

Dispatch Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 3, 2011

Parthasarathi. Ravindran
Sheshadri. Shalya
Cisco Systems, Inc.
August 30, 2010

**Session Initiation Protocol (SIP) Resource availability Event package
draft-partha-dispatch-resource-availability-00**

Abstract

Collecting Resource availability information in Session Initiation Protocol (SIP) devices in real time helps in better manageability of resources in the SIP network. Resource availability monitoring in SIP devices helps administrator to take informed decision and corrective measures to tackle under or over resource utilization in the network. Resource availability information can also be used for SIP overload control in SIP based Media servers by passing resource demand vector between devices. This document defines resource availability XML document and the mechanisms that can be used to exchange the document between SIP entities.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on March 3, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	4
2.	Terminology	4
3.	Resource availability indication mechanism	4
4.	Resource availability Event package definition	5
4.1.	Event Package name	5
4.2.	Event Package Parameters	5
4.3.	SUBSCRIBE bodies	5
4.4.	SUBSCRIBE duration	6
4.5.	NOTIFY bodies	6
4.6.	Notifier processing of SUBSCRIBE Requests	6
4.7.	Notifier generation of NOTIFY Requests	7
4.8.	Subscriber processing of NOTIFY requests	7
4.9.	Handling of forked requests	7
4.10.	Rate of Notifications	7
4.11.	State Agents	8
4.12.	Behavior of SIP Proxy	8
4.13.	Refresh SUBSCRIBE mechanism for failure handling	8
5.	Resource availability Indication (RAI) document format	8
5.1.	Contents	8
5.2.	XML data format	9
5.2.1.	Namespace	9
5.2.2.	Resource-Availability	9
5.3.	Resource	9
5.4.	Resource-subtype	10
5.4.1.	Almost-out-of-resource	10
5.5.	Total & Available	10
5.6.	Unit	10
5.7.	Timestamp	11
5.8.	Example	11
6.	Use Case for Resource Availability mechanism	12
6.1.	Subscriber acts as resource monitor only	12
6.2.	Subscriber pools for resource status	12
6.3.	Subscriber acts as overload protection	12
7.	XML Schema definition for Resource availability	12
8.	Security Considerations	14
9.	IANA Considerations	14
9.1.	Resource availability package Registration	14
9.2.	application/rai+xml MIME Registration	14
9.3.	Resource availability indication Schema Registration	15
10.	Acknowledgement	16
11.	References	16
11.1.	Normative References	16
11.2.	Informative References	16
	Authors' Addresses	17

1. Introduction

In Voice/Video over IP (VoIP) network, Session Initiation protocol (SIP) [[RFC3261](#)] is widely deployed as a signaling protocol. Resource availability information from SIP devices improves the manageability of SIP devices by providing a framework for multiple applications like Resource utilization monitoring, overload handling in SIP based media servers.

Monitoring the resource utilization pattern at any given time helps VoIP administrator to decide whether the VoIP network resources need to be expanded or it is under utilized. SIP based resource utilization is preferred within SIP network and removes the need for using any other management protocol for resource monitoring purpose.

Overload requirements for SIP are explained in [[RFC5390](#)]. In case of using Resource availability information for overload protection mechanism, SIP server indicates its resource capability and resource demand vector to the prior SIP server by which prior SIP server will be able to perform intelligent routing. The resource demand vector provides the consumption of resources for the specific type of dialog (audio, video, telepresence, etc.,). As SIP is used for overload protection, it is easy to relate the overload protection information to specific SIP entity within the SIP network without having any separate naming or addressing scheme.

2. 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 [[RFC2119](#)]. This document only uses these key words when referencing normative statements in existing RFCs.

3. Resource availability indication mechanism

Resource availability indication mechanism by UA helps resource statistics collector to collect the data and act upon them accordingly. The solution is targeted for SIP trunks or between SIP servers. By the proposed mechanism, the individual devices indicate their current resource availability information through XML document defined in this draft. Load Balancer or resource statistics collector acts as a subscriber and subscribe to the resource in the individual UA. Individual devices acts as Notifier and inform the resource information in XML.

In the resource monitor, the periodic resource information is collected from all the devices and the reports are formed. These statistics provide effective resource utilization indication for any given resource at any point of time. This information helps administrator to take informed decision and corrective measures to tackle under or over resource utilization in the network.

In the load balancing scenario, the load balancer subscribes to all the relevant SIP servers and is informed about the current resource information. Based on the resource availability indication, the call is routed to the appropriate SIP servers. In case one of the devices is reached almost out of the resource, the device notifies the load balancer immediately. This helps load balancer to decide where to route the dialog when multiple routes are available for the given dialog. The algorithm of using resource availability to achieve load balancing and resource demand vector for the individual call type is kept out of scope of this document. The threshold for the resource has to be decided by administrator by keep in mind that NOTIFY message has to be formed and send out to the external server and also resource priority dialog has to be processed without any interruption.

OPTIONS response message MAY be used for exchange resource availability information. (Using OPTIONS for this package is TBD).

4. Resource availability Event package definition

This document defines the details of the SIP event package for Resource availability based routing according to [[RFC3265](#)]

4.1. Event Package name

The name of the event package is resource-availability. This event name has to be mentioned in the Event and allow-event header as mentioned in [[RFC3265](#)]

4.2. Event Package Parameters

There is no specific event package parameter specified for this event package.

4.3. SUBSCRIBE bodies

A SUBSCRIBE request for resource availability policy MAY contain a body to request the specific resources availability notification. For example, a subscriber is interested in some specific resources only. The details of the subscription filter specification are not

yet defined.

A SUBSCRIBE request sent without a body implies the default subscription behavior as specified in the sec 4.7.

4.4. SUBSCRIBE duration

Subscription duration depends upon the deployment. The default duration for this package is 300 sec. It is not recommended to have the refresh timer less than 180 sec. Refresh of subscription ensures that the next hop is up before new dialog request is forwarded. Refresh subscription should be triggered 32 sec before the expiry time of the package. This keepalive mechanism is useful to decide whether Notifier trunk is reachable or not. The user defined value shall be used depending on the deployment scenario.

In case of SUBSCRIBE refresh failure, it is recommended that new SUBSCRIBE is initiated after the expiry of user mentioned timer.

4.5. NOTIFY bodies

The body of a NOTIFY message in this event package contains resource information of a device. The format of the NOTIFY body MUST be in one of the formats defined in the Accept header field of the SUBSCRIBE request or be the default format as specified in [[RFC3265](#)]. The default data format for the NOTIFY body of this event package is "application/rai+xml" (defined in [Section 7](#)).

This means that if no Accept header field is specified to a SUBSCRIBE request, the NOTIFY will contain a body in the "application/rai+xml" format. If the Accept header field is present in SUBSCRIBE request, it MUST include "application/rai+xml" and MAY include any other types.

4.6. Notifier processing of SUBSCRIBE Requests

Notifier has to provide the sensitive device resource information to the subscriber in the notification mechanism. The entire subscriber has to be authenticated before providing the package details. The existing authentication SIP mechanism like DIGEST, TLS, and S/MIME shall be used. In the trusted domain, the authentication may not be required and the administrator has to decide the trusted domain and non-trusted domain.

The requested SUBSCRIBER is not allowed to receive the requested information, 403 response MUST be sent by Notifier.

4.7. Notifier generation of NOTIFY Requests

NOTIFY has to be formatted as per [[RFC3265](#)], it has to contain resource availability information. NOTIFY has to be sent immediately after SUBSCRIBE is received for initial dialog or refresh and responded with 200 OK. Notifier MUST notify SUBSCRIBER whenever any of the resource reaching almost out of the resource state or going below the configured threshold. Almost out of the resource value for the given resource or system will be decided by the administrator. The threshold notification helps SUBSCRIBER to decide whether the dialog shall be forwarded or not based on SUBSCRIBER policy. Notifier shall notify the resource information in the periodic manner. When any of the resource is reaching almost out of the resource, NOTIFY will be send which contain only the resource which reached almost out of the resource.

4.8. Subscriber processing of NOTIFY requests

Subscriber updates the dialog routing logic based on the incoming NOTIFY message. In case NOTIFY message contains almost out of the resource for the specific resource or the whole system, the dialog routing logic in the subscriber has to be updated with appropriate information by which any further dialog SHOULD NOT be routed to Notifier till getting further notification with resource available indication from Notifier.

In case NOTIFY is received in the periodic manner, the resource based routing and statistics for the individual device usage at a given time is updated with the received NOTIFY message body information.

4.9. Handling of forked requests

Forking of SUBSCRIBE request is not supported for this package. In case of forking happened, forked response has to be handled as mentioned in Sec 4.4.9 of [[RFC3265](#)].

4.10. Rate of Notifications

Rate of Notification SHOULD NOT be less than once every 32 sec as the notification itself will be observed as an overhead otherwise. It is recommended that the rate of notification happens once every 120 sec. The administrator is the best person to decide the notification time based on the network topology. The rate of notification MAY be reduced or stopped when almost out of the resource is attained for the critical resources like CPU, Memory and the critical resource list is based on device.

4.11. State Agents

The resource availability indications are generated by SIP entity directly and there is no need of explicit state agent for this package. As load balancing based on the resource availability indication has to be done in the real time, the aggregation done by any state agent will introduce delay and hence defeat overload requirements.

4.12. Behavior of SIP Proxy

This subscription mechanism is recommended for neighboring SIP entities as it provides more control on overload mechanism. In case the network topology requires to pass this resource information through proxy, Proxy has to forward SUBSCRIBE/NOTIFY messages and this may be required when intermediate proxy is interworking and load balancing is decided by entity sitting behind the proxy.

4.13. Refresh SUBSCRIBE mechanism for failure handling

Apart from refresh subscription as explained in sec 4.4, Refresh subscription shall be triggered in the following network error scenarios as well:

- o ICMP message received during dialog creation from the Notifying entity
- o TCP connection failure from the Notifying entity
- o Dialog creation timeout

Refresh Subscription shall be started whenever 503 is received for any dialog creation request from Notifying entity.

Till this refresh subscription succeeds, new dialog MUST NOT be forwarded to the Notifying entity. In case of refresh SUBSCRIBE failure, new SUBSCRIBE has to be started as mentioned in sec 4.4 and no further dialog has to be forwarded till new SUBSCRIBE dialog is created.

5. Resource availability Indication (RAI) document format

5.1. Contents

Resource availability indication (RAI) document is a XML document which will be embedded as message body in NOTIFY message of resource-availability package. The document contains

- o Entity parameter in Resource-availability to specify the entity whose resource information is available as part of this message
- o Resource Tuple indicates the individual resources total and available capabilities, and also whether the resource is available for further new dialog processing (Optional)
- o Resource Subtype provides the granular information within the particular resource. (Optional)

5.2. XML data format

RAI object is a XML document. It MUST have the XML declaration and it SHOULD contain an encoding declaration in the XML declaration, e.g., "<?xml version='1.0' encoding='UTF-8'?>". If the charset parameter of the MIME content type declaration is present and it is different from the encoding declaration, the charset parameter takes precedence.

Every application conformant to this specification MUST accept the UTF-8 character encoding to ensure the minimal interoperability.

5.2.1. Namespace

The namespace URI for elements defined by this specification is a Uniform Resource Namespace (URN) [[RFC2141](#)], using the namespace identifier 'ietf' defined by [[RFC2648](#)] and extended by [[RFC3688](#)].

The URN is as follows: urn:ietf:params:xml:ns:rai

5.2.2. Resource-Availability

Resource-Availability element MUST contain an xmlns namespace attribute with value as urn:ietf:params:xml:ns:rai.

Resource-availability MUST have entity attribute which contains SIP/SIPS URI to identify the device which provides the resource information. The entity attribute SIP/SIPS URI FQDN or IP address represents the device and may not have user part.

5.3. Resource

This element indicates the individual resource status at the given time. This is an optional element wherein the resources type like CPU, Memory, DSP, DS0, available bandwidth, disk storage are mentioned in type attribute. Each resource element shall have almost out of resource status, total and available resource usage. This resource information helps in routing or load balancing or resource monitoring based on the individual resource availability. Available bandwidth mentioned in this element is the indicator of how many bits

of data the system can process per second session(audio/video).

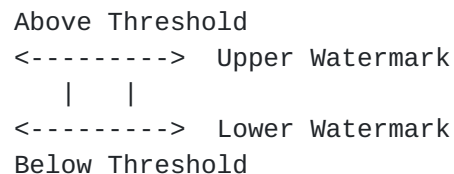
5.4. Resource-subtype

This element comes within a particular resource element to provide an internal division within a given resource. For example, Memory shall be divided into processor memory, IO memory and provides the individual status. This division helps in load balancing in case routing entity aware of the resource capability required for the specified call. This is an optional element.

5.4.1. Almost-out-of-resource

Almost-out-of-resource is a Boolean element to indicate whether system or resource is reached the threshold or not. This threshold is decided by Notifier or subscriber shall send it through other mechanism.

Almost-out-of-resource value true indicates that the threshold is reached. To avoid the back-forth movement in the threshold range, it is preferred to provide the upper watermark value which decides the above threshold limit and lower watermark value is used when the above threshold is back to normal. True value is set for almost out of resource when upper watermark is reached and when lower watermark is reached, almost out of resource is set with false.



Watermark mechanism for Threshold handling

5.5. Total & Available

Total and Available elements provide absolute value or the value specified as per unit element. These are optional element.

5.6. Unit

Unit is a string which indicates unit for the given resource or resource subtype. The default value of unit is absolute value.

5.7. Timestamp

Timestamp element contains a string indicating the date and time of the status change of this tuple. The value of this element MUST follow the IMPP datetime format [[RFC3339](#)]. Timestamps that contain 'T' or 'Z' MUST use the capitalized forms.

As a security measure, the timestamp element SHOULD be included in all tuples unless the exact time of the status change cannot be determined.

5.8. Example

The example provides all the tuples involved in rai xml body.

```
<?xml version="1.0" encoding="UTF-8"?>
<resource-Availability xmlns="urn:ietf:params:xml:ns:rai"
    entity =sip:pstngateway1.cisco.com>
  <resource type="CPU">
    <total>100</total>
    <available>50</available>
    <Unit>percentage</unit>
  </resource>
  <resource type="memory">
    <total>256</total>
    <available>153</available>
    <unit>mb</unit>
    <resourceSubType subtype="iomem">
      <total>100</total>
      <available>50</available>
      <unit>percentage</unit>
    </resourceSubType>
  </resource>
  <resource type=="DSP">
    <total>32</total>
    <available>10</available>
  </resource>
  <resource type="DS0">
    <almost-out-of-resource>true</almost-out-of-resource>
    <total>30</total>
    <available>10</available>
  </resource>
  <timestamp>2010-06-13T09:00:00Z</timestamp>
</resource-Availability>
```

RAI Example XML body

6. Use Case for Resource Availability mechanism

6.1. Subscriber acts as resource monitor only

This is the simple scenario wherein subscriber simply collects the Notifier statistics and store or display. This is useful when Administrator is interested in the usage of the VoIP network resource. Total and available element will be used by subscriber. In this scenario, subscriber will not honor the almost out of resource information from the notifier.

6.2. Subscriber pools for resource status

In this mode, Subscriber is interested in the resource status at a given time, and one shot subscription shall be used for this mechanism.

6.3. Subscriber acts as overload protection

In this deployment, subscriber stop/start the load based on the almost out of resource parameter of the body. System's almost out of resource is set to true whenever any one of the resource is reached almost out of the resource. In case the subscriber is aware that the new dialog does not require the set of resource which reached almost out resource, the new dialog shall be forwarded. For example, DS0 is almost out of the resource and the new dialog does not require DS0 resource, the new dialog shall be forwarded to Notifier. The resource requirement of Notifier for the specific dialog shall be intimated to Subscriber through other protocol mechanism or provisioning mechanism which is outside the scope of this document.

7. XML Schema definition for Resource availability

This section defines XML schema for resource availability document.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:ietf:params:xml:ns:rai"
  xmlns:tns="urn:ietf:params:xml:ns:rai"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
<!-- This import brings in the XML language attribute xml:lang-->
<xs:import namespace="http://www.w3.org/XML/1998/namespace"
  schemaLocation="http://www.w3.org/2001/xml.xsd"/>
<xs:element name="resource-availability"
  type="tns:resource-availability"/>
```



```
<xs:complexType name="resource-availability">
  <xs:sequence>
    <xs:element name="resource" type="tns:resource" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="timestamp" type="xs:dateTime"
      minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="entity" type="xs:anyURI" use="required"/>
</xs:complexType>
<xs:complexType name="resource">
  <xs:sequence>
    <xs:element name="almost-out-of-resource" type="xs:boolean"
      minOccurs="0"/>
    <xs:element name="total" type="xs:unsignedInt"
      minOccurs="0"/>
    <xs:element name="available" type="xs:unsignedInt"
      minOccurs="0"/>
    <xs:element name="resource-subtype" type="tns:resourceSubType"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="Unit" type="tns:unit" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="type" type="tns:resourceType" use="required"/>
</xs:complexType>
<xs:simpleType name="resourceType">
  <xs:restriction base="xs:string">
    <xs:pattern
      value="cpu|memory|ds0|dsp|storage|([a-z][a-z0-9])*"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="resourceSubType">
  <xs:sequence>
    <xs:element name="almost-out-of-resource" type="xs:boolean"/>
    <xs:element name="total" type="xs:unsignedInt"
      minOccurs="0"/>
    <xs:element name="available" type="xs:unsignedInt"
      minOccurs="0"/>
    <xs:element name="unit" type="tns:unit" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="subtype" type="xs:resourceSubType"
    use="required"/>
</xs:complexType>
<xs:simpleType name="resourceSubType">
  <xs:restriction base="xs:string">
    <xs:pattern
      value="procmem|iomem|user|kernel|([a-z][a-z0-9])*"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="unit">
```



```
<xs:restriction base="xs:string">
  <xs:pattern
    value="bytes|mb|gb|tb|percentage|([a-z][a-z0-9])*" />
</xs:restriction>
</xs:simpleType>
```

8. Security Considerations

Security consideration for event based framework as specified in [RFC3265] has to be considered for this draft as well. As the resource information is sensitive information, Subscribe/Notify shall use TLS transport in case subscriber and Notifier are in the public domain. In case subscriber is a untrusted entity, subscriber will be authenticated by responding with 401. The administrator provides authorization mechanism by which different entity will be provided with different level of information. For example, all SIP entity within enterprise could be provided complete resource information and only system level information could be provided towards service provider network.

The resource information provided in this mechanism is more critical. If the above specified mechanism is not secure enough, there is a scope for coming up with addition security measures (TBD).

9. IANA Considerations

This specification registers a SIP event package, a new MIME type, a new XML namespace, and a new XML schema.

9.1. Resource availability package Registration

This section registers an event package based on the registration procedures defined in [RFC3265].

Package name: resource-availability

Type: package

Published specification: This document

Person to contact: R.Parthasarathi, partr@cisco.com

9.2. application/rai+xml MIME Registration

This section registers a new MIME type based on the procedures defined in [RFC4288] and guidelines in [RFC3023].

MIME media type name: application

MIME subtype name: rai+xml

Mandatory parameters: none

Optional parameters: Same as charset parameter application/xml in [\[RFC3023\]](#)

Encoding considerations: Same as encoding considerations of application/xml in [\[RFC3023\]](#)

Security considerations: See [Section 10 of \[RFC3023\]](#) and [Section 8](#) of this specification

Interpretability considerations: None

Published Specification: This document

Applications which use this media type: Load balancing SIP entities, Resource statistics collecting SIP entities

Additional information:

Magic number: None

File extension: .xml

Macintosh file type code: 'TEXT'

Personal and email address for further information:

Parthasarathi.R, partr@cisco.com

Intended usage: COMMON

Author/Change Controller: IETF Dispatch Working Group

<sip-overload@ietf.org>, as designated by the IESG iesg@ietf.org

[9.3.](#) Resource availability indication Schema Registration

URI: urn:ietf:params:xml:schema:rai

Registrant Contact: IETF Dispatch working group, Parthasarathi.R (partr@cisco.com)

XML: the XML schema to be registered is contained in [Section 7](#).

Its first line is <?xml version="1.0" encoding="UTF-8"?> and its last line is </xs:schema>

TBD: Adding registry for resourceType and resourceSubtype tuples

10. Acknowledgement

We wish to thank Paul Kyzivat, Muthu Arul Mozhi, Paul Jones, Sanjay Sinha, Dale worley for the valuable comments. We also wish to thank Vipin Palawat, Vinay Pande, Darryl sladden, Janet Bryon for providing the initial idea on the resource availability information exchange using SIP.

11. References

11.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2141] Moats, R., "URN Syntax", [RFC 2141](#), May 1997.
- [RFC2648] Moats, R., "A URN Namespace for IETF Documents", [RFC 2648](#), August 1999.
- [RFC3023] Murata, M., St. Laurent, S., and D. Kohn, "XML Media Types", [RFC 3023](#), January 2001.
- [RFC3261] 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.
- [RFC3265] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", [RFC 3265](#), June 2002.
- [RFC3688] Mealling, M., "The IETF XML Registry", [BCP 81](#), [RFC 3688](#), January 2004.
- [RFC3339] Klyne, G., Ed. and C. Newman, "Date and Time on the Internet: Timestamps", [RFC 3339](#), July 2002.
- [RFC4288] Freed, N. and J. Klensin, "Media Type Specifications and Registration Procedures", [BCP 13](#), [RFC 4288](#), December 2005.

11.2. Informative References

- [RFC5390] Rosenberg, J., "Requirements for Management of Overload in the Session Initiation Protocol", [RFC 5390](#), December 2008.

Authors' Addresses

Parthasarathi R
Cisco Systems, Inc.
Cessna Business Park,
Kadabeesanahalli Village, Varthur Hobli,
Sarjapur-Marathahalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: partr@cisco.com

Sheshadri Shalya
Cisco Systems, Inc.
Cessna Business Park,
Kadabeesanahalli Village, Varthur Hobli,
Sarjapur-Marathahalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: svshesha@cisco.com

