strings that are not currently defined through existing SIP Response codes or Q.850 cause codes. This new Registry will be useful for future protocols used by the SIP Reason header.

This document will emphasize an existing SIP RFC [3] as the starting point for network preemption events. RFC 3312 set rules surrounding SIP interaction using a reservation protocol for QoS preconditions,

using RSVP as the example protocol. That effort did not preclude other preconditions or future protocol work from becoming a means of preconditions. NSIS is a new reservation protocol effort that specifies a preemption operation similar to RSVP's ResvErr message involving the NSIS NOTIFY message in [8] with a Transient error code 0x04000005 (Resources Pre-empted).

Note that SIP itself does not cause RSVP or NSIS reservation signaling to start or end. That operation is part of a separate API within each UA.

1.1. 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 [6].

2. Access Preemption Events

As mentioned previously, Access Preemption Events (APE) occur at the user agent. It does not matter which UA in a unicast or multicast session this happens to (the UAC or UAS of a session). If local policy dictates in a particular domain rules regarding the functionality of a UA, there must be a means by which that UA (not the user) informs the other UA(s) why a session was just torn down prematurely. The appropriate mechanism is the BYE Method. The user of the other far side UA will not understand why that session "just went away" without there being a means of informing the UA of what occurred (if this event was purposeful). Through this type of indication to the preempted UA, it can indicate to the user of that device appropriately.

The rules within a domain surrounding the UA to be informed can be different from the rules for informing the user. Local policy should determine if the user should be informed of the specific reason. This indication in SIP will provide a means for the UA to react in a locally determined way, if appropriate (play a certain tone or tone sequence, point towards a special announcement uri, cause the UA's visual display to do something, etc.).

Figure 1 illustrates the scenario. UA1 invites UA2 to a session with the Resource Priority level of 3 (levels 1 and 2 are higher in this domain, and the namespace element is not necessary for this discussion).

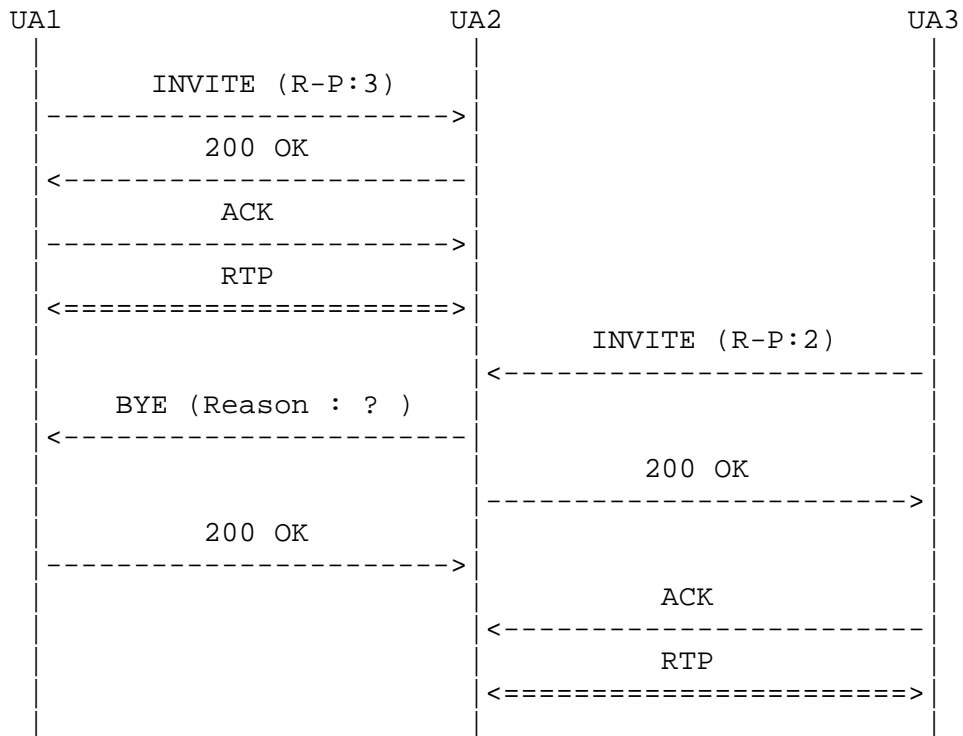


Figure 1. Access Preemption with obscure Reason

After the session between UA1 and UA2 is established, UA3 invites UA2 to a new session with an R-P of 2 (a higher priority than the current session between UA1 and UA2). Local policy within this domain dictates that UA2 must preempt all existing calls of lower priority in order to accept a higher priority call.

What Reason value could be inserted above to mean "preemption" at a UA? There are several choices: 410 "Gone", 480 "Temporarily Unavailable", 486 "Busy Here", and 503 "Service Unavailable". The use of any of these here is questionable because the session is already established. It is further complicated if there needs to be a difference in the Reason value for an Access versus a Network Preemption Event (which is a requirement here). The limits of Q.850 [5] have been stated previously in this document.

It should be possible to configure UAs receiving a preemption indication to indicate to the user that no particular type of preemption occurred. There are some domains that might prefer their users to remain unaware of the specifics of network behavior. This should not ever prevent a known preemption indication from being sent in a BYE from a UA.

2.1. Effects of Preemption at the User Agent

If 2 UAs are in a session and one UA must preempt that session to accept another session, a BYE Method message is the appropriate mechanism to perform this task. However, taking this a step further, if a UA is the common point of a 3-way (or more) ad hoc conference and must preempt all sessions in that conference due to receipt of a higher-priority session request (that this UA must accept), then a BYE message must be sent to all UAs in that ad hoc conference.

2.2. Reason Header Requirements for Access Preemption Events

The following is a list of requirements for adding an appropriate Reason value for an Access Preemption Event (APE) as described above and shown in Figure 1:

- APE_REQ#1 - create a means by which one UA can inform another UA (within the same active session) that the active session between the two devices is being purposely preempted at one UA for a higher-priority session request from another UA.
- APE_REQ#2 - create a means by which all relevant SIP elements can be informed of this Access Preemption Event to a specific session.

For example: perhaps SIP Servers that have incorporated a Record-Route header into that session set up need to be informed of this occurrence.

- APE_REQ#3 - create a means of informing all participants in an ad hoc conference that the primary UA (the mixer) has preempted the conference by accepting a higher-priority session request.
- APE_REQ#4 - create a separate indication for the access preemption event than the one used for a Network Preemption Event (described in the next section) in the session BYE message.
- APE_REQ#5 - create a means to generate a specific indication of a preemption event at the user agent to inform all relevant SIP entities, yet have the ability to generalize this indication (based on local policy) to the receiving UA such that this UA cannot display more information than the domain wants the user to see.

3. Network Preemption Events

Network Preemption Events (NPE) are instances in which an intermediate router between SIP user agents preempts one or more sessions at one of its interfaces to place a higher-priority session through that interface. Within RSVP, there exists a means to execute this functionality per [7]: ResvErr messages, which travel downstream towards appropriate receivers. The ResvErr message has the ability to carry within it a code indicating why a reservation is being torn down. The ResvErr does not travel upstream to the other UA. This document proposes that a SIP message be generated to synchronize all relevant SIP elements to this preemption event, including the upstream UA. Creating another Reason value describing that a network element preempted the session is necessary in certain domains.

Figures 2 and 3 illustrate a network preemption scenario with RSVP. NSIS, not shown in examples here, can be imagined from [8] with a NOTIFY error message indicating that a reservation has been preempted with the Transient ERROR_SPEC 0x04000005. SIP behavior will be identical using either reservation protocol.

UA1 invites UA2 to a session with the Resource Priority level of 3 (levels 1 and 2 are higher in this domain) and is accepted. This SIP signaling translated the Resource Priority value to an appropriate RSVP priority level for that flow. The link between Router 1 and Router 2 became saturated with this session reservation between UA1 and UA2 (in this example).

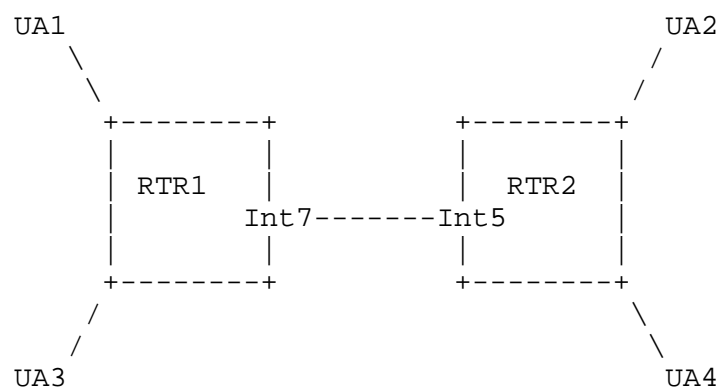


Figure 2. Network Diagram Scenario A

After the session between UA1 and UA2 is established, UA3 invites UA4 to a new session with a Resource Priority level of 2 (a higher priority than the current reservation between UA1 and UA2). Again, the priority value within the Resource-Priority header of this INVITE is translated into an appropriate RSVP priority (that is also higher

in relative priority to the UA1-UA2 session/RSVP flow). When this second, higher-priority session is signaled, one Path message goes from UA3 to UA4, resulting in the RESV message going from UA4 back to UA3. Because this link between the two routers is at capacity (at Int7 in Figure 5), Router 1 will (in this example) make the decision or will communicate with another network entity that will make the decision to preempt lower-priority BW to ensure that this higher-priority session reservation is completed. A ResvErr message is sent to UA2. The result is that UA2 will know that there has been a preemption event in a router (because the ResvErr message has a error code within it, stating "preemption"). At this point, UA1 will not know anything of this preemption. If there are any SIP Proxies between UAs 1 and 2 (perhaps that inserted a Record-Route Header), each will also need to be informed as to why this reservation was torn down.

Figure 3 shows the call flow with Router 2 from Figure 2 included at the RSVP layer sending the ResvErr message. A complete call flow including all UAs and Routers is not shown here for diagram complexity reasons. The complete signaling between UA3 and UA4 is also not included.

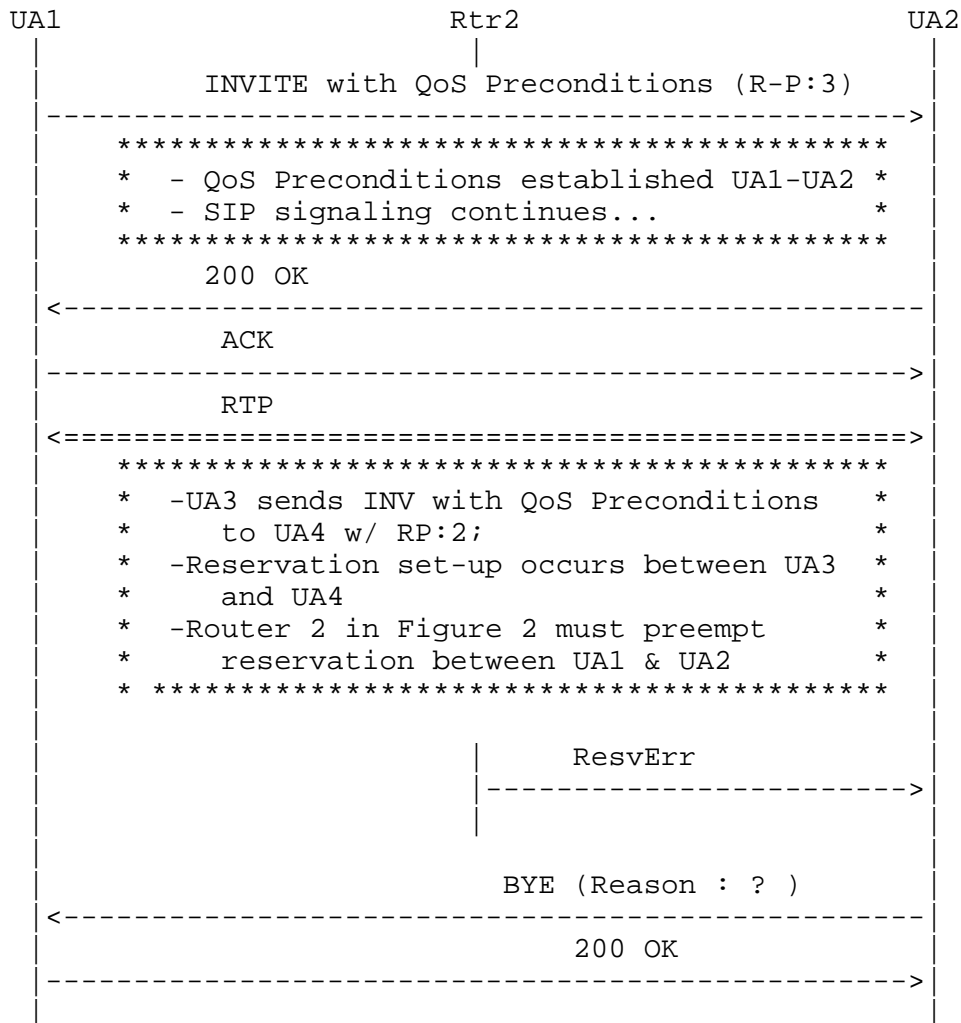


Figure 3. Network Preemption with obscure Reason

What Reason value could be inserted above to mean "preemption at a router interface"? There are several choices: 410 "Gone", 480 "Temporarily Unavailable", 486 "Busy Here", and 503 "Service Unavailable". The use of any of these here is questionable because the session is already established. It is further complicated if there needs to be a difference between the Reason value for an Access Preemption Event versus a Network Preemption Event. The limits of Q.850 [5] have already been stated previously, showing there is nothing in that spec to indicate a problem in an IP network.

To state that all preemptions are equal is possible, but will not provide adequate information. Therefore, another Reason Header value is necessary to differentiate the APE from the NPE.

3.1. Reason Header Requirements for Network Preemption Events

The following are the requirements for the appropriate SIP signaling in reaction to a Network Preemption Event (NPE):

NPE_REQ#1 - create a means of informing the far-end UA that a Network Preemption Event has occurred in an intermediate router.

NPE_REQ#2 - create a means by which all relevant SIP elements can be informed of a Network Preemption Event to a specific session.

For example: perhaps SIP Servers have incorporated a Record-Route header into that session set up.

NPE_REQ#3 - create a means of informing all participants in an ad hoc conference that the primary UA (the mixer) has been preempted by a Network Preemption Event.

NPE_REQ#4 - create a separate description of the Network Preemption Event relative to an Access Preemption Event in SIP.

4. Including a Hybrid Infrastructure

If User 1 is in a non-IP portion of infrastructure (using a TDM phone) in a session with a UA through a SIP gateway, and if the TDM portion had the ability to preempt the session and indicate to the SIP gateway when it did such a preemption, the SIP GW would need to be able to convey this preemption event into the SIP portion of this session just as if User 1 were a UA in the session. Below is a diagram of this:

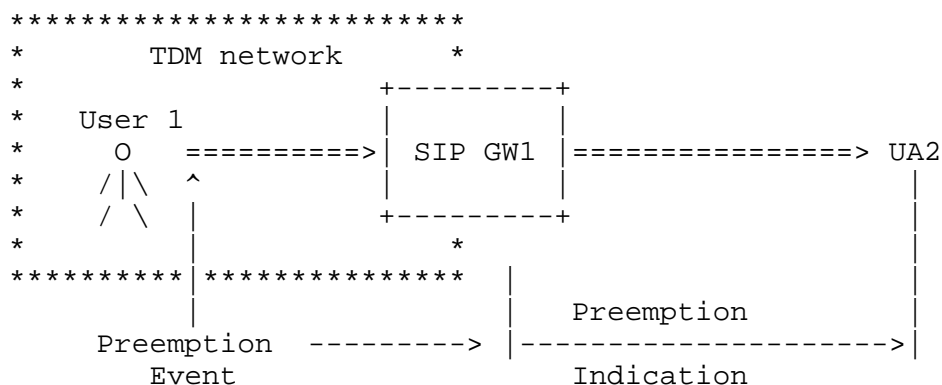


Figure 4. TDM/IP Preemption Event

4.1. Hybrid Infrastructure Requirements

The following are the requirements unique to the topology involving both IP infrastructure and TDM (or non-IP) infrastructure.

HYB_REQ#1 - create a means of informing the far-end UA in a dialog through a SIP gateway with a non-IP phone that the TDM portion of the session indicated to the SIP gateway that a preemption event terminated the session.

HYB_REQ#2 - create a means of identifying this preemption event uniquely with respect to an access preemption and network preemption event.

5. Preemption Reason Header Cause Codes and Semantics

This document defines the following new protocol value for the protocol field of the Reason header field in RFC 3326 [1]:

Preemption: The cause parameter contains a preemption cause code.

We define the following preemption cause codes:

Value	Default Text	Description
1	UA Preemption	The session has been preempted by a UA.
2	Reserved Resources Preempted	The session preemption has been initiated within the network via a purposeful RSVP preemption occurrence, and not a link error.
3	Generic Preemption	This is a limited-use preemption indication to be used on the final leg to the preempted UA to generalize the event.
4	Non-IP Preemption	The session preemption has occurred in a non-IP portion of the infrastructure, and this is the Reason cause code given by the SIP Gateway.

Example syntax for the above preemption types are as follows:

```
Reason: preemption ;cause=1 ;text="UA Preemption"
Reason: preemption ;cause=2 ;text="Reserved Resources Preempted"
Reason: preemption ;cause=3 ;text="Generic Preemption"
Reason: preemption ;cause=4 ;text="Non-IP Preemption"
```

Sections 5.1, 5.2, 5.3, and 5.4 provide use cases and extended definitions for the above four cause codes with message flow diagrams.

5.1. Access Preemption Event Reason Code

A more elaborate description of the Access Preemption Event cause=1 is as follows:

A user agent in a session has purposely preempted a session and is informing the far-end user agent, or user agents (if part of a conference), and SIP Proxies (if stateful of the session's transactions)

An example usage of this header value would be:

Reason: preemption ;cause=1 ;text="UA Preemption"

5.1.1. Access Preemption Event Call Flow

Figure 5 replicates the call flow from Figure 1, but with an appropriate Reason value indication that was proposed in Section 4.1, above:

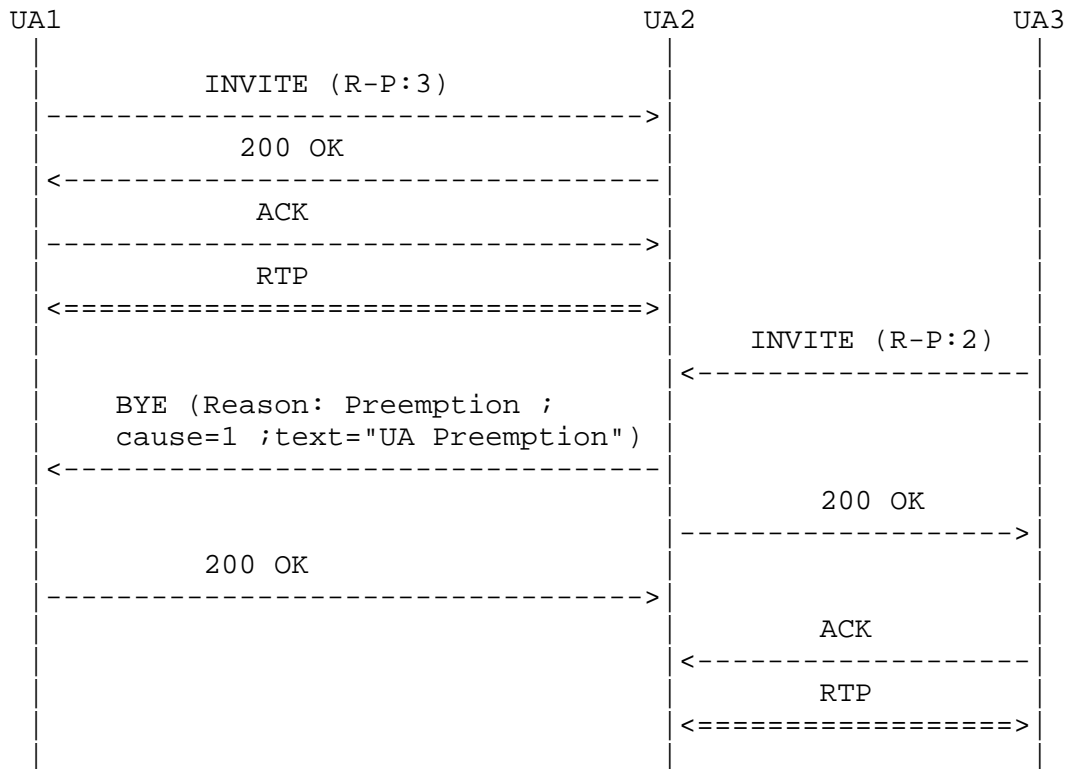


Figure 5. Access Preemption with Reason: UA Preemption

UA1 invites UA2 to a session with the Resource Priority level of 3 (levels 1 and 2 are higher in this domain). After the session between UA1 and UA2 is established, UA3 invites UA2 to a new session with an R-P of 2 (a higher priority than the current session to UA1). Local policy within this domain dictates that UA2 must preempt all existing calls of lower priority in order to accept a higher-priority call.

UA2 sends a BYE Request message with a Reason header with a value of UA Preemption. This will inform the far-end UA (UA1) and all relevant SIP elements (for example, SIP Proxies). The cause code is unique to what is proposed in the RSVP Preemption Event for differentiation purposes.

5.2. Network Preemption Events Reason Code

A more elaborate description of the Reserved Resources Preempted Event cause=2 is as follows:

A router has preempted a reservation flow and generated a reservation error message: a ResvErr traveling downstream in RSVP, and a NOTIFY in NSIS. The UA receiving the preemption error message generates a BYE request towards the far-side UA with a Reason Header with this value indicating that somewhere between two or more UAs, a router has administratively preempted this session.

An example usage of this header value would be:

Reason: Preemption :cause=2 ;text="Reserved Resources Preempted"

5.2.1. Network Preemption Event Call Flow

Figure 6 replicates the call flow from Figure 5, but with an appropriate Reason value indication that was proposed in Section 4.2, above.

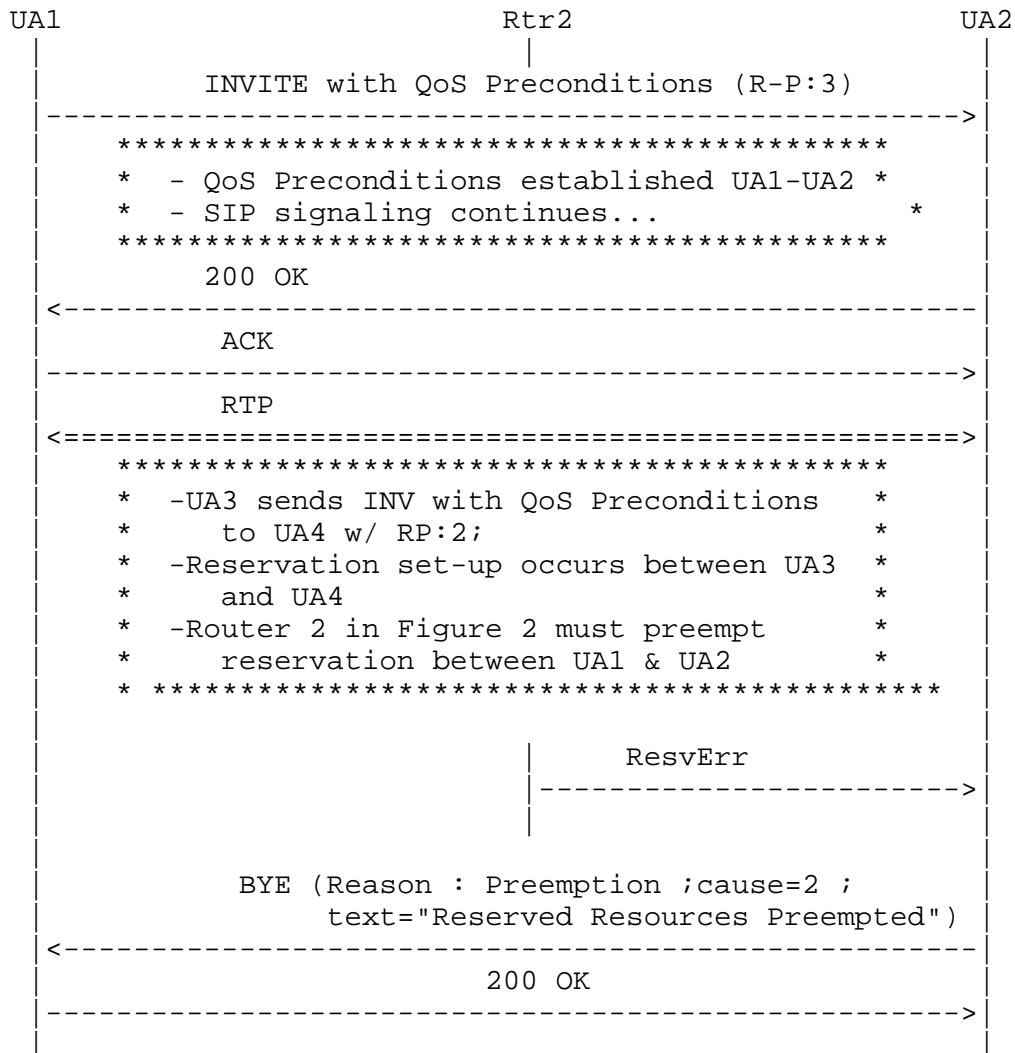


Figure 6. Network Preemption with "Reserved Resources Preempted"

Above is the call flow with Router 2 from Figure 2 included at the RSVP layer sending the Resv messages. A complete call flow including all UAs and Routers is not included for diagram complexity reasons. The signaling between UA3 and UA4 is also not included.

Upon receipt of the ResvErr message with the preemption error code, UA2 can now appropriately inform UA1 why this event occurred. This BYE message will also inform all relevant SIP elements, synchronizing them. The cause value is unique to that proposed in Section 4.1 for Access Preemption Events for differentiation purposes.

5.3. Generic Preemption Event Reason Code

A more elaborate description of the Generic Preemption Event cause=3 is as follows:

This cause code is for infrastructures that do not wish to provide the preempted UA with a more precise reason than just "preemption". It is possible that UAs will have code that will indicate the type of preemption event that is contained in the Reason header, and certain domains have expressed this as not being optimal, and wanted to generalize the indication. This MUST NOT be the initial indication within these domains, as valuable traffic analysis and other NM applications will be generalized as well. If this cause value is to be implemented, it SHOULD only be done at the final SIP Proxy in such a way that the cause value indicating which type of preemption event actually occurred is changed to this generalized preemption indication to be received by the preempted UA.

An example usage of this header value would be:

```
Reason: preemption ;cause=3 ;text="Generic Preemption"
```

5.4. Non-IP Preemption Event Reason Code

A more elaborate description of the Non-IP Preemption Event cause=4 is as follows:

A session exists in a hybrid IP/non-IP infrastructure and the preemption event occurs in the non-IP portion, and was indicated by that portion that this call termination was due to preemption. This is the indication that would be generated by a SIP Gateway towards the SIP UA that is being preempted, traversing whichever SIP Proxies are involved in session signaling (a question of server state).

An example usage of this header value would be:

```
Reason: preemption ;cause=4 ;text="Non-IP Preemption"
```


5.4.1. Non-IP Preemption Event Call Flow

Figure 7 is a simple call flow diagram of the Non-IP Preemption Event.

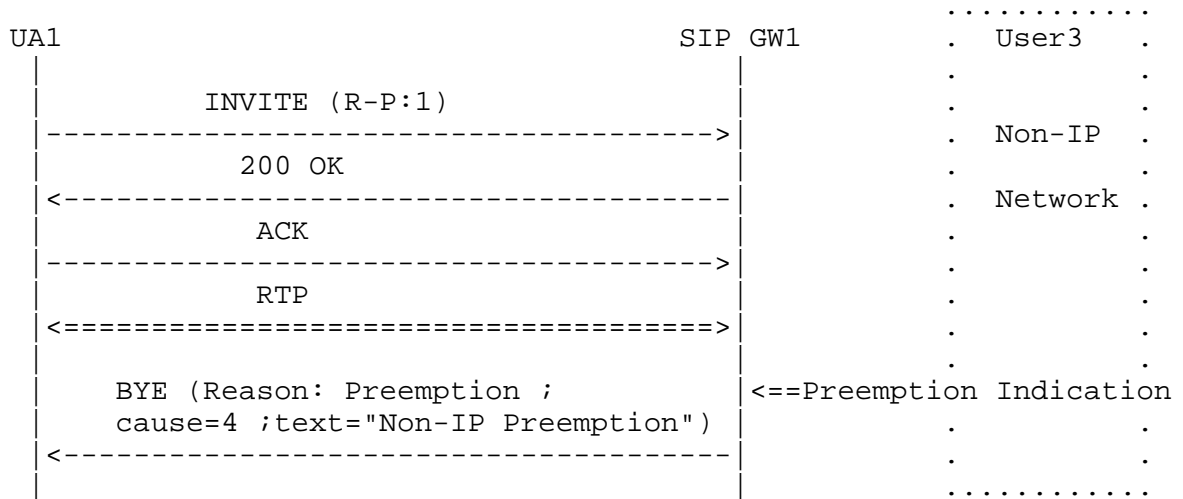


Figure 7. Non-IP Preemption Flow

In this case, UA1 signals User3 to a session. Once established, there is a preemption event in the non-IP portion of the session/call, and the TDM portion has the ability to inform the SIP GW of this type of event. This non-IP signal can be translated into SIP signaling (into the BYE session termination message). Within this BYE, there should be a Reason header indicating such an event to synchronize all SIP elements.

6. Security Considerations

Eavesdropping on this header field should not prevent proper operation of the SIP protocol, although some domains utilizing this mechanism for notifying and synchronizing SIP elements will likely want the integrity to be assured. It is therefore RECOMMENDED that integrity protection be applied when using this header to prevent unwanted changes to the field and snooping of the messages. The accepted choices for providing integrity protection in SIP are TLS and S/MIME.

7. IANA Considerations

This document adds to one existing IANA Registry and creates one new Registry. The existing IANA Registry for the SIP Reason Header is as follows:

Protocol Value	Protocol Cause	Reference
-----	-----	-----
SIP	Status code	RFC 3261
Q.850	Cause value in decimal	ITU-T Q.850

This document adds to that Registry with the following entry (including the '*' comment):

Protocol Value	Protocol Cause	Reference
-----	-----	-----
Preemption	Cause value in decimal*	RFC 4411

* See the separate "Preemption" Registry for default reason-text strings.

The cause values created by the Preemption Protocol namespace in this document are defined in Section 7.1. Each cause value has a Reason-text string as a general description of what the cause value is for. This is shown for the existing Reason header in Section 2 of RFC 3326. Before this document, the Reason-text was taken from the SIP Response code string from all SIP Response codes, or the default description from Q.850 cause codes. Currently, there is no place to register new reason-text strings other than from those two sources. Because this document defines a new Reason header protocol namespace, a new IANA Registry is created in Section 7.2 just for this and future Reason header protocol namespaces (other than SIP Response codes or Q.850 cause values) to register their respective general descriptive text strings. These text strings are non-binding and merely the default for human understanding, but they are deemed important enough to have their own Registry.

7.1. "Preemption" Namespace Registry

RFC 4411 creates the new SIP "Reason Header" [1] protocol namespace: "Preemption", with 4 defined cause codes:

In instances where this namespace is used to indicate preemption at a UA, the following syntax shall be used (the reason-text is a default string; it is not mandatory, and may be different):

```
Reason: preemption ;cause=1 ;text="UA Preemption"
```

Section 5.1 of this document describes in detail the semantics of this cause code.

The default text above is part of a new IANA Registry for default text strings for any new protocol namespace cause code. See Section 7.2 for details.

In instances where this namespace is used to indicate preemption because an RSVP ResvErr message was received at a SIP UA, the following syntax shall be used (the reason-text is a default string; it is not mandatory, and may be different):

```
Reason: preemption ;cause=2 ;text="Reserved Resources Preempted"
```

Section 5.2 of this document describes in detail the semantics of this cause code.

The default text above is part of a new IANA Registry for default text strings for any new protocol namespace cause code. See section 7.2 for details.

In instances where this namespace is used to indicate a generalized preemption event to the destination UA from a Proxy that modifies the Reason value only during this last SIP hop, the following syntax shall be used (the reason-text is a default string; it is not mandatory, and may be different):

```
Reason: preemption ;cause=3 ;text="Generic Preemption"
```

Section 5.3 of this document describes in detail the semantics of this cause code.

The default text above is part of a new IANA Registry for default text strings for any new protocol namespace cause code. See Section 7.2 for details.

In instances where this namespace is used to indicate preemption from a non-IP portion of a call leg, a SIP Gateway shall use the following syntax to inform the SIP infrastructure of this event (the reason-text is a default string; it is not mandatory, and may be different):

```
Reason: preemption ;cause=4 ;text=" Non-IP Preemption"
```

Section 5.4 of this document describes in detail the semantics of this cause code.

The default text above is part of a new IANA Registry for default text strings for any new protocol namespace cause code. See Section 7.2 for details.

Additional definitions of the preemption namespace and its cause codes MUST be defined in Standards Track documents.

7.2. Default Reason-Text IANA Registry for the SIP Reason Header

Below is a new IANA Registry for SIP Reason Header reason-text strings, associated with their respective protocol type and Reason-param cause values. Per RFC 3326, the Reason-text string is a quoted default string with only human understandability meant. These strings can be changed by local policy.

Protocol	Reason-param	Reason-Text	Reference
-----	-----	-----	-----
Preemption	Cause=1	UA Preemption	RFC 4411
Preemption	Cause=2	Reserved Resources Preempted	RFC 4411
Preemption	Cause=3	Generic Preemption	RFC 4411
Preemption	Cause=4	Non-IP Preemption	RFC 4411

8. Contributions

The following individuals contributed to this effort:

Subhasri Dhesikan
Gonzalo Camarillo
Dave Oran

The author thanks these individuals greatly for their aid in this effort.

9. Acknowledgements

To Haluk Keskiner for providing a valued sanity check. To Dean Willis, Rohan Mahy, and Allison Mankin for their belief in and backing of this effort. To Adam Roach and Arun Kumar for helpful comments to this document.

Thanks to Mike Pierce for helpful comments and catching a flaw in this spec late in the process (before it was too late).

10. References

10.1. Normative References

- [1] Schulzrinne, H., Oran, D., and G. Camarillo, "The Reason Header Field for the Session Initiation Protocol (SIP)", RFC 3326, December 2002.
- [2] 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.
- [3] Camarillo, G., Marshall, W., and J. Rosenberg, "Integration of Resource Management and Session Initiation Protocol (SIP)", RFC 3312, October 2002.
- [4] Schulzrinne, H. and J. Polk, "Communications Resource-Priority Header in the Session Initiation Protocol (SIP)", RFC 4412, February 2006.
- [5] ITU-T Recommendation Q.850 (1993)
- [6] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [7] Braden, R., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, September 1997.

10.2. Informative References

- [8] J. Manner, G. Karagiannis, A. McDonald, S. Van den Bosch, "NSLP for Quality-of-Service signalling", Work in Progress, September 2005.

Author Information

James M. Polk
Cisco Systems
2200 East President George Bush Turnpike
Richardson, Texas 75082 USA

EMail: jmpolk@cisco.com

Full Copyright Statement

Copyright (C) The Internet Society (2006).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM 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.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights 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; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat 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 implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgement

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

