SIP FORUM

SIP Forum – Fax Over IP Task Group – Problem Statement

T.38: Problems related to SIP/SDP Negotiation

While the T.38 protocol, approved by the ITU-T in 1998, was designed to allow fax machines and computer-based fax to carry forward in a transitioning communications infrastructure of both IP- and TDM-based telephony, in 2009 there are enough problems and confusion among vendors, enterprises, and service providers to significantly slow the use of IP as a real-time fax transport. The issues surrounding IP-based fax in general and the use of T.38 make it difficult for users to determine if T.38 can or will work reliably and thus offer an alternative to traditional TDM-based fax transport. To address these problems and offer solutions, the SIP Forum has chartered the FoIP Task Group (TG).

The charter of the SIP Forum FoIP task group is to investigate ongoing issues with the deployment of fax services, specifically ITU-T T.38, in SIP networks. SIP networks cannot adequately replace analog and digital PSTN in enterprises unless essential services such as fax are accommodated.

This document details a number of SDP offer/answer interoperability issues found while implementing and connecting T.38 compliant endpoints together, primarily over the SIP signaling mechanism. However, many of the issues presented here are not specific to SIP or SDP, but are in fact problems that would occur over any T.38-capable signaling mechanism.

Definitions and References

1. **ITU Recommendation T.38 (T38)**

   In this document, references to the ITU T.38 Recommendation are specific to the April, 2007 published version.

2. **Internet Facsimile Transfer (IFT)**

   As defined in T38, the method of transferring facsimile data over the Internet.

3. **Internet Facsimile Protocol (IFP)**

   As defined in T38, the protocol used to implement IFT by carrying ITU T.30 facsimile protocol signaling and data over an Internet Protocol connection.

4. **Party A or Calling Party**

   The endpoint that initiated the call in question.

5. **Party B or Called Party**

   The endpoint that received the call in question.

6. Emitting Gateway

As defined in T38, the IFP endpoint which initiates IFT service for a calling facsimile endpoint. In typical networks, this will be a media gateway or similar device.

7. Receiving Gateway

As defined in T38, the IFP endpoint which accepts an IFT service connection request from an Emitting Gateway and provides IFT service to an answering facsimile endpoint. In typical networks, this will be a media gateway or similar device.

8. Internet-Aware FAX (IAF)

A facsimile endpoint that can communicate using the T.38 IFT directly, without the need of a media gateway. IAFs were first defined in T38 Amendment 3.

Problems

While the following is not an all-inclusive list, it presents the highest-priority issues as determined by the Task Group.

1. Triggering of T.38 switchover

Problems

T38 does not indicate which party (Party A or Party B) is responsible for detecting that a FAX transmission is being attempted and initiating the switch from audio mode to T.38 mode. In practice, this results in some common scenarios:

- A Receiving Gateway may initiate T.38 after detecting CED/ANS tones generated by the Party B endpoint.
- A Receiving Gateway may initiate T.38 after detecting the V.21 HDLC flags (preamble) generated by the party B endpoint.
- A Receiving Gateway may initiate T.38 after detecting CNG tone generated by the Party A Endpoint.
- An Emitting Gateway may initiate T.38 after detecting CNG tone generated by the Party A endpoint.

In some cases, gateways may attempt to detect tones generated by the far endpoint, which may be unreliable if the audio connection between the endpoints is using a highly compressed voice codec.

While it is generally accepted that the Receiving Gateway should initiate the switch to T.38, and it should only do this after detecting the V.21 HDLC flags generated by the endpoint it services (to ensure that the answering device is in fact a facsimile endpoint, and not a data modem or other device that might also generate CED/ANS tones), in practice this is not the case, and actual devices may act in any of the fashions listed above. This can easily result in 'glare', where both gateways attempt to switch to T.38 (nearly) simultaneously, or a complete lack of T.38 switchover if the detection method in use is not adequately able to detect the far endpoint's generated tones. In a 'glare' situation, if the gateways do not properly implement backoff procedures as defined in RFC 3261, the call will likely fail.

2. T.38 requested on initial INVITE

Problems

IAF devices and some gateways may send an initial INVITE containing a T.38 media stream offer. In most cases, this media stream will be marked as 'inactive', and will be accompanied by an active audio stream offer. In other cases, no audio stream offer will be present at all (primarily IAF devices). If no audio stream is present, a receiving endpoint may reject the INVITE because it cannot determine whether T.38 will be supported by the session's eventual destination until after some call routing processes have completed.

3. T.38 session parameters changed during T.30 session

Problems

Some endpoints will send re-INVITE messages containing modified T.38 session parameters in their SDP offer **after** the endpoints have previously agreed on T.38 session parameters and the T.38 session has begun. If the receiver of this offer accepts it, the implication is that the active T.38 session's parameters will be modified to conform to the new offer, which is unlikely to be possible since the session is already active. If the receiver of the offer rejects it, the sender of the offer is likely to drop the call.

4. Minimum/maximum redundancy IFPs in UDPTL frames

Problems

T38 documents an error correction scheme (referred to as 'error protection') that involves sending redundant copies of previously-transmitted IFP messages in subsequent UDPTL frames. This affords the receiver the opportunity to recover frames lost in transit. Indication of the support of this mode is made by setting the T38FaxUdpEC parameter to 't38UDPRedundancy". However, T38 makes no recommendation about the number of redundancy messages that should be included in a UDPTL packet, nor does it have any language to take into account the redundancy messages when computing the maximum IFP size that can be transmitted to the receiving endpoint (based on its reported T38FaxMaxBuffer and/or T38FaxMaxDatagram parameters).

5. Suppression of audio during T.38 switchover

Problems

When a Receiving Gateway senses that the endpoint it is servicing is attempting to initiate a facsimile connection, and the gateway intends to switch to T.38 to service that connection, it may not suppress the audio stream from the endpoint towards the Emitting Gateway. If it does not do so, the endpoints will attempt to negotiate a T.30 facsimile connection over the audio stream while the T.38 session is being established in the signaling path. While the T.38 session negotiation process, under normal circumstances, should occur rapidly enough to prevent the endpoints from exchanging DIS and DCS, if this does occur, then when the T.38 session begins the facsimile connection will have progressed too far to be recovered and it will fail.

6. T.38 Session Parameter: T38FaxVersion

Problems

This parameter appears clear and simple at first sight, but recent discussions show that some people try to read more into this parameter than they should. Some people interpret the version as more of a functionality level indicator, so 3 implies the T.38 device supports V.34. T38 does not appear to support this interpretation, though it offers no clear way for a T.38 device to assess if a remote T.38 device supports V.34 (though T38MaxBitRate may infer it).

Testing if T38FaxVersion is zero vs non-zero is a critically important distinction terminals need to make, otherwise they can't interpret the ASN.1 they receive. Implementations must support all versions up to the one they advertise as T38FaxVersion. If one side says it supports up to version X, and the other side says it supports up to version Y, when Y > X, the communication will happen both ways in version X mode. That is, the highest version they have in common will be used.

A missing T38FaxVersion field implies version 0, and this appears to achieve natural compatibility with the very oldest implementations.

Even today, version 0 is still by far the most common in use. Most T.38 entities support nothing else.

7. T.38 Session Parameter: T38MaxBitRate

Problems

This parameter indicates the maximum fax image bit rate supported by the endpoint. This is an odd way to express things, as the endpoints vary by which modem standards they support, rather than which bit rates. However, in practice the following implications seem to work out OK for all known implementations of T.38:

T38MaxBitRate:9600 implies V.29 and V.27ter support.

T38MaxBitRate:14400 implies V.17, V.29 and V.27ter support.

T38MaxBitRate:33600 implies V.34, V.17, V.29 and V.27ter support.

There appears to be some confusion that this parameter has some bandwidth management purpose. T38 doesn't support that. It simply says it is the maximum FAX bit rate. The actual bandwidth in the IP channel may vary greatly, depending on chunk sizes and the level of redundancy/FEC used.

8. T.38 Session Parameter: T38FaxFillBitRemoval

Problems

This parameter indicates that it is acceptable for the Emitting Gateway to remove all fill bits from a non-ECM image, because the receiving side will reinsert the appropriate minimum amount. This seems clear, though it is seldom supported.

Removing 100% of the fill bits requires deep inspection of the image, but removing >95% is actually a very lightweight processing task, and can save some worthwhile bandwidth for non-ECM calls. The receiving side needs to adjust the fill bits, for flow control, so reimposing a minimum on a bit stream stripped of fill bits at the same time is a minor additional task.

9. T.38 Session Parameter: T38FaxTranscodingMMR

Problems

This parameter indicates the ability to transcode MH/MR from/to a facsimile endpoint to MMR data between the T.38 gateways. It is unclear whether this is supposed to work only for non-ECM transmission, or for non-ECM and ECM. Since most ECM transmission uses T.6 or JBIG encoding, it may not be a big issue, but it isn't properly specified. It is easy to implement this for non-ECM faxing, but it would be impracticable for ECM. This kind of transcoding does have the potential to cut the bandwidth requirement between the T.38 gateways significantly. However, nearly all fax machines are able to do MMR compression these days, so machines that don't use it do so by configuration choice. It is questionable, therefore, whether the T.38 gateways should override such user choices.

10. T.38 Session Parameter: T38FaxTranscodingJBIG

Problems

It seems impossible to use the T38FaxTranscodingJBIG option, as T38 specifies it so vaguely. It is supposed to indicate the ability to send JBIG data between T.38 gateways, when the facsimile endpoints connected to those gateways are using some other (presumably poorer) bi-level compression. However, T38 says nothing about how this is supposed to work. JBIG can only be used with guaranteed-delivery transport (such as TCP).

11. T.38 Session Parameter: T38FaxMaxBuffer

Problems

This parameter tells each end how much buffer space the other end has. The values for the two directions are completely independent. However, the exact meaning of "buffer space" is not clarified in T38. Is it received UDPTL/RTP/TPKT packets? Is it IFP messages? Is it the data extracted from IFP messages? Also, the exact playout status of the far end is never known, as it needs to time various things for itself. T38 gives no guidance about the timing of the start of data relative to the carrier start indicators, which makes this buffer issue quite serious. It is never possible to determine exactly how much might be in the far end's buffer. Therefore, this parameter seems to be of little practical value. Most T.38

implementations advertise only a small buffer, so it is important not to flood them with data, which could overrun such a small buffer. In practice, most implementations just seem to avoid sending anything until its time appears to be due, and hope for the best.

12. T.38 Session Parameter: T38FaxMaxDatagram

Problems

This parameter seems to be interpreted in different ways by different implementations of T.38. It is tied in with the greatest weakness of the T.38 spec right now - the lack of adequate chunking and timing guidance. The lack of this gives the designer of the receiving side only a vague idea of what to expect from the sending side, seriously dumbing down what the receiving side can do. It also mightily confuses the designer of the sending side as to what is appropriate to send for maximum compatibility.

T38 table B.1 says "This option indicates the maximum size of a UDPTL packet or the maximum size of the payload within an RTP packet that can be accepted by the remote device."

T38 D.2.1.3.1 says "The maximum size of the payload within an RTP packet that can be accepted by the remote device."

T38 D.2.3.5 says "This parameter signals the largest acceptable datagram for the offering endpoint and the answering endpoint (i.e. the maximum size of the RTP payload). The answering endpoint may accept a larger or smaller datagram than the offering endpoint. Each endpoint should be considerate of the maximum datagram size of the opposite endpoint."

What exactly is the RTP payload? Before adding redundancy, or after? What about UDPTL? A transmitted RTP packet can obviously be larger than T38FaxMaxDatagram, as you need to add the framing words to the payload. What happens in the case of UDPTL, where the framing and redundancy coalesce? The meaning of T38FaxMaxDatagram appears to depend on the transport type, which is a rather odd design.

In practice it seems some systems treat T38FaxMaxDatagram as the maximum IFP length, and some treat it as the maximum UDPTL length. I infer this, because some systems use a number too small for it to be anything but the maximum IFP length. Either that, or they are not prepared to accept any redundancy/FEC data. The number is also much smaller than the same box is prepared to accept for an RTP packet of audio, which seems to imply its buffers can accept much bigger UDP packets.

The only other real guidance about chunking seems to be in T38 7.5. This says the following, in recent versions of T.38, as a supposed clarification of earlier wording:

"Limitation of V.21 frame packet size

To reduce the gateway processing delay, the use of smaller V.21 frame data packets is more beneficial for interconnected gateways to flexibly perform jitter buffer adjustment according to the network situation and compatibility of the facsimile terminal.

The maximum V.21 packet size shall be 7 bytes, except for IAF devices. Larger V.21 frames shall be sent in multiple packets."

What does "The maximum V.21 packet size" mean? Presumably packet here refers to an IFP, but is it the total IFP packet that should be <=7 bytes or its V.21 payload? The description seems to imply the total packet length, yet its the payload length we are trying to constrain. T38 7.5 seems to confuse more than clarify.

In practice a lot of ATAs are not happy if the V.21 data IFPs contain more than 1 byte of V.21 data. Most gateways send one byte per frame, and this is reasonably harmless. The annoying (though fully workable) case is sending T.38 from a terminating T.38 entity. Here 100% of the HDLC frame's content is known at the instant frame transmission starts, and the frames are always fairly small. Still, we end up sending a messy and inefficient stream of packets, with one byte of the HDLC frame in each, to maximize compatibility.

13. Media Stream Configuration after T.38 switchover from audio

Problems

When a call begins as audio and then switches to T.38, the media stream configuration in place after the switchover can vary greatly depending on the endpoint implementations. There are examples in the field of:

1. The call begins with only an audio stream, and after switchover there is only a T.38 stream.
2. The call begins with an audio stream (active) and a T.38 stream (inactive); after switchover, the audio stream is inactive and the T.38 stream Is active.
3. The call begins with only an audio stream, but after switchover there are **both** audio and T.38 streams, with both marked as active. In this case, it is highly unlikely that the endpoint sending such an offer is actually prepared to receive audio and T.38 simultaneously.
4. The call begins with only an audio stream, but after switchover that stream is converted to T.38 and a **new** audio stream is added to the SDP. While the endpoint may in fact be expecting this to be treated as retaining the existing audio stream (active or inactive), the SDP RFCs define behavior based on the position that the stream appears in the offer or answer, and “moving” the audio stream from the first position to the second position in fact makes it a new (different) stream from the one that was in place prior to the switchover. Receivers of such an offer will likely treat the offer in the way that the offerer intended (retain the existing audio stream, possibly changing its active state, and adding a T.38 stream), but this is only because implementers have learned to do so to increase interoperability, not because the standards mandate, or even suggest, such behavior.