SIPCORE
Internet-Draft
Intended status: Standards Track
Expires: January 24, 2019

E. Burger
Georgetown University
July 23, 2018

A Session Initiation Protocol (SIP) Response Code for Rejected Calls
draft-burger-sipcore-rejected-01

Abstract

This document defines the 608 (Rejected) SIP response code. This response code enables calling parties to learn their call was rejected by an intermediary and will not be answered. As a 6xx code, the caller will be aware that future attempts to contact the same UAS will be likely to fail. The present use case driving the need for the 608 response code is when the intermediary is an analytics engine. In this case, the rejection is by a machine or other process. This contrasts with the 607 (Unwanted) SIP response code, which a human at the target UAS indicated the call was not wanted. In some jurisdictions this distinction is important and may have additional requirements beyond the 607 response code. Specifically, this document defines the use of the Call-Info header in 608 responses to enable rejected callers to contact entities that blocked their calls in error.

Status of This Memo

Copyright Notice

1. Introduction

The IETF has been addressing numerous issues surrounding how to handle unwanted and, depending on the jurisdiction, illegal calls [RFC5039]. Technologies such as STIR [RFC7340] and SHAKEN [SHAKEN] address cryptographic signing and attestation, respectively, of signaling to ensure the integrity and authenticity of the asserted identity.

This document describes a new SIP response code, 608, which allows calling parties to learn an intermediary rejected their call. As described below, we need a distinct indicator to differentiate between a user rejection and an intermediary's rejection of a call. In some jurisdictions, calls, even if unwanted by the user, may not be blocked unless there is an explicit user request. Moreover, users may misidentify the nature of a caller. For example, a legitimate caller may call a user who finds the call to be unwanted. However, instead of marking the call as unwanted, the user may mark the call as illegal. With that information, an analytics engine may determine that all calls from that source should be blocked. However, in some jurisdictions blocking calls from that source for other users may not be legal. Likewise, one can envision jurisdictions that allow an operator to block such calls, but only if there is a remediation mechanism in place to address false positives.

Some call blocking services may return responses such as 604 (Does Not Exist Anywhere). This might be a strategy to attempt to get a destination's address removed from a calling database. However, other network elements might interpret this to mean the user truly does not exist and result in the user not being able to receive calls from anyone, even if wanted. As well, in many jurisdictions, providing false signaling is illegal.

The 608 response code addresses this need of remediating falsely blocked calls. Specifically, this code informs the UAC an intermediary blocked the call and, to satisfy some jurisdictional requirements for providing a redress mechanism, how to contact the operator of the intermediary.

In the call handling ecosystem, users can explicitly reject a call or later mark a call as being unwanted by issuing a 607 SIP response code (Unwanted) [RFC8197]. Figure 1 and Figure 2 shows the operation of the 607 SIP response code. The UAS indicates the call was unwanted. As RFC8197 explains, not only does the called party desire to reject that call, they may wish to let their proxy know they might consider future calls from that source unwanted. Upon receipt of the 607 response from the UAS, the proxy may send call information to a call analytics engine. For various reasons described in RFC8197, if a network operator receives multiple reports of unwanted calls, that may indicate the entity placing the calls is likely to be a source of unwanted calls for many people. As such, other users of the service provider’s service may wish the service provider to automatically reject calls on their behalf based on that and other analytics.

Another value of the 607 rejection is presuming the proxy forwards the response code to the UAC, the calling UAC or intervening proxies know the user is not interested in receiving calls from that sender.

```
                      +-----------+
                      |   Call    |
                      | Analytics |
                      |  Engine   |
                      +-----------+
                         ^     | (likely not SIP)
                         |     v
                      +-----------+
   +-----+    607     |  Called   |    607    +-----+
   | UAC | <--------- |  Party    | <-------- | UAS |
   +-----+            |  Proxy    |           +-----+
                      +-----------+
```

Figure 1: Unwanted (607) Call Flow

For calls rejected with a 607 from a legitimate caller, receiving a 607 response code can inform the caller to stop attempting to call the user. Moreover, if the legitimate caller believes the user is rejecting their calls in error, they can use other channels to contact the user. For example, if a pharmacy calls a user to let them know their prescription is available for pickup and the user mistakenly thinks the call is unwanted and issues a 607 response code, the pharmacy, having an existing relationship with the customer, can send the user an email, also noting they might consider not rejecting their calls in the future.

Moreover, many systems that allow the user to mark the call unwanted (e.g., with the 607 response code) also allow the user to change their mind and unmark such calls. This is relatively easy to implement as the user usually has a direct relationship with the provider of the blocking service.

```
                       +--------+         +-----------+
                       | Called |         |   Call    |
      +-----+          | Party  |         | Analytics |   +-----+
      | UAC |          | Proxy  |         |  Engine   |   | UAS |
      +-----+          +--------+         +-----------+   +-----+
         |  INVITE         |                    |            |
         | --------------> |  INVITE            |            |
         |                 | ------------------------------> |
         |                 |                    |            |
         |                 |                    |       607  |
         |                 | <------------------------------ |
         |                 |                    |            |
         |                 |  Unwanted call     |            |
         |            607  | -----------------> |            |
         | <-------------- |  indicator         |            |
         |                 |                    |            |
```

Figure 2: Unwanted (607) Ladder Diagram

However, things get more complicated if an intermediary, such as a third-party provider of call management services that classify calls based on the relative likelihood the call is unwanted, misidentifies the call as unwanted. Figure 3 shows this case. Note the UAS typically does not receive an INVITE as the proxy rejects the call on behalf of the user. In this situation, it would be beneficial for the caller to be able to learn who rejected the call, so they might be able to correct the misidentification.

In this situation, one might be tempted to have the intermediary use the 607 response code. 607 indicates to the caller the subscriber did not get the call and they do not want the call. However, RFC8197 specifies that one of the uses of 607 is to inform analytics engines that a user (human) has rejected a call. The problem here is network elements downstream from the intermediary might interpret the 607 as a user (human) marking the call as unwanted, as opposed to a statistical, machine learning, vulnerable to the base rate fallacy [BaseRate] algorithm rejecting the call. In other words, those downstream entities should not be relying on another entity 'deciding' the call is unwanted. By distinguishing between a (human) user rejection and an intermediary's statistical rejection, a downstream network element that sees a 607 response code can weight it as a human rejection in its call analytics.

```
                          +-----------+
                          |   Call    |
                          | Analytics |
                          |  Engine   |
                          +-----------+
                             ^     | (likely not SIP)
                             |     v
                          +-----------+
      +-----+    608      |  Called   |           +-----+
      | UAC | <---------  |  Party    |           | UAS |
      +-----+             |  Proxy    |           +-----+
                          +-----------+
```

Figure 3: Rejected (608) Call Flow

It is useful for blocked callers to have a redress mechanism. One can imagine that some jurisdictions will require it. However, we must be mindful that most of the calls that will be blocked will, in fact, be illegal and eligible for blocking. Thus, providing alternate contact information for a user would be counterproductive to protecting that user from illegal communications. This is another reason we do not propose to simply allow alternate contact information in a 607 response message.

One might ask why we cannot use the same mechanism an analytics service provider offers their customers that lets them correct a call blocked in error? The reason is whilst there is an existing relationship between the customer (called party) and the analytics service provider, it is unlikely there is a relationship between the caller and the analytics service provider. Moreover, there are numerous call blocking providers in the ecosystem. As such, we need a mechanism for indicating an intermediary rejected a call while providing contact information for the operator of the intermediary that provides call rejection services to the called party, without exposing the target user’s contact information.

2. Terminology

This document uses the terms "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" as described in BCP14 [RFC2119][RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Protocol Operation

For clarity, this section uses the term 'intermediary' as the entity that acts as a SIP User Agent Server (UAS) on behalf of the user in the network, as opposed to the user's UAS (colloquially, but not necessarily, their phone). The intermediary could be a back-to-back user agent (B2BUA) or a SIP Proxy.

Figure 4 shows an overview of the call flow for a rejected call.

```
                       +--------+         +-----------+
                       | Called |         |   Call    |
      +-----+          | Party  |         | Analytics |   +-----+
      | UAC |          | Proxy  |         |  Engine   |   | UAS |
      +-----+          +--------+         +-----------+   +-----+
         |  INVITE           |                    |            |
         | --------------> |  Information from  |            |
         |                 | -----------------> |            |
         |                 |  INVITE            |            |
         |                 |           Reject   |            |
         |            608  | <----------------- |            |
         | <-------------- |           call     |            |
         |                 |                    |            |
```

Figure 4: Rejected (608) Ladder Diagram

3.1. Intermediary Operation

An intermediary MAY issue the 608 response code in a failure response for an INVITE, MESSAGE, SUBSCRIBE, or other out-of-dialog SIP [RFC3261] request to indicate that an intermediary rejected the offered communication as unwanted by the user. An intermediary MAY issue the 608 as the value of the "cause" parameter of a SIP reason-value in a Reason header field [RFC3326].

Unless there are indicators the calling party will use the contents of the Call-Info header for malicious purposes (see Section 6), if an intermediary issues a 608 code, the intermediary MUST include a Call-Info header in the response.

If there is a Call-Info header, it MUST have the 'purpose' parameter of 'card'. The value of the Call-Info header MUST refer to a valid vCard [RFC6350] object.

The vCard referenced in the Call-Info header MUST include at least one of the URL, EMAIL, TEL, or ADR properties. UACs supporting this specification MUST be prepared to receive a full vCard. Call originators (at the UAC) can use the information returned by the vCard to contact the intermediary that rejected the call to appeal the intermediary’s blocking of the call attempt. What the intermediary does if the blocked caller contacts the intermediary is outside the scope of this document.

Proxies need to be mindful that a downstream intermediary may reject the attempt with a 608 while other paths may still be in progress. In this situation, the requirements stated in Section 16.7 of RFC3261 [RFC3261] apply. Specifically, the proxy should cancel pending transactions and must not create any new branches. Note this is not a new requirement but simply pointing out the existing 6xx protocol mechanism in SIP.

3.2. UAC Operation

A UAC conforming to this specification MUST include the sip.608 feature capability tag in the INVITE request.

Upon receiving a 608 response, UACs perform normal SIP processing for 6xx responses.

3.3. Legacy Interoperation

If the UAC indicates support for 608 and the intermediary issues a 608, life is good as the UAC will receive all the information it needs to remediate an erroneous block by an intermediary. However, what if the UAC does not understand 608? Besides a UAC predating this specification, the could occur for callers from the legacy, non-SIP public switched network connecting to the SIP network via a media gateway.

We address this situation by having the first network element that conforms with this specification play an announcement in the media. See Section 3.4 for requirements on the announcement. The simple rule is a network element that inserts the sip.608 feature capability MUST be able to convey at a minimum whom to contact, ideally how to contact, the operator of the intermediary that rejected the call attempt.

The degenerate case is the intermediary is the only element that understands the semantics of the 608 response code. Obviously, any SIP device will understand that a 608 response code is a 6xx error. However, there are no other elements in the call path that understand the meaning of the value of the Call-Info header. The intermediary knows this is the case as the INVITE request will not have the sip.608 feature capability. In this case, one can consider the intermediary to be the element ’inserting’ a virtual sip.608 feature capability. As such, the intermediary MUST play the announcement, with the caveats described in Section 3.4 and Section 6.

Now we take the case where a network element that understands the 608 response code receives an INVITE for further processing. A network element conforming with this specification MUST insert the sip.608 feature capability, per the behaviors described in Section 4.2 of [RFC6809]. This information will be in the vCard referenced by the Call-Info header in the 608 response message. Note this specification does not specify the mechanism for such notification to the UAC (see Section 3.4).

Do note that even if a network element plays an announcement describing the contents of the 608 response message, the network element MUST also send the 608 response code message as the final response to the INVITE.

One aspect of using a feature capability is only the network elements that will consume (UAC) or play an announcement (media gateway, SBC, or proxy) need understand the sip.608 feature capability. All other (existing) infrastructure can remain without modification, assuming they are conformant to Section 16.6 of [RFC3261], specifically they will pass headers such as "Feature-Capability: sip.608" unmodified.

3.4. Announcement Requirements

There are a few requirements on the element that will be doing the announcement for legacy interoperation.

As noted above, the element that inserts the sip.608 feature capability is responsible for conveying the information referenced by the Call-Info header in the 608 response message. However, this specification does not mandate the modality for conveying that information.

Let us take the case where a telecommunications service provider controls the element inserting the sip.608 feature capability. It would be reasonable to expect the service provider would play an actual announcement in the media path towards the UAC (caller). It is important to note the network element should be mindful of the media type requested by the UAC as it formulates the announcement. For example, it would make sense for an INVITE that only indicated audio codecs in the SDP [RFC4566] to result in an audio announcement. However, if the INVITE only indicated a real-time text codec, for example, the network element SHOULD send the information in a text format, not an audio format, unless the network element is unable to render the information in the requested media format.

It is also possible for the network element inserting the sip.608 feature capability to be under the control of the same entity that controls the UAC. For example, a large call center might have legacy UACs, but have a modern outbound calling proxy that understands the full semantics of the 608 response code. In this case, it is enough for the outbound calling proxy to digest the Call-Info information and handle the information digitally, rather than 'transcoding' the Call-Info information for presentation to the caller.

4. Examples

These examples are not normative, for clarity do not include all protocol elements, and may have errors. Review the protocol documents for actual syntax and semantics of the protocol elements.

Given an INVITE (shamelessly taken from [SHAKEN]):

```
INVITE sip:+12155551213@tel.example1.net SIP/2.0
Max-Forwards: 69
Contact: <sip:+12155551212@69.241.19.12:50207;rinstance=9da3088f36cc>
To: <sip:+12155551213@tel.example1.net>
From: "Alice" <sip:+12155551212@tel.example2.net>;tag=614bdb40
Call-ID: 79048YzkxNDA5NTI1MzA0OWFjOTFkMmFlODhiNTI2OWQ1ZTI
P-Asserted-Identity: "Alice"<sip:+12155551212@tel.example2.net>,
    <tel:+12155551212>
CSeq: 2 INVITE
Allow: SUBSCRIBE, NOTIFY, INVITE, ACK, CANCEL, BYE, REFER, INFO,
    MESSAGE, OPTIONS
Content-Type: application/sdp
Date: Tue, 16 Aug 2016 19:23:38 GMT
Feature-Caps: sip.608
Identity:
eyJhbGciOiJFUzI1NiIsInR5cCI6InBhc3Nwb3J0IiwicHB0Ijoic2hha2VuIiwieDV1I
joiaHR0cDovL2NlcnQtYXV0aC5wb2Muc3lzLmNvbWNhc3QubmV0L2V4YW1wbGUuY2VydC
J9eyJhdHRlc3QiOiJBIiwiZGVzdCI6eyJ0biI6IisxMjE1NTU1MTIxMyJ9LCJpYXQiOiI
xNDcxMzc1NDE4Iiwib3JpZyI6eyJ0biI64oCdKzEyMTU1NTUxMjEyIn0sIm9yaWdpZCI6
IjEyM2U0NTY3LWU4OWItMTJkMy1hNDU2LTQyNjY1NTQ0MDAwMCJ9._28kAwRWnheXyA6n
Y4MvmK5JKHZH9hSYkWI4g75mnq9Tj2lW4WPm0PlvudoGaj7wM5XujZUTb_3MA4modoDtC
A;info=<http://cert.example2.net/example.cert>;alg=ES256
Content-Length: 153

v=0
o=- 13103070023943130 1 IN IP4 192.0.2.177
c=IN IP4 192.0.2.177
t=0 0
m=audio 54242 RTP/AVP 0
a=sendrecv
```

An intermediary could reply:

```
SIP/2.0 608 Rejected
Via: SIP/2.0/UDP 192.0.2.177:60012;branch=z9hG4bK-524287-1
From: "Alice" <sip:+12155551212@tel.example2.net>;tag=614bdb40
To: <sip:+12155551213@tel.example1.net>
Call-ID: 79048YzkxNDA5NTI1MzA0OWFjOTFkMmFlODhiNTI2OWQ1ZTI
CSeq: 2 INVITE
Call-Info: <https://blocker.example.net/complaints.vcf>;purpose=card
```

A minimal vCard, in this example at https://blocker.example.net/complaints.vcf, could contain:

```
BEGIN:VCARD
VERSION:4.0
FN:Robocall Adjudication
EMAIL;TYPE=work:bitbucket@blocker.example.net
END:VCARD
```

For an intermediary that provides a Web site for adjudication, the vCard could contain:

```
BEGIN:VCARD
VERSION:4.0
FN:Robocall Adjudication
URL;TYPE=work:https://blocker.example.net/adjudication-form
END:VCARD
```

For an intermediary that provides a telephone number and a postal address, the vCard could contain:

```
BEGIN:VCARD
VERSION:4.0
FN:Robocall Adjudication
ADR;TYPE=work;Argument Clinic;12 Main St;Anytown;AP;000000;Somewhere
TEL;VALUE=uri;TYPE=work:tel:+1-555-555-1212
END:VCARD
```

Note that it is up to the UAC to decide which vCard contact modality, if any, it will use.

Figure 5 depicts a call flow illustrating legacy interoperability. In this non-normative example, we see a UAC that does not support the full semantics for 608. However, there is an SBC that does support 608. Per RFC6809 [RFC6809], the SBC can insert "sip.608" into the Feature-Caps header for the INVITE. When the intermediary, labeled "Called Party Proxy" in the figure, rejects the call, it knows it can simply perform the processing described in this document. Since the intermediary saw the sip.608 feature capability, it knows it does not need to send any media describing whom to contact in the event of an erroneous rejection.

```
     +---------+
     |  Call   |
     |Analytics|
     | Engine  |
     +---------+
        ^   |
        |   v
     +---------+
     | Called  |     +-----+     +-----+     +---+     +-----+     +---+
     | Party   | <---|Proxy| <---|Proxy| <---|SBC| <---|Proxy| <---|UAC|
     | Proxy   |     +-----+     +-----+     +---+     +-----+     +---+
     +---------+                               |                     |
          |                                    |                     |
          |                                    |              INVITE |
          |                             INVITE |<--------------------|
          |<-----------------------------------|                     |
          |              Feature-Caps: sip.608 |                     |
          |                                    |                     |
          | 608 Rejected                       |                     |
          |----------------------------------->| 183                 |
          | Call-Info: <...>                   |-------------------->|
          |     [path for Call-Info elided     | SDP for media       |
          |      for illustration purposes]    |                     |
          |                                    |=== Announcement ===>|
          |                                    |                     |
          |                                    | 608                 |
          |                                    |-------------------->|
          |                                    | Call-Info: <...>    |
```

Figure 5: Legacy Operation

When the SBC receives the 608 response code, it correlates that with the original INVITE from the UAC. The SBC remembers that it inserted the sip.608 feature capability, which means it is responsible for somehow alerting the UAC the call failed and whom to contact. At this point the SBC can play a prompt, either natively or through a mechanism such as NETANN [RFC4240], that sends the relevant information in the appropriate media to the UAC.

As an example, the SBC could extract the FN and TEL vCard fields and play something like a special information tone (see Telcordia SR-2275 [SR-2275] section 6.21.2.1 or ITU-T E.180 [ITU.E.180.1998] section 7), followed by "Your call has been rejected by ...", followed by a text-to-speech translation of the FN text, followed by "You can reach them on", followed by a text-to-speech translation of the telephone number in the TEL field.

Note the SBC also still sends the full 608 response code, including the Call-Info header, towards the UAC.

5. IANA Considerations

5.1. SIP Response Code

This document defines a new SIP response code, 608. Please register the response code in the "Response Codes" subregistry of the "Session Initiation Protocol (SIP) Parameters" registry at <http://www.iana.org/assignments/sip-parameters>.

Response code: 608

Description: Rejected

Reference: [RFCXXXX]

5.2. SIP Feature-Capability Indicator

This document defines the feature capability sip.608 in the "SIP Feature-Capability Indicator Registration Tree" registry defined in [RFC6809].

Name: sip.608

Description: This feature capability indicator, when included in a Feature-Caps header field of an INVITE request, indicates that the entity that inserted the sip.608 Feature-Caps value will be responsible for indicating to the caller any information contained in the 608 SIP response code, specifically the value referenced by the Call-Info header.

Reference: [RFCXXXX]

6. Security Considerations

Intermediary operators need to be mindful of whom they are sending the 608 response to. There is a risk that a truly malicious caller is being rejected. This raises two issues. The first is the caller, being alerted their call is being automatically rejected, may change their call behavior to defeat call blocking systems. The second, and more significant risk, is that by providing a contact modality in the Call-Info field, the intermediary may be giving the malicious caller a vector for attack. In other words, the intermediary will be publishing an address that a malicious actor may use to launch an attack on the intermediary. Because of this, intermediary operators may wish to configure their response to only include a Call-Info field for INVITE or other initiating methods that are signed and pass validation by STIR [RFC8224].

Another risk is for an attacker to purposely not include the sip.608 feature capability in a flood of INVITE requests, direct those requests to stateless proxies, and direct the Contact header to a victim device. Because the mechanism described here can result in an audio file being sent to the target of the Contact header, an attacker could use the mechanism described by this document as an amplification attack, given a SIP INVITE can be under 1 kilobyte and an audio file can be hundreds of kilobytes. One remediation for this is for devices that insert a sip.608 feature capability only transmit media to what is highly likely to be the actual source of the call attempt. A method for this is to only play media in response to an INVITE that is signed and passed validation by STIR [RFC8224].

7. Acknowledgements

This document liberally lifts from [RFC8197] in its text and structure. However, the mechanism and purpose of 608 is quite different than 607. Any errors are the current editor’s and not the editor of RFC8197. Thanks also go to Ken Carlberg of the FCC, Russ Housley, Paul Kyzivat, and Tolga Asveren for their suggestions on improving the draft. Tolga’s suggestion to provide a mechanism for legacy interoperability served to expand the draft by 50%.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <https://www.rfc-editor.org/info/rfc3261>.

[RFC3326] Schulzrinne, H., Oran, D., and G. Camarillo, "The Reason Header Field for the Session Initiation Protocol (SIP)", RFC 3326, DOI 10.17487/RFC3326, December 2002, <https://www.rfc-editor.org/info/rfc3326>.

[RFC6350] Perreault, S., "vCard Format Specification", RFC 6350, DOI 10.17487/RFC6350, August 2011, <https://www.rfc-editor.org/info/rfc6350>.

[RFC6809] Holmberg, C., Sedlacek, I., and H. Kaplan, "Mechanism to Indicate Support of Features and Capabilities in the Session Initiation Protocol (SIP)", RFC 6809, DOI 10.17487/RFC6809, November 2012, <https://www.rfc-editor.org/info/rfc6809>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

8.2. Informative References

[BaseRate]
Bar-Hillel, M., "The Base-Rate Fallacy in Probability Judgements", 4 1977, <http://www.dtic.mil/get-tr-doc/pdf?AD=ADA045772>.

[ITU.E.180.1998]
International Telecommunications Union, "Technical characteristics of tones for the telephone service", ITU Recommendation E.180/Q.35, March 1998.

[RFC4240] Burger, E., Ed., Van Dyke, J., and A. Spitzer, "Basic Network Media Services with SIP", RFC 4240, DOI 10.17487/RFC4240, December 2005, <https://www.rfc-editor.org/info/rfc4240>.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <https://www.rfc-editor.org/info/rfc4566>.

[RFC5039] Rosenberg, J. and C. Jennings, "The Session Initiation Protocol (SIP) and Spam", RFC 5039, DOI 10.17487/RFC5039, January 2008, <https://www.rfc-editor.org/info/rfc5039>.

[RFC7340] Peterson, J., Schulzrinne, H., and H. Tschofenig, "Secure Telephone Identity Problem Statement and Requirements", RFC 7340, DOI 10.17487/RFC7340, September 2014, <https://www.rfc-editor.org/info/rfc7340>.

[RFC8197] Schulzrinne, H., "A SIP Response Code for Unwanted Calls", RFC 8197, DOI 10.17487/RFC8197, July 2017, <https://www.rfc-editor.org/info/rfc8197>.

[RFC8224] Peterson, J., Jennings, C., Rescorla, E., and C. Wendt, "Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 8224, DOI 10.17487/RFC8224, February 2018, <https://www.rfc-editor.org/info/rfc8224>.

[SHAKEN] Alliance for Telecommunications Industry Solutions (ATIS) and the SIP Forum, "Signature-based Handling of Asserted information using toKENs (SHAKEN)", ATIS 1000074, 1 2017, <https://www.sipforum.org/download/sip-forum-twg-10-signature-based-handling-of-asserted-information-using-tokens-shaken-pdf/?wpdmdl=2813>.

[SR-2275] Telcordia, "Bellcore Notes on the Networks", Telcordia SR-2275, October 2000.

Author's Address

Eric W. Burger
Georgetown University
37th & O St, NW
Washington, DC 20057
USA

Email: eburger@standardstrack.com