Internet Engineering Task Force (IETF)
Request for Comments: 6076
Category: Standards Track
ISSN: 2070-1721

D. Malas
CableLabs
A. Morton
AT&T Labs

January 2011

Basic Telephony SIP End-to-End Performance Metrics

Abstract

This document defines a set of metrics and their usage to evaluate the performance of end-to-end Session Initiation Protocol (SIP) for telephony services in both production and testing environments. The purpose of this document is to combine a standard set of common metrics, allowing interoperable performance measurements, easing the comparison of industry implementations.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6076.

Copyright Notice

Table of Contents

1. Introduction and Scope

SIP has become a widely used standard among many service providers, vendors, and end users in the telecommunications industry. Although there are many different standards for measuring the performance of telephony signaling protocols, such as Signaling System 7 (SS7), none of the metrics specifically address SIP.

The scope of this document is limited to the definitions of a standard set of metrics for measuring and reporting SIP performance from an end-to-end perspective in a telephony environment. The metrics introduce a common foundation for understanding and quantifying performance expectations between service providers, vendors, and the users of services based on SIP. The intended audience for this document can be found among network operators, who often collect information on the responsiveness of the network to customer requests for services.

Measurements of the metrics described in this document are affected by variables external to SIP. The following is a non-exhaustive list of examples:

- Network connectivity
- Switch and router performance
- Server processes and hardware performance

This document defines a list of pertinent metrics for varying aspects of a telephony environment. They may be used individually or as a set based on the usage of SIP within the context of a given telecommunications service.

The metrics defined in this document DO NOT take into consideration the impairment or failure of actual application processing of a request or response. The metrics do not distinguish application processing time from other sources of delay, such as packet transfer delay.

Metrics designed to quantify single device application processing performance are beyond the scope of this document.

This document does not provide any numerical objectives or acceptance threshold values for the SIP performance metrics defined below, as these items are beyond the scope of IETF activities, in general.

The metrics defined in this document are applicable in scenarios where the SIP messages launched (into a network under test) are dedicated messages for testing purposes, or where the messages are user-initiated and a portion of the live is traffic present. These two scenarios are sometimes referred to as active and passive measurement, respectively.

2. Terminology

The following terms and conventions will be used throughout 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 [RFC2119].

End-to-End - This is described as two or more elements utilized for initiating a request, receiving the request, and responding to the request. It encompasses elements as necessary to be involved in a session dialog between the originating user agent client (UAC), destination user agent server (UAS), and any interim proxies (may also include back-to-back user agents (B2BUAs)). This may be relative to a single operator's set of elements or may extend to encompass all elements (if beyond a single operator's network) associated with a session.

Session - As described in RFC 3261 [RFC3261], SIP is used primarily to request, create, and conclude sessions. "These sessions include Internet telephone calls, multimedia distribution, and multimedia conferences". The metrics within this document measure the performance associated with the SIP dialogs necessary to establish these sessions; therefore, they are titled as Session Request Delay, Session Disconnect Delay, etc. Although the titles of many of the metrics include this term, they are specifically measuring the signaling aspects only. Each session is identified by a unique "Call-ID", "To", and "From" header field tag.

Session Establishment - Session establishment occurs when a 200 OK response from the target UA has been received, in response to the originating UA's INVITE setup request, indicating the session setup request was successful.

Session Setup - As referenced within the sub-sections of Section 4.2 in this document, session setup is the set of messages and included parameters directly related to the process of a UA requesting to establish a session with a corresponding UA. This is also described as a set of steps in order to establish "ringing" [RFC3261].

3. Time Interval Measurement and Reporting

Many of the metrics defined in this memo utilize a clock to assess the time interval between two events. This section defines time-related terms and reporting requirements.

t1 - start time

This is the time instant (when a request is sent) that begins a continuous time interval. t1 occurs when the designated request has been processed by the SIP application and the first bit of the request packet has been sent from the UA or proxy (and is externally observable at some logical or physical interface).

t1 represents the time at which each request-response test begins, and SHALL be used to designate the time of day when a particular measurement was conducted (e.g., the Session Request Delay at "t1" (at some specific UA interface) was measured to be X ms).

t4 - end time

This is the time instant that concludes the continuous time interval begun when the related request is sent. t4 occurs when the last bit of the designated response is received by the SIP application at the requesting device (and is externally observable at some logical or physical interface).

> Note: The designations t2 and t3 are reserved for future use at another interface involved in satisfying a request.

Section 10.1 of [RFC2330] describes time-related issues in measurements, and defines the errors that can be attributed to the clocks themselves. These definitions are used in the material below.

Time-of-Day Accuracy

As defined above, t1 is associated with the start of a request and also serves as the time-of-day stamp associated with a single specific measurement. The clock offset [RFC2330] is the difference between t1 and a recognized primary source of time, such as UTC (offset = t1 - UTC).

When measurement results will be correlated with other results or information using time-of-day stamps, then the time clock that supplies t1 SHOULD be synchronized to a primary time source, to minimize the clock's offset. The clocks used at the different measurement points SHOULD be synchronized to each other, to minimize the relative offset (as defined in RFC2330). The clock's offset and the relative offset MUST be reported with each measurement.

Time Interval Accuracy

The accuracy of the t4-t1 interval is also critical to maintain and report. The difference between a clock's offsets at t1 and t4 is one source of error for the measurement and is associated with the clock's skew [RFC2330].

A stable and reasonably accurate clock is needed to make the time interval measurements required by this memo. This source of error SHOULD be constrained to less than +/- 1 ms, implying 1-part-per-1000 frequency accuracy for a 1-second interval. This implies that greater stability is required as the length of the t4-t1 increases, in order to constrain the error to be less than +/- 1 ms.

There are other important aspects of clock operation:

1. Synchronization protocols require some ability to make adjustments to the local clock. However, these adjustments (clock steps or slewing) can cause large errors if they occur during the t1 to t4 measurement interval. Clock correction SHOULD be suspended during a t1 to t4 measurement interval, unless the time interval accuracy requirement above will be met. Alternatively, a measurement SHOULD NOT be performed during clock correction, unless the time interval accuracy requirement above will be met.

2. If a free-running clock is used to make the time interval measurement, then the time of day reported with the measurement (which is normally timestamp t1) SHOULD be derived from a different clock that meets the time-of-day accuracy requirements described above.

The physical operation of reading time from a clock may be constrained by the delay to service the interrupt. Therefore, if the accuracy of the time stamp read at t1 or t4 includes the interrupt delay, this source of error SHOULD be known and included in the error assessment.

4. SIP Performance Metrics

In regard to all of the following metrics, t1 begins with the first associated SIP message sent by either UA, and is not reset if the UA must retransmit the same message, within the same transaction, multiple times. The first associated SIP message indicates the t1 associated with the user or application expectation relative to the request.

Some metrics are calculated using messages from different transactions in order to measure across actions such as redirection and failure recovery. The end time is typically based on a successful end-to-end provisional response, a successful final response, or a failure final response for which there is no recovery. The individual metrics detail which message to base the end time on.

The authentication method used to establish the SIP dialog will change the message exchanges. The example message exchanges used do not attempt to describe all of the various authentication types. Since authentication is frequently used, SIP Digest authentication was used for example purposes.

In regard to all of the metrics, the accuracy and granularity of the output values are related to the accuracy and granularity of the input values. Some of the metrics below are defined by a ratio. When the denominator of this ratio is 0, the metric is undefined.

While these metrics do not specify the sample size, this should be taken into consideration. These metrics will provide a better indication of performance with larger sample sets. For example, some SIP Service Providers (SSPs) [RFC5486] may choose to collect input over an hourly, daily, weekly, or monthly timeframe, while another SSP may choose to perform metric calculations over a varying set of SIP dialogs.

4.1. Registration Request Delay (RRD)

Registration Request Delay (RRD) is a measurement of the delay in responding to a UA REGISTER request. RRD SHALL be measured and reported only for successful REGISTER requests, while Ineffective Registration Attempts (Section 4.2) SHALL be reported for failures. This metric is measured at the originating UA. The output value of this metric is numerical and SHOULD be stated in units of milliseconds. The RRD is calculated using the following formula:

RRD = Time of Final Response - Time of REGISTER Request

In a successful registration attempt, RRD is defined as the time interval from when the first bit of the initial REGISTER message containing the necessary information is passed by the originating UA to the intended registrar, until the last bit of the 200 OK is received indicating the registration attempt has completed successfully. This dialog includes an expected authentication challenge prior to receiving the 200 OK as described in the following registration flow examples.

The following message exchange provides an example of identifiable events necessary for inputs in calculating RRD during a successful registration completion:

```
                  UA1                 Registrar
                   |                      |
                   |REGISTER              |
            t1---->|--------------------->|
               /\  |                   401|
               ||  |<---------------------|
              RRD  |REGISTER              |
               ||  |--------------------->|
               \/  |                   200|
            t4---->|<---------------------|
                   |                      |
```

Note: Networks with elements using primarily Digest authentication will exhibit different RRD characteristics than networks with elements primarily using other authentication mechanisms (such as Identity). Operators monitoring RRD in networks with a mixture of authentication schemes should take note that the RRD measurements will likely have a multimodal distribution.

4.2. Ineffective Registration Attempts (IRAs)

Ineffective registration attempts are utilized to detect failures or impairments causing the inability of a registrar to receive a UA REGISTER request. This metric is measured at the originating UA. The output value of this metric is numerical and SHOULD be reported as a percentage of registration attempts.

This metric is calculated as a percentage of total REGISTER requests. The IRA percentage is calculated using the following formula:

```
                          # of IRAs
     IRA % = ----------------------------- x 100
               Total # of REGISTER Requests
```

A failed registration attempt is defined as a final failure response to the initial REGISTER request. It usually indicates a failure received from the destination registrar or interim proxies, or failure due to a timeout of the REGISTER request at the originating UA. A failure response is described as a 4XX (excluding 401, 402, and 407 non-failure challenge response codes), 5XX, or possible 6XX message. A timeout failure is identified by the Timer F expiring. IRAs may be used to detect problems in downstream signaling functions, which may be impairing the REGISTER message from reaching the intended registrar; or, it may indicate a registrar has become overloaded and is unable to respond to the request.

The following message exchange provides a timeout example of an identifiable event necessary for input as a failed registration attempt:

```
                  UA1                Registrar
                   |                      |
                   |REGISTER              |
                   |--------------------->|
                   |REGISTER              |
                   |--------------------->|
                   |REGISTER              |
                   |--------------------->|
                   |                      |
      Failure ---->|***Timer F Expires    |
                   |                      |
```

In the previous message exchange, UA1 retries a REGISTER request multiple times before the timer expires, indicating the failure. Only the first REGISTER request MUST be used for input to the calculation and an IRA. Subsequent REGISTER retries are identified by the same transaction identifier (the same topmost Via header field branch parameter value) and MUST be ignored for purposes of metric calculation. This ensures an accurate representation of the metric output.

The following message exchange provides a registrar servicing failure example of an identifiable event necessary for input as a failed registration attempt:

```
                  UA1                Registrar
                   |                      |
                   |REGISTER              |
                   |--------------------->|
                   |                      |
                   |                      |
                   |                      |
                   |                      |
                   |                   503|
      Failure ---->|<---------------------|
                   |                      |
```

4.3. Session Request Delay (SRD)

Session Request Delay (SRD) is utilized to detect failures or impairments causing delays in responding to a UA session request. SRD is measured for both successful and failed session setup requests as this metric usually relates to a user experience; however, SRD for session requests ending in a failure MUST NOT be combined in the same result with successful requests. The duration associated with success and failure responses will likely vary substantially, and the desired output time associated with each will be significantly different in many cases. This metric is similar to Post-Selection Delay defined in [E.721], and it is measured at the originating UA only. The output value of this metric MUST indicate whether the output is for successful or failed session requests and SHOULD be stated in units of seconds. The SRD is calculated using the following formula:

   SRD = Time of Status Indicative Response - Time of INVITE

4.3.1. Successful Session Setup SRD

In a successful request attempt, SRD is defined as the time interval from when the first bit of the initial INVITE message containing the necessary information is sent by the originating user agent to the intended mediation or destination agent, until the last bit of the first provisional response is received indicating an audible or visual status of the initial session setup request. (Note: In some cases, the initial INVITE may be forked. Section 5.4 provides information for consideration on forking.) In SIP, the message indicating status would be a non-100 Trying provisional message received in response to an INVITE request. In some cases, a non-100 Trying provisional message is not received, but rather a 200 message is received as the first status message instead. In these situations, the 200 message would be used to calculate the interval. In most circumstances, this metric relies on receiving a non-100 Trying message. The use of the Provisional Response ACKnowledgement (PRACK) method [RFC3262] MAY improve the quality and consistency of the results.

The following message exchange provides an example of identifiable events necessary for inputs in calculating SRD during a successful session setup request without a redirect (i.e., 3XX message):

```
                  UA1                    UA2
                   |                      |
                   |INVITE                |
            t1---->|--------------------->|
               /\  |                      |
               ||  |                      |
              SRD  |                      |
               ||  |                      |
               \/  |                   180|
            t4---->|<---------------------|
                   |                      |
```

The following message exchange provides an example of identifiable events necessary for inputs in calculating SRD during a successful session setup with a redirect (e.g., 302 Moved Temporarily):

```
                  UA1             Redirect Server              UA2
                   |                      |                      |
                   |INVITE                |                      |
            t1---->|--------------------->|                      |
               /\  |                   302|                      |
               ||  |<---------------------|                      |
               ||  |ACK                   |                      |
              SRD  |--------------------->|                      |
               ||  |INVITE                                       |
               ||  |-------------------------------------------->|
               \/  |                                          180|
            t4---->|<--------------------------------------------|
```

4.3.2. Failed Session Setup SRD

In a failed request attempt, SRD is defined as the time interval from when the first bit of the initial INVITE message containing the necessary information is sent by the originating agent or user to the intended mediation or destination agent, until the last bit of the first provisional response or a failure indication response. A failure response is described as a 4XX (excluding 401, 402, and 407 non-failure challenge response codes), 5XX, or possible 6XX message. A change in the metric output might indicate problems in downstream signaling functions, which may be impairing the INVITE message from reaching the intended UA or may indicate changes in end-point behavior. While this metric calculates the delay associated with a failed session request, the metric Ineffective Session Attempts (Section 4.8) is used for calculating a ratio of session attempt failures.

The following message exchange provides an example of identifiable events necessary for inputs in calculating SRD during a failed session setup attempt without a redirect (i.e., 3XX message):

```
                  UA1                    UA2
                   |                      |
                   |INVITE                |
            t1---->|--------------------->|
               /\  |                      |
               ||  |                      |
              SRD  |                      |
               ||  |                      |
               \/  |                   480|
            t4---->|<---------------------|
                   |                      |
```

The following message exchange provides an example of identifiable events necessary for inputs in calculating SRD during a failed session setup attempt with a redirect (e.g., 302 Moved Temporarily):

```
                  UA1             Redirect Server               UA2
                   |                      |                      |
                   |INVITE                |                      |
            t1---->|--------------------->|                      |
               /\  |                   302|                      |
               ||  |<---------------------|                      |
               ||  |ACK                   |                      |
              SRD  |--------------------->|                      |
               ||  |INVITE                                       |
               ||  |-------------------------------------------->|
               \/  |                                          480|
            t4---->|<--------------------------------------------|
```

4.4. Session Disconnect Delay (SDD)

This metric is utilized to detect failures or impairments delaying the time necessary to end a session. SDD is measured for both successful and failed session disconnects; however, SDD for session disconnects ending in a failure MUST NOT be combined in the same result with successful disconnects. The duration associated with success and failure results will likely vary substantially, and the desired output time associated with each will be significantly different in many cases. It can be measured from either end-point UA involved in the SIP dialog. The output value of this metric is numerical and SHOULD be stated in units of milliseconds. The SDD is calculated using the following formula:

   SDD = Time of 2XX or Timeout - Time of Completion Message (BYE)

SDD is defined as the interval between the first bit of the sent session completion message, such as a BYE, and the last bit of the subsequently received 2XX response. In some cases, a recoverable error response, such as a 503 Retry-After, may be received. In such situations, these responses should not be used as the end time for this metric calculation. Instead, the successful (2XX) response related to the recovery message is used. The following message exchanges provide an example of identifiable events necessary for inputs in calculating SDD during a successful session completion:

Measuring SDD at the originating UA (UA1) -

```
                  UA1                    UA2
                   |                      |
                   |INVITE                |
                   |--------------------->|
                   |                   180|
                   |<---------------------|
                   |                   200|
                   |<---------------------|
                   |ACK                   |
                   |--------------------->|
                   |BYE                   |
            t1---->|--------------------->|
               /\  |                      |
               ||  |                      |
              SDD  |                      |
               ||  |                      |
               \/  |                   200|
            t4---->|<---------------------|
```

Measuring SDD at the target UA (UA2) -

```
                  UA1                    UA2
                   |                      |
                   |INVITE                |
                   |--------------------->|
                   |                   180|
                   |<---------------------|
                   |                   200|
                   |<---------------------|
                   |ACK                   |
                   |--------------------->|
                   |                   BYE|
                   |<---------------------|<----t1
                   |                      |  /\
                   |                      |  ||
                   |                      | SDD
                   |                      |  ||
                   |200                   |  \/
                   |--------------------->|<----t4
```

In some cases, no response is received after a session completion message is sent and potentially retried. In this case, the completion message, such as a BYE, results in a Timer F expiration. Sessions ending in this manner SHOULD be excluded from the metric calculation.

4.5. Session Duration Time (SDT)

This metric is used to detect problems (e.g., poor audio quality) causing short session durations. SDT is measured for both successful and failed session completions. It can be measured from either end-point UA involved in the SIP dialog. This metric is similar to Call Hold Time, and it is traditionally calculated as Average Call Hold Time (ACHT) in telephony applications of SIP. The output value of this metric is numerical and SHOULD be stated in units of seconds. The SDT is calculated using the following formula:

```
SDT = Time of BYE or Timeout - Time of 200 OK response to INVITE
```

This metric does not calculate the duration of sessions leveraging early media. For example, some automated response systems only use early media by responding with a SIP 183 Session Progress message with the Session Description Protocol (SDP) connecting the originating UA with the automated message. Usually, in these sessions the originating UA never receives a 200 OK, and the message exchange ends with the originating UA sending a CANCEL.

4.5.1. Successful Session Duration SDT

In a successful session completion, SDT is calculated as an average and is defined as the duration of a dialog defined by the interval between receipt of the first bit of a 200 OK response to an INVITE, and receipt of the last bit of an associated BYE message indicating dialog completion. Retransmissions of the 200 OK and ACK messages due to network impairments do not reset the metric timers.

The following message exchanges provide an example of identifiable events necessary for inputs in calculating SDT during a successful session completion. (The message exchanges are changed between the originating and target UAs to provide varying examples.):

Measuring SDT at the originating UA (UA1) -

```
                  UA1                   UA2
                   |                      |
                   |INVITE                |
                   |--------------------->|
                   |                   180|
                   |<---------------------|
                   |                   200|
            t1---->|<---------------------|
               /\  |ACK                   |
               ||  |--------------------->|
               ||  |                      |
              SDT  |                      |
               ||  |                      |
               ||  |                      |
               \/  |                   BYE|
            t4---->|<---------------------|
                   |                      |
```

When measuring SDT at the target UA (UA2), it is defined by the interval between sending the first bit of a 200 OK response to an INVITE, and receipt of the last bit of an associated BYE message indicating dialog completion. If UA2 initiates the BYE, then it is defined by the interval between sending the first bit of a 200 OK response to an INVITE, and sending the first bit of an associated BYE message indicating dialog completion. This is illustrated in the following example message exchange:

```
                  UA1                   UA2
                   |                      |
                   |INVITE                |
                   |--------------------->|
                   |                   180|
                   |<---------------------|
                   |                   200|
                   |<---------------------|<----t1
                   |ACK                   |  /\
                   |--------------------->|  ||
                   |                      |  ||
                   |                      |  SDT
                   |                      |  ||
                   |                      |  ||
                   |                   BYE|  \/
                   |<---------------------|<----t4
                   |                      |
```

(In these two examples, t1 is the same even if either UA receives the BYE instead of sending it.)

4.5.2. Failed Session Completion SDT

In some cases, no response is received after a session completion message is sent and potentially retried. In this case, SDT is defined as the interval between receiving the first bit of a 200 OK response to an INVITE, and the resulting Timer F expiration. The following message exchanges provide an example of identifiable events necessary for inputs in calculating SDT during a failed session completion attempt:

Measuring SDT at the originating UA (UA1) -

```
                  UA1                    UA2
                   |                      |
                   |INVITE                |
                   |--------------------->|
                   |                   180|
                   |<---------------------|
                   |                   200|
            t1---->|<---------------------|
               /\  |ACK                   |
               ||  |--------------------->|
               ||  |BYE                   |
              SDT  |--------------------->|
               ||  |BYE                   |
               ||  |--------------------->|
               \/  |                      |
            t4---->|***Timer F Expires    |
```

When measuring SDT at UA2, SDT is defined as the interval between sending the first bit of a 200 OK response to an INVITE, and the resulting Timer F expiration. This is illustrated in the following example message exchange:

```
                  UA1                    UA2
                   |                      |
                   |INVITE                |
                   |--------------------->|
                   |                   180|
                   |<---------------------|
                   |                   200|
                   |<---------------------|<----t1
                   |                   ACK|  /\
                   |--------------------->|  ||
                   |                   BYE|  ||
                   |<---------------------|  SDT
                   |                   BYE|  ||
                   |<---------------------|  ||
                   |                      |  \/
                   |    Timer F Expires***|<----t4
```

Note that in the presence of message loss and retransmission, the value of this metric measured at UA1 may differ from the value measured at UA2 up to the value of Timer F.

4.6. Session Establishment Ratio (SER)

This metric is used to detect the ability of a terminating UA or downstream proxy to successfully establish sessions per new session INVITE requests. SER is defined as the ratio of the number of new session INVITE requests resulting in a 200 OK response, to the total number of attempted INVITE requests less INVITE requests resulting in a 3XX response. This metric is similar to the Answer Seizure Ratio (ASR) defined in [E.411]. It is measured at the originating UA only. The output value of this metric is numerical and SHOULD be adjusted to indicate a percentage of successfully established sessions. The SER is calculated using the following formula:

```
             # of INVITE Requests w/ associated 200 OK
SER = --------------------------------------------------------- x 100
          (Total # of INVITE Requests) -
                   (# of INVITE Requests w/ 3XX Response)
```

The following message exchange provides an example of identifiable events necessary for inputs in determining session establishment as described above:

```
                           UA1                 UA2
                            |INVITE             |
               +----------->|------------------>|
               |            |                180|
               |            |<------------------|
      Session Established   |                   |
               |            |                200|
               +----------->|<------------------|
                            |                   |
```

The following is an example message exchange including a SIP 302 Redirect response.

```
                            UA1                 UA2                 UA3
                             |INVITE             |                   |
                +----------->|------------------>|                   |
                |            |                   |                   |
      INVITE w/ 3XX Response |                   |                   |
                |            |                302|                   |
                +----------->|<------------------|                   |
                             |                                       |
                             |INVITE                                 |
                +----------->|-------------------------------------->|
                |            |                                       |
                |            |                                    180|
       Session Established   |<--------------------------------------|
                |            |                                       |
                |            |                                    200|
                +----------->|<--------------------------------------|
                             |                                       |
```

4.7. Session Establishment Effectiveness Ratio (SEER)

This metric is complimentary to SER, but is intended to exclude the potential effects of an individual user of the target UA from the metric. SEER is defined as the ratio of the number of INVITE requests resulting in a 200 OK response and INVITE requests resulting in a 480, 486, 600, or 603; to the total number of attempted INVITE requests less INVITE requests resulting in a 3XX response. The response codes 480, 486, 600, and 603 were chosen because they clearly indicate the effect of an individual user of the UA. It is possible an individual user could cause a negative effect on the UA. For example, they may have misconfigured the UA, causing a response code not directly related to an SSP, but this cannot be easily determined from an intermediary B2BUA somewhere between the originating and terminating UAs. With this in consideration, response codes such as 401, 407, and 420 (not an exhaustive list) were not included in the numerator of the metric. This metric is similar to the Network Effectiveness Ratio (NER) defined in [E.411]. It is measured at the originating UA only. The output value of this metric is numerical and SHOULD be adjusted to indicate a percentage of successfully established sessions less common UAS failures.

The SEER is calculated using the following formula:

```
SEER =

 # of INVITE Requests w/ associated 200, 480, 486, 600, or 603
 ------------------------------------------------------------- x 100
         (Total # of INVITE Requests) -
                   (# of INVITE Requests w/ 3XX Response)
```

Reference the example flows in Section 4.6.

4.8. Ineffective Session Attempts (ISAs)

Ineffective session attempts occur when a proxy or agent internally releases a setup request with a failed or overloaded condition. This metric is similar to Ineffective Machine Attempts (IMAs) in telephony applications of SIP, and was adopted from Telcordia GR-512-CORE [GR-512]. The output value of this metric is numerical and SHOULD be adjusted to indicate a percentage of ineffective session attempts. The following failure responses provide a guideline for this criterion:

- 408 Request Timeout

- 500 Server Internal Error

- 503 Service Unavailable

- 504 Server Time-out

This set was derived in a similar manner as described in Section 4.7. In addition, 408 failure responses may indicate an overloaded state with a downstream element; however, there are situations other than overload that may cause an increase in 408 responses.

This metric is calculated as a percentage of total session setup requests. The ISA percentage is calculated using the following formula:

```
                            # of ISAs
          ISA % = ----------------------------- x 100
                   Total # of Session Requests
```

The following dialog [RFC3665] provides an example describing message exchanges of an ineffective session attempt:

```
          UA1           Proxy 1          Proxy 2            UA2
           |                |                |                |
           |INVITE          |                |                |
           |--------------->|                |                |
           |             407|                |                |
           |<---------------|                |                |
           |ACK             |                |                |
           |--------------->|                |                |
           |INVITE          |                |                |
           |--------------->|INVITE          |                |
           |             100|--------------->|INVITE          |
           |<---------------|             100|--------------->|
           |                |<---------------|                |
           |                |                |INVITE          |
           |                |                |--------------->|
           |                |                |                |
           |                |                |INVITE          |
           |                |                |--------------->|
           |                |                |                |
           |                |             408|                |
           |             408|<---------------|                |
           |<---------------|ACK             |                |
           |                |--------------->|                |
           |ACK             |                |                |
           |--------------->|                |                |
```

4.9. Session Completion Ratio (SCR)

A session completion is defined as a SIP dialog, which completes without failing due to a lack of response from an intended proxy or UA. This metric is similar to the Call Completion Ratio (CCR) in telephony applications of SIP. The output value of this metric is numerical and SHOULD be adjusted to indicate a percentage of successfully completed sessions.

This metric is calculated as a percentage of total sessions completed successfully. The SCR percentage is calculated using the following formula:

```
                   # of Successfully Completed Sessions
         SCR % = --------------------------------------- x 100
                        Total # of Session Requests
```

The following dialog [RFC3665] provides an example describing the necessary message exchanges of a successful session completion:

```
          UA1           Proxy 1          Proxy 2             UA2
           |                |                |                |
           |INVITE          |                |                |
           |--------------->|                |                |
           |             407|                |                |
           |<---------------|                |                |
           |ACK             |                |                |
           |--------------->|                |                |
           |INVITE          |                |                |
           |--------------->|INVITE          |                |
           |             100|--------------->|INVITE          |
           |<---------------|             100|--------------->|
           |                |<---------------|                |
           |                |                |             180|
           |                |            180 |<---------------|
           |             180|<---------------|                |
           |<---------------|                |             200|
           |                |             200|<---------------|
           |             200|<---------------|                |
           |<---------------|                |                |
           |ACK             |                |                |
           |--------------->|ACK             |                |
           |                |--------------->|ACK             |
           |                |                |--------------->|
           |                Both Way RTP Media                |
           |<================================================>|
           |                |                |             BYE|
           |                |             BYE|<---------------|
           |             BYE|<---------------|                |
           |<---------------|                |                |
           |200             |                |                |
           |--------------->|200             |                |
           |                |--------------->|200             |
           |                |                |--------------->|
           |                |                |                |
```

5. Additional Considerations

5.1. Metric Correlations

These metrics may be used to determine the performance of a domain and/or user. The following is an example subset of dimensions for providing further granularity per metric:

- To "user"
- From "user"
- Bi-direction "user"
- To "domain"
- From "domain"
- Bi-direction "domain"

5.2. Back-to-Back User Agent (B2BUA)

A B2BUA may impact the ability to collect these metrics with an end-to-end perspective. It is necessary to realize that a B2BUA may act as an originating UAC and terminating UAS, or it may act as a proxy. In some cases, it may be necessary to consider information collected from both sides of the B2BUA in order to determine the end-to-end perspective. In other cases, the B2BUA may act simply as a proxy allowing data to be derived as necessary for the input into any of the listed calculations.

5.3. Authorization and Authentication

During the process of setting up a SIP dialog, various authentication methods may be utilized. These authentication methods will add to the duration as measured by the metrics, and the length of time will vary based on those methods. The failures of these authentication methods will also be captured by these metrics, since SIP is ultimately used to indicate the success or failure of the authorization and/or authentication attempt. The metrics in Section 3 are inclusive of the duration associated with this process, even if the method is external to SIP. This was included purposefully, due to its inherent impact on the protocol and the subsequent SIP dialogs.

5.4. Forking

Forking SHOULD be considered when determining the messages associated with the input values for the described metrics. If all of the forked dialogs were used in the metric calculations, the numbers would skew dramatically. There are two different points of forking, and each MUST be considered. First, forking may occur at a proxy downstream from the UA that is being used for metric input values. The downstream proxy is responsible for forking a message. Then, this proxy will send provisional (e.g., 180) messages received from the requests and send the accepted (e.g., 200) response to the UA.

Second, in the cases where the originating UA or proxy is forking the messages, then it MUST parse the message exchanges necessary for input into the metrics. For example, it MAY utilize the first INVITE or set of INVITE messages sent and the first accepted 200 OK. Tags will identify this dialog as distinct from the other 200 OK responses, which are acknowledged, and an immediate BYE is sent. The application responsible for capturing and/or understanding the input values MUST utilize these tags to distinguish between dialog requests.

Note that if an INVITE is forked before reaching its destination, multiple early dialogs are likely, and multiple confirmed dialogs are possible (though unlikely). When this occurs, an SRD measurement should be taken for each dialog that is created (early or confirmed).

5.5. Data Collection

The input necessary for these calculations may be collected in a number of different manners. It may be collected or retrieved from call detail records (CDRs) or raw signaling information generated by a proxy or UA. When using records, time synchronization MUST be considered between applicable elements.

If these metrics are calculated at individual elements (such as proxies or endpoints) instead of by a centralized management system, and the individual elements use different measurement sample sizes, then the metrics reported for the same event at those elements may differ significantly.

The information may also be transmitted through the use of network management protocols like the Simple Network Management Protocol (SNMP) and via future extensions to the SIP Management Information Base (MIB) modules [RFC4780], or through a potential undefined new performance metric event package [RFC3265] retrieved via SUBSCRIBE requests.

Data may be collected for a sample of calls or all calls, and may also be derived from test call scenarios. These metrics are flexible based on the needs of the application.

For consistency in calculation of the metrics, elements should expect to reveal event inputs for use by a centralized management system, which would calculate the metrics based on a varying set sample size of inputs received from elements compliant with this specification.

5.6. Testing Documentation

In some cases, these metrics will be used to provide output values to signify the performance level of a specific SIP-based element. When using these metrics in a test environment, the environment MUST be accurately documented for the purposes of replicating any output values in future testing and/or validation.

6. Conclusions

This document provides a description of common performance metrics and their defined use with SIP. The use of these metrics will provide a common viewpoint across all vendors, service providers, and users. These metrics will likely be utilized in production telephony SIP environments for providing input regarding Key Performance Indicators (KPI) and Service Level Agreement (SLA) indications; however, they may also be used for testing end-to-end SIP-based service environments.

7. Security Considerations

Security should be considered in the aspect of securing the relative data utilized in providing input to the above calculations. All other aspects of security should be considered as described in RFC 3261 [RFC3261].

Implementers of these metrics MUST realize that these metrics could be used to describe characteristics of customer and user usage patterns, and privacy should be considered when collecting, transporting, and storing them.

8. Contributors

The following people made substantial contributions to this work:

Carol Davids	Illinois Institute of Technology
Marian Delkinov	Ericsson
Adam Uzelac	Global Crossing
Jean-Francois Mule	CableLabs
Rich Terpstra	Level 3 Communications

9. Acknowledgements

We would like to thank Robert Sparks, John Hearty, and Dean Bayless for their efforts in reviewing the document and providing insight regarding clarification of certain aspects described throughout the document. We also thank Dan Romascanu for his insightful comments and Vijay Gurbani for agreeing to perform the role of document shepherd.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[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.

[RFC3262] Rosenberg, J. and H. Schulzrinne, "Reliability of Provisional Responses in Session Initiation Protocol (SIP)", RFC 3262, June 2002.

[RFC3265] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.

[RFC3665] Johnston, A., Donovan, S., Sparks, R., Cunningham, C., and K. Summers, "Session Initiation Protocol (SIP) Basic Call Flow Examples", BCP 75, RFC 3665, December 2003.

[RFC4780] Lingle, K., Mule, J-F., Maeng, J., and D. Walker, "Management Information Base for the Session Initiation Protocol (SIP)", RFC 4780, April 2007.

10.2. Informative References

[E.411] ITU-T, "Series E: Overall Network Operation, Telephone Service, Service Operation and Human Factors", E.411 , March 2000.

[E.721] ITU-T, "Series E: Overall Network Operation, Telephone Service, Service Operation and Human Factors", E.721 , May 1999.

[GR-512] Telcordia, "LSSGR: Reliability, Section 12", GR-512-CORE Issue 2, January 1998.

[RFC2330] Paxson, V., Almes, G., Mahdavi, J., and M. Mathis, "Framework for IP Performance Metrics", RFC 2330, May 1998.

[RFC5486] Malas, D. and D. Meyer, "Session Peering for Multimedia Interconnect (SPEERMINT) Terminology", RFC 5486, March 2009.

Authors' Addresses

Daryl Malas
CableLabs
858 Coal Creek Circle
Louisville, CO 80027
US

Phone: +1 303 661 3302
EMail: d.malas@cablelabs.com

Al Morton
AT&T Labs
200 Laurel Avenue South
Middletown, NJ 07748
US

Phone: +1 732 420 1571
EMail: acmorton@att.com