Audio/Video Transport (avt) H. Schulzrinne
Internet-Draft Columbia U.
Obsoletes: 2833 (if approved) T. Taylor
Expires: December 17, 2006 Nortel
June 15, 2006

RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals
draft-ietf-avt-rfc2833bis-15

Status of this Memo

Copyright Notice

Abstract

This memo describes how to carry dual-tone multifrequency (DTMF) signaling, other tone signals and telephony events in RTP packets. It obsoletes RFC 2833.

This memo captures and expands upon the basic framework defined in RFC 2833, but retains only the most basic event codes. It sets up an IANA registry to which other event code assignments may be added.

Companion documents add event codes to this registry relating to modem, fax, text telephony, and channel-associated signalling events. The remainder of the event codes defined in RFC 2833 are conditionally reserved in case other documents revive their use. See Section 7 and Appendix A for details.

[RFC Editor: please change the Informative reference following RFC 4103 to the RFC number assigned to draft-ietf-avt-rfc2833bisdata-08.txt, one of the "companion documents" referred to above.]

In terms of procedure, this document provides a number of clarifications to the original document. However, it specifically differs from RFC 2833 by removing the requirement that all compliant implementations support the DTMF events. Instead, compliant implementations taking part in out-of-band negotiations of media stream content indicate what events they support. As well, this memo adds three new procedures to the RFC 2833 framework: subdivision of long events into segments, reporting of multiple events in a single packet, and the concept and reporting of state events.

Table of Contents

1. Introduction

1.1. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 [1].

This document uses the following abbreviations:

ANSam Answer tone (amplitude modulated) [24]

DTMF   Dual-Tone Multifrequency [10]

IVR   Integrated Voice Response unit

PBX   Private branch exchange (telephone system)

PSTN   Public Switched (circuit) Telephone Network

RTP   Real-time Transport Protocol [6]

SDP   Session Description Protocol [3]

1.2. Overview

This memo defines two RTP [6] payload formats, one for carrying dual-tone multifrequency (DTMF) digits and other line and trunk signals as events (Section 2), and a second one to describe general multi-frequency tones in terms only of their frequency and cadence (Section 4). Separate RTP payload formats for telephony tone signals are desirable since low-rate voice codecs cannot be guaranteed to reproduce these tone signals accurately enough for automatic recognition. In addition, tone properties such as the phase reversals in the ANSam tone will not survive speech coding. Defining separate payload formats also permits higher redundancy while maintaining a low bit rate. Finally, some telephony events such as "on-hook" occur out-of-band and cannot be transmitted as tones.

The remainder of this section provides the motivation for defining the payload types described in this document. Section 2 defines the payload format and associated procedures for use of named events. Section 3 describes the events for which event codes are defined in this document. Section 4 describes the payload format and associated procedures for tone representations. Section 5 provides some examples of encoded events, tones, and combined payloads. Section 6 deals with security considerations. Section 7 defines the IANA requirements for registration of event codes for named telephone

events, establishes the initial content of that registry, and provides the media type registrations for the two payload formats. Appendix A describes the changes from RFC 2833 [12] and in particular indicates the disposition of the event codes defined in [12].

1.3. Potential Applications

The payload formats described here may be useful in a number of different scenarios.

On the sending side, there are two basic possibilities: either the sending side is an end system which originates the signals itself, or it is a gateway with the task of propagating incoming telephone signals into the Internet.

On the receiving side there are more possibilities. The first is that the receiver must propagate tone signalling accurately into the PSTN for machine consumption. One example of this is a gateway passing DTMF tones to an IVR. In this scenario, frequencies, amplitudes, tone durations, and the durations of pauses between tones are all significant, and individual tone signals must be delivered reliably and in order.

In a second receiving scenario, the receiver must play out tones for human consumption. Typically, rather than a series of tone signals each with its own meaning, the content will consist of a single tone played out continuously or a single sequence of tones and possibly silence, repeated cyclically for some period of time. Often the end of the tone playout will be triggered by an event fed back in the other direction, using either in- or out-of-band means. Examples of this are dial tone or busy tone.

The relationship between position in the network and the tones to be played out is a complicating factor in this scenario. In the phone network, tones are generated at different places, depending on the switching technology and the nature of the tone. This determines, for example, whether a person making a call to a foreign country hears her local tones she is familiar with or the tones as used in the country called.

For analog lines, dial tone is always generated by the local switch. ISDN terminals may generate dial tone locally and then send a Q.931 [22] SETUP message containing the dialed digits. If the terminal just sends a SETUP message without any Called Party digits, then the switch does digit collection (provided by the terminal as KEYPAD key press digit information within Called Party or Keypad Facility IEs of INFORMATION messages), and provides dial tone over the B-channel. The terminal can either use the audio signal on the B-channel or can

use the Q.931 messages to trigger locally generated dial tone.

Ringing tone (also called ringback tone) is generated by the local switch at the callee, with a one-way voice path opened up as soon as the callee's phone rings. (This reduces the chance of clipping the called party's response just after answer. It also permits pre-answer announcements or in-band call-progress indications to reach the caller before or in lieu of a ringing tone.) Congestion tone and

special information tones can be generated by any of the switches along the way, and may be generated by the caller's switch based on ISUP messages received. Busy tone is generated by the caller's switch, triggered by the appropriate ISUP message, for analog instruments, or the ISDN terminal.

In the third scenario, an end system is directly connected to the Internet and processes the incoming media stream directly. There is no need to regenerate tone signals, so that time alignment and power levels are not relevant. These systems rely on sending systems to generate events in place of tones and do not perform their own audio waveform analysis. An example of such a system is an Internet interactive voice-response (IVR) system.

In circumstances where exact timing alignment between the audio stream and the DTMF digits or other events is not important and data is sent unicast, as in the IVR example, it may be preferable to use a reliable control protocol rather than RTP packets. In those circumstances, this payload format would not be used.

Note that in a number of these cases it is possible that the gateway or end system will be both a sender and receiver of telephone signals. Sometimes the same class of signals will be sent as received -- in the case of "RTP trunking" or voiceband data, for instance. In other cases, such as that of an end system serving analogue lines, the signals sent will be in a different class from those received.

1.4. Events, States, Tone Patterns, and Voice Encoded Tones

This document provides the means for in-band transport over the Internet of two broad classes of signalling information: in-band tones or tone sequences, and signals sent out-of-band in the PSTN. Tone signals can be carried using any of the three methods listed below. Depending on the application, it may be desirable to carry the signalling information in more than one form at once.

1. The gateway or end system can upspeed to a higher-bandwidth codec such as G.711 [19] when tone signals are to be conveyed. See new ITU-T Recommendation V.152 [26] for a formal treatment of this

approach. Alternatively, for FAX, text, or modem signals

respectively, a specialized transport such as T.38 [23], RFC 4103 [15], or V.150.1 modem relay [25] may be used. Finally, 64 kbit/s channels may be carried transparently using the RFC 4040 CLEARMODE payload type [14]. These methods are out of scope of the present document, but may be used along with the payload types defined here.

2. The sending gateway can simply measure the frequency components of the voice band signals and transmit this information to the RTP receiver using the tone representation defined in this document (Section 4). In this mode, the gateway makes no attempt to discern the meaning of the tones, but simply distinguishes tones from speech signals. An end system may use the same approach using configured rather than measured frequencies.

   All tone signals in use in the PSTN and meant for human consumption are sequences of simple combinations of sine waves, either added or modulated. (However, some modem signals such as the ANSam tone [24] or systems dependent on phase shift keying cannot be conveyed so simply.)

3. As a third option, a sending gateway can recognize tones such as ringing or busy tone or DTMF digit '0', and transmit a code which identifies them using the telephone-event payload defined in this document (Section 2). The receiver then produces a tone signal or other indication appropriate to the signal. Generally, since the recognition of signals at the sender often depends on their on/off pattern or the sequence of several tones, this recognition can take several seconds. On the other hand, the gateway may have access to the actual signaling information that generates the tones and thus can generate the RTP packet immediately, without the detour through acoustic signals.

The third option (use of named events) is the only feasible method for transmitting out-of-band PSTN signals as content within RTP sessions.

2. RTP Payload Format for Named Telephone Events

2.1. Introduction

The RTP payload format for named telephone events is designated as "telephone-event", the media type as "audio/telephone-event". In accordance with current practice, this payload format does not have a static payload type number, but uses a RTP payload type number established dynamically and out-of-band. The default clock frequency is 8000 Hz, but the clock frequency can be redefined when assigning the dynamic payload type.

Named telephone events are carried as part of the audio stream, and MUST use the same sequence number and time-stamp base as the regular audio channel to simplify the generation of audio waveforms at a gateway. The named telephone events payload type can be considered to be a very highly-compressed audio codec, and is treated the same as other codecs.

2.2. Use of RTP Header Fields

2.2.1. Timestamp

The event duration described in Section 2.5 extends forwards from the time given by the RTP timestamp. For events that span multiple RTP packets, the RTP timestamp identifies the beginning of the event, i.e., several RTP packets may carry the same timestamp. For long-lasting events that have to be split into segments (see below, Section 2.5.1.3), the timestamp indicates the beginning of the segment.

2.2.2. Marker Bit

The RTP marker bit indicates the beginning of a new event. For long-lasting events that have to be split into segments (see below, Section 2.5.1.3), only the first segment will have the marker bit set.

2.3. Payload Format

The payload format for named telephone events is shown in Figure 1.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     event     |E|R| volume    |          duration             |
```

+-+

Figure 1: Payload Format for Named Events

2.3.1. Event Field

The event field is a number between 0 and 255 identifying a specific telephony event. An IANA registry of event codes for this field has been established (see IANA Considerations, Section 7). The initial content of this registry consists of the events defined in Section 3.

2.3.2. E ("End") Bit

If set to a value of one, the "end" bit indicates that this packet contains the end of the event. For long-lasting events that have to be split into segments (see below, Section 2.5.1.3), only the final packet for the final segment will have the "E" bit set.

2.3.3. R Bit

This field is reserved for future use. The sender MUST set it to zero, the receiver MUST ignore it.

2.3.4. Volume Field

For DTMF digits and other events representable as tones, this field describes the power level of the tone, expressed in dBm0 after dropping the sign. Power levels range from 0 to -63 dBm0. Thus, larger values denote lower volume. This value is defined only for events for which the documentation indicates that volume is applicable. For other events, the sender MUST set volume to zero and the receiver MUST ignore the value.

2.3.5. Duration Field

The duration field indicates the duration of the event or segment being reported, in timestamp units, expressed as an unsigned integer in network byte order. For a non-zero value, the event or segment began at the instant identified by the RTP timestamp and has so far lasted as long as indicated by this parameter. The event may or may not have ended. If the event duration exceeds the maximum

representable by the duration field, the event is split into several contiguous segments as described below (Section 2.5.1.3).

The special duration value of zero is reserved to indicate that the event lasts "forever", i.e., is a state and is considered to be effective until updated. A sender MUST NOT transmit a zero duration for events other than those defined as states. The receiver SHOULD ignore an event report with zero duration if the event is not a state.

Events defined as states MAY contain a non-zero duration, indicating that the sender intends to refresh the state before the time duration has elapsed ("soft state").

> For a sampling rate of 8000 Hz, the duration field is sufficient to express event durations of up to approximately 8 seconds.

2.4. Optional media type Parameters

As indicated in the media type registration for named events in Section 7.1.1, the telephone-event media type supports two optional parameters: the "events" parameter, and the "rate" parameter.

The "events" parameter lists the events supported by the implementation. Events are listed as one or more comma-separated elements. Each element can either be a single integer providing the value of an event code or an integer followed by a hyphen and a larger integer, presenting a range of consecutive event code values. The list does not have to be sorted. No white space is allowed in the argument. The union of all of the individual event codes and event code ranges designates the complete set of event numbers supported by the implementation.

The "rate" parameter describes the sampling rate, in Hertz, and hence the units for the RTP timestamp and event duration fields. The number is written as an integer. If omitted, the default value is 8000 Hz.

2.4.1. Relationship to SDP

The recommended mapping of media type optional parameters to SDP is given in section 3 of RFC 3555 [7]. The "rate" media type parameter

for the named event payload type follows this convention: it is expressed as usual as the <clock rate> component of the a=rtpmap: attribute line.

The "events" media type parameter deviates from the convention suggested in RFC 3555 because it omits the string "events=" before the list of supported events.

```
a=fmtp:<format> <list of values>
```

The list of values has the format and meaning described above.

For example, if the payload format uses the payload type number 100, and the implementation can handle the DTMF tones (events 0 through 15) and the dial and ringing tones (assuming as an example that these were defined as events with codes 66 and 70 respectively), it would

include the following description in its SDP message:

```
m=audio 12346 RTP/AVP 100
a=rtpmap:100 telephone-event/8000
a=fmtp:100 0-15,66,70
```

The following sample media type definition corresponds to the SDP example above:

```
audio/telephone-event;events="0-15,66,70";rate="8000"
```

2.5. Procedures

This section defines the procedures associated with the named event payload type. Additional procedures may be specified in the documentation associated with specific event codes.

2.5.1. Sending Procedures

2.5.1.1. Negotiation of Payloads

Events are usually sent in combination with or alternating with other payload types. Payload negotiation may specify separate event and other payload streams, or may specify a combined stream that mixes other payload types with events using RFC 2198 [2] redundancy

headers. The purpose of using a combined stream may be for debugging or to ease the transition between general audio and events.

Negotiation of payloads between sender and receiver is achieved by out-of-band means, using SDP, for example.

The sender SHOULD indicate what events it supports, using the optional "events" parameter associated with the telephone-events media type. If the sender receives an "events" parameter from the receiver, it MUST restrict the set of events it sends to those listed in the received "events" parameter. For backward compatibility, if no "events" parameter is received, the sender SHOULD assume support for the DTMF events 0-15 but for no other events.

Events MAY be sent in combination with older events using RFC 2198 [2] redundancy. Section 2.5.1.4 describes how this can be used to avoid packet and RTP header overheads when retransmitting final event reports. Section 2.6 discusses the use of additional levels of RFC 2198 redundancy to increase the probability that at least one copy of the report of the end of an event reaches the receiver. The following SDP shows an example of such usage, where G.711 audio appears in a separate stream, and the primary component of the redundant payload is events.


```
m=audio 12344 RTP/AVP 99
a=rtpmap:99 pcmu/8000
m=audio 12346 RTP/AVP 100 101
a=rtpmap:100 red/8000/1
a=fmtp:100 101/101/101
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
```

When used in accordance with the offer-answer model (RFC 3264 [5]), the SDP a=ptime: attribute indicates the packetization period that the author of the session description expects when receiving media. This value does not have to be the same in both directions. The appropriate period may vary with the application, since increased packetization periods imply increased end-to-end response times in instances where one end responds to events reported from the other.

Negotiation of telephone-events sessions using SDP MAY specify such differences by separating events corresponding to different

applications into different streams. In the example below, events 0-15 are DTMF events, which have a fairly wide tolerance on timing. Events 32-49 and 52-60 are events related to data transmission and are subject to end-to-end response time considerations. As a result, they are assigned a smaller packetization period than the DTMF events.

```
m=audio 12344 RTP/AVP 99
a=rtpmap:99 telephone-event/8000
a=fmtp:99 0-15
a=ptime:50
m=audio 12346 RTP/AVP 100
a=rtpmap:100 telephone-event/8000
a=fmtp:100 32-49,52-60
a=ptime:30
```

For further discussion of packetization periods see Section 2.6.3.

2.5.1.2. Transmission of Event Packets

DTMF digits and other named telephone events are carried as part of the audio stream, and MUST use the same sequence number and time-stamp base as the regular audio channel to simplify the generation of audio waveforms at a gateway.

An audio source SHOULD start transmitting event packets as soon as it recognizes an event, and continue to send updates until the event has ended. The update packets MUST have the same RTP timestamp value as the initial packet for the event, but the duration MUST be increased to reflect the total cumulative duration since the beginning of the

event.

The first packet for an event MUST have the "M" bit set. The final packet for an event MUST have the "E" bit set, but setting of the "E" bit MAY be deferred until the final packet is retransmitted (see Section 2.5.1.4). Intermediate packets for an event MUST NOT have either the "M" bit or the "E" bit set.

Sending of a packet with the "E" bit set is OPTIONAL if the packet reports two events which are defined as mutually exclusive states, or if the final packet for one state is immediately followed by a packet

reporting a mutually exclusive state. (For events defined as states, the appearance of a mutually exclusive state implies the end of the previous state.)

A source has wide latitude as to how often it sends event updates. A natural interval is the spacing between non-event audio packets. (Recall that a single RTP packet can contain multiple audio frames for frame-based codecs and that the packet interval can vary during a session.) Alternatively, a source MAY decide to use a different spacing for event updates, with a value of 50 ms RECOMMENDED.

Timing information is contained in the RTP timestamp, allowing precise recovery of inter-event times. Thus, the sender does not in theory need to maintain precise or consistent time intervals between event packets. However, the sender SHOULD minimize the need for buffering at the receiving end by sending event reports at constant intervals.

> DTMF digits and other tone events are sent incrementally to avoid having the receiver wait for the completion of the event. In some cases (for example, data session startup protocols), waiting to the end of a tone before reporting it will cause the session to fail. In other cases, it will simply cause undesirable delays in playout at the receiving end.

For robustness, the sender SHOULD retransmit "state" events periodically.

2.5.1.3. Long Duration Events

If an event persists beyond the maximum duration expressible in the duration field (0xFFFF), the sender MUST send a packet reporting this maximum duration but MUST NOT set the "E" bit in this packet. The sender MUST then begin reporting a new "segment" with the RTP timestamp set to the time at which the previous segment ended and the duration set to the cumulative duration of the new segment. The "M" bit of the first packet reporting the new segment MUST NOT be set.

The sender MUST repeat this procedure as required until the end of the complete event has been reached. The final packet for the complete event MUST have the "E" bit set (either on initial transmission or on retransmission as described below).

2.5.1.3.1. Exceptional Procedure For Combined Payloads

If events are combined as a redundant payload with another payload type using RFC 2198 [2] redundancy, the above procedure SHALL be applied, but using a maximum duration which ensures that the timestamp offset of the oldest generation of events in an RFC 2198 packet never exceeds 0x3FFF. If the sender is using a constant packetization period, the maximum segment duration can be calculated from the following formula:

    maximum duration = 0x3FFF - (R-1)*(packetization period in
    timestamp units)

where R is the highest redundant layer number consisting of event payload.

> The RFC 2198 redundancy header timestamp offset value is only 14 bits, compared with the 16 bits in the event payload duration field. Since with other payloads the RTP timestamp typically increments for each new sample, the timestamp offset value becomes limiting on reported event duration. The limit becomes more constraining when older generations of events are also included in the combined payload.

2.5.1.4. Retransmission of Final Packet

The final packet for each event and for each segment SHOULD be sent a total of three times at the interval used by the source for updates. This ensures that the duration of the event or segment can be recognized correctly even if an instance of the last packet is lost.

A sender MAY use RFC 2198 [2] with up to two levels of redundancy to combine retransmissions with reports of new events, thus saving on header overheads. In this usage, the primary payload is new event reports, while the first and (if necessary) second levels of redundancy report first and second retransmissions of final event reports. Within a session negotiated to allow such usage, packets containing the RFC 2198 payload SHOULD NOT be sent except when both primary and retransmitted reports are to be included. All other packets of the session SHOULD contain only the simple, non-redundant telephone-event payload. Note that the expected proportion of simple versus redundant packets affects the order in which they should be specified on an SDP m= line.

There is little point in sending initial or interim event reports redundantly because each succeeding packet describes the event fully (except for typically irrelevant variations in volume).

A sender MAY delay setting the "E" bit until retransmitting the last packet for a tone, rather than setting the bit on its first transmission. This avoids having to wait to detect whether the tone has indeed ended. Once the sender has set the "E" bit for a packet, it MUST continue to set the "E" bit for any further retransmissions of that packet.

2.5.1.5. Packing Multiple Events Into One Packet

Multiple named events can be packed into a single RTP packet if and only if the events are consecutive and contiguous, i.e., occur without overlap and without pause between them, and if the last event packed into a packet occurs quickly enough to avoid excessive delays at the receiver.

This approach is similar to having multiple frames of frame-based audio in one RTP packet.

The constraint that packed events not overlap implies that events designated as states can be followed in a packet only by other state events which are mutually exclusive to them. The constraint itself is needed so that the beginning time of each event can be calculated at the receiver.

In a packet containing events packed in this way, the RTP timestamp MUST identify the beginning of the first event or segment in the packet. The "M" bit MUST be set if the packet records the beginning of at least one event. (This will be true except when the packet carries the end of one segment and the beginning of the next segment of the same long-lasting event.) The "E" bit and duration for each event in the packet MUST be set using the same rules as if that event were the only event contained in the packet.

2.5.1.6. RTP Sequence Number

The RTP sequence number MUST be incremented by one in each successive RTP packet sent. Incrementing applies to retransmitted as well as initial instances of event reports, to permit the receiver to detect lost packets for RTCP receiver reports.

2.5.2. Receiving Procedures

2.5.2.1. Indication of Receiver Capabilities using SDP

Receivers can indicate which named events they can handle, for example, by using the Session Description Protocol (RFC 2327 [3]). SDP descriptions using the event payload MUST contain an fmtp format attribute that lists the event values that the receiver can process.

2.5.2.2. Playout of Tone Events

In the gateway scenario, an Internet telephony gateway connecting a packet voice network to the PSTN recreates the DTMF or other tones and injects them into the PSTN. Since, for example, DTMF digit recognition takes several tens of milliseconds, the first few milliseconds of a digit will arrive as regular audio packets. Thus, careful time and power (volume) alignment between the audio samples and the events is needed to avoid generating spurious digits at the receiver. The receiver may also choose to delay playout of the tones by some small interval after playout of the preceding audio has ended, to ensure that downstream equipment can discriminate the tones properly.

Some implementations send events and encoded audio packets (e.g., PCMU or the codec used for speech signals) for the same time instant for the duration of the event. It is RECOMMENDED that gateways render only the telephone-event payload once it is received, since the audio may contain spurious tones introduced by the audio compression algorithm. However, it is anticipated that these extra tones in general should not interfere with recognition at the far end.

Receiver implementations MAY use different algorithms to create tones, including the two described here. (Note that not all implementations have the need to recreate a tone; some may only care about recognizing the events.) With either algorithm, a receiver may impose a playout delay to provide robustness against packet loss or delay. The tradeoff between playout delay and other factors is discussed further in Section 2.6.3.

In the first algorithm, the receiver simply places a tone of the given duration in the audio playout buffer at the location indicated

by the timestamp. As additional packets are received that extend the same tone, the waveform in the playout buffer is extended accordingly. (Care has to be taken if audio is mixed, i.e., summed, in the playout buffer rather than simply copied.) Thus, if a packet in a tone lasting longer than the packet interarrival time gets lost and the playout delay is short, a gap in the tone may occur.

Alternatively, the receiver can start a tone and play it until one of

the following occurs:

o it receives a packet with the "E" bit set;

o it receives the next tone, distinguished by a different timestamp value (noting that new segments of long-duration events also appear with a new timestamp value);

o it receives an alternative non-event media stream (assuming none was being received while the event stream was active); or

o a given time period elapses.

This is more robust against packet loss, but may extend the tone beyond its original duration if all retransmissions of the last packet in an event are lost. Limiting the time period of extending the tone is necessary to avoid that a tone "gets stuck". This algorithm is not a license for senders to set the duration field to zero; it MUST be set to the current duration as described, since this is needed to create accurate events if the first event packet is lost, among other reasons.

Regardless of the algorithm used, the tone SHOULD NOT be extended by more than three packet interarrival times. A slight extension of tone durations and shortening of pauses is generally harmless.

A receiver SHOULD NOT restart a tone once playout has stopped. It MAY do so if the tone is of a type meant for human consumption or is one for which interruptions will not cause confusion at the receiving device.

If a receiver receives an event packet for an event which it is not currently playing out and the packet does not have the "M" bit set,

earlier packets for that event have evidently been lost. This can be confirmed by gaps in the RTP sequence number. The receiver MAY determine on the basis of retained history and the timestamp and event code of the current packet that it corresponds to an event already played out and lapsed. In that case further reports for the event MUST be ignored, as indicated in the previous paragraph.

If, on the other hand, the event has not been played out at all, the receiver MAY attempt to play the event out to the complete duration indicated in the event report. The appropriate behaviour will depend on the event type concerned, and requires consideration of the relationship of the event to audio media flows and whether correct event duration is essential to the correct operation of the media session.

A receiver SHOULD NOT rely on a particular event packet spacing, but instead MUST use the event timestamps and durations to determine timing and duration of playout.

The receiver MUST calculate jitter for RTCP receiver reports based on all packets with a given timestamp. Note: The jitter value should primarily be used as a means for comparing the reception quality between two users or two time-periods, not as an absolute measure.

If a zero volume is indicated for an event for which the volume field is defined, then the receiver MAY reconstruct the volume from the volume of non-event audio or MAY use the nominal value specified by the ITU Recommendation or other document defining the tone. This ensures backwards compatibility with RFC 2833 [12], where the volume field was defined only for DTMF events.

2.5.2.3. Long Duration Events

If an event report is received with duration equal to the maximum duration expressible in the duration field (0xFFFF) and the "E" bit for the report is not set, the event report may mark the end of a segment generated according to the procedures of Section 2.5.1.3. If another report for the same event type is received, the receiver MUST compare the RTP timestamp for the new event with the sum of the RTP timestamp of the previous report plus the duration (0xFFFF). The receiver uses the absence of a gap between the events to detect that

it is receiving a single long-duration event.

The total duration of a long duration event is (obviously) the sum of the durations of the segments used to report it. This is equal to the duration of the final segment (as indicated in the final packet for that segment), plus 0xFFFF multiplied by the number of segments preceding the final segment.

2.5.2.3.1. Exceptional Procedure For Combined Payloads

If events are combined as a redundant payload with another payload type using RFC 2198 [2] redundancy, segments are generated at intervals of 0x3FFF or less, rather than 0xFFFF, as required by the procedures of Section 2.5.1.3.1 in this case. If a receiver is using the events component of the payload, event duration may be only an approximate indicator of division into segments, but the lack of an E-bit and the adjacency of two reports with the same event code are strong indicators in themselves.

2.5.2.4. Multiple Events In a Packet

The procedures of Section 2.5.1.5 require that if multiple events are

reported in the same packet, they are contiguous and non-overlapping. As a result, it is not strictly necessary for the receiver to know the start times of the events following the first one in order to play them out -- it needs only to respect the duration reported for each event. Nevertheless, if knowledge of the start time for a given event after the first one is required, it is equal to the sum of the start time of the preceding event plus the duration of the preceding event.

2.5.2.5. Soft States

If the duration of a soft state event expires, the receiver SHOULD consider the value of the state to be "unknown" unless otherwise indicated in the event documentation.

2.6. Congestion and Performance

Packet transmission through the internet is marked by occasional periods of congestion lasting in the order of seconds, during which

network delay, jitter, and packet loss are all much higher than they are in between these periods. [28] provides a typical characterization of this phenomenon. Well-behaved applications are expected, preferably, to reduce their demands on the network during such periods of congestion. At the least they should not increase their demands. This section explores both application performance and the possibilities for good behaviour in the face of congestion.

2.6.1. Performance Requirements

Typically an implementation of the telephone-events payload will aim to limit the rate at which each of the following impairments occurs:

a. an event encoded at the sender fails to be played out at the receiver, either because the event report is lost or because it arrives after playout of later content has started;

b. the start of playout of an event at the receiver is delayed relative to other events or other media operating on the same timestamp base;

c. the duration of playout of a given event differs from the correct duration as detected at the sender by more than a given amount;

d. gaps occur in playout of a given event;

e. end-to-end delay for the media stream exceeds a given value.

The relative importance of these constraints varies between

applications.

2.6.2. Reliability Mechanisms

One reliability mechanism is available to every payload type including telephone-events. This is to use a jitter buffer (i.e., impose a playout delay) at the receiving end. This mechanism addresses the first four requirements listed above, but at the expense of the last one.

The named event procedures provide two complementary redundancy mechanisms to deal with lost packets:

a. Intra-event updates:

Events that last longer than one packetization period (e.g., 50 ms) are updated periodically, so that the receiver can reconstruct the event and its duration if it receives any of the update packets, albeit with delay.

During an event, the RTP event payload format provides incremental updates on the event. The error resiliency afforded by this mechanism depends on whether the first or second algorithm in Section 2.5.2.2 is used and on the playout delay at the receiver. For example, if the receiver uses the first algorithm and only places the current duration of tone signal in the playout buffer, for a playout delay of 120 ms and a packetization interval of 50 ms, two packets in a row can get lost without causing a premature end of the tone generated.

b. Repeat last event packet:

As described in Section 2.5.1.4, the last report for an event is transmitted a total of three times. This mechanism adds robustness to the reporting of the end of an event.

It may be necessary to extend the level of redundancy to achieve objective a) in a specific network environment. Taking the 25-30% loss rate during congestion periods illustrated in [28] as typical, and setting an objective that at least 99% of end-of-event reports will eventually get through to the receiver under these conditions, simple probability calculations indicate that each event completion has to be reported four times. This is one more level of redundancy than required by the basic "Repeat last event packet" algorithm. Of course, the objective is probably unrealistically stringent; it was chosen to make a point.

Where Section 2.5.1.4 indicates that it is appropriate to use the

RFC 2198 [2] audio redundancy mechanism to carry retransmissions of final event reports, this mechanism MAY also be used to extend the number of final report retransmissions. This is done by using more than two levels of redundancy when necessary. The use of RFC 2198 helps to mitigate the extra bandwidth demands that

would be imposed simply by retransmitting final event packets more than three times.

These two redundancy mechanisms clearly address requirement a) in the previous section. They also help meet requirement c), to the extent that the redundant packets arrive before playout of the events they report is due to expire. They are not helpful in meeting the other requirements, although they do not directly cause impairments themselves in the way that a large jitter buffer increases end-to-end delay.

The playout algorithm is an additional mechanism for meeting the performance requirements. In particular, using the second algorithm in Section 2.5.2.2 will meet requirement d) of the previous section by preventing gaps in playout, but at the potential cost of increases in duration (requirement c)).

Finally, there is an interaction between the packetization period used by a sender, the playout delay used by the receiver, and the vulnerability of an event flow to packet losses. Assuming packet losses are independent, a shorter packetization interval means that the receiver can use a smaller playout delay to recover from a given number of consecutive packet losses, at any stage of event playout. This improves end-to-end delays in applications where that matters.

In view of the tradeoffs between the different reliability mechanisms, documentation of specific events SHOULD include a discussion of the appropriate design decisions for the applications of those events. This mandate is repeated in the section on IANA Considerations.

2.6.3. Adjusting To Congestion

So far the discussion has been about meeting performance requirements. However, there is also the question of whether applications of events can adapt to congestion to the point that they reduce their demands on the networks during congestion. In theory this can be done for events by increasing the packetization interval, so that fewer packets are sent per second. This has to be accompanied by an increased playout delay at the receiving end. Coordination between the two ends for this purpose is an interesting issue in itself. If it is done, however, such an action implies a one-time gap or extended playout of an event when the packetization

interval is first extended, as well as increased end-to-end delay during the whole period of increased playout delay.

The benefit from such a measure varies primarily depending on the average duration of the events being handled. In the worst case, as a first example shows, the reduction in aggregate bandwidth usage due to an increased packetization interval may be quite modest. Suppose the average event duration is 3.33 ms (V.21 bits, for instance). Suppose further that four transmissions in total are required for a given event report to meet the loss objective. Table 1 shows the impact of varying packetization interval on the aggregate bit rate of the media stream.

Packetization Interval (ms)	Packets/s	IP Packet Size (bits)	Total IP Bit Rate (bits/s)
50	20	2440	48800
33.3	30	1800	54000
25	40	1480	59200
20	50	1288	64400

Table 1: Data Rate At the IP Level versus Packetization Interval (three retransmissions, 3.33 ms per event)

As can be seen, a doubling of the interval (from 25 to 50 ms) drops aggregate bit rate by about 20% while increasing end-to-end delay by 25 ms and causing a one-time gap of the same amount. (Extending the playout of a specific V.21 tone event is out of the question, so the first algorithm of Section 2.5.2.2 must be used in this application.) The reduction in number of packets per second with longer packetization periods is countered by the increase in packet size due to the increase in number of events per packet.

For events of longer duration, the reduction in bandwidth is more proportional to the increase in packetization interval. The loss of final event reports may also be less critical, so that lower redundancy levels are acceptable. Table 2 shows similar data to Table 1, but assuming 70 ms events separated by 50 ms of silence (as in an idealized DTMF-based text messaging session) with only the basic two retransmissions for event completions.

Packetization Interval (ms)	Packets/s	IP Packet Size (bits)	Total IP Bit Rate (bits/s)
50	20	448/520	10040
33.3	30	448/520	14280
25	40	448/520	18520
20	50	448	22400

Table 2: Data Rate At the IP Level versus Packetization Interval (two retransmissions, 70 ms per event, 50 ms between events)

In the third column of the table, packet size is 448 bits when only one event is being reported and 520 bits when the previous event is also included. No more than one level of redundancy is needed up to a packetization interval of 50 ms, although at that point most packets are reporting two events. Longer intervals require a second level of redundancy in at least some packets.

3. Specification of Event Codes For DTMF Events

This document defines one class of named events: DTMF tones.

3.1. DTMF Applications

DTMF signalling [10] is typically generated by a telephone set or possibly by a PBX (Private branch telephone exchange). DTMF digits may be consumed by entities such as gateways or application servers in the IP network, or by entities such as telephone switches or IVRs in the circuit switched network.

The DTMF events support two possible applications at the sending end:

1. the Internet telephony gateway detects DTMF on the incoming circuits and sends the RTP payload described here instead of regular audio packets. The gateway likely has the necessary digital signal processors and algorithms, as it often needs to detect DTMF, e.g., for two-stage dialing. Having the gateway detect tones relieves the receiving Internet end system from having to do this work and also avoids having low bit-rate codecs like G.723.1 [20] render DTMF tones unintelligible.

2. an Internet end system such as an "Internet phone" can emulate DTMF functionality without concerning itself with generating precise tone pairs and without imposing the burden of tone recognition on the receiver.

A similar distinction occurs at the receiving end.

1. In the gateway scenario, an Internet telephony gateway connecting a packet voice network to the PSTN recreates the DTMF tones or other telephony events and injects them into the PSTN.

2. In the end system scenario, the DTMF events are consumed by the

receiving entity itself.

In the most common application, DTMF tones are sent in one direction only, typically from the calling end. The consuming device is most commonly an IVR. DTMF may alternate with voice from either end. In most cases the only constraint on tone duration is that it exceed a minimum value. However, in some cases a long-duration tone (in excess of 1-2 seconds) has special significance.

ITU-T Recommendation Q.24 [11], Table A-1, indicates that the legacy switching equipment in the countries surveyed expects a minimum recognizable signal duration of 40 ms, a minimum pause between signals of 40 ms, and a maximum signalling rate of 8 to 10

digits per second depending on the country. Human-generated DTMF signals, of course, are generally longer with larger pauses between them.

DTMF tones may also be used for text telephony. This application is documented in ITU-T Recommendation V.18 [27] Annex B. In this case, DTMF is sent alternately from either end (half-duplex mode), with a minimum 300 ms turn-around time. The only constraints on tone durations in this application are that they and the pauses between them must exceed specified minimum values. It is RECOMMENDED that a gateway at the sending end be capable of detecting DTMF signals as specified by V.18 Annex B (tones and pauses >=40 ms) but should send event durations corresponding to those of a V.18 DTMF sender (tones >=70 ms, pauses >=50 ms). This may occasionally imply some degree of buffering of outgoing events, but if the source terminal conforms to V.18 Annex B this should not get out of hand.

Since minor increases in tone duration are harmless for all applications of DTMF, but unintended breaks in playout of a DTMF digit can confuse the receiving endpoint by creating the appearance of extra digits, receiving applications that are converting DTMF events back to tones SHOULD use the second playout algorithm rather than the first one in Section 2.5.2.2. This provides some robustness against packet loss or congestion.

3.2. DTMF Events

Table 3 shows the DTMF-related event codes within the telephone-event

payload format. The DTMF digits 0-9 and * and # are commonly supported. DTMF digits A through D are less frequently encountered, typically in special applications such as military networks.

Event	Code	Type	Volume?
0--9	0--9	tone	yes
*	10	tone	yes
#	11	tone	yes
A--D	12--15	tone	yes

Table 3: DTMF named events

3.3. Congestion Considerations

The key considerations for the delivery of DTMF events are reliability and avoidance of unintended breaks within the playout of a given tone. End-to-end round-trip delay is not a major consideration except in the special case where DTMF tones are being used for text telephony. Assuming that, as recommended in Section 3.1 above, the second playout algorithm of Section 2.5.2.2 is in use, a temporary increase in packetization interval to as much as 100 ms or double the normal interval, whichever is less, should be harmless.

4. RTP Payload Format for Telephony Tones

4.1. Introduction

As an alternative to describing tones and events by name, as described in Section 2, it is sometimes preferable to describe them by their waveform properties. In particular, recognition is faster than for naming signals since it does not depend on recognizing durations or pauses.

There is no single international standard for telephone tones such as dial tone, ringing (ringback), busy, congestion ("fast-busy"), special announcement tones or some of the other special tones, such as payphone recognition, call waiting or record tone. However, ITU-T

Recommendation E.180 [18] notes that across all countries, these tones share a number of characteristics:

o Telephony tones consist of either a single tone, the addition of two or three tones or the modulation of two tones. (Almost all tones use two frequencies; only the Hungarian "special dial tone" has three.) Tones that are mixed have the same amplitude and do not decay.

o In-band tones for telephony events are in the range of 25 Hz (ringing tone in Angola) to 2600 Hz (the tone used for line signalling in SS No. 5 and R1). The in-band telephone frequency range is limited to 3400 Hz. R2 defines a 3825 Hz out-of-band tone for line signalling on analogue trunks. (The piano has a range from 27.5 to 4186 Hz.)

o Modulation frequencies range between 15 (ANSam tone) to 480 Hz (Jamaica). Non-integer frequencies are used only for frequencies of 16 2/3 and 33 1/3 Hz.

o Tones that are not continuous have durations of less than four seconds.

o ITU Recommendation E.180 [18] notes that different telephone companies require a tone accuracy of between 0.5 and 1.5%. The Recommendation suggests a frequency tolerance of 1%.

4.2. Examples of Common Telephone Tone Signals

As an aid to the implementor, Table 4 summarizes some common tones. The rows labeled "ITU ..." refer to ITU-T Recommendation E.180 [18]. In these rows, the on and off durations are suggested ranges within which local standards would set specific values. The symbol "+" in the table indicates addition of the tones, without modulation, while

"*" indicates amplitude modulation.

Tone Name	Frequency	On Time (s)	Off Time (s)
CNG	1100	0.5	3.0

V.25 CT	1300	0.5	2.0
CED	2100	3.3	--
ANS	2100	3.3	--
ANSam	2100*15	3.3	--
V.21 bit	980 or 1180 or 1650 or 1850	0.00333	--
-------------	----------	--------	--------
ITU dial tone	425	--	--
U.S. dial tone	350+440	--	--
ITU ringing tone	425	0.67-1.5	3-5
U.S. ringing tone	440+480	2.0	4.0
ITU busy tone	425	0.1-0.6	0.1-0.7
U.S. busy tone	480+620	0.5	0.5
ITU congestion tone	425	0.1-0.6	0.1-0.7
U.S. congestion tone	480+620	0.25	0.25

Table 4: Examples of telephony tones

4.3. Use of RTP Header Fields

4.3.1. Timestamp

The RTP timestamp reflects the measurement point for the current packet. The event duration described in Section 4.3.3 extends forwards from that time.

4.3.2. Marker Bit

The tones payload type uses the marker bit to distinguish the first RTP packet reporting a given instance of a tone from succeeding packets for that tone. The marker bit SHOULD be set to 1 for the first packet, and to 0 for all succeeding packets relating to the same tone.

4.3.3. Payload Format

Based on the characteristics described above, this document defines an RTP payload format called "tone" that can represent tones consisting of one or more frequencies. (The corresponding media type is "audio/tone".) The default timestamp rate is 8000 Hz, but other rates may be defined. Note that the timestamp rate does not affect the interpretation of the frequency, just the durations.

In accordance with current practice, this payload format does not have a static payload type number, but uses a RTP payload type number established dynamically and out-of-band.

The payload format is shown in Figure 2.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    modulation   |T|  volume   |          duration             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|R R R R|       frequency       |R R R R|       frequency       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|R R R R|       frequency       |R R R R|       frequency       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    ......

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|R R R R|       frequency       |R R R R|      frequency        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Payload Format for Tones

The payload contains the following fields:

modulation:

The modulation frequency, in Hz. The field is a 9-bit unsigned integer, allowing modulation frequencies up to 511 Hz. If there is no modulation, this field has a value of zero. Note that the amplitude of modulation is not indicated in the payload and must

be determined by out-of-band means

T:

If the "T" bit is set (one), the modulation frequency is to be divided by three. Otherwise, the modulation frequency is taken as is.

This bit allows frequencies accurate to 1/3 Hz, since modulation frequencies such as 16 2/3 Hz are in practical use.

volume:

The power level of the tone, expressed in dBm0 after dropping the sign, with range from 0 to -63 dBm0. (Note: A preferred level range for digital tone generators is -8 dBm0 to -3 dBm0.)

duration:

The duration of the tone, measured in timestamp units and presented in network byte order. The tone begins at the instant identified by the RTP timestamp and lasts for the duration value. The value of zero is not permitted and tones with such a duration SHOULD be ignored.

The definition of duration corresponds to that for sample-based codecs, where the timestamp represents the sampling point for the first sample.

frequency:

The frequencies of the tones to be added, measured in Hz and represented as a 12-bit unsigned integer. The field size is sufficient to represent frequencies up to 4095 Hz, which exceeds the range of telephone systems. A value of zero indicates silence. A single tone can contain any number of frequencies. If no frequencies are specified, the packet reports a period of silence.

R:

This field is reserved for future use. The sender MUST set it to zero, the receiver MUST ignore it.

4.3.4. Optional media type Parameters

The "rate" parameter describes the sampling rate, in Hertz. The number is written as an integer. If omitted, the default value is

8000 Hz.

4.4. Procedures

This section defines the procedures associated with the tones payload type.

4.4.1. Sending Procedures

The sender MAY send an initial tones packet as soon as a tone is recognized, or MAY wait until a pre-negotiated packetization period has elapsed. The first RTP packet for a tone SHOULD have the marker bit set to 1.

In the case of longer-duration tones, the sender SHOULD generate multiple RTP packets for the same tone instance. The RTP timestamp MUST be updated for each packet generated (in contrast, for instance, to the timestamp for packets carrying telephone-events). Subsequent packets for the same tone SHOULD have the marker bit set to 0, and the RTP timestamp in each subsequent packet MUST equal the sum of the timestamp and the duration in the preceding packet.

A final RTP packet MAY be generated as soon as the end of the tone is detected, without waiting for the latest packetization period to elapse.

The telephone-event payload described in Section 2 is inherently redundant, in that later packets for the same event carry all of the earlier history of the event except for variations in volume. In contrast, each packet for the tone payload type stands alone; a lost packet means a gap in the information available at the receiving end. Thus, for increased reliability, the sender SHOULD combine new and old tone reports in the same RTP packet using RFC 2198 [2] audio redundancy.

4.4.2. Receiving Procedures

Receiving implementations play out the tones as received, typically with a playout delay to allow for lost packets. When playing out successive tone reports for the same tone (marker bit is zero, the RTP timestamp is contiguous with that of the previous RTP packet, and payload content is identical), the receiving implementation SHOULD continue the tone without change or a break.

4.4.3. Handling Of Congestion

If the sender determines that packets are being lost due to congestion (e.g., through RTCP receiver reports), it SHOULD increase

the packetization interval for initial and interim tone reports so as to reduce traffic volume to the receiver. The degree to which this is possible without causing damaging consequences at the receiving end depends both upon the playout delay used at that end and upon the specific application associated with the tones. Both the maximum packetization interval and maximum increase in packetization interval at any one time are therefore a matter of configuration or out-of-band negotiation.

5. Examples

Consider a DTMF dialling sequence, where the user dials the digits "911" and a sending gateway detects them. The first digit is 200 ms long (1600 timestamp units) and starts at time 0; the second digit lasts 250 ms (2000 timestamp units) and starts at time 880 ms (7040 timestamp units); the third digit is pressed at time 1.4 s (11,200 timestamp units) and lasts 220 ms (1760 timestamp units). The frame duration is 50 ms.

Table 5 shows the complete sequence of events assuming that only the telephone-events payload type is being reported. For simplicity: the timestamp is assumed to begin at 0, the RTP sequence number at 1, and volume settings are omitted.

Time (ms)	Event	M bit	Time stamp	Seq No	Event Code	Duration	E bit
0	"9" starts						

50	RTP packet 1 sent	"1"	0	1	9	400	"0"
100	RTP packet 2 sent	"0"	0	2	9	800	"0"
150	RTP packet 3 sent	"0"	0	3	9	1200	"0"
200	RTP packet 4 sent	"0"	0	4	9	1600	"0"
200	"9" ends						
250	RTP packet 4 first retrans mission	"0"	0	5	9	1600	"1"

300	RTP packet 4 second retrans mission	"0"	0	6	9	1600	"1"
880	First "1" starts						
930	RTP packet 5 sent	"1"	7040	7	1	400	"0"
...	...	...	...	...	...	...	...

1130	RTP packet 9 sent	"0"	7040	11	1	2000	"0"
1130	First "1" ends						
1180	RTP packet 9 first retrans mission	"0"	7040	12	1	2000	"1"
1230	RTP packet 9 second retrans mission	"0"	7040	13	1	2000	"1"
1400	Second "1" starts						
1450	RTP packet 10 sent	"1"	11200	14	1	400	"0"
...	...	...	...	...	...	...	...
1620	Second "1" ends						

1650	RTP packet 14 sent	"0"	11200	18	1	1760	"1"
1700	RTP packet 14 first retrans	"0"	11200	19	1	1760	"1"

	mission							
1750	RTP packet 14 second retrans mission	"0"	11200	20	1	1760	"1"

Table 5: Example of Event Reporting

Table 6 shows the same sequence assuming that only the tone payload type is being reported. This looks somewhat different. For simplicity: the timestamp is assumed to begin at 0, the sequence number at 1. Volume, the T bit, and the modulation frequency are omitted. The latter two are always 0.

Time (ms)	Event	M bit	Time stamp	Seq No	Duration	Freq 1 (Hz)	Freq 2 (Hz)
0	"9" starts						
50	RTP packet 1 sent	"1"	0	1	400	852	1477
100	RTP packet 2 sent	"0"	400	2	400	852	1477
...	...	...	...	...	...	...	...
200	RTP packet 4 sent	"0"	1200	4	400	852	1477
200	"9" ends						
880	First "1" starts						
930	RTP packet 5 sent	"1"	7040	5	400	697	1209
980	RTP packet 6 sent	"0"	7440	6	400	697	1209
...	...	...	...	...	...	...	...
1130	First "1" ends						
1400	Second "1" starts						
1450	RTP packet 10 sent	"1"	11200	10	400	697	1209
...	...	...	...	...	...	...	...

1620	Second "1" ends						
1650	RTP packet 14 sent	"0"	12800	14	160	697	1209

Table 6: Example of Tone Reporting

Now consider a combined payload, where the tone payload is the primary payload type and the event payload is treated as a redundant encoding (one level of redundancy). Because the primary payload is tones, the tone payload rules determine the setting of the RTP header fields. This means that the RTP timestamp always advances. As a corollary, the timestamp offset for the events payload in the RFC 2198 header increases by the same amount.

One issue that has to be considered in a combined payload is how to handle retransmissions of final event reports. The tones payload specification does not recommend retransmissions of final packets, so it is unclear what to put in the primary payload fields of the combined packet. In the interests of simplicity it is suggested that the retransmitted packets copy the fields relating to the primary payload (including the RTP timestamp) from the original packet. The same principle can be applied if the packet includes multiple levels of event payload redundancy.

The figures below all illustrate "RTP packet 14" in the above tables. Figure 3 shows an event-only payload, corresponding to Table 5. Figure 4 shows an event-only payload, corresponding to Table 6. Finally, Figure 5 shows a combined payload, with tones primary and events as a single redundant layer. Note that the combined payload has the RTP sequence numbers shown in Table 5, because the transmitted sequence includes the retransmitted packets.

Figure 3 assumes that the following SDP specification was used. This session description provides for separate streams of G.729 [21] audio and events. Packets reported within the G.729 stream are not considered here.

```
m=audio 12344 RTP/AVP 99
a=rtpmap:99 G729/8000
a=ptime:20
m=audio 12346 RTP/AVP 100
a=rtpmap:100 telephone-event/8000
```

```
   a=fmtp:100 0-15
   a=ptime:50
```



```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |V=2|P|X|  CC   |M|     PT      |       sequence number         |
   | 2 |0|0|   0   |0|    100      |            18                 |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           timestamp                           |
   |                             11200                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           synchronization source (SSRC) identifier            |
   |                            0x5234a8                           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     event     |E R| volume    |          duration             |
   |       1       |1 0|    20     |             1760              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: Example RTP Packet For Event Payload

Figure 4 assumes that an SDP specification similar to that of the previous case was used.

```
   m=audio 12344 RTP/AVP 99
   a=rtpmap:99 G729/8000
   a=ptime:20
   m=audio 12346 RTP/AVP 101
   a=rtpmap:101 tone/8000
   a=ptime:50
```

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |V=2|P|X|  CC   |M|     PT      |       sequence number         |
   | 2 |0|0|   0   |0|    101      |             14                |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           timestamp                           |
   |                             12800                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           synchronization source (SSRC) identifier            |
```

```
      |                            0x5234a8                           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |    modulation    |T|  volume    |          duration          |
      |        0         |0|    20      |             160            |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |R R R R|       frequency        |R R R R|       frequency      |
      |0 0 0 0|          697           |0 0 0 0|         1209         |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```


Figure 4: Example RTP Packet For Tone Payload

Figure 5, for the combined payload, assumes the following SDP session description:

```
m=audio 12344 RTP/AVP 99
a=rtpmap:99 G729/8000
a=ptime:20
m=audio 12346 RTP/AVP 102 101 100
a=rtpmap:102 red/8000/1
a=fmtp:102 101/100
a=rtpmap:101 tone/8000
a=rtpmap:100 telephone-event/8000
a=fmtp:100 0-15
a=ptime:50
```

For ease of presentation, Figure 5 presents the actual payloads as if they began on 32-bit boundaries. In the actual packet, they follow immediately after the end of the RFC 2198 header, and thus are displaced one octet into successive words.

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |V=2|P|X|  CC   |M|     PT      |       sequence number         |
   | 2 |0|0|   0   |0|    102      |            18                 |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           timestamp                           |
   |                             12800                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           synchronization source (SSRC) identifier            |
   |                            0x5234a8                           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |F|   block PT  |  timestamp offset         |   block length    |
   |1|      100    |       1600                |         4         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |F|   block PT  |   event payload begins ...                    /
   |0|      101    |                                               \
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Event payload

```
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     event     |E R| volume    |          duration             |
   |       1       |1 0|    20     |            1760               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Tone payload

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     modulation    |T|  volume     |          duration         |
|         0         |0|    20       |             160           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|R R R R|       frequency       |R R R R|       frequency       |
|0 0 0 0|          697          |0 0 0 0|         1209          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: Example RTP Packet For Combined Tone and Event Payloads

6. Security Considerations

RTP packets using the payload formats defined in this specification are subject to the security considerations discussed in the RTP specification (RFC 3550 [6]), and any appropriate RTP profile (for example RFC 3551 [13]). The RFC 3550 discussion focusses on requirements for confidentiality. Additional security considerations relating to implementation are described in RFC 2198 [2].

The telephony-event payload defined in this specification is highly compressed. A change in value of just one bit can result in a major change in meaning as decoded at the receiver. Thus message integrity MUST be provided for the telephony-event payload type.

To meet the need for protection both of confidentiality and integrity, compliant implementations SHOULD implement the Secure Real-time Transport Protocol (SRTP) [8].

Note that the appropriate method of key distribution for SRTP may vary with the specific application.

In some deployments it may be preferable to use other means to provide protection equivalent to that provided by SRTP.

Provided that gateway design includes robust, low overhead tone generation, this payload type does not exhibit any significant non-uniformity in the receiver side computational complexity for packet processing to cause a potential denial-of-service threat.

7. IANA Considerations

This document updates the descriptions of two RTP payload formats, 'telephone-event' and 'tone', and associated Internet media types, audio/telephone-event and audio/tone. It also documents the event codes for DTMF tone events.

Within the audio/telephone-event type, events MUST be registered with IANA. Registrations are subject to the policies "Specification Required" and "Expert Review" as defined in RFC 2434 [4]. The IETF-appointed expert must ensure that:

a. the meaning and application of the proposed events is clearly documented;

b. the events cannot be represented by existing event codes, possibly with some minor modification of event definitions;

c. the number of events is the minimum necessary to fulfil the purpose of their application(s).

The expert is further responsible for providing guidance on the allocation of event codes to the proposed events. Specifically, the expert must indicate whether the event appears to be the same as one defined in RFC 2833 but not specified in any new document. In this case the event code specified in RFC 2833 for that event SHOULD be assigned to the proposed event. Otherwise event codes MUST be assigned from the set of available event codes listed below. If this set is exhausted, the criterion for assignment from the reserved set of event codes is to first assign those that appear to have the lowest probability of being revived in their RFC 2833 meaning in a new specification.

The documentation for each event MUST indicate whether the event is a state, tone, or other type of event (e.g., an out-of-band electrical event such as on-hook or an indication that will not itself be played out as tones at the receiving end). For tone events, the documentation MUST indicate whether the volume field is applicable or must be set to 0.

In view of the tradeoffs between the different reliability mechanisms discussed in Section 2.6, documentation of specific events SHOULD include a discussion of the appropriate design decisions for the applications of those events.

Legal event codes range from 0 to 255. The initial registry content is shown in Table 7, and consists of the sixteen events defined in Section 3 of this document. The remaining codes have the following

disposition:

o codes 17-22, 50-51, 90-95, 113-120, 169, and 206-255 are available for assignment;

o codes 23-40, 49, and 52-63 are reserved for events defined in [16];

o codes 121-137 and 174-205 are reserved for for events defined in [17];

o codes 16, 41-48, 64-88, 96-112, 138-168, and 170-173 are reserved in the first instance for specifications reviving the corresponding RFC 2833 events, and in the second instance for general assignment after all other codes have been assigned.

Event Code	Event Name	Reference
0	DTMF digit "0"	<This RFC>
1	DTMF digit "1"	<This RFC>
2	DTMF digit "2"	<This RFC>
3	DTMF digit "3"	<This RFC>
4	DTMF digit "4"	<This RFC>
5	DTMF digit "5"	<This RFC>
6	DTMF digit "6"	<This RFC>
7	DTMF digit "7"	<This RFC>
8	DTMF digit "8"	<This RFC>
9	DTMF digit "9"	<This RFC>
10	DTMF digit "*"	<This RFC>
11	DTMF digit "#"	<This RFC>
12	DTMF digit "A"	<This RFC>
13	DTMF digit "B"	<This RFC>
14	DTMF digit "C"	<This RFC>
15	DTMF digit "D"	<This RFC>

Table 7: audio/telephone-event Event Code Registry

7.1. Media type registrations

7.1.1. Registration of media type audio/telephone-event

This registration is done in accordance with [7] and [9].

Type name: audio

Subtype name: telephone-event

Required parameters: none.

Optional parameters:

The "events" parameter lists the events supported by the implementation. Events are listed as one or more comma-separated elements. Each element can either be a single integer providing the value of an event code or an integer followed by a hyphen and a larger integer, presenting a range of consecutive event code values. The list does not have to be sorted. No white space is allowed in the argument. The union of all of the individual event codes and event code ranges designates the complete set of event numbers supported by the implementation. If the "events" parameter is omitted, support for events 0-15 (the DTMF tones) is assumed.

The "rate" parameter describes the sampling rate, in Hertz. The number is written as an integer. If omitted, the default value is 8000 Hz.

Encoding considerations:

In the terminology defined by [9] section 4.8, this type is framed and binary.

Security considerations:

See the "Security Considerations" section (Section 6) in this document.

Interoperability considerations: none.

Published specification: this document.

Applications which use this media:

The telephone-event audio subtype supports the transport of events occuring in telephone systems over the Internet.

Additional information:

Magic number(s): N/A.

File extension(s): N/A.

Macintosh file type code(s): N/A.

Person & email address to contact for further information:

Tom Taylor, taylor@nortel.com.

IETF AVT Working Group.

Intended usage: COMMON.

Restrictions on usage:

This type is defined only for transfer via RTP [6].

Author: IETF Audio/Video Transport Working Group.

Change controller:

IETF Audio/Video Transport Working Group as delegated from the IESG.

7.1.2. Registration of media type audio/tone

This registration is done in accordance with [7] and [9].

Type name: audio

Subtype name: tone

Required parameters: none

Optional parameters:

The "rate" parameter describes the sampling rate, in Hertz. The number is written as an integer. If omitted, the default value is 8000 Hz.

Encoding considerations:

In the terminology defined by [9] section 4.8, this type is framed and binary.

Security considerations:

See the "Security Considerations" section (Section 6) in this document.

Interoperability considerations: none

Published specification: this document.

Applications which use this media:

The tone audio subtype supports the transport of pure composite tones, for example those commonly used in the current telephone system to signal call progress.

Additional information:

Magic number(s): N/A.

File extension(s): N/A.

Macintosh file type code(s): N/A.

Person & email address to contact for further information:

Tom Taylor, taylor@nortel.com.

IETF AVT Working Group.

Intended usage: COMMON.

Restrictions on usage:

This type is defined only for transfer via RTP [6].

Author: IETF Audio/Video Transport Working Group.

Change controller:

IETF Audio/Video Transport Working Group as delegated from the IESG.

8. Acknowledgements

Scott Petrack was the original author of RFC 2833. Henning Schulzrinne later loaned his expertise to complete the document, but Scott must be credited with the energy behind the idea of a compact encoding of tones over IP.

In RFC 2833, the suggestions of the Megaco working group were acknowledged. Colin Perkins and Magnus Westerland, Chairs of the AVT Working Group, provided helpful advice in the formation of the present document. Over the years, detailed advice and comments for RFC 2833, this document, or both were provided by Hisham Abdelhamid, Flemming Andreasen, Fred Burg, Steve Casner, Dan Deliberato, Fatih Erdin, Bill Foster, Mike Fox, Mehryar Garakani, Gunnar Hellstrom, Rajesh Kumar, Terry Lyons, Steve Magnell, Zarko Markov, Kai Miao, Satish Mundra, Kevin Noll, Vern Paxson, Oren Peleg, Raghavendra Prabhu, Moshe Samoha, Todd Sherer, Adrian Soncodi, Yaakov Stein, Mira Stevanovic, Tim Melanchuk, Alex Urquizo and Herb Wildfeur.

9. References

9.1. Normative References

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

[2] Perkins, C., Kouvelas, I., Hodson, O., Hardman, V., Handley, M., Bolot, J., Vega-Garcia, A., and S. Fosse-Parisis, "RTP Payload for Redundant Audio Data", RFC 2198, September 1997.

[3] Handley, M. and V. Jacobson, "SDP: Session Description

Protocol", RFC 2327, April 1998.

[4] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[5] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

[6] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[7] Casner, S. and P. Hoschka, "MIME Type Registration of RTP Payload Formats", RFC 3555, July 2003.

[8] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[9] Freed, N. and J. Klensin, "Media Type Specifications and Registration Procedures", BCP 13, RFC 4288, December 2005.

[10] International Telecommunication Union, "Technical features of push-button telephone sets", ITU-T Recommendation Q.23, November 1988.

[11] International Telecommunication Union, "Multifrequency push-button signal reception", ITU-T Recommendation Q.24, November 1988.

9.2. Informative References

[12] Schulzrinne, H. and S. Petrack, "RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals", RFC 2833, May 2000.

[13] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, July 2003.

[14] Kreuter, R., "RTP Payload Format for a 64 kbit/s Transparent Call", RFC 4040, April 2005.

[15] Hellstrom, G. and P. Jones, "RTP Payload for Text Conversation", RFC 4103, June 2005.

[16] Schulzrinne, H. and T. Taylor, "Definition of Events For Modem, FAX, and Text Telephony Signals", RFC yyyy, June 2006.

[17] Schulzrinne, H. and T. Taylor, "Definition of Events For Channel-Oriented Telephony Signalling", Work In Progress , November 2005.

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

[19] International Telecommunication Union, "Pulse code modulation (PCM) of voice frequencies", ITU-T Recommendation G.711, November 1988.

[20] International Telecommunication Union, "Speech coders : Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s", ITU-T Recommendation G.723.1, March 1996.

[21] International Telecommunication Union, "Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)", ITU-T Recommendation G.729, March 1996.

[22] International Telecommunication Union, "ISDN user-network interface layer 3 specification for basic call control", ITU-T Recommendation Q.931, May 1998.

[23] International Telecommunication Union, "Procedures for real-time Group 3 facsimile communication over IP networks", ITU-T Recommendation T.38, July 2003.

[24] International Telecommunication Union, "Procedures for starting sessions of data transmission over the public switched telephone network", ITU-T Recommendation V.8, November 2000.

[25] International Telecommunication Union, "Modem-over-IP networks: Procedures for the end-to-end connection of V-series DCEs", ITU-T Recommendation V.150.1, January 2003.

[26] International Telecommunication Union, "Procedures for supporting Voice-Band Data over IP Networks", ITU-T Recommendation V.152, January 2005.

[27] International Telecommunication Union, "Operational and interworking requirements for {DCEs operating in the text telephone mode", ITU-T Recommendation V.18, November 2000.

See also Recommendation V.18 Amendment 1, Nov. 2002.

[28] VOIP Troubleshooter LLC, "Indepth: Packet Loss Burstiness", 2005, <http://www.voiptroubleshooter.com/indepth/burstloss.html>.

Appendix A. Summary of Changes from RFC 2833

The memo has been significantly restructured, incorporating a large number of clarifications to the specification. With the exception of those items noted below, the changes to the memo are intended to be backwards compatible clarifications. However, due to inconsistencies and unclear definitions in RFC 2833 [12] it is likely that some implementations interpreted that memo in ways that differ from this version.

RFC 2833 required that all implementations be capable of receiving the DTMF events (event codes 0-15). Section 2.5.1.1 of the present document requires that a sender transmit only the events that the receiver is capable of receiving. In the absence of a knowledge of receiver capabilities, the sender SHOULD assume support of the DTMF events but of no other events. The sender SHOULD indicate what events it can send. Section 2.5.2.1 requires that a receiver signalling its capabilities using SDP MUST indicate which events it can receive.

Non-zero values in the Volume field of the payload were applicable only to DTMF tones in RFC 2833, and for other events the receiver was required to ignore them. The present memo requires that the definition of each event indicate whether the Volume field is applicable to that event. The last paragraph of Section 2.5.2.2 indicates what a receiver may do if it receives volumes with zero values for events to which the Volume field is applicable. Along with the RFC 2833 receiver rule, this ensures backward compatibility in both directions of transmission.

Section 2.5.1.3 and Section 2.5.2.3 introduce a new procedure for reporting and playing out events whose duration exceeds the capacity of the payload Duration field. This procedure may cause momentary confusion at an old (RFC 2833) receiver, because the timestamp is updated without setting the E bit of the preceding event report and without setting the M bit of the new one.

Section 2.5.1.5 and Section 2.5.2.4 introduce a new procedure whereby a sequence of short-duration events may be packed into a single event report. If an old (RFC 2833) receiver receives such a report, it may discard the packet as invalid, since the packet holds more content than the receiver was expecting. In any event, the additional events

in the packet will be lost.

Section 2.3.5 introduces the possibility of "state" events and defines procedures for setting the Duration field for reports of such events. Section 2.5.1.2 defines special exemptions from the setting of the E bit for state events. Three more sections mention

procedures related to these events.

The Security Considerations section is updated to mention the requirement for protection of integrity. More importantly, it makes implementation of SRTP [8] mandatory for compliant implementations, without specifying a mandatory-to-implement method of key distribution.

Finally, this document establishes an IANA registry for event codes and establishes criteria for their documentation. This document provides an initial population for the new registry, consisting solely of the sixteen DTMF events. Two companion documents [16] and [17] describe events related to modems, fax, and text telephony and to channel-associated telephony signalling respectively. Some changes were made to the latter because of errors and redundancies in the RFC 2833 assignments. The remaining events defined in RFC 2833 are deprecated because they do not appear to have been implemented, but their codes have been conditionally reserved in case any of them is needed in the future. Table 8 indicates the disposition of the event codes in detail. Event codes not mentioned in this table were not allocated by RFC 2833 and continue to be unused.

Event Codes	RFC 2833 Description	Disposition
0-15	DTMF digits	This RFC
16	Line flash (deprecated)	Reserved
23-31	Unused	[16]
32-40	Data and fax	[16]
41-48	Data and fax (V.8bis, deprecated)	Reserved
52-63	Unused	[16]
64-89	E.182 line events (deprecated)	Reserved
96-112	Country-specific line events (deprecated)	Reserved
121-127	Unused	[17]
128-137	Trunks: MF 0-9	[17]
138-143	Trunks: other MF (deprecated)	Reserved
144-159	Trunks: ABCD signalling	[17]

160-168	Trunks: various (deprecated)	Reserved
170-173	Trunks: various (deprecated)	Reserved
174-205	Unused	[17]

Table 8: Disposition of RFC 2833-defined event codes

Authors' Addresses

Henning Schulzrinne
Columbia U.
Dept. of Computer Science
Columbia University
1214 Amsterdam Avenue
New York, NY 10027
US

Email: schulzrinne@cs.columbia.edu

Tom Taylor
Nortel
1852 Lorraine Ave
Ottawa, Ontario K1H 6Z8
CA

Email: taylor@nortel.com

Intellectual Property Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.